

# Package `canoniclayout`\*

Claudio Beccari<sup>†</sup>

January 22, 2011

## Abstract

This package is a small extension to the memoir class that allows to determine the page parameters for a canonical layout where the text block is placed onto the page in such a way as: (a) its diagonal coincides with the page relevant diagonal, and (b) the circle inscribed within the text block is also tangent to the page sides. The text block width to height ratio is kept equal to the page ratio, as well as the inner to outer and the top to bottom margin ratios.

The canonical layout determined by this package is laid down on this very page; of course the actual text block does not fit the text block grid, because this documentation is typeset by means of the `ltxdoc` document class, that uses different layout parameters. At the same time the classical layout, conceived for two sided printing, gets properly compared with a standard (one sided printing) L<sup>A</sup>T<sub>E</sub>X layout.<sup>1</sup>

<b>Contents</b>	<b>3</b>	<b>Usage</b>	<b>4</b>
<b>1 Introduction</b>	<b>1</b>	<b>4 Font normal size</b>	<b>4</b>
<b>2 The mathematics of the canonical layout</b>	<b>3</b>	<b>5 Conclusion</b>	<b>5</b>
	<b>3</b>	<b>6 Documented code</b>	<b>6</b>

## 1 Introduction

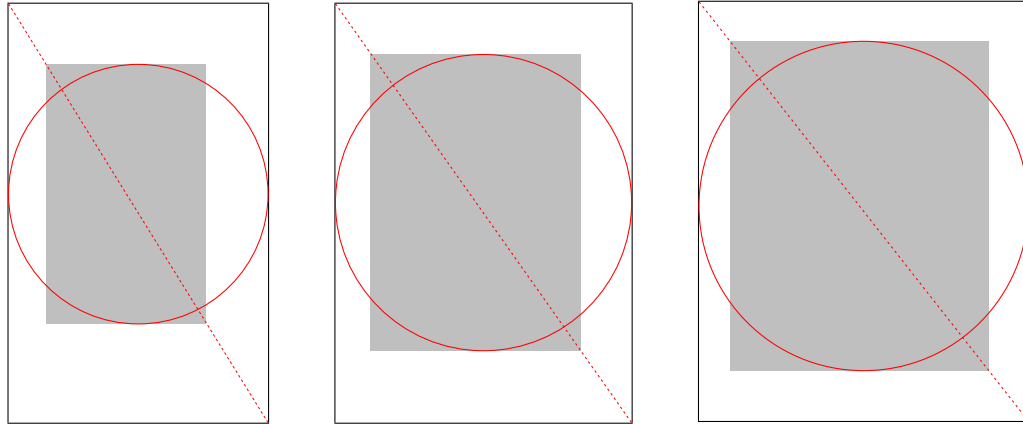
As any user of the memoir class knows well, this class allows to customize the page layout virtually without any limitation; such limitations are connected mainly with the user inconsistent data that miss to specify necessary information or give unnecessary and contrasting information.

---

\*Version number v.0.2; last revision 2010/12/04.

<sup>†</sup>e-mail: `claudio dot beccari at gmail dot com`

<sup>1</sup>Actually the `ltxdoc` class default paper size is the letter paper; here we rounded a little bit the letter paper dimensions to integer millimeter values of 220 mm × 280 mm. See the discussion about the convenience of such layout when the main font size is considered.



Golden number canonical layout    ISO canonical layout    Letter paper canonical layout

Figure 1: A right page with the relevant diagonal, the text block and the canonical circle. In this figure the important information is the page proportions, not the scale; matter of fact the letter paper is 17.6 mm shorter than the A4 paper, but the drawings to the same height emphasize the relative proportions of the various page parts.

Many packages exist in order to specify a named layout; this package `canonical` is another one that computes the page parameters to determine the horizontal and vertical dimensions of the margins and the text block. All other details shall be specified by the user, for example the dimensions of the stock paper, the trimming widths, and therefore the position of the page onto the stock paper, and so on. This package deals only with the trimmed page, not with the page and stock paper relationship.

Figure 1 depicts the situation described above; the gray rectangle represents the type block; the dotted page diagonal, as it can be seen, passes through the diagonal vertices of the type block; at the same time the circle is tangent to both the type block and the page contour.

