

L'extension geometry

Hideo Umeki
latexgeometry@gmail.com

2010/09/12 v5.6

Résumé

Cette extension met à disposition une interface flexible et simple pour traiter les dimensions des pages. Vous pouvez ainsi changer la mise en page avec des paramètres intuitifs. Par exemple, si vous voulez fixer une marge de 2 cm pour chaque bord de la feuille, vous pouvez juste taper `\usepackage[margin=2cm]{geometry}`. L'aspect de la page peut être changé en cours de document avec la commande `\newgeometry`.

1 Préface à la version 5

- **Changement de la mise en page en cours de document.**
Les nouvelles commandes `\newgeometry{...}` et `\restoregeometry` permettent à l'utilisateur de changer les dimensions de la page en cours de document. `\newgeometry` est assez similaire à `\geometry`, à ceci près que `\newgeometry` désactive toutes les options indiquées dans le préambule mais conserve les options liées à la taille du papier : `landscape`, `portrait` et les options de format (telles que `papersize`, `paper=a4paper` et ainsi de suite).
- **Un nouvel ensemble d'options pour spécifier la zone de mise en page.**
Les options spécifiant la zone dans laquelle sont calculées les dimensions de la page sont ajoutées : `layout`, `layoutsizex`, `layoutwidth`, `layoutheight` et ainsi de suite. Ces options aident à imprimer la mise en forme pour une autre taille de papier. Par exemple, avec `a4paper` et `layout=a5paper`, l'extension `geometry` utilise la mise en forme A5 pour calculer les marges avec un format de papier A4.
- **Une nouvelle option de pilote xetex.**
Une nouvelle option de pilote, `xetex`, est ajoutée. La routine d'auto-détection des pilotes a été revue pour éviter une erreur avec les commandes non définies. Notez que « `geometry.cfg` » (dans T_EX Live), qui désactive la routine d'auto-détection et sélectionne `pdftex`, n'est plus nécessaire et ne crée plus de problème quand bien même il existe toujours. Sélectionner `xetex` est fortement recommandé avec X_YL^AT_EX.
- **Nouveaux formats de papier prédéfinis pour les formats JIS B et les formats ISO C**
Les formats de papier prédéfinis `b0j` à `b6j` pour les formats JIS (Japanese Industrial Standards) B et `c0paper` à `c6paper` pour les formats ISO C (v5.4~) sont ajoutés.
- **Changement des valeurs par défaut pour les marges sous-définies.**
Dans les versions précédentes, si une seule marge était spécifiée, `bottom=1cm` par exemple, alors `geometry` fixait l'autre marge avec le ratio de marge (1:1 par défaut pour les dimensions verticales) et obtenait `top=1cm` dans ce cas. La version 5 fixe la taille du corps du texte avec la valeur par défaut `scale` (= 0.7) et détermine la marge non spécifiée. (Voir Section 6.5)
- **Les options `showframe` et `showcrop` fonctionnent sur chaque page.**
Avec l'option `showframe`, le cadre de la page est affiché sur chaque page. En complément, une nouvelle option `showcrop` affiche les traits de coupe à chaque coin de la zone de mise en page sur chaque page. Notez que les marques seront invisibles sans spécification d'une taille de mise en page plus petite que la taille du papier. La version 5.4 introduit un nouveau processus de débordement `\shipout` utilisant l'extension `atbegshi` impliquant le chargement de cette extension quand les options `showframe` ou `showcrop` sont sélectionnées.
- **Le chargement de `geometry.cfg` précède le traitement des options de classe.**
La version antérieure chargeait `geometry.cfg` après avoir traité les options de classe du document. Maintenant que le fichier de configuration est chargé avant, vous pouvez changer le comportement indiqué dans `geometry.cfg` en ajoutant des options dans `\documentclass` aussi bien que dans `\usepackage` et `\geometry`.
- **Options supprimées : `compat2` et `twosideshift`.** La version 5 n'est plus compatible avec ces options `compat2` et `twosideshift` pour des raisons de simplification.

2 Introduction

Fixer des dimensions pour une mise en page sous L^AT_EX n'est pas immédiat. Vous devez ajuster plusieurs dimensions inhérentes à L^AT_EX pour placer une zone de texte là où vous le souhaitez. Si vous voulez centrer la zone de texte dans un format de papier que vous utilisez, vous devez par exemple spécifier les dimensions fondamentales suivantes :

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

Sans l'extension `calc`, l'exemple ci-dessus demanderait des réglages plus fastidieux. L'extension `geometry` présente une façon simple de fixer des paramètres de mise en page. Dans l'exemple, ce que vous avez à faire est juste

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

En plus du centrage, le réglage des marges par rapport à chaque bord de la feuille est également problématique. Mais `geometry` simplifie aussi ce point. Si vous souhaitez fixer chaque marge à 1,8 cm, vous pouvez saisir

```
\usepackage[margin=3.8cm]{geometry}
```

De fait, l'extension `geometry` dispose d'un mécanisme d'autocomplétion dans lequel les dimensions manquantes sont automatiquement déterminées. L'extension `geometry` sera également utile lorsque vous avez à traiter la mise en page en suivant des instructions strictes. Par exemple,

La largeur maximale de la zone de texte est de 16,5 cm de large par 22,25 cm de haut. La marge haute sur chaque page doit être de 3 cm depuis le bord haut de la feuille. La marge gauche doit être de 2,3 cm depuis le bord gauche. Le pied de page avec le numéro de la page doit être en bas de la zone de texte.

Dans ce cas, avec `geometry` vous saisissez

```
\usepackage[total={16.5cm,22.25cm},
top=3cm, left=2.3cm, includefoot]{geometry}.
```

Définir une zone de texte sur papier dans un système de préparation de document présente certaines analogies avec le placement d'une fenêtre dans un système d'environnement de bureau. Le nom « `geometry` » provient de l'option `-geometry` utilisée pour spécifier la taille et la localisation d'une fenêtre dans l'environnement de bureau X Window.

