

The geometry package

Hideo Umeki

翻译 virhuiai@qq.com

<https://github.com/davidcarlisle/geometry>

<https://github.com/virhuiai/Latex-Typesetting-Hub/tree/main/> / geometry

2020/03/24 1.8d

1 Introduction

引言

To set dimensions for page layout in \LaTeX is not straightforward. You need to adjust several \LaTeX native dimensions to place a text area where you want. If you want to center the text area in the paper you use, for example, you have to specify native dimensions as follows:

```
\usepackage{calc}
\setlength\textwidth{7in}
\setlength\textheight{10in}
\setlength\oddsidemargin{(\paperwidth-\textwidth)/2 - 1in}
\setlength\topmargin{(\paperheight-\textheight-\headheight-\headsep-\footskip)/2 - 1in}
```

Without package *calc*, the above example would need more tedious settings. Package **geometry** provides an easy way to set page layout parameters. In this case, what you have to do is just

```
\usepackage[text={7in,10in},centering]{geometry}.
```

Besides centering problem, setting margins from each edge of the paper is also troublesome. But **geometry** also make it easy. If you want to set each margin to 1.5in, you can type

```
\usepackage[margin=1.5in]{geometry}
```

Thus, the **geometry** package has an auto-completion mechanism, in which unspecified dimensions are automatically determined. The **geometry** package will be also useful when you have to set page layout obeying the following strict instructions: for example,

The total allowable width of the text area is 6.5 inches wide by 8.75 inches high. The top margin on each page should be 1.2 inches from the top edge of the page. The left margin should be 0.9 inch from the left edge. The footer with page number should be at the bottom of the text area.

In this case, using **geometry** you can type

```
\usepackage[total={6.5in,8.75in},
top=1.2in, left=0.9in, includefoot]{geometry}.
```

Setting a text area on the paper in document preparation system has some analogy to placing a window on the background in the window system. The name ‘geometry’ comes from the **-geometry** option used for specifying a size and location of a window in X Window System.

在 \LaTeX 中设置页面布局并不简单。你需要调整几个 \LaTeX 的内部尺寸，以将文本区域放置在你想要的位置。例如，如果你想要在纸张上居中显示文本区域，你需要指定以下内部尺寸：

如果没有使用 *calc* 宏包，上面的示例将需要更繁琐的设置。宏包 **geometry** 提供了一种简便的方式来设置页面布局参数。在这种情况下，您只需要做以下操作：

```
\usepackage[text={7in,10in},centering]{geometry}
```

除了居中问题，设置页面边距也会变得麻烦。但是 **geometry** 也可以轻松解决这个问题。如果您想将每个边缘的边距设置为 1.5 英寸，可以输入：

```
\usepackage[margin=1.5in]{geometry}
```

因此，**geometry** 宏包具有自动完成机制，未指定的尺寸将自动确定。当您需要按照严格的说明设置页面布局时，**geometry** 宏包也非常有用。例如：

文本区域的总可允许宽度为 6.5 英寸，高度为 8.75 英寸。每页的顶部边距应距离页面顶部边缘 1.2 英寸。左边距应距离左边缘 0.9 英寸。页脚与页码应位于文本区域底部。

在这种情况下，使用 **geometry** 宏包，您可以输入：

```
\usepackage[total={6.5in,8.75in},
top=1.2in, left=0.9in, includefoot]{geometry}
```

在文档准备系统中，设置纸张上的文本区域类似于在窗口系统中将窗口放置在背景上。“geometry” 这个名称源于在 X Window System 中用于指定窗口大小和位置的 **-geometry** 选项。

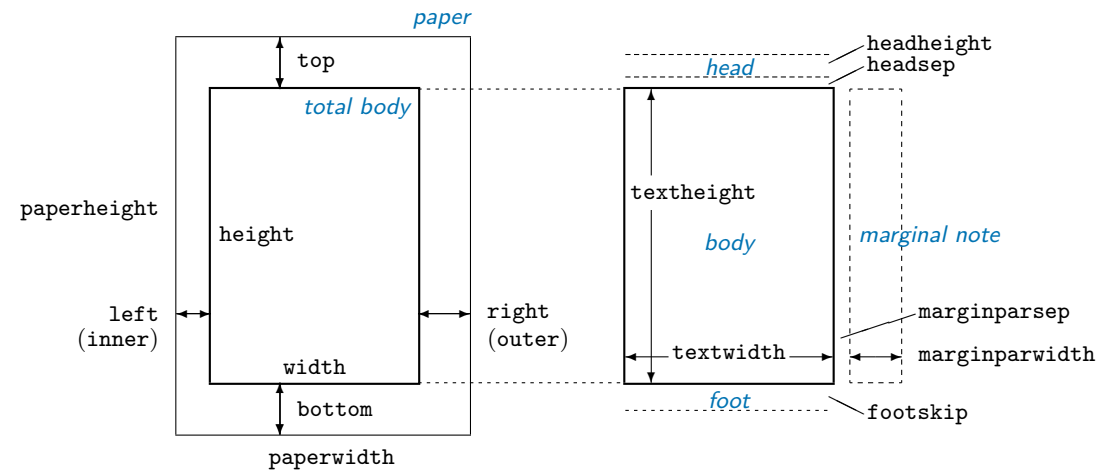


图 1: Dimension names used in the `geometry` package. `width = textwidth` and `height = textheight` by default. `left`, `right`, `top` and `bottom` are margins. If margins on verso pages are swapped by `twoside` option, margins specified by `left` and `right` options are used for the inside and outside margins respectively. `inner` and `outer` are aliases of `left` and `right` respectively.
`geometry` 宏包中使用的尺寸名称如下：默认情况下，`width = textwidth` 和 `height = textheight`。 `left`、`right`、`top` 和 `bottom` 是边距。如果通过 `twoside` 选项交换了底稿页的边距，那么通过 `left` 和 `right` 选项指定的边距将分别用于内边距和外边距。`inner` 和 `outer` 分别是 `left` 和 `right` 的别名。

2 Page geometry

页面布局

Figure 1 shows the page layout dimensions defined in the `geometry` package. The page layout contains a *total body* (printable area) and *margins*. The *total body* consists of a *body* (text area) with an optional *header*, *footer* and marginal notes (*marginpar*). There are four margins: *left*, *right*, *top* and *bottom*. For twosided documents, horizontal margins should be called *inner* and *outer*.

图 1展示了 `geometry` 宏包中定义的页面布局尺寸。页面布局包括一个**总体正文**（可打印区域）和**页边距**。**总体正文**包括一个**正文**（文本区域），还可以有可选的**页眉**、**页脚**和**边注**（`marginpar`）。有四个边距：**左边距**、**右边距**、**顶边距**和**底边距**。对于双面文档，水平的边距应称为**内边距**和**外边距**。

paper : *total body* and *margins*
total body : *body* (text area) (optional *head*, *foot* and *marginpar*)
margins : *left (inner)*, *right (outer)*, *top* and *bottom*

纸张 : 总体正文和页边距
 总体正文 : 正文（文本区域）（可选的页眉、页脚和边注）
 页边距 : 左边距（内边距）、右边距（外边距）、顶边距和底边距

Each margin is measured from the corresponding edge of a paper. For example, left margin (inner margin) means a horizontal distance between the left (inner) edge of the paper and that of the total body. Therefore the left and top margins defined in `geometry` are different from the native dimensions `\leftmargin` and `\topmargin`. The size of a body (text area) can be modified by `\textwidth` and `\textheight`. The dimensions for paper, total body and margins have the following relations.

每个边距都是从纸张的相应边缘测量的。例如，左边距（内边距）表示纸张的左（内）边缘与总体正文的左（内）边缘之间的水平距离。因此，在 `geometry` 中定义的左边距和顶边距与原生尺寸 `\leftmargin` 和 `\topmargin` 是不同的。正文（文本区域）的大小可以通过 `\textwidth` 和 `\textheight` 进行修改。纸张、总体正文和边距的尺寸具有以下关系。

$$\text{paperwidth} = \text{left} + \text{width} + \text{right} \quad (1) \quad \text{纸张宽度} = \text{左边距} + \text{正文宽度} + \text{右边距} \quad (1)$$

$$\text{paperheight} = \text{top} + \text{height} + \text{bottom} \quad (2) \quad \text{纸张高度} = \text{顶边距} + \text{正文高度} + \text{底边距} \quad (2)$$

The total body `width` and `height` would be defined:

$$\text{width} := \text{textwidth} (+ \text{marginparsep} + \text{marginparwidth}) \quad (3) \quad \text{宽度} := \text{正文宽度} (+ \text{边注间距} + \text{边注宽度}) \quad (3)$$

$$\text{height} := \text{textheight} (+ \text{headheight} + \text{headsep} + \text{footskip}) \quad (4) \quad \text{高度} := \text{正文高度} (+ \text{页眉高度} + \text{页眉与正文的距离} + \text{页脚与正文的距离}) \quad (4)$$

In Equation (3) `width:=textwidth` by default, while `marginparsep` and `marginparwidth` are included in `width` if `includemp` option is set `true`. In Equation (4), `height:=textheight` by default. If `includehead` is set to `true`, `headheight` and `headsep` are considered as a part of `height`. In the same way, `includefoot` takes `footskip` into `height`. Figure 2 shows how these options work in the vertical direction.

在方程 (3) 中，默认情况下，`width:=textwidth`，而如果将 `includemp` 选项设置为 `true`，则 `marginparsep` 和 `marginparwidth` 将包括在 `width` 中。在方程 (4) 中，默认情况下，`height:=textheight`。如果将 `includehead` 设置为 `true`，则将考虑 `headheight` 和 `headsep` 作为 `height` 的一部分。同样，`includefoot` 将 `footskip` 包括在 `height` 中。图 2展示了这些选项在垂直方向上的工作方式。

Thus, the page layout consists of three parts (lengths) in each direction: one body and two margins. If the two of them are explicitly specified, the other length is obvious and no need to be specified. Figure 3 shows a simple model of page dimensions.

因此，页面布局在每个方向上由三个部分（长度）组成：一个正文和两个边距。如果其中两个长度被明确指定，另一个长度是显而易见的，无需指定。图 3展示了页面尺寸的简单模型。当给定一个长度 L，并

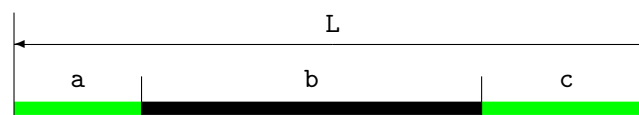


图 3: A simple model of page dimensions.

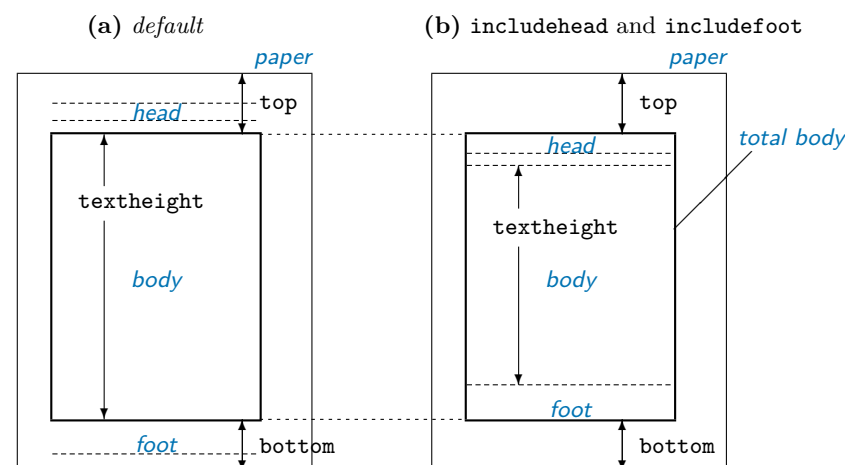


图 2: `includehead` and `includefoot` include the head and foot respectively into *total body*. (a) $\text{height} = \text{textheight}$ (default). (b) $\text{height} = \text{textheight} + \text{headheight} + \text{headsep} + \text{footskip}$ if `includehead` and `includefoot`. If the top and bottom margins are specified, `includehead` and `includefoot` result in shorter `textheight`.
`includehead` 和 `includefoot` 分别将页眉和页脚包括在总体正文中。(a) $\text{height} = \text{textheight}$ (默认情况下)。(b) 如果设置了 `includehead` 和 `includefoot`, 则 $\text{height} = \text{textheight} + \text{headheight} + \text{headsep} + \text{footskip}$ 。如果指定了顶部和底部边距, 则 `includehead` 和 `includefoot` 会导致较短的 `textheight`。

When a length L is given and is partitioned into the body b , the margins a and c , it's obvious that

$$L = a + b + c \quad (5)$$

将其分成正文 b 、边距 a 和 c 时, 很显然有以下关系:

The specification with two of the three (a, b and c) fixed explicitly is solvable. If two or more are left unspecified or ‘under-specified’, Equation (5) cannot be solved without any other relation between them. If all of them are specified, then it needs to check whether or not they satisfy Equation (5), that is too much specification or ‘overspecified’.