It must also be noticed, in the sketch on the left, that the page rectangle is a golden one; therefore also the type block is a golden rectangle; the inner to the outer margin ratio is the golden section value so as this same ratio exists between the top and the bottom margin. This canonical layout implies that the type block height is equal to the page width. Therefore this canonical layout is not suited when the page width is small compared to the page height: with the golden section<sup>2</sup> this ratio is 0.618, with the ISO paper shape, in the sketch on the center, this ratio is 0,707, and so on. But it is equally unsuitable with squarish pages, such as the sketch on the right where the page proportions conform with

<sup>2</sup>Here we denote  $\phi = 1,618\dots$  with the name of *golden number*, and its reciprocal  $\varphi = 0,618\dots$  with the name of *golden section*.

the USA letter paper size: the type block leaves little space for the margins, as it is evident from the properties of the canonical circle, which is tangent to the page lateral sides and to the type block horizontal sides; to the limit in a square page the type block is so large that there is no room for the margins.

## 2 The mathematics of the canonical layout

If one plays a little with the drawing of figure 1 and applies what has been described in the Introduction, may easily derive the following formulas:

$$\text{Page shape ratio:} \quad x = b/h \quad (1)$$

$$\text{Inner margin:} \quad I = bx(1-x)/(1+x) \quad (2)$$

$$\text{External margin:} \quad E = b(1-x)/(1+x) \quad (3)$$

$$\text{Top margin:} \quad T = hx(1-x)/(1+x) \quad (4)$$

$$\text{Bottom margin:} \quad B = h(1-x)/(1+x) \quad (5)$$

$$\text{Text width:} \quad W = xb \quad (6)$$

$$\text{Text height:} \quad H = b \quad (7)$$

$$\text{Circle center abscissa:} \quad C_x = 0.5b \quad (8)$$

$$\text{Circle center ordinate:} \quad C_y = h(1-x)/(1+x) + b/2 \quad (9)$$

$$\text{Circle radius:} \quad R = 0.5b \quad (10)$$

whose results are summarized in table 1 for three common values of the shape factor and for the letter paper shape ratio.

Actually the formulas 1–10, and their numerical evaluations reported in table 1, are just for general information. The actual computations are performed by this

---

Table 1: Canonical layout ratios referred to the the base (or width,  $b$ ) of the trimmed page and its height ( $h$ ).

Quantity	Canon			
	$\phi : 1$	$3 : 2$	$\sqrt{2} : 1$	$11 : 8.5$
Page shape ratio	0,618	0,666	0,707	0,773
Inner margin	0,146 <i>b</i>	0,133 <i>b</i>	0,121 <i>b</i>	0,099 <i>b</i>
External margin	0,236 <i>b</i>	0,200 <i>b</i>	0,172 <i>b</i>	0,128 <i>b</i>
Top margin	0,146 <i>h</i>	0,133 <i>h</i>	0,121 <i>h</i>	0,099 <i>h</i>
Bottom margin	0,236 <i>h</i>	0,200 <i>h</i>	0,172 <i>h</i>	0,128 <i>h</i>
Text width	0,618 <i>b</i>	0,666 <i>b</i>	0,707 <i>b</i>	0,773 <i>b</i>
Text height	<i>b</i>	<i>b</i>	<i>b</i>	<i>b</i>
Circle center abscissa	<i>b</i> /2	<i>b</i> /2	<i>b</i> /2	<i>b</i> /2
Circle center ordinate	0,545 <i>h</i>	0,533 <i>h</i>	0,526 <i>h</i>	0,515 <i>h</i>
Circle radius	<i>b</i> /2	<i>b</i> /2	<i>b</i> /2	<i>b</i> /2

package, by means of the `memoir` built-in facilities. But it can be easily seen that as the trimmed page becomes more squarish, the page size ratio tends to one, and the margins become smaller and smaller. Therefore it seems that this layout is most suited for intermediate page ratios, such as the ISO one or the common European book size of 170 mm × 240 mm with a ratio of 0,708.

### 3 Usage

This package is invoked as usual with:

```
\usepackage{canoniclayout}
```

No options are necessary; but it is necessary to invoke the package only after the page dimensions are known, be they specified by an option passed to the class, such as, for example, `a4paper`, or by means of a `pagesize` specification, for example `pagebv` for calling the ISO B5 size.

At any convenient moment after calling this package, the trimming margins with respect to the stock paper may be computed and the layout rechecked and fixed for consistency.