3 Anatomie d'une page

La Figure 1 montre les dimensions associées à la mise en page telle que définies par l'extension `geometry`. Le gabarit¹ contient un *corps intégral* (zone imprimable) et des *marges*. Le *corps intégral* se compose d'un *corps* (zone de texte) et d'éléments optionnels : un *en-tête*, un *pied de page* et une note marginale (`marginpar`). Il y a quatre marges : *gauche*, *droite*, *haute* et *basse*. Pour les documents en recto-verso, les marges horizontales devraient être appelées *intérieure* et *extérieure*.

<i>papier</i>	:	<i>corps intégral</i> et <i>marges</i>
<i>corps intégral</i>	:	<i>corps</i> (zone de texte) (<i>en-tête</i> , <i>pied de page</i> et <i>note marginale</i> optionnels)
<i>marges</i>	:	<i>gauche</i> (<i>intérieure</i>), <i>droite</i> (<i>extérieure</i>), <i>haute</i> et <i>basse</i>

Chaque marge est mesurée à partir du bord du papier associé. Par exemple, la marge gauche (marge intérieure) correspond à distance horizontale entre le bord gauche (intérieur) du papier et celui du corps total. C'est pourquoi les marges gauche et haute définies dans `geometry` diffèrent des dimensions usuelles `\leftmargin` et `\topmargin`. La taille du corps (zone de texte) peut être modifiée avec `\textwidth` et `\textheight`.

Les dimensions du papier, du corps intégral et des marges ont les relations suivantes.

$$\text{paperwidth} = \text{left} + \text{width} + \text{right} \quad (1)$$

$$\text{paperheight} = \text{top} + \text{height} + \text{bottom} \quad (2)$$

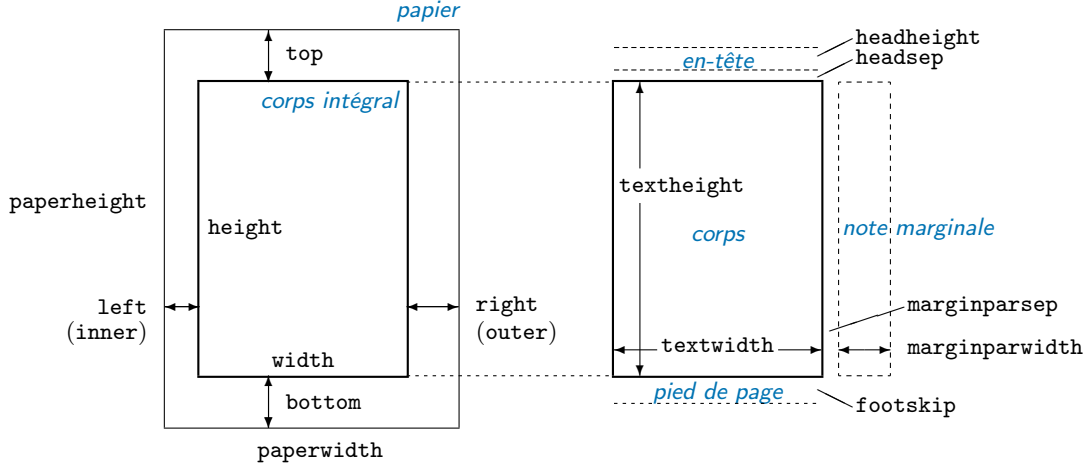


FIGURE 1 – Noms des dimensions utilisées dans l’extension `geometry`. `width` = `textwidth` et `height` = `textheight` par défaut. `left`, `right`, `top` et `bottom` sont des marges. Si les marges des pages au verso sont interverties par l’option `twoside`, les marges spécifiées par les options `left` et `right` sont utilisées pour les marges intérieures et extérieures respectivement. `inner` et `outer` sont des alias pour `left` et `right` respectivement.

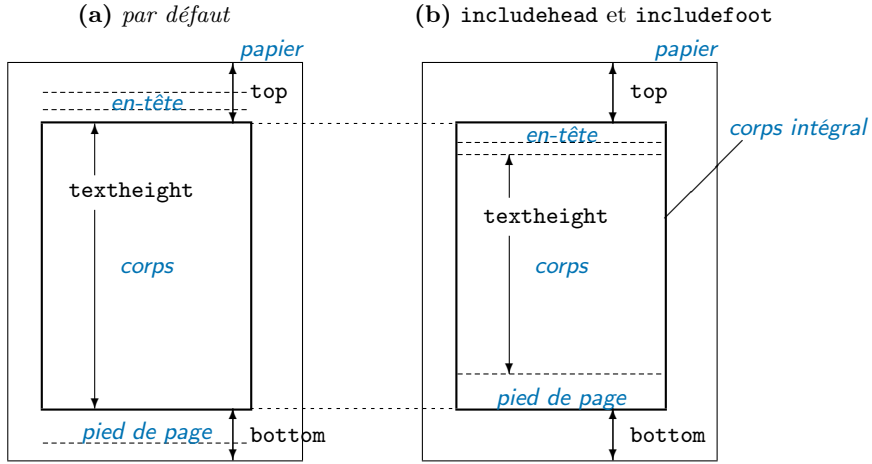


FIGURE 2 – `includehead` et `includefoot` incluent respectivement l’en-tête et le pied de page dans le `corps intégral`. (a) `height` = `textheight` (par défaut). (b) `height` = `textheight` + `headheight` + `headsep` + `footskip` si `includehead` et `includefoot` valent `true`. Si les marges haute et basse sont spécifiées, `includehead` et `includefoot` conduisent à un `textheight` plus court.