The `geometry` package has auto-completion mechanism that saves the trouble of specifying the page layout dimensions. For example, you can set

```
\usepackage[width=14cm, left=3cm]{geometry}
```

on A4 paper. In this case you don't have to set the right margin The details of auto-completion will be described in Section 5.5.

如果明确指定了三个中的两个 (a, b 和 c), 则可以解决规范问题。如果有两个或更多未指定或“不完全规定”, 则在没有其他关系的情况下, 方程 (5) 无法解决。如果全部都被指定, 那么就需要检查它们是否满足方程 (5), 即是否过度规定或“超规定”。

`geometry` 宏包具有自动补全机制, 可以省去指定页面布局尺寸的麻烦。例如, 您可以设置如下:

```
\usepackage[width=14cm, left=3cm]{geometry}
```

在 A4 纸上。在这种情况下, 您不需要设置右边距。自动补全的详细信息将在第 5.5 节中描述。

3 User interface

用户接口

3.1 Commands

命令

The `geometry` package provides the following commands:

- `\geometry{<options>}`
- `\newgeometry{<options>}` and `\restoregeometry`
- `\savegeometry{<name>}` and `\loadgeometry{<name>}`

`geometry` 宏包提供以下命令:

- `\geometry{<options>}`
- `\newgeometry{<options>}` 和 `\restoregeometry`
- `\savegeometry{<name>}` 和 `\loadgeometry{<name>}`

`\geometry{⟨options⟩}` changes the page layout according to the options specified in the argument. This command, if any, should be placed only in the preamble (before `\begin{document}`).

The `geometry` package may be used as part of a class or another package you use in your document. The command `\geometry` can overwrite some of the settings in the preamble. Multiple use of `\geometry` is allowed and then processed with the options concatenated. If `geometry` is not yet loaded, you can use only `\usepackage[⟨options⟩]{geometry}` instead of `\geometry`.

`\newgeometry{⟨options⟩}` changes the page layout mid-document. `\newgeometry` is almost similar to `\geometry` except that `\newgeometry` disables all the options specified by `\usepackage` and `\geometry` in the preamble and skips papersize-related options. `\restoregeometry` restores the page layout specified in the preamble. This command has no arguments. See Section 6 for details.

`\savegeometry{⟨name⟩}` saves the page dimensions as `⟨name⟩` where you put this command. `\loadgeometry{⟨name⟩}` loads the page dimensions saved as `⟨name⟩`. See Section 6 for details.

3.2 Optional argument

The `geometry` package adopts keyval interface ‘`⟨key⟩=⟨value⟩`’ for the optional argument to `\usepackage`, `\geometry` and `\newgeometry`.

The argument includes a list of comma-separated keyval options and has basic rules as follows:

- Multiple lines are allowed, while blank lines are not.
- Any spaces between words are ignored.
- Options are basically order-independent. (There are some exceptions. See Section 5.2 for details.)

For example,

```
\usepackage[a5paper,hmargin={3cm,.8in},height=10in]{geometry}
```

is equivalent to

```
\usepackage[height=10in,a5paper,hmargin={3cm,0.8in}]{geometry}
```

Some options are allowed to have sub-list, e.g. `{3cm,0.8in}`. Note that the order of values in the sub-list is significant. The above setting is also equivalent to the followings:

```
\usepackage{geometry}
\geometry{height=10in,a5paper,hmargin={3cm,0.8in}}
```

or

```
\usepackage[a5paper]{geometry}
\geometry{hmargin={3cm,0.8in},height=8in}
\geometry{height=10in}.
```

Thus, multiple use of `\geometry` just appends options. `geometry` supports package `calc`¹. For example,

```
\usepackage{calc}
\usepackage[textheight=20\baselineskip+10pt]{geometry}
```

`\geometry{⟨options⟩}` 根据参数中指定的选项更改页面布局。如果有的话，此命令应该只放在导言区（`\begin{document}` 之前）。

`geometry` 宏包可以作为文档中使用的类或其他宏包的一部分。命令 `\geometry` 可以覆盖导言区中的一些设置。可以多次使用 `\geometry` 命令，并将选项连接在一起进行处理。如果 `geometry` 尚未加载，可以使用 `\usepackage[⟨options⟩]{geometry}` 代替 `\geometry`。

`\newgeometry{⟨options⟩}` 可以在文档中更改页面布局。`\newgeometry` 与 `\geometry` 几乎相同，除了会禁用导言区中由 `\usepackage` 和 `\geometry` 指定的所有选项，并且跳过与页面尺寸相关的选项外。`\restoregeometry` 命令将恢复导言区中指定的页面布局。此命令没有参数。详见第 6 节。

`\savegeometry{⟨name⟩}` 将页面尺寸保存为 `⟨name⟩`。`\loadgeometry{⟨name⟩}` 加载保存为 `⟨name⟩` 的页面尺寸。详见第 6 节。

可选参数

`geometry` 宏包在 `\usepackage`、`\geometry` 和 `\newgeometry` 的可选参数中采用了 keyval 接口的 ‘`⟨key⟩=⟨value⟩`’ 形式。

参数包括一个由逗号分隔的 keyval 选项列表，并有以下基本规则：

- 允许有多行，但不允许有空行。
- 单词之间的空格会被忽略。
- 选项基本上是无序的。（有一些例外情况，请参见第 5.2 节了解详细信息。）

例如，

```
\usepackage[a5paper,hmargin={3cm,.8in},height=10in]{geometry}
```

等同于

```
\usepackage[height=10in,a5paper,hmargin={3cm,0.8in}]{geometry}
```

某些选项允许有子列表，例如 `{3cm,0.8in}`。请注意，子列表中的值的顺序是重要的。上述设置也等同于以下设置：

```
\usepackage{geometry}
\geometry{height=10in,a5paper,hmargin={3cm,0.8in}}
```

或者

```
\usepackage[a5paper]{geometry}
\geometry{hmargin={3cm,0.8in},height=8in}
\geometry{height=10in}.
```

因此，多次使用 `\geometry` 只会追加选项。`geometry` 支持 `calc`¹ 宏包。例如，

```
\usepackage{calc}
\usepackage[textheight=20\baselineskip+10pt]{geometry}
```

¹CTAN: macros/latex/required/tools

¹CTAN: macros/latex/required/tools

3.3 Option types

geometry options are categorized into four types:

1. **Boolean type** takes a boolean value (`true` or `false`). If no value, `true` is set by default.

$\langle key \rangle = \text{true} \mid \text{false}.$
 $\langle key \rangle$ with no value is equivalent to $\langle key \rangle = \text{true}.$

Examples: `verbose=true, includehead, twoside=false.`

Paper name is the exception. The preferred paper name should be set with no values. Whatever value is given, it is ignored. For instance, `a4paper=XXX` is equivalent to `a4paper`.

2. **Single-valued type** takes a mandatory value.

$\langle key \rangle = \langle value \rangle.$

Examples: `width=7in, left=1.25in, footskip=1cm, height=.86\paperheight.`

3. **Double-valued type**

takes a pair of comma-separated values in braces. The two values can be shortened to one value if they are identical.

$\langle key \rangle = \{ \langle value1 \rangle, \langle value2 \rangle \}.$
 $\langle key \rangle = \langle value \rangle$ is equivalent to $\langle key \rangle = \{ \langle value \rangle, \langle value \rangle \}$

Examples: `hmargin={1.5in,1in}, scale=0.8, body={7in,10in}.`

4. **Triple-valued type**

takes three mandatory, comma-separated values in braces.

$\langle key \rangle = \{ \langle value1 \rangle, \langle value2 \rangle, \langle value3 \rangle \}$

Each value must be a dimension or null. When you give an empty value or ‘*’, it means null and leaves the appropriate value to the auto-completion mechanism. You need to specify at least one dimension, typically two dimensions. You can set nulls for all the values, but it makes no sense. *Examples:*

`hdivide={2cm,*,1cm}, vdivide={3cm,19cm, }, divide={1in,*,1in}.`

4 Option details

This section describes all options available in `geometry`. Options with a dagger [†] are not available as arguments of `\newgeometry` (See Section 6).

4.1 Paper size

The options below set paper/media size and orientation.

[†] `paper` | `papername`

specifies the paper size by name. `paper=⟨paper-name⟩`. For convenience, you can specify the paper name without `paper=`. For example, `a4paper` is equivalent to `paper=a4paper`.

3.3 选项类型

geometry 的选项被分为四种类型：

1. 布尔类型可以取布尔值（`true` 或 `false`）。如果没有给出值，则默认为 `true`。

$\langle key \rangle = \text{true} \mid \text{false}.$
 $\langle key \rangle$ 没有给出值, 等同于 $\langle key \rangle = \text{true}.$

示例： `verbose=true, includehead, twoside=false.`

纸张名称是一个例外。应该使用没有值来设置首选纸张名称。无论给出什么值，都会被忽略。例如，`a4paper=XXX` 等同于 `a4paper`。

2. 单值类型必须给出一个值。

$\langle key \rangle = \langle value \rangle.$

示例： `width=7in, left=1.25in, footskip=1cm, height=.86\paperheight.`

3. 双值类型

需要用括号括起来的逗号分隔的一对值。如果两个值相同，可以缩写为一个值

$\langle key \rangle = \{ \langle value1 \rangle, \langle value2 \rangle \}.$
 $\langle key \rangle = \langle value \rangle$ 等同于 $\langle key \rangle = \{ \langle value \rangle, \langle value \rangle \}$

示例： `hmargin={1.5in,1in}, scale=0.8, body={7in,10in}.`

4. 三值类型

需要给出三个必填的逗号分隔值。

$\langle key \rangle = \{ \langle value1 \rangle, \langle value2 \rangle, \langle value3 \rangle \}$

每个值必须是一个长度或空值。当给出空值或 ‘*’ 时，表示为空值，并将相应的值留给自动补全机制。您需要至少指定一个长度，通常是两个长度。您可以为所有值设置为空值，但这没有意义。

示例：

`hdivide={2cm,*,1cm}, vdivide={3cm,19cm, }, divide={1in,*,1in}.`

选项详细信息

本节描述了 `geometry` 中所有可用的选项。带有 † 标记的选项不能作为 `\newgeometry` 的参数使用（参见第 6 节）。

4.1 纸张大小

以下选项设置纸张/媒体大小和方向。

[†] `paper` | `papername`

通过名称指定纸张大小。`paper=⟨纸张名称⟩`。为了方便起见，您可以省略 `paper=` 直接指定纸张名称。例如，`a4paper` 等同于 `paper=a4paper`。

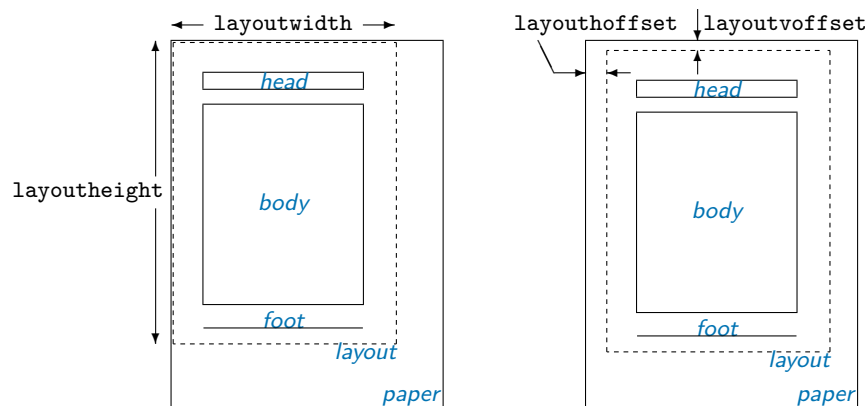


图 4: The dimensions related to the layout size. Note that the layout size defaults to the same size as the paper, so you don't have to specify layout-related options explicitly in most cases.

[†] `a0paper`, `a1paper`, `a2paper`, `a3paper`, `a4paper`, `a5paper`, `a6paper`,
`b0paper`, `b1paper`, `b2paper`, `b3paper`, `b4paper`, `b5paper`, `b6paper`,
`c0paper`, `c1paper`, `c2paper`, `c3paper`, `c4paper`, `c5paper`, `c6paper`,
`b0j`, `b1j`, `b2j`, `b3j`, `b4j`, `b5j`, `b6j`,
`ansipaper`, `ansibpaper`, `ansicpaper`, `ansidpaper`, `ansiepaper`,
`letterpaper`, `executivepaper`, `legalpaper`

specifies paper name. The value part is ignored even if any. For example, the followings have the same effect:
`a5paper`, `a5paper=true`, `a5paper=false` and so forth. `a[0-6]paper`, `b[0-6]paper` and `c[0-6]paper` are
ISO A, B and C series of paper sizes respectively. The JIS (Japanese Industrial Standards) A-series is
identical to the ISO A-series, but the JIS B-series is different from the ISO B-series. `b[0-6]j` should be used
for the JIS B-series.