### 4 Font normal size

At the same time if this page layout is chosen, the font normal size should be selected on the actual text block width; it is possible that the golden ratio type block, on a golden rectangle paper obtained by trimming an ISO A4 stock paper, might be suitable with a 10pt normal font size (exactly 66 characters per line with 10pt Latin Modern fonts), while a full ISO A4 paper yields a type block that might be more suited for a 12pt normal font size (86 characters with 10pt Latin Modern fonts). The computations might be left to the `memoir` class commands, but it's better the user is aware of this particular point. For his/her sake this package provides the command `\currentfontletters` (to be used only after `\begin{document}`, because specific font packages might be loaded after the page layout has been computed; in any case the command is “text producing”, therefore it can't be used while in the preamble!) that types out a sentence such as “The number of characters in one text line with the current font is *nn*”, where the *nn* is the actual number printed out. If such number is much larger than the ideal Bringhurst range of  $66 \pm 5$ , it's wiser to choose a larger normal font size. In any case the alternative command `\charactersperpage` types out the same information, the number of lines and the estimated number of characters per page. These commands are supposed to be useful while setting up the complete preliminary parameters for the page layout.

Just to have a comparison, table 2 displays the number of characters in one line of text for several fonts and for page sizes A4 and A5.

It can be seen that the bold measures satisfy the Bringhurst rule with a number of characters in the range  $66 \pm 5$ ; the measures typeset in medium series type

Table 2: Text line lengths measured in average number of characters in different fonts

Font collection	A4 paper size			A5 paper size			B5 paper size		
	10 pt	11 pt	12 pt	10 pt	11 pt	12 pt	10 pt	11 pt	12 pt
Times eXtended	<i>92</i>	<i>84</i>	<i>77</i>	<b>64</b>	59	<i>54</i>	<i>77</i>	<b>70</b>	<b>64</b>
Latin Modern	<i>86</i>	<i>79</i>	<i>73</i>	60	<i>55</i>	<i>51</i>	72	<b>66</b>	<b>61</b>
Kepler Project	<i>83</i>	76	<b>69</b>	58	<i>53</i>	<i>48</i>	<b>69</b>	<b>63</b>	58
Palatino eXtended	<i>82</i>	75	<b>69</b>	58	<i>53</i>	<i>48</i>	<b>69</b>	<b>63</b>	57

are out of the optimal Bringhurst range, but may be considered acceptable; the measures typeset in italics are completely out of the interval  $66 \pm 10$  and should not be considered acceptable. This means that the large A4 paper size may be best used with intrinsically large fonts such as the Palatino eXtended (package `pxfonts`) and the Kepler Project (package `kpfonts`). With the smaller A5 paper size with 10 pt size fonts the best ones are the Times eXtended (package `txfonts`), but substantially all the examined fonts are acceptable, while with 12 pt fonts none are acceptable.

With an intermediate sized paper, such as a B5 ISO size, all the considered fonts are acceptable inside or just outside the optimal Bringhurst interval.

Many books are trimmed to a size close the B5 paper size, therefore the page layout produced with this package is most useful with medium and smaller sizes (at least with ISO sizes).

## 5 Conclusion

I don't think this package should set also the header and footer distances, for example, or should set any specification for the page "decorations", such as underlined headers, or overlined footers, or the size, font family, series and shape of the header and footer contents. These specifications are of course important, but have nothing to do with the relationship between the paper size, the page size and the type block size.

As a final remark it's worth noting that a single pass of `pdflatex` on this `canoniclayout.dtx` file produces at the same time both the `canoniclayout.sty` package file and the documentation `canoniclayout.pdf` file. After running `pdflatex` move `canoniclayout.sty` to a `tex/latex/canoniclayout/` folder and `canoniclayout.pdf` to a `doc/latex/canoniclayout/` one; if such folders don't exist, create them on your personal T<sub>E</sub>X tree. After moving these files remember to refresh the file name data base.

## 6 Documented code

The package preliminaries are already in place. We are going to make some computations in order to determine the page shape factor  $x$ ; to this end we exploit the dimensional computing capabilities of  $\varepsilon$ -TeX, that are already embedded in the most recent implementations of the program *pdftex*; in fact it must be remarked that these facilities are being embedded into the *pdftex* typesetting engine since the TeX Live distribution of the year 2005. If these facilities are not available, either because the TeX distribution is a little too old, or because the real *tex* typesetting engine, not *pdftex*, is being used, then, this entire package contents is skipped and nothing is done, except issuing a suitable warning.

```
1 \*package)
2 \expandafter\ifx\cename eTeXversion\endcsname\relax
3 \PackageWarning{canoniclayout}{%
4 The typesetting engine is not suited\Messagebreak
5 for working in extended mode.\Messagebreak}
6 This package is being skipped}
7 \expandafter\endinput
8 \fi
9 \ifnum\eTeXversion>\z@\else
10 \PackageWarning{canoniclayout}{%
11 The typesetting engine is not working in extended mode.\MessageBreak
12 This package is being skipped}
13 \expandafter\endinput
14 \fi
```