La largeur `width` et la hauteur `height` du corps intégral sont définies par :

$$\text{width} := \text{textwidth} \quad (+ \text{marginparsep} + \text{marginparwidth}) \quad (3)$$

$$\text{height} := \text{textheight} \quad (+ \text{headheight} + \text{headsep} + \text{footskip}) \quad (4)$$

Dans l’Équation (3), `width` égale `textwidth` par défaut, tandis que `marginparsep` et `marginparwidth` sont inclus dans `width` si l’option `includemp` est fixée à `true`. Dans l’Équation (4), `height` égale `textheight` par défaut. Si `includehead` est fixée à `true`, `headheight` et `headsep` sont considérées comme des parties de `height`. De la même manière, `includefoot` place `footskip` dans `height`. La Figure 2 montre comment ces options fonctionnent verticalement.

Ainsi, le gabarit se décompose en trois parties (longueurs) dans chaque direction : un corps et deux marges. Si deux d’entre elles sont explicitement spécifiées, l’autre longueur est alors évidente et n’a pas besoin d’être spécifiée. La Figure 3 montre un modèle simple de dimensions d’une page. Quand une longueur L est donnée et est décomposée entre le corps b , les marges a et c , il est évident que

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

1. Les noms des fonctions présentées ci-après reprenant parfois les appellations anglaises, le terme gabarit se traduit par *layout*, celui de corps intégral se traduit par *total body*, celui d’en-tête par *head* (ou *header*), celui de pied de page par *foot* (ou *footer*), celui de marge par *margin*, celui de note marginale par *marginal note*.

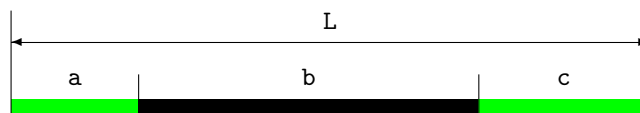


FIGURE 3 – Un modèle simple des dimensions de la page.

Une spécification avec deux des trois longueurs (a , b et c) fixées explicitement est résolvable. Si deux longueurs ou plus sont laissées non spécifiées ou « sous-spécifiées », l'Équation (5) ne peut être résolue sans une autre relation entre elles. Si elles sont toutes spécifiées autrement dit trop spécifiées ou « sur-spécifiées », alors nous devons vérifier si elles satisfont ou pas l'Équation (5).

L'extension `geometry` dispose d'un mécanisme d'autocomplétion qui évite les problèmes liés à la spécification des dimensions de la mise en page. Par exemple, vous pouvez retenir

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

sur papier A4. Dans ce cas, nous n'avez pas à définir la marge droite. Les détails de cette autocomplétion sont décrits dans la Section 6.5.

4 Interface utilisateur

4.1 Commandes

L'extension `geometry` propose les commandes suivantes :

- `\geometry{<options>}`
- `\newgeometry{<options>}` et `\restoregeometry`
- `\savegeometry{<nom>}` et `\loadgeometry{<nom>}`

`\geometry{<options>}` change la mise en page selon les options spécifiées en argument. Cette commande, plus que toute autre, devrait être placée uniquement dans le préambule (avant `\begin{document}`).

L'extension `geometry` peut être utilisée comme une partie d'une classe ou d'une autre extension que vous utilisez dans votre document. La commande `\geometry` peut écraser certains réglages dans l'en-tête. L'utilisation de plusieurs `\geometry` est autorisée et conduit à un traitement de l'ensemble des options concaténées. Si `geometry` n'est pas encore chargé, vous pouvez seulement utiliser `\usepackage[<options>]{geometry}` au lieu de `\geometry`.

`\newgeometry{<options>}` change la mise en page en cours de document. `\newgeometry` est similaire à `\geometry` à ceci près que `\newgeometry` annule toutes les options spécifiées par `\usepackage` et `\geometry` en préambule et ne tient pas compte des options liées à la taille du papier. `\restoregeometry` restaure la mise en page spécifiée dans le préambule. Cette commande n'a pas d'argument. Voir la Section 7 pour plus d'informations.

`\savegeometry{<nom>}` sauvegarde sous le nom `<nom>` les dimensions de la page à l'endroit où la commande est placée. `\loadgeometry{<nom>}` charge les dimensions de la page sauvegardées sous le nom `<nom>`. Voir la Section 7 pour plus d'informations.

4.2 Argument optionnel

L'extension `geometry` adopte l'interface `keyval` « `<clé>=<valeur>` » pour l'argument optionnel de `\usepackage`, `\geometry` et `\newgeometry`.

L'argument inclut une liste d'options `keyval` séparées par des virgules et a les règles de base suivantes :

- des lignes multiples sont autorisées mais pas les lignes vides ;
- les espaces entre les mots sont ignorés ;
- les options sont traitées indépendamment de leur ordre de saisie. (Il y a ici quelques exceptions.

Voir la Section 6.2 pour plus d'informations.)

Par exemple,

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

est équivalent à

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

Certaines options sont autorisées à présenter une sous-liste, par exemple `{3cm,0.8in}`. Notez que l'ordre des valeurs dans la sous-liste est significatif. Le réglage ci-dessus est équivalent aux suivants :

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

ou

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

Ainsi, l'utilisation de plusieurs `\geometry` cumule juste les options.

`geometry` supporte l'extension *calc*². Par exemple,

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

4.3 Types d'option

Les options de `geometry` sont réparties en quatre types :

1. Type booléen

prend une valeur booléenne (vrai `true` ou faux `false`). En l'absence de valeur, `true` est sélectionné par défaut.

$\langle clé \rangle = \text{true} \mid \text{false}.$
 $\langle clé \rangle$ sans valeur est équivalent à $\langle clé \rangle = \text{true}.$

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

Le format (ou nom) du papier est ici l'exception. Le format du papier désiré devrait être mis dans valeur. Quelle que soit la valeur donnée, elle est ignorée. Par exemple, `a4paper=XXX` est équivalent à `a4paper`.

2. Type à valeur unique

prend une valeur obligatoire.

$\langle clé \rangle = \langle valeur \rangle.$

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

3. Type à valeur double

prend une paire de valeurs obligatoires séparées par une virgule et regroupées entre accolades. Les deux valeurs peuvent être réduites à une valeur unique si elles sont identiques.