[†] `screen` a special paper size with (W,H) = (225mm,180mm). For presentation with PC and video projector,
“`screen,centering`” with ‘slide’ documentclass would be useful.

[†] `paperwidth` width of the paper. `paperwidth=<length>`.

[†] `paperheight` height of the paper. `paperheight=<length>`.

[†] `papersize` width and height of the paper. `papersize={<width>,<height>}` or `papersize=<length>`.

[†] `landscape` switches the paper orientation to landscape mode.

[†] `portrait` switches the paper orientation to portrait mode. This is equivalent to `landscape=false`.

The options for paper names (e.g., `a4paper`) and orientation (`portrait` and `landscape`) can be set as document class options.
For example, you can set `\documentclass[a4paper,landscape]{article}`, then `a4paper` and `landscape` are processed in
geometry as well. This is also the case for `twoside` and `twocolumn` (see also Section 4.5).

4.2 Layout size

You can specify the layout area with options described in this section regardless of the paper size. The options would help to
print the specified layout to a different sized paper. For example, with `a4paper` and `layout=a5paper`, the package uses ‘A5’
layout to calculate margins on ‘A4’ paper. The layout size defaults to the same as the paper. The options for the layout size
are available in `\newgeometry`, so that you can change the layout size in the middle of the document. The paper size itself
can't be changed though. Figure 4 shows what the difference between `layout` and `paper` is.

`layout` specifies the layout size by paper name. `layout=<paper-name>`. All the paper names defined in `geometry` are
available. See Section 4.1 for details.

`layoutwidth` width of the layout. `layoutwidth=<length>`.

`layoutheight` height of the layout. `layoutheight=<length>`.

`layoutsiz` width and height of the layout. `layoutsiz={<width>,<height>}` or `layoutsiz=<length>`.

[†] `a0paper`, `a1paper`, `a2paper`, `a3paper`, `a4paper`, `a5paper`, `a6paper`,
`b0paper`, `b1paper`, `b2paper`, `b3paper`, `b4paper`, `b5paper`, `b6paper`,
`c0paper`, `c1paper`, `c2paper`, `c3paper`, `c4paper`, `c5paper`, `c6paper`,
`b0j`, `b1j`, `b2j`, `b3j`, `b4j`, `b5j`, `b6j`,
`ansipaper`, `ansibpaper`, `ansicpaper`, `ansidpaper`, `ansiepaper`,
`letterpaper`, `executivepaper`, `legalpaper`

指定纸张名称。即使有值部分，也会被忽略。例如，下面的选项具有相同的效果：

`a5paper`, `a5paper=true`, `a5paper=false` 等等。`a[0-6]paper`, `b[0-6]paper` 和
`c[0-6]paper` 分别是 ISO A、B 和 C 系列纸张大小。JIS（日本工业标准）A 系列与
ISO A 系列相同，但 JIS B 系列与 ISO B 系列不同。应该使用 `b[0-6]j` 来表示 JIS B
系列。

[†] `screen` 一种特殊的纸张尺寸，宽度 (W) 为 225mm，高度 (H) 为 180mm。对于使用个人电脑
和视频投影仪进行演示，使用带有 ‘slide’ 文档类的 “`screen,centering`” 会很有用。

[†] `paperwidth` 纸张的宽度。 `paperwidth=<长度>`。

[†] `paperheight` 纸张的高度。 `paperheight=<长度>`。

[†] `papersize` 纸张的宽度和高度。 `papersize={<宽度>,<高度>}` 或 `papersize=<长度>`。

[†] `landscape` 将纸张方向切换为横向模式。

[†] `portrait` 将纸张方向切换为纵向模式。这相当于 `landscape=false`。

纸张名称选项（例如，`a4paper`）和方向选项（`portrait` 和 `landscape`）可以作为文档类选项进行设
置。例如，您可以设置 `\documentclass[a4paper,landscape]{article}`，那么 `a4paper` 和 `landscape`
也会在 `geometry` 中进行处理。对于 `twoside` 和 `twocolumn` 也是如此（详见第 4.5 节）。

4.2 布局尺寸

无论纸张大小如何，您都可以使用本节中描述的选项来指定布局区域。这些选项可以帮助您将指定的布
局打印到不同大小的纸张上。例如，使用 `a4paper` 和 `layout=a5paper`，该宏包会在 “A4” 纸上使用
“A5” 布局来计算边距。布局尺寸默认与纸张相同。布局尺寸的选项在 `\newgeometry` 中也可用，因此
您可以在文档的中间更改布局尺寸。但是纸张尺寸本身无法更改。图 4 展示了 `layout` 和 `paper` 之间
的区别。

`layout` 按纸张名称指定布局尺寸。 `layout=<纸张名称>`。所有在 `geometry` 中定义的纸张名称
都可用。详情请参阅第 4.1 节。

`layoutwidth` 布局的宽度。 `layoutwidth=<长度>`。

`layoutheight` 布局的高度。 `layoutheight=<长度>`。

`layoutsiz` 布局的宽度和高度。 `layoutsiz={<宽度>,<高度>}` 或 `layoutsiz=<长度>`。

hscale	ratio of width of <i>total body</i> to \paperwidth. hscale= $\langle h-scale \rangle$, e.g., hscale=0.8 is equivalent to width=0.8\paperwidth. (0.7 by default)
vscale	ratio of height of <i>total body</i> to \paperheight, e.g., vscale= $\langle v-scale \rangle$. (0.7 by default) vscale=0.9 is equivalent to height=0.9\paperheight.
scale	ratio of <i>total body</i> to the paper. scale={ $\langle h-scale \rangle, \langle v-scale \rangle$ } or scale= $\langle scale \rangle$. (0.7 by default)
width totalwidth	width of <i>total body</i> . width= $\langle length \rangle$ or totalwidth= $\langle length \rangle$. This dimension defaults to textwidth, but if includemp is set to true, width \geq textwidth because width includes the width of the marginal notes. If textwidth and width are specified at the same time, textwidth takes priority over width.
height totalheight	height of <i>total body</i> , excluding header and footer by default. If includehead or includefoot is set, height includes the head or foot of the page as well as textheight. height= $\langle length \rangle$ or totalheight= $\langle length \rangle$. If both textheight and height are specified, height will be ignored.
total	width and height of <i>total body</i> . total={ $\langle width \rangle, \langle height \rangle$ } or total= $\langle length \rangle$.
textwidth	specifies \textwidth, the width of <i>body</i> (the text area). textwidth= $\langle length \rangle$.
textheight	specifies \textheight, the height of <i>body</i> (the text area). textheight= $\langle length \rangle$.
text body	specifies both \textwidth and \textheight of the body of page. body={ $\langle width \rangle, \langle height \rangle$ } or text= $\langle length \rangle$.
lines	enables users to specify \textheight by the number of lines. lines= $\langle integer \rangle$.
includehead	includes the head of the page, \headheight and \headsep, into <i>total body</i> . It is set to false by default. It is opposite to ignorehead. See Figure 2 and Figure 5.
includefoot	includes the foot of the page, \footskip, into <i>total body</i> . It is opposite to ignorefoot. It is false by default. See Figure 2 and Figure 5.
includeheadfoot	sets both includehead and includefoot to true, which is opposite to ignoreheadfoot. See Figure 2 and Figure 5.
includemp	includes the margin notes, \marginparwidth and \marginparsep, into <i>body</i> when calculating horizontal calculation.
includeall	sets both includeheadfoot and includemp to true. See Figure 5.
ignorehead	disregards the head of the page, headheight and headsep, in determining vertical layout, but does not change those lengths. It is equivalent to includehead=false. It is set to true by default. See also includehead.
ignorefoot	disregards the foot of page, footskip, in determining vertical layout, but does not change that length. This option defaults to true. See also includefoot.
ignoreheadfoot	sets both ignorehead and ignorefoot to true. See also includeheadfoot.
ignoremp	disregards the marginal notes in determining the horizontal margins (defaults to true). If marginal notes overrun the page, the warning message will be displayed when verbose=true. See also includemp and Figure 5.
ignoreall	sets both ignoreheadfoot and ignoremp to true. See also includeall.

hscale	总正文 的宽度与 \paperwidth 的比例。hscale= $\langle h-scale \rangle$ ，例如，hscale=0.8 等同于 width=0.8\paperwidth。（默认为 0.7）
vscale	总正文 的高度与 \paperheight 的比例。例如，vscale= $\langle v-scale \rangle$ 。（默认为 0.7）vscale=0.9 等同于 height=0.9\paperheight。
scale	总正文 与纸张的比例。scale={ $\langle h-scale \rangle, \langle v-scale \rangle$ } 或 scale= $\langle scale \rangle$ 。（默认为 0.7）
width totalwidth	总正文 的宽度。width= $\langle \text{长度} \rangle$ 或 totalwidth= $\langle \text{长度} \rangle$ 。该尺寸默认为 textwidth，但如果将 includemp 设置为 true，则 width \geq textwidth，因为 width 包括边注的宽度。如果同时指定 textwidth 和 width，则 textwidth 优先于 width。
height totalheight	总正文 的高度，默认不包括页眉和页脚。如果设置了 includehead 或 includefoot，则 height 包括页眉或页脚以及 textheight。height= $\langle \text{长度} \rangle$ 或 totalheight= $\langle \text{长度} \rangle$ 。如果同时指定了 textheight 和 height，则会忽略 height。
total	总正文 的宽度和高度。total={ $\langle \text{宽度} \rangle, \langle \text{高度} \rangle$ } 或 total= $\langle \text{长度} \rangle$ 。
textwidth	指定 \textwidth，即正文（文本区域）的宽度。textwidth= $\langle \text{长度} \rangle$ 。
textheight	指定 \textheight，即正文（文本区域）的高度。textheight= $\langle \text{长度} \rangle$ 。
text body	指定正文（文本区域）的 \textwidth 和 \textheight。body={ $\langle \text{宽度} \rangle, \langle \text{高度} \rangle$ } 或 text= $\langle \text{长度} \rangle$ 。
lines	允许用户通过行数指定 \textheight。lines= $\langle \text{整数} \rangle$ 。
includehead	将页眉，\headheight 和 \headsep 包含到 总正文 中。默认为 false。与 ignorehead 相反。参见图2 和图5。
includefoot	将页脚，\footskip，包含到 总正文 中。与 ignorefoot 相反。默认为 false。参见图 2 和图 5。
includeheadfoot	将 includehead 和 includefoot 都设置为 true，与 ignoreheadfoot 相反。参见图2 和图 5。
includemp	当计算水平布局时，将边注，\marginparwidth 和 \marginparsep，包含到 正文 中。
includeall	将 includeheadfoot 和 includemp 都设置为 true。参见图 5。
ignorehead	在确定垂直布局时忽略页眉，即 \headheight 和 \headsep，但不更改这些长度。等同于 includehead=false。默认为 true。参见 includehead。
ignorefoot	在确定垂直布局时忽略页脚，即 \footskip，但不更改该长度。默认为 true。参见 includefoot。
ignoreheadfoot	将 ignorehead 和 ignorefoot 都设置为 true。参见 includeheadfoot。
ignoremp	在确定水平边距时忽略边注（默认为 true）。如果边注超出页面，则在 verbose=true 时将显示警告消息。参见 includemp 和图5。
ignoreall	将 ignoreheadfoot 和 ignoremp 都设置为 true。参见 includeall。

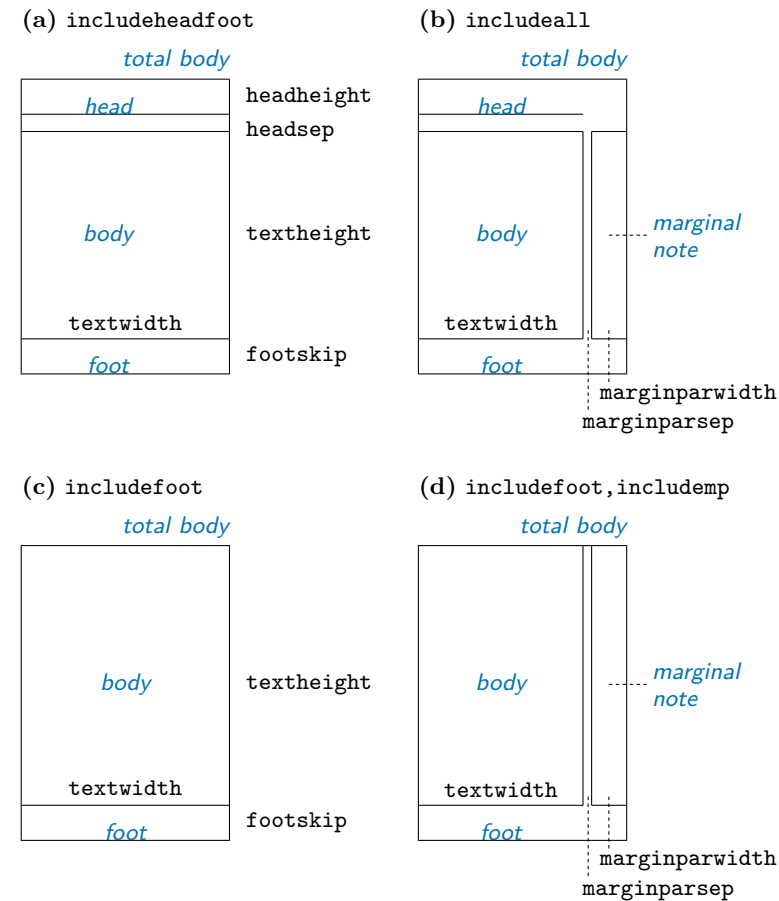


图 5: Sample layouts for *total body* with different switches. (a) `includeheadfoot`, (b) `includeall`, (c) `includefoot` and (d) `includefoot,includemp`. If `reversemp` is set to `true`, the location of the marginal notes are swapped on every page. Option `twoside` swaps both margins and marginal notes on verso pages. Note that the marginal note, if any, is printed despite `ignoremp` or `includemp=false` and overrun the page in some cases.