Now we can determine the page shape factor and we save it into the control sequence `\CLshape`. The trick of multiplying by `p@`, a length of 1 pt, is a simple trick for reducing the shape ratio to a length whose numerical value is exactly the ratio we are interested in. By stripping the ‘pt’ part from this dimension value we remain with the real number we want to keep in memory. Of course we need also the reciprocal value so that we have to perform this calculation two times with different factors. This “strange” way of computing real ratios derives from the ability of the  $\varepsilon$ -TeX engine to scale lengths with a higher computational precision than “normal” TeX can do with the standard dimension and integer operations. In fact all the computational operations are performed in the CPU registers and the intermediate first result of the multiplication is kept into a double length register, so that no bits are rounded or truncated; the division operation is performed then with a double length numerator and a single length denominator, so that, again, almost no information is lost at the inevitable end of the division algorithm with a non vanishing remainder.

```
15 \dimen@=\dimexpr \p@ * \paperwidth / \paperheight\relax
16 \edef\CLshape{\strip@pt\dimen@}
17 \dimen@=\dimexpr \p@ * \p@ / \dimen@\relax
18 \edef\CLinvshape{\strip@pt\dimen@}
```

We then assign the text height, thanks to the fact that the canonical layout has the block height just long as the page width, due to the circle property mentioned in

the Introduction; in any case the formulas 1–10 and table 1 confirm this property for any shape.

```
19 \textheight=\paperwidth
```

This done, we can use the internal memoir commands to determine the other layout dimensions.

```
20 \settypeblocksize{\textheight}{*}{\CLshape}
```

```
21 \setlrmargins{*}{*}{\CLinvshape}
```

```
22 \setulmargins{*}{*}{\CLinvshape}
```

```
23 \setmarginnotes{7pt}{\dimexpr \foremargin-3\marginparsep}{\onelineskip}
```

```
24
```

We now use the `\checkandfixthelayout` so that it prints the page parameters to the `.log` file.

```
25 \checkandfixthelayout
```

Just for testing the number of character in a line the following macros are provided. A couple of scratch dimension registers will be used `dimen@` and `\abcwidth`; `\abcwidth` will hold the current lowercase alphabet length, and `\dimen@` will hold several values in succession. The scratch counter `\count254` will receive the final result to be printed out.  $\varepsilon$ -TeX arithmetics will be used throughout, but we know this is possible, since we already tested it at the very beginning.

Another service macro for the user’s sake is `\charactersperpage`; this macro typesets the number of characters per line, the numbers of lines per page, and eventually their product, a good estimate of the number of characters per page. Many test editors have facilities for counting a “compuscript” number o words and characters; with this further information the user may easily estimate the number of pages of his/her document; of course s/he must make a certain allowance for the front and back matter pages, so that s/he can evaluate the total amount of signatures of the whole typeset document; of course, by typesetting it with *(pdf)latex* s/he knows exactly the number of pages of the final document, but the typographer wants to know a good estimate before the source TeX files are finished.

```
26 \newlength\abc@width
```

```
27 \newcommand*\currentfontletters{%
```

```
28 \settowidth\abc@width{abcdefghijklmnopqrstuvwxyz}\dimen0=26\p@
```

```
29 \dimen0=\dimexpr \textwidth * \dimen0 / \abc@width \relax
```

```
30 \count254=\numexpr \dimen0 / \p@ \relax
```

```
31 \par The number of characters in one text line with the current font is
```

```
32   \the\count254.\par}
```

```
33
```

```
34 \newcommand*\charactersperpage{%
```

```
35 \settowidth\abc@width{abcdefghijklmnopqrstuvwxyz}\dimen0=26\p@
```

```
36 \dimen0=\dimexpr \textwidth * \dimen0 / \abc@width \relax
```

```
37 \count254=\numexpr \dimen0 / \p@ \relax
```

```
38 \count252=\numexpr \textheight / \onelineskip \relax
```

```
39 \count250=\numexpr \count252 * \count254\relax
```

```
40 \quote The number of characters in one text line with the current
```

```
41   font is \the\count254;
```

```
42   this text block contains \the\count252\ lines;
43   a rough estimate of the number of characters per page is
44   \the\count250.\par
45   This can be used to evaluate the total number of pages of
46   a document.
47 \endquote}
48
49 </package>
```