$\langle clé \rangle = \{ \langle valeur1 \rangle, \langle valeur2 \rangle \}.$
 $\langle clé \rangle = \langle valeur \rangle$ est équivalent à $\langle clé \rangle = \{ \langle valeur \rangle, \langle valeur \rangle \}.$

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

4. Type à triple valeur

prend trois valeurs obligatoires séparées par une virgule et regroupées entre accolades.

$\langle clé \rangle = \{ \langle valeur1 \rangle, \langle valeur2 \rangle, \langle valeur3 \rangle \}$

Chaque valeur doit être une dimension ou être nulle. Quand vous indiquez une valeur vide ou « * », cela correspond à un nul et laisse la valeur souhaitée aux bons soins du mécanisme d'autocomplétion. Vous devez spécifier au moins une dimension, typiquement deux dimensions. Vous pouvez indiquer des nuls pour toutes les valeurs mais cela n'a pas de sens. *Exemples :*

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

5 Détail des options

Cette section décrit toutes les options disponibles dans `geometry`. Les options avec une dague[†] ne sont pas disponibles comme argument de `\newgeometry` (voir la Section 7).

2. CTAN : `macros/latex/required/tools`

5.1 Taille du papier

Les options ci-dessous définissent la taille du papier/médium et son orientation.

- † **paper** | **papername** spécifie la taille du papier par son format. **paper**=*<format>*. Par simplification, vous pouvez spécifier cette taille sans **paper**=. Par exemple, **a4paper** est équivalent à **paper=a4paper**.
- † **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**, **ansipaper**, **ansicpaper**, **ansicpaper**, **ansicpaper**, **ansicpaper**, **letterpaper**, **executivepaper**, **legalpaper** spécifie le format du papier. La valeur assignée à cette option est toujours ignorée. Par exemple, les cas suivants ont le même effet : **a5paper**, **a5paper=true**, **a5paper=false** et ainsi de suite. **a[0-6]paper**, **b[0-6]paper** et **c[0-6]paper** sont les formats ISO A, B et C respectivement. Les formats JIS (Japanese Industrial Standards) A sont identiques aux formats ISO A mais les formats JIS B sont différents des formats ISO B. **b[0-6]j** devrait être utilisé pour les formats JIS B.
- † **screen** un format spécial avec (L,H) = (225mm,180mm). Pour les présentations sur ordinateur ou pour projecteur, « **screen,centering** » avec la classe de document « **slide** » devrait être utile.
- † **paperwidth** largeur du papier. **paperwidth**=*<longueur>*.
- † **paperheight** hauteur du papier. **paperheight**=*<longueur>*.
- † **papersize** largeur et hauteur du papier. **papersize**=*<largeur>*,*<hauteur>* ou **papersize**=*<longueur>*.
- † **landscape** bascule l'orientation du papier en mode paysage.
- † **portrait** bascule l'orientation du papier en mode portrait. Ceci est équivalent à **landscape=false**.

Les options de format de papier (telle **a4paper**) et d'orientation (**portrait** et **landscape**) peuvent être déclarée dans les options de la classe du document. Par exemple, si vous utilisez

```
\documentclass[a4paper,landscape]{article}
```

alors **a4paper** et **landscape** seront traités dans **geometry** également. C'est également le cas pour **twoside** et **twocolumn** (voir également la Section 5.5).

5.2 Taille du gabarit

Vous pouvez spécifier le gabarit (ou zone de mise en page) avec les options décrites dans cette section indépendamment de la taille du papier. Ces options aident à imprimer un gabarit spécifique sur un papier de taille différente. Par exemple, avec **a4paper** et **layout=a5paper**, l'extension utilise le gabarit « A5 » pour calculer les marges sur un papier « A4 ». La taille du gabarit est par défaut celle du papier. Les options pour la taille du gabarit sont disponibles dans **\newgeometry**, de telle manière à ce que vous puissiez changer la taille du gabarit en cours de document. La taille du papier ne peut cependant être modifiée. La Figure 4 montre la différence entre **layout** et **paper**.

- layout** spécifie la taille du gabarit par nom de format. **layout**=*<format>*. Tous les formats de papier définis dans **geometry** sont disponibles. Voir la Section 5.1 pour plus d'informations.
- layoutwidth** largeur du gabarit. **layoutwidth**=*<longueur>*.
- layoutheight** hauteur du gabarit. **layoutheight**=*<longueur>*.
- layoutsizes** largeur et hauteur du gabarit. **layoutsizes**=*<largeur>*,*<hauteur>* ou **layoutsizes**=*<longueur>*.
- layouthoffset** spécifie le décalage horizontal par rapport au bord gauche du papier. **layouthoffset**=*<longueur>*.
- layoutvoffset** spécifie le décalage vertical par rapport au bord haut du papier. **layoutvoffset**=*<longueur>*.
- layoutoffset** spécifie le décalage horizontal et le décalage vertical. **layoutoffset**=*<hoffset>*,*<voffset>* ou **layoutsizes**=*<longueur>*.

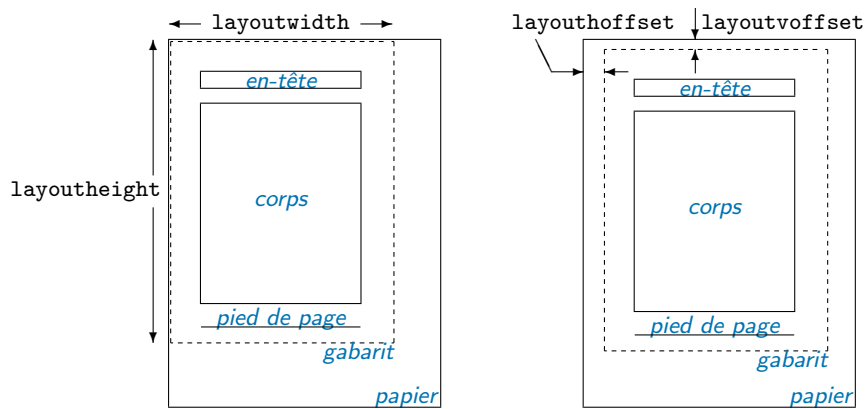


FIGURE 4 – Les dimensions liées à la taille du gabarit. Notez que la taille du gabarit est par défaut celle du papier, ce qui fait que vous n’avez pas à spécifier explicitement les options liées au gabarit la plupart du temps.