使用不同开关的**总体正文**的示例布局。(a) `includeheadfoot`, (b) `includeall`, (c) `includefoot` 和 (d) `includefoot,includemp`。如果将 `reversemp` 设置为 `true`, 则边注的位置将在每一页上交换。选项 `twoside` 在背面页面上交换两边的边距和边注。请注意, 如果有边注, 则会在某些情况下打印出来, 尽管设置了 `ignoremp` 或 `includemp=false`, 并可能超出页面。

heightrounded

This option rounds `\textheight` to n -times (n : an integer) of `\baselineskip` plus `\topskip` to avoid “underfull vbox” in some cases. For example, if `\textheight` is 486pt with `\baselineskip` 12pt and `\topskip` 10pt, then

$$(39 \times 12\text{pt} + 10\text{pt} =) 478\text{pt} < 486\text{pt} < 490\text{pt} (= 40 \times 12\text{pt} + 10\text{pt}),$$

as a result `\textheight` is rounded to 490pt. `heightrounded=false` by default.

heightrounded

此选项将 `\textheight` 舍入为 n 倍 (n : 整数) 的 `\baselineskip` 加上 `\topskip`, 以避免在某些情况下出现 “underfull vbox”。例如, 如果 `\textheight` 是 486pt, `\baselineskip` 是 12pt, `\topskip` 是 10pt, 则

$$(39 \times 12\text{pt} + 10\text{pt} =) 478\text{pt} < 486\text{pt} < 490\text{pt} (= 40 \times 12\text{pt} + 10\text{pt}),$$

结果 `\textheight` 被舍入为 490pt。默认情况下, `heightrounded=false`。

Figure 5 illustrates various layouts with different layout modes. The dimensions for a header and a footer can be controlled by `nohead` or `nofoot` mode, which sets each length to 0pt directly. On the other hand, options with the prefix `ignore` do *not* change the corresponding native dimensions.

The following options can specify body and margins simultaneously with three comma-separated values in braces.

hdivide horizontal partitions (left,width,right). `hdivide={⟨left margin⟩,⟨width⟩,⟨right margin⟩}`. Note that you should not specify all of the three parameters. The best way of using this option is to specify two of three and leave the rest with null(nothing) or ‘*’. For example, when you set `hdivide={2cm,15cm, }`, the margin from the right-side edge of page will be determined calculating `paperwidth-2cm-15cm`.

图 5 展示了不同布局模式下的各种布局。页眉和页脚的尺寸可以通过 `nohead` 或 `nofoot` 模式进行控制, 这会直接将每个长度设置为 0pt。另一方面, 以 `ignore` 为前缀的选项不会改变对应的原始尺寸。

以下选项可以同时指定正文和边距, 使用花括号内的三个逗号分隔的值。

hdivide 水平分割 (左边距, 宽度, 右边距)。`hdivide={⟨左边距⟩,⟨宽度⟩,⟨右边距⟩}`。请注意, 不应同时指定这三个参数。使用此选项的最佳方法是指定其中的两个, 并将剩下的一个设为 null (空) 或 ‘*’。例如, 当设置 `hdivide={2cm,15cm, }` 时, 页面右侧边缘的边距将通过计算 `paperwidth-2cm-15cm` 来确定。

`vdivide` vertical partitions (top,height,bottom). `vdivide={⟨top margin⟩,⟨height⟩,⟨bottom margin⟩}`.
`divide` `divide={A,B,C}` is interpreted as `hdivide={A,B,C}` and `vdivide={A,B,C}`.

4.4 Margin size

The options specifying the size of the margins are listed below.

<code>left</code>	<code>lmargin</code>	<code>inner</code>	left margin (for oneside) or inner margin (for twoside) of <i>total body</i> . In other words, the distance between the left (inner) edge of the paper and that of <i>total body</i> . <code>left=⟨length⟩</code> . <code>inner</code> has no special meaning, just an alias of <code>left</code> and <code>lmargin</code> .
<code>right</code>	<code>rmargin</code>	<code>outer</code>	right or outer margin of <i>total body</i> . <code>right=⟨length⟩</code> .
<code>top</code>	<code>tmargin</code>		top margin of the page. <code>top=⟨length⟩</code> . Note this option has nothing to do with the native dimension <code>\topmargin</code> .
<code>bottom</code>	<code>bmargin</code>		bottom margin of the page. <code>bottom=⟨length⟩</code> .
<code>hmargin</code>			left and right margin. <code>hmargin={⟨left margin⟩,⟨right margin⟩}</code> or <code>hmargin=⟨length⟩</code> .
<code>vmargin</code>			top and bottom margin. <code>vmargin={⟨top margin⟩,⟨bottom margin⟩}</code> or <code>vmargin=⟨length⟩</code> .
<code>margin</code>			<code>margin={A,B}</code> is equivalent to <code>hmargin={A,B}</code> and <code>vmargin={A,B}</code> . <code>margin=A</code> is automatically expanded to <code>hmargin=A</code> and <code>vmargin=A</code> .
<code>hmarginratio</code>			horizontal margin ratio of <code>left</code> (inner) to <code>right</code> (outer). The value of <code>⟨ratio⟩</code> should be specified with colon-separated two values. Each value should be a positive integer less than 100 to prevent arithmetic overflow, e.g., 2:3 instead of 1:1.5. The default ratio is 1:1 for oneside, 2:3 for twoside.
<code>vmarginratio</code>			vertical margin ratio of <code>top</code> to <code>bottom</code> . The default ratio is 2:3.
<code>marginratio</code>	<code>ratio</code>		horizontal and vertical margin ratios. <code>marginratio={⟨horizontal ratio⟩,⟨vertical ratio⟩}</code> or <code>marginratio=⟨ratio⟩</code> .
<code>hcentering</code>			sets auto-centering horizontally and is equivalent to <code>hmarginratio=1:1</code> . It is set to <code>true</code> by default for oneside. See also <code>hmarginratio</code> .
<code>vcentering</code>			sets auto-centering vertically and is equivalent to <code>vmarginratio=1:1</code> . The default is <code>false</code> . See also <code>vmarginratio</code> .
<code>centering</code>			sets auto-centering and is equivalent to <code>marginratio=1:1</code> . See also <code>marginratio</code> . The default is <code>false</code> . See also <code>marginratio</code> .
<code>twoside</code>			switches on twoside mode with left and right margins swapped on verso pages. The option sets <code>\@twoside</code> and <code>\@mparswitch</code> switches. See also <code>asymmetric</code> .
<code>asymmetric</code>			implements a twosided layout in which margins are not swapped on alternate pages (by setting <code>\oddsidemargin</code> to <code>\evensidemargin + bindingoffset</code>) and in which the marginal notes stay always on the same side. This option can be used as an alternative to the <code>twoside</code> option. See also <code>twoside</code> .
<code>bindingoffset</code>			removes a specified space from the lefthand-side of the page for oneside or the inner-side for twoside. <code>bindingoffset=⟨length⟩</code> . This is useful if pages are bound by a press binding (glued, stitched, stapled ...).

<code>vdivide</code>	垂直分割 (上边距, 高度, 下边距)。 <code>vdivide={⟨上边距⟩,⟨高度⟩,⟨下边距⟩}</code> 。
<code>divide</code>	<code>divide={A,B,C}</code> 被解释为 <code>hdivide={A,B,C}</code> 和 <code>vdivide={A,B,C}</code> 。

4.4 边距大小

下面列出了指定边距大小的选项。

<code>left</code>	<code>lmargin</code>	<code>inner</code>	正文的左边距（单页模式）或内部边距（双页模式）。换句话说，纸张左（内）边缘与正文左（内）边缘之间的距离。 <code>left=⟨长度⟩</code> 。 <code>inner</code> 没有特殊含义，只是 <code>left</code> 和 <code>lmargin</code> 的别名。
<code>right</code>	<code>rmargin</code>	<code>outer</code>	正文的右边距或外部边距。 <code>right=⟨长度⟩</code> 。
<code>top</code>	<code>tmargin</code>		页面的上边距。 <code>top=⟨长度⟩</code> 。 请注意，此选项与原始尺寸 <code>\topmargin</code> 无关。
<code>bottom</code>	<code>bmargin</code>		页面的下边距。 <code>bottom=⟨长度⟩</code> 。
<code>hmargin</code>			左边距和右边距。 <code>hmargin={⟨左边距⟩,⟨右边距⟩}</code> 或 <code>hmargin=⟨长度⟩</code> 。
<code>vmargin</code>			上边距和下边距。 <code>vmargin={⟨上边距⟩,⟨下边距⟩}</code> 或 <code>vmargin=⟨长度⟩</code> 。
<code>margin</code>			<code>margin={A,B}</code> 等同于 <code>hmargin={A,B}</code> 和 <code>vmargin={A,B}</code> 。 <code>margin=A</code> 会自动展开为 <code>hmargin=A</code> 和 <code>vmargin=A</code> 。
<code>hmarginratio</code>			<code>left</code> （内）与 <code>right</code> （外）的水平边距比例。 <code>⟨比例⟩</code> 的值应使用冒号分隔的两个值来指定。每个值应为小于 100 的正整数，以防止算术溢出，例如使用 2:3 而不是 1:1.5。对于单页模式，默认比例是 1:1，对于双页模式，默认比例是 2:3。
<code>vmarginratio</code>			<code>top</code> 与 <code>bottom</code> 的垂直边距比例。默认比例是 2:3。
<code>marginratio</code>	<code>ratio</code>		水平和垂直边距比例。 <code>marginratio={⟨水平比例⟩,⟨垂直比例⟩}</code> 或 <code>marginratio=⟨比例⟩</code> 。
<code>hcentering</code>			设置水平自动居中，等同于 <code>hmarginratio=1:1</code> 。默认情况下，单页模式下设置为 <code>true</code> 。见 <code>hmarginratio</code> 。
<code>vcentering</code>			设置垂直自动居中，等同于 <code>vmarginratio=1:1</code> 。默认情况下为 <code>false</code> 。见 <code>vmarginratio</code> 。
<code>centering</code>			设置自动居中，等同于 <code>marginratio=1:1</code> 。默认情况下为 <code>false</code> 。见 <code>marginratio</code> 。
<code>twoside</code>			打开双面模式，交换背面页面的左右边距。此选项设置 <code>\@twoside</code> 和 <code>\@mparswitch</code> 开关。见 <code>asymmetric</code> 。
<code>asymmetric</code>			实现了一个双面布局，其中边距在交替页面上不会交换（通过将 <code>\oddsidemargin</code> 设置为 <code>\evensidemargin + bindingoffset</code> ），并且边注始终停留在同一侧。此选项可用作 <code>twoside</code> 选项的替代方案。见 <code>twoside</code> 。
<code>bindingoffset</code>			从页面的左侧（单页模式或双页模式的内部侧）移除指定的空间。 <code>bindingoffset=⟨长度⟩</code> 。如果页面通过压合装订（胶合、缝合、订书钉等）装订，则这很有用。见图6。

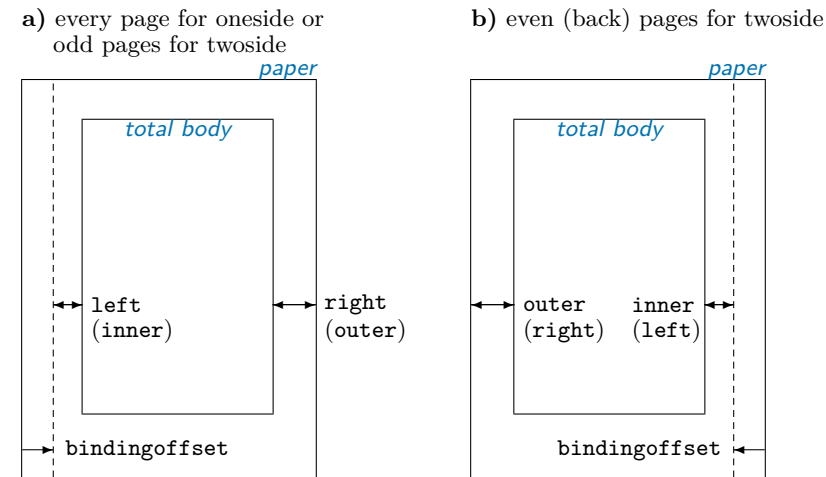


图 6: The option `bindingoffset` adds the specified length to the inner margin. Note that `twoside` option swaps the horizontal margins and the marginal notes together with `bindingoffset` on even pages (see **b**)), but `asymmetric` option suppresses the swap of the margins and marginal notes (but `bindingoffset` is still swapped).