5.3 Taille du corps

Les options spécifiant la taille du *corps intégral* sont décrites dans cette section.

<code>hscale</code>	rapport de largeur entre le <i>corps intégral</i> et <code>\paperwidth</code> noté <code>hscale=\langle rapport-h \rangle</code> . Par exemple, <code>hscale=0.8</code> est équivalent à <code>width=0.8\paperwidth</code> . (0.7 par défaut)
<code>vscale</code>	rapport de hauteur entre le <i>corps intégral</i> et <code>\paperheight</code> noté <code>vscale=\langle rapport-v \rangle</code> . Par exemple, <code>vscale=0.9</code> est équivalent à <code>height=0.9\paperheight</code> . (0.7 par défaut)
<code>scale</code>	rapport de taille entre le <i>corps intégral</i> et le papier. <code>scale=\{\langle rapport-h \rangle, \langle rapport-v \rangle\}</code> ou <code>scale=\langle rapport \rangle</code> . (0.7 par défaut)
<code>width</code> <code>totalwidth</code>	largeur du <i>corps intégral</i> . <code>width=\langle longueur \rangle</code> ou <code>totalwidth=\langle longueur \rangle</code> . Cette dimension vaut par défaut <code>textwidth</code> , mais si <code>includemp</code> vaut <code>true</code> , <code>width ≥ textwidth</code> car <code>width</code> inclut la largeur de la note marginale. Si <code>textwidth</code> et <code>width</code> sont spécifiés en même temps, <code>textwidth</code> est prioritaire par rapport à <code>width</code> .
<code>height</code> <code>totalheight</code>	hauteur du <i>corps intégral</i> , excluant l’en-tête et le pied de page par défaut. Si <code>includehead</code> ou <code>includefoot</code> est sélectionné, <code>height</code> inclut l’en-tête ou le pied de page tout comme il contient <code>textheight</code> . <code>height=\langle longueur \rangle</code> ou <code>totalheight=\langle longueur \rangle</code> . Si <code>textheight</code> et <code>height</code> sont spécifiés en même temps, <code>height</code> sera ignoré.
<code>total</code>	largeur et hauteur du <i>corps intégral</i> . <code>total=\{\langle largeur \rangle, \langle hauteur \rangle\}</code> ou <code>total=\langle longueur \rangle</code> .
<code>textwidth</code>	spécifie <code>\textwidth</code> , la largeur du <i>corps</i> (la zone de texte). <code>textwidth=\langle longueur \rangle</code> .
<code>textheight</code>	spécifie <code>\textheight</code> , la hauteur du <i>corps</i> (la zone de texte). <code>textheight=\langle longueur \rangle</code> .
<code>text</code> <code>body</code>	spécifie à la fois <code>\textwidth</code> et <code>\textheight</code> du corps de la page. <code>body=\{\langle largeur \rangle, \langle hauteur \rangle\}</code> ou <code>text=\langle longueur \rangle</code> .
<code>lines</code>	permet à l’utilisateur de spécifier <code>\textheight</code> par un nombre de lignes. <code>lines=\langle nombre-entier \rangle</code> .
<code>includehead</code>	inclut l’en-tête de la page, <code>\headheight</code> et <code>\headsep</code> , dans le <i>corps intégral</i> . Cette option est mise à <code>false</code> par défaut. Elle s’oppose à <code>ignorehead</code> . Voir la Figure 2 et la Figure 5.
<code>includefoot</code>	inclut le pied de page, <code>\footskip</code> , dans le <i>corps intégral</i> . Cette option est mise à <code>false</code> par défaut. Elle s’oppose à <code>ignorefoot</code> . Voir la Figure 2 et la Figure 5.
<code>includeheadfoot</code>	fixe <code>includehead</code> et <code>includefoot</code> à <code>true</code> , ce qui est l’opposé de <code>ignoreheadfoot</code> . Voir la Figure 2 et la Figure 5.
<code>includemp</code>	inclut la note marginale, <code>\marginparwidth</code> and <code>\marginparsep</code> , dans le <i>corps</i> lors de calculs d’éléments horizontaux.
<code>includeall</code>	fixe <code>includeheadfoot</code> et <code>includemp</code> à <code>true</code> . Voir la Figure 5.
<code>ignorehead</code>	ne tient pas compte de l’en-tête, <code>headheight</code> et <code>headsep</code> , dans la détermination de la mise en page verticale, mais ne change pas ces longueurs. Cette option est équivalente à <code>includehead=false</code> . Elle est mise à <code>true</code> par défaut. Voir aussi <code>includehead</code> .

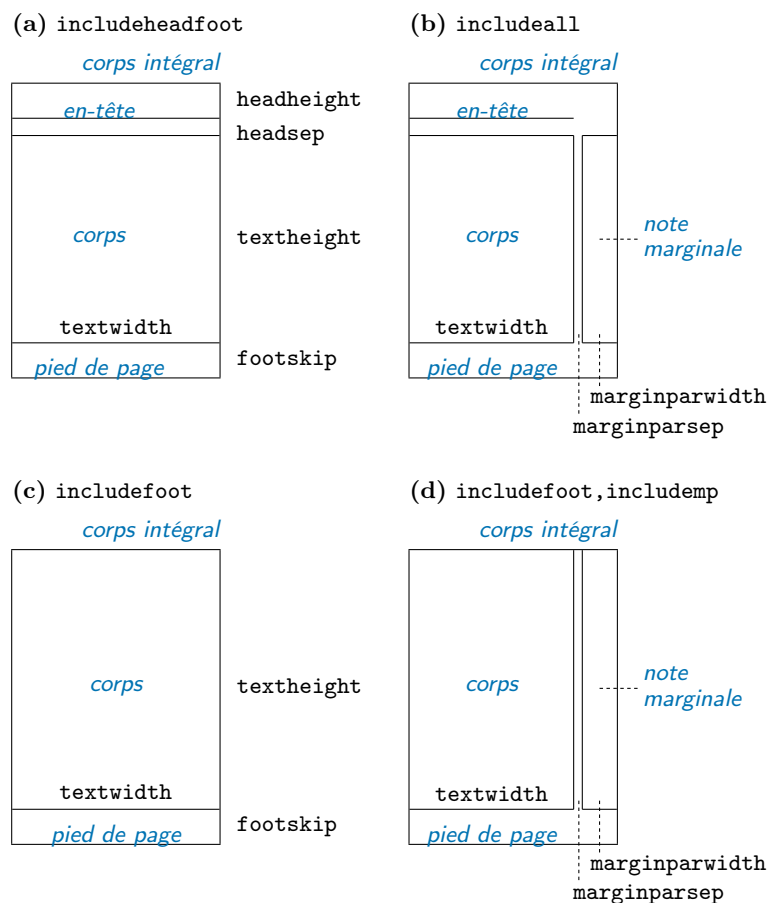


FIGURE 5 – Exemples de mise en page avec différents modes de présentation du *corps intégral* (a) `includeheadfoot`, (b) `includeall`, (c) `includefoot` et (d) `includefoot,includemp`. Si `reversemp` est mis à `true`, l'emplacement de la note marginale est interverti sur chaque page. L'option `twoside` intervertit à la fois les marges et les notes marginales sur les pages au verso. Notez que la note marginale, si elle existe, est composée malgré `ignoremp` ou `includemp=false` et dépasse de la page dans certains cas.

- `ignorefoot` ne tient pas compte du pied de page, `footskip`, dans la détermination de la mise en page verticale, mais ne change pas cette longueur. Cette option est mise à `true` par défaut. Voir aussi `includefoot`.
- `ignoreheadfoot` fixe `ignorehead` et `ignorefoot` à `true`. Voir aussi `includeheadfoot`.
- `ignoremp` ne tient pas compte de la note marginale dans la détermination des marges horizontales (la valeur par défaut étant `true`). Si la note marginale dépasse de la page, un message d'alerte sera affiché si `verbose=true`. Voir aussi `includemp` et la Figure 5.
- `ignoreall` fixe `ignoreheadfoot` et `ignoremp` à `true`. Voir aussi `includeall`.
- `heightrounded`

Cette option arrondit `\textheight` à n -times (n : un entier) de `\baselineskip` plus `\topskip` pour éviter l'erreur « *underfull vbox* » dans certains cas. Par exemple, si `\textheight` vaut 486pt avec `\baselineskip` 12pt et `\topskip` 10pt, alors

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

Ici, `\textheight` est au final arrondi à 490pt. `heightrounded=false` par défaut.

La Figure 5 illustre différents modes de mise en page du corps intégral. Les dimensions pour l'en-tête et le pied de page peuvent être contrôlées par les modes `nohead` or `nofoot` mode, qui fixe chaque longueur à 0pt directement. Par ailleurs, les options avec le préfixe `ignore` ne changent pas les dimensions de base correspondantes.

Les options suivantes peuvent spécifier le corps et les marges simultanément avec, entre accolages, trois valeurs séparées par des virgules.

- `hdivide` partition horizontale (marge gauche, largeur, marge droite).
`hdivide={\langle marge-gauche \rangle, \langle largeur \rangle, \langle marge-droite \rangle}`. Notez que vous n'avez pas besoin

de spécifier l'ensemble des trois paramètres. La meilleure façon d'utiliser cette option est de spécifier deux paramètres seulement en laissant le dernier vide ou renseigné à « * ». Par exemple, quand vous saisissez `hdivide={2cm,15cm, }`, la marge partant du bord droit de la page sera déterminée par le calcul de `paperwidth-2cm-15cm`.

`vdivide` partition verticale (marge haute, hauteur, marge basse).
`vdivide={⟨marge-haute⟩,⟨hauteur⟩,⟨marge-basse⟩}`.
`divide` `divide={A,B,C}` est interprété comme `hdivide={A,B,C}` et `vdivide={A,B,C}`.

5.4 Taille des marges

Les options spécifiant la taille des marges sont listées ci-dessous.