选项 `bindingoffset` 将指定的长度添加到内边距。请注意, `twoside` 选项会在偶数页上与 `bindingoffset` 一起交换水平边距和边注 (参见 **b**)), 但 `asymmetric` 选项会取消交换边距和边注 (但 `bindingoffset` 仍然会被交换)。

See Figure 6.

`hdivide` See description in Section 4.3.
`vdivide` See description in Section 4.3.
`divide` See description in Section 4.3.

`hdivide` 见第 4.3 节中的描述。
`vdivide` 见第 4.3 节中的描述。
`divide` 见第 4.3 节中的描述。

4.5 Native dimensions

The options below overwrite \LaTeX native dimensions and switches for page layout (See the right-hand side in Figure 1).

`headheight` | `head`
 modifies `\headheight`, height of header. `headheight=<length>` or `head=<length>`.
`headsep` modifies `\headsep`, separation between header and text (body). `headsep=<length>`.
`footskip` | `foot`
 modifies `\footskip`, distance separation between baseline of last line of text and baseline of footer. `footskip=<length>` or `foot=<length>`.
`nohead` eliminates spaces for the head of the page, which is equivalent to both `\headheight=0pt` and `\headsep=0pt`.
`nofoot` eliminates spaces for the foot of the page, which is equivalent to `\footskip=0pt`.
`noheadfoot` equivalent to `nohead` and `nofoot`, which means that `\headheight`, `\headsep` and `\footskip` are all set to 0pt.
`footnotesep` changes the dimension `\skip\footins`, separation between the bottom of text body and the top of footnote text.
`marginparwidth` | `marginpar`
 modifies `\marginparwidth`, width of the marginal notes. `marginparwidth=<length>`.
`marginparsep` modifies `\marginparsep`, separation between body and marginal notes. `marginparsep=<length>`.
`nomarginpar` shrinks spaces for marginal notes to 0pt, which is equivalent to `\marginparwidth=0pt` and `\marginparsep=0pt`.
`columnsep` modifies `\columnsep`, the separation between two columns in `twocolumn` mode.

4.5 原生尺寸

下面的选项可以覆盖 \LaTeX 的原生尺寸和页面布局开关 (见图 1 的右侧)。

`headheight` | `head`
 修改 `\headheight`, 页眉的高度。 `headheight=<length>` 或 `head=<length>`。
`headsep` 修改 `\headsep`, 页眉和正文 (主体) 之间的距离。 `headsep=<length>`。
`footskip` | `foot`
 修改 `\footskip`, 正文最后一行的基线与页脚基线之间的距离。 `footskip=<length>` 或 `foot=<length>`。
`nohead` 消除页头的空白, 等效于 `\headheight=0pt` 和 `\headsep=0pt`。
`nofoot` 消除页脚的空白, 等效于 `\footskip=0pt`。
`noheadfoot` 等效于 `nohead` 和 `nofoot`, 即将 `\headheight`、`\headsep` 和 `\footskip` 都设为 0pt。
`footnotesep` 修改 `\skip\footins` 的尺寸, 即正文底部与脚注文本顶部之间的距离。
`marginparwidth` | `marginpar`
 修改 `\marginparwidth`, 边注的宽度。 `marginparwidth=<length>`。
`marginparsep` 修改 `\marginparsep`, 正文和边注之间的距离。 `marginparsep=<length>`。
`nomarginpar` 将边注的空白缩小为 0pt, 等效于 `\marginparwidth=0pt` 和 `\marginparsep=0pt`。
`columnsep` 修改 `\columnsep`, 在双栏模式下两栏之间的距离。

<code>hoffset</code>	modifies <code>\hoffset</code> . <code>hoffset=<length></code> .
<code>voffset</code>	modifies <code>\voffset</code> . <code>voffset=<length></code> .
<code>offset</code>	horizontal and vertical offset. <code>offset={\<hoffset>,\<voffset>}</code> or <code>offset=<length></code> .
<code>twocolumn</code>	sets <code>twocolumn</code> mode with <code>\@twocolumntrue</code> . <code>twocolumn=false</code> denotes <code>onecolumn</code> mode with <code>\@twocolumnfalse</code> . Instead of <code>twocolumn=false</code> , you can specify <code>onecolumn</code> (which defaults to <code>true</code>)
<code>onecolumn</code>	works as <code>twocolumn=false</code> . On the other hand, <code>onecolumn=false</code> is equivalent to <code>twocolumn</code> .
<code>twoside</code>	sets both <code>\@twosidetrue</code> and <code>\@mparswitchtrue</code> . See Section 4.4.
<code>textwidth</code>	sets <code>\textwidth</code> directly. See Section 4.3.
<code>textheight</code>	sets <code>\textheight</code> directly. See Section 4.3.
<code>reversemp</code> <code>reversemarginpar</code>	makes the marginal notes appear in the left (inner) margin with <code>\@reversemargintrue</code> . The option doesn't change <code>includemp</code> mode. It's set <code>false</code> by default.

4.6 Drivers

The package supports drivers `dvips`, `dvipdfm`, `pdftex`, `luatex`, `xetex` and `vtex`. You can also set `dvipdfm` for `dvipdfmx` and `xdvipdfmx` The options `dvipdfmx` and `xdvipdfmx` are also supported as aliases for the `dvipdfm` option. `pdftex` for `pdflatex`, and `vtex` for `VTEX` environment. The driver options are exclusive. The driver can be set by either `driver=<driver name>` or any of the drivers directly like `pdftex`. By default, `geometry` guesses the driver appropriate to the system in use. Therefore, you don't have to set a driver in most cases. However, if you want to use `dvipdfm`, you should specify it explicitly.

[†] <code>driver</code>	specifies the driver with <code>driver=<driver name></code> . <code>dvips</code> , <code>dvipdfm</code> , <code>pdftex</code> , <code>luatex</code> , <code>vtex</code> , <code>xetex</code> , <code>auto</code> and <code>none</code> are available as a driver name. The names except for <code>auto</code> and <code>none</code> can be specified directly with the name without <code>driver=</code> . <code>driver=auto</code> makes the auto-detection work whatever the previous setting is. <code>driver=none</code> disables the auto-detection and sets no driver, which may be useful when you want to let other package work out the driver setting. For example, if you want to use <code>crop</code> package with <code>geometry</code> , you should call <code>\usepackage[driver=none]{geometry}</code> before the <code>crop</code> package.
[†] <code>dvips</code>	writes the paper size in dvi output with the <code>\special</code> macro. If you use <code>dvips</code> as a DVI-to-PS driver, for example, to print a document with <code>\geometry{a3paper,landscape}</code> on A3 paper in landscape orientation, you don't need options “ <code>-t a3 -t landscape</code> ” to <code>dvips</code> .
[†] <code>dvipdfm</code>	works like <code>dvips</code> except for landscape correction. You can set this option when using <code>dvipdfmx</code> and <code>xdvipdfmx</code> to process the dvi output.
[†] <code>pdftex</code>	sets <code>\pdfpagewidth</code> and <code>\pdfpageheight</code> internally.
[†] <code>luatex</code>	sets <code>\pagewidth</code> and <code>\pageheight</code> internally.
[†] <code>xetex</code>	is the same as <code>pdftex</code> except for ignoring <code>\pdf{h,v}origin</code> undefined in <code>X_qL^AT_EX</code> . This option is introduced in the version 5. Note that ‘ <code>geometry.cfg</code> ’ in <code>T_EX Live</code> , which disables the auto-detection routine and sets <code>pdftex</code> , is no longer necessary, but has no problem even though it's left undeleted. Instead of <code>xetex</code> , you can specify <code>dvipdfm</code> with <code>X_qL^AT_EX</code> if you want to use specials of <code>dvipdfm</code> <code>X_qT_EX</code> supports.

<code>hoffset</code>	修改 <code>\hoffset</code> 。 <code>hoffset=<length></code> 。
<code>voffset</code>	修改 <code>\voffset</code> 。 <code>voffset=<length></code> 。
<code>offset</code>	水平和垂直偏移。 <code>offset={\<hoffset>,\<voffset>}</code> 或 <code>offset=<length></code> 。
<code>twocolumn</code>	使用 <code>\@twocolumntrue</code> 设置双栏模式。 <code>twocolumn=false</code> 表示单栏模式，等效于 <code>\@twocolumnfalse</code> 。你也可以使用 <code>onecolumn</code> 来代替 <code>twocolumn=false</code> （默认为 <code>true</code> ）。
<code>onecolumn</code>	等效于 <code>twocolumn=false</code> 。另一方面， <code>onecolumn=false</code> 等效于 <code>twocolumn</code> 。
<code>twoside</code>	同时设置 <code>\@twosidetrue</code> 和 <code>\@mparswitchtrue</code> 。见第 <code>refsec:margin</code> 节。
<code>textwidth</code>	直接设置 <code>\textwidth</code> 。见第4.3节。
<code>textheight</code>	直接设置 <code>\textheight</code> 。见第 <code>refsec:body</code> 节。
<code>reversemp</code> <code>reversemarginpar</code>	这个选项使用 <code>\@reversemargintrue</code> 让边注出现在左侧（内侧）边距。这个选项不会改变 <code>includemp</code> 模式。默认情况下，该选项为 <code>false</code> 。

4.6 Driver 驱动

这个选项用于设置驱动程序（driver），可以使用的驱动程序有：`dvips`、`dvipdfm`、`pdftex`、`luatex`、`xetex` 和 `vtex`。你也可以将 `dvipdfmx` 用于 `dvipdfmx` 和 `xdvipdfmx`，选项 `pdftex` 用于 `pdflatex`，选项 `vtex` 用于 `VTEX`。这些驱动程序选项是互斥的。你可以通过 `driver=<driver name>` 或直接使用驱动程序进行设置，如 `pdftex`。默认情况下，`geometry` 会猜测适合当前系统的驱动程序，所以在大多数情况下你不需要设置驱动程序。但是，如果你想使用 `dvipdfm`，你需要明确指定它。

[†] <code>driver</code>	使用 <code>driver=<driver name></code> 来指定驱动程序。可用的驱动程序名称有 <code>dvips</code> 、 <code>dvipdfm</code> 、 <code>pdftex</code> 、 <code>luatex</code> 、 <code>vtex</code> 、 <code>xetex</code> 、 <code>auto</code> 和 <code>none</code> 。除了 <code>auto</code> 和 <code>none</code> 外的名称可以直接使用不带 <code>driver=</code> 的名称指定。 <code>driver=auto</code> 会使自动检测驱动程序的功能生效，无论之前的设置是什么。 <code>driver=none</code> 会禁用自动检测，并且不设置驱动程序，这在你想让其他包来决定驱动程序设置时可能会有用。例如，如果你想在使用 <code>crop</code> 宏包时使用 <code>geometry</code> ，你应该在 <code>crop</code> 宏包之前调用 <code>\usepackage[driver=none]{geometry}</code> 。
[†] <code>dvips</code>	使用 <code>\special</code> 宏来在 dvi 输出中写入纸张大小。如果你使用 <code>dvips</code> 作为 DVI 到 PS 的驱动程序，例如在 A3 纸上打印 <code>\geometry{a3paper,landscape}</code> 的文档时，你不需要在 <code>dvips</code> 中使用 “ <code>-t a3 -t landscape</code> ” 选项。
[†] <code>dvipdfm</code>	与 <code>dvips</code> 相同，只是不会进行横向纠正。当使用 <code>dvipdfmx</code> 和 <code>xdvipdfmx</code> 处理 dvi 输出时，可以设置此选项。
[†] <code>pdftex</code>	内部设置 <code>\pdfpagewidth</code> 和 <code>\pdfpageheight</code> 。
[†] <code>luatex</code>	内部设置 <code>\pagewidth</code> 和 <code>\pageheight</code> 。
[†] <code>xetex</code>	与 <code>pdftex</code> 相同，只是在 <code>X_qL^AT_EX</code> 中忽略 <code>\pdf{h,v}origin</code> 未定义。此选项在版本 5 中引入。注意，在 <code>T_EX Live</code> 中的 ‘ <code>geometry.cfg</code> ’，它禁用了自动检测例程并设置为 <code>pdftex</code> ，虽然不再需要，但即使未删除也没有问题。如果你想在 <code>X_qL^AT_EX</code> 中使用 <code>dvipdfm</code> <code>X_qT_EX</code> 支持的特殊命令，可以用 <code>dvipdfm</code> 替代 <code>xetex</code> 。

[†] `vtex` sets dimensions `\mediawidth` and `\mediaheight` for \TeX . When this driver is selected (explicitly or automatically), `geometry` will auto-detect which output mode (DVI, PDF or PS) is selected in \TeX , and do proper settings for it.