<code>left</code> <code>lmargin</code> <code>inner</code>	marge gauche (pour du recto uniquement en mode <i>oneside</i>) ou marge intérieure (pour du recto-verso en mode <i>twoside</i>) du <i>corps intégral</i> . En d'autres termes, la distance entre le bord gauche (intérieur) du papier et celui du <i>corps intégral</i> . <code>left=⟨longueur⟩</code> . <code>inner</code> n'a pas de sens particulier, n'étant qu'un alias de <code>left</code> et <code>lmargin</code> .
<code>right</code> <code>rmargin</code> <code>outer</code>	marge droite ou extérieure du <i>corps intégral</i> . <code>right=⟨longueur⟩</code> .
<code>top</code> <code>tmargin</code>	marge haute de la page. <code>top=⟨longueur⟩</code> . Notez que cette option n'a rien à voir avec la dimensions de base <code>\topmargin</code> .
<code>bottom</code> <code>bmargin</code>	marge basse de la page. <code>bottom=⟨longueur⟩</code> .
<code>hmargin</code>	marges gauche et droite. <code>hmargin={⟨marge gauche⟩,⟨marge droite⟩}</code> ou <code>hmargin=⟨longueur⟩</code> .
<code>vmargin</code>	marges haute et basse. <code>vmargin={⟨marge haute⟩,⟨marge basse⟩}</code> ou <code>vmargin=⟨longueur⟩</code> .
<code>margin</code>	<code>margin={A,B}</code> est équivalent à <code>hmargin={A,B}</code> et <code>vmargin={A,B}</code> . <code>margin=A</code> est automatiquement développé en <code>hmargin=A</code> et <code>vmargin=A</code> .
<code>hmarginratio</code>	rapport entre la marge horizontale de gauche (intérieure) et celle de droite (extérieure). La valeur du <code>⟨rapport⟩</code> devrait être spécifiée par deux valeurs séparées par deux points. Chaque valeur devrait être un entier positif inférieur à 100 pour éviter les dépassements de calculs arithmétiques, par exemple 2:3 au lieu de 1:1.5. Le rapport par défaut est 1:1 pour du recto uniquement, 2:3 pour du recto-verso.
<code>vmarginratio</code>	rapport entre les marges verticales <code>top</code> et <code>bottom</code> . Le rapport par défaut est 2:3.
<code>marginratio</code> <code>ratio</code>	rapport pour les marges horizontales et verticales. <code>marginratio={⟨rapport horizontal⟩,⟨rapport vertical⟩}</code> ou <code>marginratio=⟨rapport⟩</code> .
<code>hcentering</code>	fixe un centrage horizontal automatique et est équivalent à <code>hmarginratio=1:1</code> . Cette option est mise à <code>true</code> par défaut pour du recto uniquement (<i>oneside</i>). Voir aussi <code>hmarginratio</code> .
<code>vcentering</code>	fixe un centrage vertical automatique et est équivalent à <code>vmarginratio=1:1</code> . La valeur par défaut est <code>false</code> . Voir aussi <code>vmarginratio</code> .
<code>centering</code>	applique un centrage automatique horizontalement et verticalement et est équivalent à <code>marginratio=1:1</code> . Voir aussi <code>marginratio</code> . La valeur par défaut est <code>false</code> . Voir aussi <code>marginratio</code> .
<code>twoside</code>	bascule en mode recto-verso avec les marges gauche et droite interverties sur les pages au verso. L'option active les bascules <code>\@twoside</code> et <code>\@mparswitch</code> . Voir aussi <code>asymmetric</code> .
<code>asymmetric</code>	implémente une mise en page recto-verso dans laquelle les marges ne sont pas interverties pour chaque verso (en définissant <code>\oddsidemargin</code> comme <code>\evensidemargin + bindingoffset</code>) et dans laquelle les notes marginales restent toujours du même côté. Cette peut être utilisée comme une alternative à l'option <code>twoside</code> . Voir aussi <code>twoside</code> .
<code>bindingoffset</code>	retranche un espace défini du côté gauche de la page en recto uniquement ou du côté intérieur en recto-verso. <code>bindingoffset=⟨longueur⟩</code> . Ceci est utile si les pages sont traitées avec une méthode de reliure (qu'elles soient collées, cousues, agrafées ...) en permettant de définir la zone perdue du fait de la reliure. Voir Figure 6.
<code>hdivide</code>	Voir la description en Section 5.3.
<code>vdivide</code>	Voir la description en Section 5.3.
<code>divide</code>	Voir la description en Section 5.3.

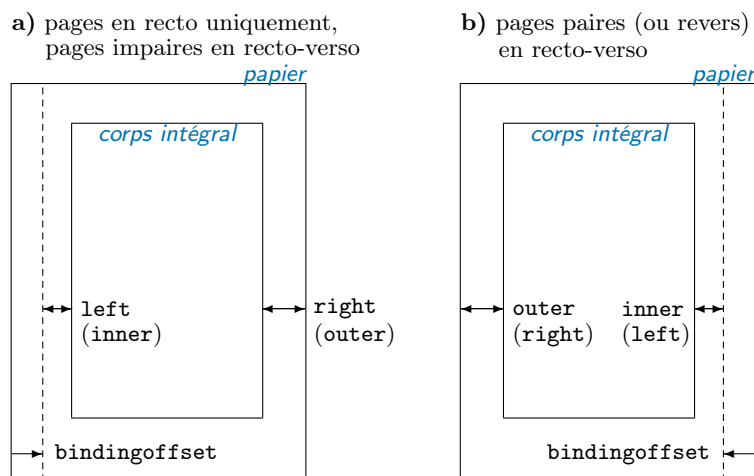


FIGURE 6 – L’option `bindingoffset` ajoute un espace de largeur spécifiée à la marge intérieure. Notez que l’option `twoside` intervertit les marges horizontales et les notes marginales avec l’effet de `bindingoffset` sur les pages paires (voir **b**)), mais l’option `asymmetric` supprime cette intervention des marges et des notes marginales (mais `bindingoffset` reste toujours interverti puisqu’il est lié à une question de reliure des pages).

5.5 Dimensions fondamentales

Les options ci-dessous écrasent les dimensions fondamentales de L^AT_EX et autres bascules associées à la mise en page (Voir le côté droit de la Figure 1).