If explicit driver setting is mismatched with the typesetting program in use, the default driver `dvips` would be selected.

4.7 Other options

The other useful options are described here.

[†] `verbose` displays the parameter results on the terminal. `verbose=false` (default) still puts them into the log file.

[†] `reset` sets back the layout dimensions and switches to the settings before `geometry` is loaded. Options given in `geometry.cfg` are also cleared. Note that this cannot reset `pass` and `mag` with `truedimen`. `reset=false` has no effect and cannot cancel the previous `reset(=true)` if any. For example, when you go

```
\documentclass[landscape]{article}
\usepackage[twoside,reset,left=2cm]{geometry}
```

with `\ExecuteOptions{scale=0.9}` in `geometry.cfg`, then as a result, `landscape` and `left=2cm` remain effective, and `scale=0.9` and `twoside` are ineffective.

[†] `mag` sets magnification value (`\mag`) and automatically modifies `\hoffset` and `\voffset` according to the magnification. `mag=<value>`. Note that `<value>` should be an integer value with 1000 as a normal size. For example, `mag=1414` with `a4paper` provides an enlarged print fitting in `a3paper`, which is 1.414 ($=\sqrt{2}$) times larger than `a4paper`. Font enlargement needs extra disk space. **Note that setting `mag` should precede any other settings with ‘true’ dimensions, such as `1.5truein`, `2truecm` and so on.** See also `truedimen` option.

[†] `truedimen` changes all internal explicit dimension values into *true* dimensions, e.g., `1in` is changed to `1truein`. Typically this option will be used together with `mag` option. Note that this is ineffective against externally specified dimensions. For example, when you set “`mag=1440, margin=10pt, truedimen`”, margins are not ‘true’ but magnified. If you want to set exact margins, you should set like “`mag=1440, margin=10truept, truedimen`” instead.

[†] `pass` disables all of the `geometry` options and calculations except `verbose` and `showframe`. It is order-independent and can be used for checking out the page layout of the `documentclass`, other packages and manual settings without `geometry`.

[†] `showframe` shows visible frames for the text area and page, and the lines for the head and foot on the first page.

[†] `showcrop` prints crop marks at each corner of user-specified layout area.

5 Processing options

5.1 Order of loading

If there’s `geometry.cfg` somewhere \TeX can find it, `geometry` loads it first. For example, in `geometry.cfg` you may write `\ExecuteOptions{a4paper}`, which specifies A4 paper as the default paper. Basically you can use all the options defined in `geometry` with `\ExecuteOptions{}`.

The order of loading in the preamble of your document is as follows:

[†] `vtex` 为 \TeX 设置 `\mediawidth` 和 `\mediaheight` 尺寸。当选择了此驱动程序（显式或自动），`geometry` 将自动检测在 \TeX 中选择的输出模式（DVI、PDF 或 PS）并进行适当的设置。

如果显式设置的驱动程序与正在使用的排版程序不匹配，则会选择默认的驱动程序 `dvips`。

4.7 其他选项

这里介绍了其他有用的选项。

[†] `verbose` 在终端上显示参数结果。`verbose=false`（默认值）仍然会将结果放入日志文件中。

[†] `reset` 将布局尺寸设置回到加载 `geometry` 之前的设置。`geometry.cfg` 中给出的选项也会被清除。注意，这不能重置带有 `truedimen` 的 `pass` 和 `mag`。`reset=false` 没有效果，也无法取消之前的 `reset(=true)` 设置。例如，当你使用 `\ExecuteOptions{scale=0.9}` 在 `geometry.cfg` 中进行如下设置：

```
\documentclass[landscape]{article}
\usepackage[twoside,reset,left=2cm]{geometry}
```

那么结果是 `landscape` 和 `left=2cm` 仍然生效，而 `scale=0.9` 和 `twoside` 不生效。

[†] `mag` 设置放大倍数 (`\mag`) 并根据放大倍数自动修改 `\hoffset` 和 `\voffset`。`mag=<value>`。注意，`<value>` 应该是一个整数值，以 1000 作为正常大小。例如，`mag=1414` 结合 `a4paper` 提供了一个适合于 `a3paper` 的放大打印，即比 `a4paper` 大 1.414 ($=\sqrt{2}$) 倍。字体的放大需要额外的磁盘空间。注意，设置 `mag` 应该在任何其他带有真实尺寸的设置之前，例如 `1.5truein`、`2truecm` 等。还请参见 `truedimen` 选项。

[†] `truedimen` 将所有内部显式尺寸值更改为真实尺寸，例如将 `1in` 更改为 `1truein`。通常，此选项将与 `mag` 选项一起使用。注意，这对于外部指定的尺寸是无效的。例如，当你设置 `mag=1440, margin=10pt, truedimen` 时，边距不是 ‘true’，而是被放大了。如果你想设置精确的边距，应该改为 `mag=1440, margin=10truept, truedimen`。

[†] `pass` 禁用所有的几何选项和计算，除了 `verbose` 和 `showframe`。它是无序的，可以用于检查文档类、其他包和手动设置的页面布局，而不使用 `geometry`。

[†] `showframe` 在文本区域和页面上显示可见的框架，以及第一页上的页眉和页脚线条。

[†] `showcrop` 在用户指定的布局区域的每个角上打印出裁剪标记。

处理选项

5.1 加载顺序

如果 \TeX 能找到 `geometry.cfg` 文件，`geometry` 会首先加载它。例如，在 `geometry.cfg` 文件中，您可以写上 `\ExecuteOptions{a4paper}`，这将把 A4 纸设置为默认纸张。基本上，您可以使用 `geometry` 中定义的所有选项与 `\ExecuteOptions{}` 一起使用。

在您的文档的导言部分加载的顺序如下：

1. `geometry.cfg` if it exists.
2. Options specified with `\documentclass[⟨options⟩]{...}`.
3. Options specified with `\usepackage[⟨options⟩]{geometry}`
4. Options specified with `\geometry{⟨options⟩}`, which can be called multiple times. (`reset` option will cancel the specified options ever given in `\usepackage{geometry}` or `\geometry`.)

5.2 Order of options

The specification of `geometry` options is order-independent, and overwrites the previous one for the same setting. For example,

`[left=2cm, right=3cm]` is equivalent to `[right=3cm, left=2cm]`.

The options called multiple times overwrite the previous settings. For example,

`[verbose=true, verbose=false]` results in `verbose=false`.

`[hmargin={3cm,2cm}, left=1cm]` is the same as `hmargin={1cm,2cm}`, where the left (or inner) margin is overwritten by `left=1cm`.

`reset` and `mag` are exceptions. The `reset` option removes all the `geometry` options (except `pass`) before it. If you set

```
\documentclass[landscape]{article}
\usepackage[margin=1cm,twoside]{geometry}
\geometry{a5paper, reset, left=2cm}
```

then `margin=1cm`, `twoside` and `a5paper` are removed, and is eventually equivalent to

```
\documentclass[landscape]{article}
\usepackage[left=2cm]{geometry}
```

The `mag` option should be set in advance of any other settings with ‘true’ length, such as `left=1.5truecm`, `width=5truein` and so on. The `\mag` primitive can be set before this package is called.

5.3 Priority

There are several ways to set dimensions of the *body*: `scale`, `total`, `text` and `lines`. The `geometry` package gives higher priority to the more concrete specification. Here is the priority rule for *body*.

$$\begin{array}{c} \text{priority:} \quad \text{low} \quad \longrightarrow \quad \text{high} \\ \left\{ \begin{array}{c} \text{hscale} \\ \text{vscale} \\ \text{scale} \end{array} \right\} < \left\{ \begin{array}{c} \text{width} \\ \text{height} \\ \text{total} \end{array} \right\} < \left\{ \begin{array}{c} \text{textwidth} \\ \text{textheight} \\ \text{text} \end{array} \right\} < \text{lines.} \end{array}$$

For example,

```
\usepackage[hscale=0.8, textwidth=7in, width=18cm]{geometry}
```

is the same as `\usepackage[textwidth=7in]{geometry}`. Another example:

```
\usepackage[lines=30, scale=0.8, text=7in]{geometry}
```

1. 如果存在`geometry.cfg`, 加载之。
2. 使用`\documentclass[⟨options⟩]{...}` 指定的选项。
3. 使用`\usepackage[⟨options⟩]{geometry}` 指定的选项。
4. 使用`\geometry{⟨options⟩}` 指定的选项, 可以多次调用。(`reset` 选项将取消在`\usepackage{geometry}` 或`\geometry` 中指定的选项。)

5.2 选项的顺序

`geometry` 选项的指定是无序的, 并且会覆盖先前相同设置的选项。例如,

`[left=2cm, right=3cm]` 等同于 `[right=3cm, left=2cm]`。

多次调用的选项会覆盖先前的设置。例如,

`[verbose=true, verbose=false]` 的结果是 `verbose=false`。

`[hmargin={3cm,2cm}, left=1cm]` 等同于 `hmargin={1cm,2cm}`, 其中左侧 (或内侧) 边距被 `left=1cm` 覆盖。

`reset` 和 `mag` 是例外。`reset` 选项会在其之前删除所有的几何选项 (除了 `pass`)。如果您设置

```
\documentclass[landscape]{article}
\usepackage[margin=1cm,twoside]{geometry}
\geometry{a5paper, reset, left=2cm}
```

那么 `margin=1cm`、`twoside` 和 `a5paper` 被删除, 并最终等同于

```
\documentclass[landscape]{article}
\usepackage[left=2cm]{geometry}
```

`mag` 选项应该在任何其他带有 ‘true’ 长度的设置之前进行设置, 例如 `left=1.5truecm`、`width=5truein` 等等。`\mag` 原语可以在调用此宏包之前进行设置。

5.3 优先级

有多种方式可以设置正文的尺寸: `scale`、`total`、`text` 和 `lines`。`geometry` 宏包对更具体的规格给予更高的优先级。以下是正文的优先级规则。

$$\begin{array}{c} \text{优先级:} \quad \text{低} \quad \longrightarrow \quad \text{高} \\ \left\{ \begin{array}{c} \text{hscale} \\ \text{vscale} \\ \text{scale} \end{array} \right\} < \left\{ \begin{array}{c} \text{width} \\ \text{height} \\ \text{total} \end{array} \right\} < \left\{ \begin{array}{c} \text{textwidth} \\ \text{textheight} \\ \text{text} \end{array} \right\} < \text{lines.} \end{array}$$

例如,

```
\usepackage[hscale=0.8, textwidth=7in, width=18cm]{geometry}
```

等同于 `\usepackage[textwidth=7in]{geometry}`。另一个例子:

```
\usepackage[lines=30, scale=0.8, text=7in]{geometry}
```

results in `[lines=30, textwidth=7in]`.

5.4 Defaults

This section sums up the default settings for the auto-completion described later. The default vertical margin ratio is 2/3, namely,

$$\text{top} : \text{bottom} = 2 : 3 \quad \text{default.} \tag{6}$$

As for the horizontal margin ratio, the default value depends on whether the document is onesided or twosided,

$$\text{left (inner)} : \text{right (outer)} = \begin{cases} 1 : 1 & \text{default for oneside,} \\ 2 : 3 & \text{default for twoside.} \end{cases} \tag{7}$$

Obviously the default horizontal margin ratio for oneside is ‘centering’. The `geometry` package has the following default setting for *onesided* documents:

- `scale=0.7` (*body* is $0.7 \times \textit{paper}$)
- `marginratio={1:1, 2:3}` (1:1 for horizontal and 2:3 for vertical margins)
- `ignoreall` (the header, footer, marginal notes are excluded when calculating the size of *body*.)

For *twosided* document with `twoside` option, the default setting is the same as *onesided* except that the horizontal margin ratio is set to 2:3 as well. Additional options overwrite the previous specified dimensions.

5.5 Auto-completion

Figure 7 shows schematically how many specification patterns exist and how to solve the ambiguity of the specifications. Each axis shows the numbers of lengths explicitly specified for body and margins. $S(m,b)$ presents the specification with a set of numbers (*margin, body*) = (m,b). For example, the specification `width=14cm, left=3cm` is categorized into $S(1,1)$, which is an adequate specification. If you add `right=4cm`, it would be in $S(2,1)$ and overspecified. If only `width=14cm` is given, it’s in $S(0,1)$, underspecified. The `geometry` package has the auto-completion mechanism, in which if the layout parameters are underspecified or overspecified, `geometry` works out the ambiguity using the defaults and other relations. Here are the specifications and the completion rules.

S(0,0) Nothing is specified. The `geometry` package sets *body* with the default `scale` (= 0.7). For example, `width` is set to be $0.7 \times \text{layoutwidth}$. Note that by default `layoutwidth` and `layoutheight` will be equal to `\paperwidth` and `\paperheight` respectively. Thus $S(0,0)$ goes to $S(0,1)$. See $S(0,1)$.

S(0,1) Only *body* is specified, such as `width=7in, lines=20, body={20cm,24cm}, scale=0.9` and so forth. Then `geometry` sets margins with the margin ratio. If the margin ratio is not specified, the default is used. The default vertical margin ratio is defined as

$$\text{top} : \text{bottom} = 2 : 3 \quad \text{default.} \tag{8}$$

As for the horizontal margin ratio, the default value depends on whether the document is onesided or

的结果是 `[lines=30, textwidth=7in]`.

5.4 默认设置

本节总结了后面描述的自动补全的默认设置。默认的垂直边距比例是 2/3，即

$$\text{top} : \text{bottom} = 2 : 3 \quad \text{default.} \tag{5}$$

至于水平边距比例，默认值取决于文档是单面还是双面的，

$$\text{left (inner)} : \text{right (outer)} = \begin{cases} 1 : 1 & \text{单面默认,} \\ 2 : 3 & \text{双面默认.} \end{cases} \tag{6}$$

显然，单面的默认水平边距比例是“居中”。对于 `geometry` 宏包，默认的设置对于单面文档如下：

- `scale=0.7` (*正文*是 $0.7 \times \textit{纸张}$)
- `marginratio={1:1, 2:3}`（水平边距比例为 1:1，垂直边距比例为 2:3）
- `ignoreall`（在计算*正文*的大小时，头部、底部和边注都被排除在外。）

对于带有 `twoside` 选项的双面文档，默认设置与单面相同，只是水平边距比例也设置为 2:3。附加选项会覆盖先前指定的尺寸。

5.5 自动补全

图 7 简要展示了有多少种规格模式以及如何解决规格的模糊性。每个轴表示正文和边距明确指定的长度的数量。 $S(m,b)$ 表示一组数字 (*边距, 正文*) = (m,b) 的规格。例如，规格 `width=14cm, left=3cm` 归类为 $S(1,1)$ ，这是一个合适的规格。如果添加 `right=4cm`，则属于 $S(2,1)$ ，即过度规定。如果只给出 `width=14cm`，则属于 $S(0,1)$ ，即不足规定。`geometry` 宏包具有自动补全机制，如果布局参数不足或过度规定，`geometry` 会使用默认值和其他关系解决模糊性。以下是规格和补全规则。

S(0,0) 未指定任何内容。`geometry` 宏包使用默认值 `scale` (= 0.7) 设置 *正文*。例如，`width` 被设置为 $0.7 \times \text{layoutwidth}$ 。需要注意的是，默认情况下，`layoutwidth` 和 `layoutheight` 分别等于 `\paperwidth` 和 `\paperheight`。因此， $S(0,0)$ 转向 $S(0,1)$ 。请参见 $S(0,1)$ 。

S(0,1) 只指定了 *正文*，例如 `width=7in, lines=20, body={20cm,24cm}, scale=0.9` 等。然后，`geometry` 使用边距比例设置边距。如果未指定边距比例，则使用默认值。默认的垂直边距比例定义为

$$\text{top} : \text{bottom} = 2 : 3 \quad \text{默认值.} \tag{7}$$

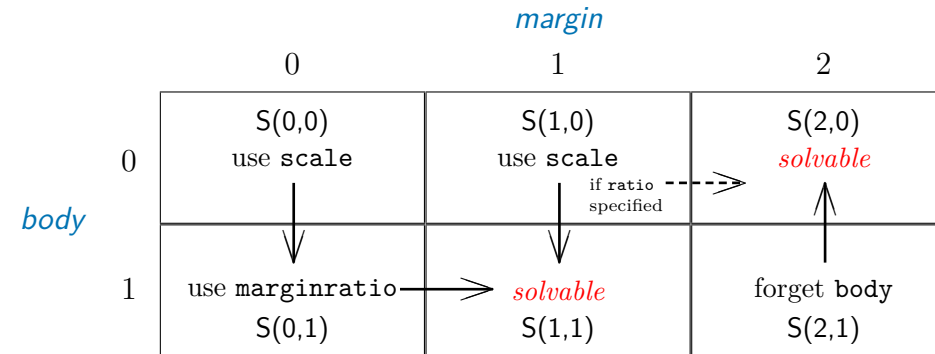


图 7: Specifications $S(0,0)$ to $S(2,1)$ and the completion rules (arrows). Column and row numbers denote the number of explicitly specified lengths for margin and body respectively.

$S(m,b)$ denote a specification with a set of the numbers $(margin, body) = (m, b)$.

规格 $S(0,0)$ 到 $S(2,1)$ 和补全规则 (箭头)。列和行号分别表示边距和正文明确指定长度的数量。 $S_{(m,b)}$ 表示一组数字 (边距, 正文) = (m, b) 的规格。

twosided,

$$\text{left (inner)} : \text{right (outer)} = \begin{cases} 1 : 1 & \text{default for oneside,} \\ 2 : 3 & \text{default for twoside.} \end{cases} \quad (9)$$

For example, if `height=22cm` is specified on A4 paper, geometry calculates `top` margin as follows:

$$\begin{aligned} \text{top} &= (\text{layoutheight} - \text{height}) \times 2/5 \\ &= (29.7 - 22) \times 2/5 = 3.08(\text{cm}) \end{aligned} \quad (10)$$

Thus `top` margin and body `height` have been determined, the specification for the vertical goes to $S(1,1)$ and all the parameters can be solved.

S(1,0) Only one margin is specified, such as `bottom=2cm`, `left=1in`, `top=3cm`, and so forth.

- If the margin ratio is *not* specified, geometry sets *body* with the default `scale` (= 0.7). For example, if `top=2.4cm` is specified, geometry sets

$$\text{height} = 0.7 \times \text{layoutheight} \quad (= 0.7 \backslash \text{paperheight by default}),$$

then $S(1,0)$ goes to $S(1,1)$, in which `bottom` is calculated with `layoutheight - (height + top)` and results in 6.51cm on A4 paper if the layout size is equal to the paper size.

- If the margin ratio is specified, such as `hmarginratio={1:2}`, `vratio={3:4}` and so forth, geometry sets the otherFor example, if a set of options “`top=2.4cm, vratio={3:4}`” is specified, geometry sets `bottom` to be 3.2cm calculating

$$\text{bottom} = \text{top}/3 \times 4 = 3.2\text{cm}$$

Thus $S(1,0)$ goes to $S(2,0)$.

Note that the version 4 or earlier used to set the other margin with the margin ratio. In the version 5, therefore, with the same specification, the result will be different from the one in the version 4. For example, if only `top=2.4cm` is specified, you got `bottom=2.4cm` in the version 4 or earlier, but you will get `bottom=6.51cm` in the version 5.

S(2,1) The *body* and two *margins* are all specified, such as `vdivide={1in,8in,1.5in}`,

至于水平边距比例，其默认值取决于文档是单面还是双面的情况，

$$\text{left (inner)} : \text{right (outer)} = \begin{cases} 1 : 1 & \text{单面的默认值,} \\ 2 : 3 & \text{双面的默认值.} \end{cases} \quad (8)$$

例如，在 A4 纸上指定 `height=22cm`, geometry 计算 `top` 边距如下：

$$\begin{aligned} \text{top} &= (\text{layoutheight} - \text{height}) \times 2/5 \\ &= (29.7 - 22) \times 2/5 = 3.08(\text{cm}) \end{aligned} \quad (9)$$

因此，确定了 `top` 边距和正文 `height`，垂直规格转到 $S(1,1)$ ，所有参数都可以解决。

S(1,0) 只指定了一个边距，例如 `bottom=2cm`, `left=1in`, `top=3cm` 等。

- 如果未指定边距比例，geometry 使用默认值 `scale` (= 0.7) 设置 *正文*。例如，如果指定了 `top=2.4cm`, geometry 设置

$$\text{height} = 0.7 \times \text{layoutheight} \quad (= 0.7 \backslash \text{paperheight 默认情况下}),$$

然后， $S(1,0)$ 转向 $S(1,1)$ ，在其中使用 `layoutheight - (height + top)` 计算 `bottom`，如果布局大小等于纸张大小，则在 A4 纸上结果为 6.51cm。

- 如果指定了边距比例，例如 `hmarginratio={1:2}`, `vratio={3:4}` 等，geometry 使用指定的边距比例设置另一个边距。例如，如果指定了一组选项 “`top=2.4cm, vratio={3:4}`”，geometry 将 `bottom` 设置为 3.2cm，计算如下：

$$\text{bottom} = \text{top}/3 \times 4 = 3.2\text{cm}$$

因此， $S(1,0)$ 转向 $S(2,0)$ 。

需要注意的是，版本 4 或更早的版本通常使用边距比例设置另一个边距。因此，在版本 5 中，如果使用相同的规格，结果将与版本 4 中的结果不同。例如，如果只指定 `top=2.4cm`，在版本 4 或更早的版本中，你将得到 `bottom=2.4cm`，但在版本 5 中，你将得到 `bottom=6.51cm`。

S(2,1) 指定了 *正文* 和两个 *边距*，例如 `vdivide={1in,8in,1.5in}`,

“left=3cm,width=13cm,right=4cm” and so forth. Since geometry basically gives priority to *margins* if dimensions are overspecified, geometry forgets and resets *body*. For example, if you specify

```
\usepackage[a4paper,left=3cm,width=13cm,right=4cm]{geometry},
```

width is reset to be 14cm because the width of a A4 paper is 21cm long.

6 Changing layout mid-document

The version 5 provides the new commands `\newgeometry{...}` and `\restoregeometry`, which allow you to change page dimensions in the middle of the document. Unlike `\geometry` in the preamble, `\newgeometry` is available only after `\begin{document}`, resets all the options ever specified except for the papersize-related options: `landscape`, `portrait`, and paper size options (such as `papersize`, `paper=a4paper` and so forth), which can't be changed with `\newgeometry`. The command `\restoregeometry` restores the page layout specified in the preamble (before `\begin{document}`) with the options to `\usepackage{geometry}` and `\geometry`. Note that both `\newgeometry` and `\restoregeometry` insert `\clearpage` where they are called.

Below is an example of changing layout mid-document. The layout L1 specified with `hmargin=3cm` (left and right margins are 3cm long) is changed to L2 with `left=3cm`, `right=1cm` and `bottom=0.1cm`. The layout L1 is restored with `\restoregeometry`.

```
\usepackage[hmargin=3cm]{geometry}
\begin{document}
  Layout L1

\newgeometry{left=3cm,right=1cm,bottom=0.1cm}

  Layout L2 (new)

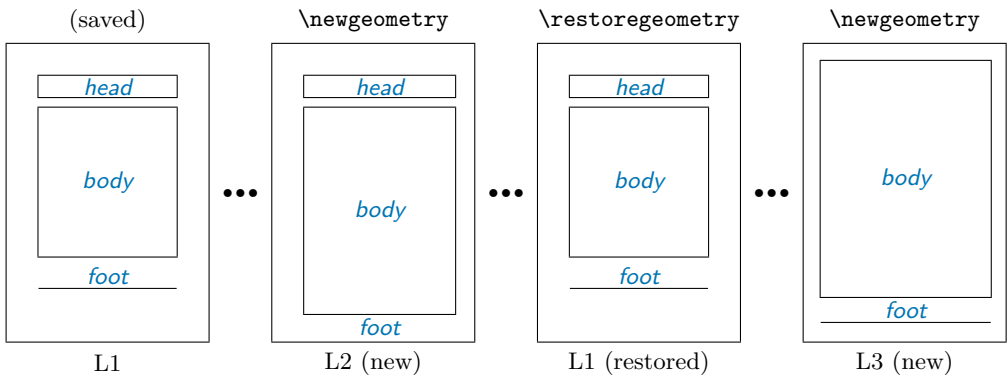
\restoregeometry

  Layout L1 (restored)

\newgeometry{margin=1cm,includefoot}

  Layout L3 (new)

\end{document}
```



“left=3cm,width=13cm,right=4cm” 等。由于 geometry 基本上优先考虑 *边距*，如果尺寸被超指定，geometry 将忽略并重新设置 *正文*。例如，如果你指定

```
\usepackage[a4paper,left=3cm,width=13cm,right=4cm]{geometry},
```

width 将被重置为 14cm，因为 A4 纸的宽度是 21cm。

在文档中间改变页面布局

版本 5 提供了新的命令 `\newgeometry...` 和 `\restoregeometry`，允许您在文档中间更改页面尺寸。与导言区的 `\geometry` 命令不同，`\newgeometry` 只能在 `\begin{document}` 之后使用，并且会重置所有以前指定的选项，但保留与纸张尺寸相关的选项：`landscape`、`portrait` 和纸张大小选项（如 `papersize`、`paper=a4paper` 等）。这些选项不能使用 `\newgeometry` 进行更改。命令 `\restoregeometry` 会恢复在导言区（`\begin{document}` 之前）使用 `\usepackage{geometry}` 和 `\geometry` 指定的页面布局。

请注意，`\newgeometry` 和 `\restoregeometry` 命令都会在调用它们的位置插入 `\clearpage`，因此它们会开始一个新的页面。以下是一个在文档中间改变布局的示例。布局 L1 使用 `hmargin=3cm`（left 和 right 边距为 3cm）指定，然后将其更改为 L2，使用 `left=3cm`、`right=1cm` 和 `bottom=0.1cm`。然后使用 `\restoregeometry` 恢复布局 L1。

```
\usepackage[hmargin=3cm]{geometry}
\begin{document}
  Layout L1

\newgeometry{left=3cm,right=1cm,bottom=0.1cm}

  Layout L2 (new)

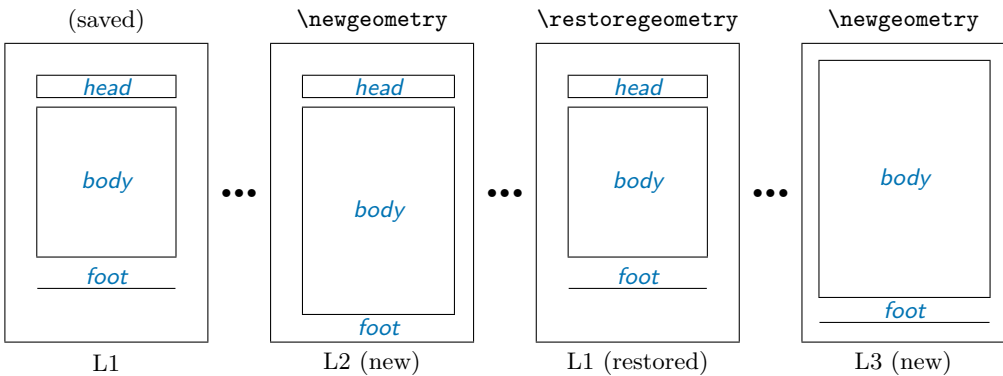
\restoregeometry

  Layout L1 (restored)

\newgeometry{margin=1cm,includefoot}

  Layout L3 (new)

\end{document}
```



A set of commands `\savegeometry{⟨name⟩}` and `\loadgeometry{⟨name⟩}` is handy if you want to reuse more different layouts in your document. For example,

```
\usepackage[hmargin=3cm]{geometry}
\begin{document}
    L1
\newgeometry{left=3cm,right=1cm,bottom=0.1cm}
\savegeometry{L2}
    L2 (new, saved)
\restoregeometry
    L1 (restored)
\newgeometry{margin=1cm,includefoot}
    L3 (new)
\loadgeometry{L2}
    L2 (loaded)
\end{document}
```

如果您想在文档中重用多个不同的布局,一组命令`\savegeometry{⟨name⟩}`和`\loadgeometry{⟨name⟩}`会很方便。例如:

```
\usepackage[hmargin=3cm]{geometry}
\begin{document}
    L1
\newgeometry{left=3cm,right=1cm,bottom=0.1cm}
\savegeometry{L2}
    L2 (new, saved)
\restoregeometry
    L1 (restored)
\newgeometry{margin=1cm,includefoot}
    L3 (new)
\loadgeometry{L2}
    L2 (loaded)
\end{document}
```

7 Examples

示例

1. A onesided page layout with the text area centered in the paper. The examples below have the same result because the horizontal margin ratio is set 1:1 for oneside by default.

- centering
- marginratio=1:1
- vcentering

2. A twosided page layout with the inside offset for binding set to 1cm.

- twoside, bindingoffset=1cm

In this case, `textwidth` is shorter than that of the default twosided document by $0.7 \times 1\text{cm}$ ($= 0.7\text{cm}$) because the default width of *body* is set with `scale=0.7`, which means $\text{width} = 0.7 \times \text{layoutwidth}$ ($= 0.7 \times \text{paperwidth}$ by default).

3. A layout with the left, right, and top margin 3cm, 2cm and 2.5in respectively, with `textheight` of 40 lines, and with the head and foot of the page included in *total body*. The two examples below have the same result.

- left=3cm, right=2cm, lines=40, top=2.5in, includeheadfoot
- hmargin={3cm,2cm}, tmargin=2.5in, lines=40, includeheadfoot

4. A layout with the height of *total body* 10in, the bottom margin 2cm, and the default width. The top margin will be calculated automatically. Each solution below results in the same page layout.

- vdivide={*, 10in, 2cm}
- bmargin=2cm, height=10in
- bottom=2cm, textheight=10in

1. 单页布局, 文本区域在纸张中央。以下示例的结果相同, 因为默认情况下, 单页的水平边距比例设置为 1:1。

- centering
- marginratio=1:1
- vcentering

2. 双页布局, 内部偏移量设置为 1cm。

- twoside, bindingoffset=1cm

在这种情况下, `textwidth` 比默认的双页文档短了 $0.7 \times 1\text{cm}$ ($= 0.7\text{cm}$), 因为 *body* 的默认宽度设置为 `scale=0.7`, 意味着 $\text{width} = 0.7 \times \text{layoutwidth}$ (默认为 $= 0.7 \times \text{paperwidth}$)。

3. 左、右和上边距分别设置为 3cm、2cm 和 2.5in, 文本高度设置为 40 行, 页眉和页脚包括在 *total body* 中。以下两个示例的结果相同。

- left=3cm, right=2cm, lines=40, top=2.5in, includeheadfoot
- hmargin={3cm,2cm}, tmargin=2.5in, lines=40, includeheadfoot

4. 高度设置为 *total body* 的高度为 10in, 底边距设置为 2cm, 默认宽度。顶边距将自动计算。以下两个解决方案的页面布局相同。

- vdivide={*, 10in, 2cm}
- bmargin=2cm, height=10in
- bottom=2cm, textheight=10in

Note that dimensions for *head* and *foot* are excluded from *height* of *total body*. An additional `includefoot` makes `\footskip` included in `totalheight`. Therefore, in the two cases below, `textheight` in the former layout is shorter than the latter (with 10in exactly) by `\footskip`. In other words, `height = textheight + footskip` when `includefoot=true` in this case.

- `bmargin=2cm, height=10in, includefoot`
- `bottom=2cm, textheight=10in, includefoot`

5. A layout with `textwidth` and `textheight` 90% of the paper and with *body* centered. Each solution below results in the same page layout as long as `layoutwidth` and `layoutheight` are not modified from the default.

- `scale=0.9, centering`
- `text={.9\paperwidth,.9\paperheight}, ratio=1:1`
- `width=.9\paperwidth, vmargin=.05\paperheight, marginratio=1:1`
- `hdivide={*,0.9\paperwidth,*}, vdivide={*,0.9\paperheight,*}` (as for onesided documents)
- `margin={0.05\paperwidth,0.05\paperheight}`

You can add `heightrounded` to avoid an “underfull vbox warning” like

```
Underfull \vbox (badness 10000) has occurred while \output is active.
```

See Section 4.3 for the detailed description about `heightrounded`.

6. A layout with the width of marginal notes set to 3cm and included in the width of *total body*. The following examples are the same.

- `marginparwidth=3cm, includemp`
- `marginpar=3cm, ignoremp=false`

7. A layout where *body* occupies the whole paper with A5 paper in landscape. The following examples are the same.

- `a5paper, landscape, scale=1.0`
- `landscape=TRUE, paper=a5paper, margin=0pt`

8. A screen size layout appropriate for presentation with PC and video projector.

```
\documentclass{slide}
\usepackage[screen,margin=0.8in]{geometry}
...
\begin{slide}
...
\end{slide}
```

9. A layout with fonts and spaces both enlarged from A4 to A3. In the case below, the resulting paper size is A3.

- `a4paper, mag=1414.`

请注意, *head*和*foot*的尺寸不包括在*total body*的高度中。添加 `includefoot` 会将 `\footskip` 包含在 `totalheight` 中。因此, 在下面的两种情况下, 前一种布局中的 `textheight` 比后一种布局中的 `textheight` (确切为 10in) 要短 `\footskip`。换句话说, 当在这种情况下 `includefoot=true` 时, `height = textheight + footskip`。

- `bmargin=2cm, height=10in, includefoot`
- `bottom=2cm, textheight=10in, includefoot`

5. 宽度为纸张宽度的 90%, 高度为纸张高度的 90%, *body* 居中。只要 `layoutwidth` 和 `layoutheight` 没有修改, 默认情况下, 以下解决方案会得到相同的页面布局。

- `scale=0.9, centering`
- `text={.9\paperwidth,.9\paperheight}, ratio=1:1`
- `width=.9\paperwidth, vmargin=.05\paperheight, marginratio=1:1`
- `hdivide={*,0.9\paperwidth,*}, vdivide={*,0.9\paperheight,*}` (as for onesided documents)
- `margin={0.05\paperwidth,0.05\paperheight}`

您可以添加 `heightrounded` 来避免出现“未填满的 vbox 警告”, 例如:

```
Underfull \vbox (badness 10000) has occurred while \output is active.
```

有关 `heightrounded` 的详细描述, 请参阅第 4.3 节。

6. 一个布局, 边注的宽度设置为 3cm, 并包含在 *total body* 的宽度中。以下示例相同。

- `marginparwidth=3cm, includemp`
- `marginpar=3cm, ignoremp=false`

7. 一个布局, 其中 *body* 占据整个纸张, 使用横向的 A5 纸张。以下示例相同。

- `a5paper, landscape, scale=1.0`
- `landscape=TRUE, paper=a5paper, margin=0pt`

8. 适合使用个人电脑和投影仪进行演示的屏幕尺寸布局。

```
\documentclass{slide}
\usepackage[screen,margin=0.8in]{geometry}
...
\begin{slide}
...
\end{slide}
```

9. 从 A4 到 A3 的布局, 字体和间距都放大。在下面的示例中, 生成的纸张尺寸为 A3。

- `a4paper, mag=1414.`

If you want to have a layout with two times bigger fonts, but without changing paper size, you can type

- `letterpaper, mag=2000, truedimen`.

You can add `dvips` option, that is useful to preview it with proper paper size by `dviout` or `xdvi`.

10. Changing the layout of the first page and leaving the others as default before loading `geometry`. Use `pass` option, `\newgeometry` and `\restoregeometry`.

```
\documentclass{book}
\usepackage[pass]{geometry}
% 'pass' disregards the package layout,
% so the original 'book' layout is memorized here.
\begin{document}
\newgeometry{margin=1cm}% changes the first page dimensions.
Page 1
\restoregeometry % restores the original 'book' layout.
Page 2 and more
\end{document}
```

11. A complex page layout.

```
\usepackage[a5paper, landscape, twocolumn, twoside,
left=2cm, hmarginratio=2:1, includemp, marginparwidth=43pt,
bottom=1cm, foot=.7cm, includefoot, textheight=11cm, heightrounded,
columnsep=1cm, dvips, verbose]{geometry}
```

Try typesetting it and checking out the result yourself. :-)

8 Known problems

- With `mag` \neq 1000 and `truedimen`, `paperwidth` and `paperheight` shown in verbose mode are different from the real size of the resulted PDF. The PDF itself is correct anyway.
- With `mag` \neq 1000, `no truedimen` and `hyperref`, `hyperref` should be loaded before `geometry`. Otherwise the resulted PDF size will become wrong.
- With `crop` package and `mag` \neq 1000, `center` option of `crop` doesn't work well.

如果您想要一个字体大小增大两倍的布局, 但是不改变纸张尺寸, 您可以输入:

- `letterpaper, mag=2000, truedimen`.

您可以添加 `dvips` 选项, 以便使用 `dviout` 或 `xdvi` 以正确的纸张尺寸预览。

10. 在加载 `geometry` 之前,更改第一页的布局,并保留其他页的默认布局。使用 `pass` 选项, `\newgeometry` 和 `\restoregeometry`。

```
\documentclass{book}
\usepackage[pass]{geometry}
% 'pass' disregards the package layout,
% so the original 'book' layout is memorized here.
\begin{document}
\newgeometry{margin=1cm}% changes the first page dimensions.
Page 1
\restoregeometry % restores the original 'book' layout.
Page 2 and more
\end{document}
```

11. 一个复杂的页面布局。

```
\usepackage[a5paper, landscape, twocolumn, twoside,
left=2cm, hmarginratio=2:1, includemp, marginparwidth=43pt,
bottom=1cm, foot=.7cm, includefoot, textheight=11cm, heightrounded,
columnsep=1cm, dvips, verbose]{geometry}
```

尝试排版并检查结果。:-)

已知问题

- 当 `mag` \neq 1000 且 `truedimen` 时, 在详细模式下显示的 `paperwidth` 和 `paperheight` 与生成的 PDF 的实际尺寸不同。但是生成的 PDF 本身是正确的。
- 当 `mag` \neq 1000, 没有 `truedimen` 和 `hyperref` 时, 应在加载 `geometry` 之前加载 `hyperref`。否则生成的 PDF 尺寸将错误。
- 使用 `crop` 宏包和 `mag` \neq 1000 时, `crop` 的 `center` 选项无法正常工作。