<code>headheight</code>	<code>head</code>	modifie <code>\headheight</code> , hauteur de l’en-tête. <code>headheight=<longueur></code> ou <code>head=<longueur></code> .
<code>headsep</code>		modifie <code>\headsep</code> , espace séparant l’en-tête et le texte (corps). <code>headsep=<longueur></code> .
<code>footskip</code>	<code>foot</code>	modifie <code>\footskip</code> , distance séparant la ligne de base de la dernière ligne de texte et la ligne de base du pied de page. <code>footskip=<longueur></code> ou <code>foot=<longueur></code> .
<code>nohead</code>		supprime l’espace pour l’en-tête, ce qui est équivalent à la fois à <code>\headheight=0pt</code> et <code>\headsep=0pt</code> .
<code>nofoot</code>		supprime l’espace pour le pied de page, ce qui est équivalent à <code>\footskip=0pt</code> .
<code>noheadfoot</code>		équivalent à <code>nohead</code> et <code>nofoot</code> , ce qui signifie que <code>\headheight</code> , <code>\headsep</code> et <code>\footskip</code> sont tous fixés à <code>0pt</code> .
<code>footnotesep</code>		change la dimension <code>\skip\footins</code> , espace séparant entre le bas du corps du texte et le haut du texte des notes de bas de page.
<code>marginparwidth</code>	<code>marginpar</code>	modifie <code>\marginparwidth</code> , largeur des notes marginales. <code>marginparwidth=<longueur></code> .
<code>marginparsep</code>		modifie <code>\marginparsep</code> , espace séparant le corps et les notes marginales. <code>marginparsep=<longueur></code> .
<code>nomarginpar</code>		réduit l’espace des notes marginales à <code>0pt</code> , ce qui équivaut à <code>\marginparwidth=0pt</code> et <code>\marginparsep=0pt</code> .
<code>columnsep</code>		modifie <code>\columnsep</code> , espace séparant les deux colonnes du mode <code>twocolumn</code> .
<code>hoffset</code>		modifie <code>\décalage-h</code> . <code>hoffset=<longueur></code> .
<code>voffset</code>		modifie <code>\décalage-v</code> . <code>voffset=<longueur></code> .
<code>offset</code>		décalage horizontal et vertical. <code>offset={<décalage-h>,<décalage-v>}</code> ou <code>offset=<longueur></code> .
<code>twocolumn</code>		fait passer en mode <code>twocolumn</code> avec <code>\@twocolumntrue</code> . <code>twocolumn=false</code> revient à un mode à une colonne avec <code>\@twocolumnfalse</code> . Au lieu de <code>twocolumn=false</code> , vous pouvez spécifier <code>onecolumn</code> (ce qui vaut par défaut <code>true</code>).
<code>onecolumn</code>		fonctionne comme <code>twocolumn=false</code> . <code>onecolumn=false</code> est équivalent à <code>twocolumn</code> .
<code>twoside</code>		fixe à la fois <code>\@twosidetrue</code> et <code>\@mparswitchtrue</code> . Voir Section 5.4.
<code>textwidth</code>		définit <code>\textwidth</code> directement. Voir Section 5.3.
<code>textheight</code>		définit <code>\textheight</code> directement. Voir Section 5.3.
<code>reversemp</code>	<code>reversemarginpar</code>	place les notes marginales dans la marge gauche (intérieure) avec <code>\@reversemargintrue</code> . L’option ne change pas le mode <code>includemp</code> . Elle vaut <code>false</code> par défaut.

5.6 Pilotes

L'extension gère les pilotes `dvips`, `dvipdfm`, `pdftex`, `xetex` et `vtex`. Vous pouvez également utiliser `dvipdfm` pour `dvipdfmx` et `xdvipdfmx`. `pdftex` sert pour `pdflatex` et `vtex` pour l'environnement `VTEX`. Les options associées aux pilotes sont exclusives. Le pilote peut être sélectionné soit par `driver=<nom-du-pilote>` soit directement par le nom du pilote tel `pdftex`. Par défaut, `geometry` devine le pilote approprié au système utilisé. C'est pourquoi vous n'avez pas à sélectionner le pilote dans la plupart des cas. Cependant, si vous souhaitez utiliser `dvipdfm`, vous devriez spécifier explicitement ce pilote.

[†] <code>driver</code>	spécifie le pilote avec <code>driver=<nom-du-pilote></code> . <code>dvips</code> , <code>dvipdfm</code> , <code>pdftex</code> , <code>vtex</code> , <code>xetex</code> , <code>auto</code> et <code>none</code> sont disponibles comme nom de pilote. Les noms, exceptions faites de <code>auto</code> et <code>none</code> peuvent être spécifiés directement avec le nom sans <code>driver=</code> . <code>driver=auto</code> impose l'auto-détection quelle que soit le réglage précédent. <code>driver=none</code> empêche l'auto-détection et ne sélectionne aucun pilote, ce qui peut être utile lorsque vous voulez qu'une autre extension détermine le paramétrage du pilote. Par exemple, si vous souhaitez utiliser l'extension <code>crop</code> avec <code>geometry</code> , vous devriez saisir <code>\usepackage[driver=none]{geometry}</code> avant l'appel de l'extension <code>crop</code> package.
[†] <code>dvips</code>	inclut la dimension du papier dans la sortie dvi avec la commande <code>\special</code> . Si vous utilisez <code>dvips</code> comme un pilote DVI-vers-PS, par exemple, imprimer un document avec <code>\geometry{a3paper,landscape}</code> sur du papier A3 en orientation paysage, vous n'avez alors pas besoin des options « <code>-t a3 -t landscape</code> » avec <code>dvips</code> .
[†] <code>dvipdfm</code>	fonctionne comme <code>dvips</code> sauf pour l'orientation paysage. Vous pouvez sélectionner cette option en utilisant <code>dvipdfmx</code> et <code>xdvipdfmx</code> pour obtenir la sortie dvi.
[†] <code>pdftex</code>	fixe <code>\pdfpagewidth</code> et <code>\pdfpageheight</code> en interne.
[†] <code>xetex</code>	est équivalent à <code>pdftex</code> à ceci près qu'il ignore <code>\pdf{h,v}origin</code> non défini avec <code>X_YL_AT_EX</code> . Cette option est introduite dans la version 5. Notez que « <code>geometry.cfg</code> » dans <code>T_EX Live</code> , qui désactive la routine d'auto-détection et sélectionne <code>pdftex</code> , n'est plus nécessaire mais ne crée pas de problème quand bien même il n'est pas supprimé. Au lieu de <code>xetex</code> , vous pouvez spécifier <code>dvipdfm</code> avec <code>X_YL_AT_EX</code> si vous souhaitez utiliser les éléments spécifiques de <code>dvipdfm</code> que <code>X_YL_AT_EX</code> supporte.
[†] <code>vtex</code>	fixe les dimensions <code>\mediawidth</code> et <code>\mediaheight</code> pour <code>VT_EX</code> . Quand ce pilote est sélectionné (explicitement ou automatiquement), <code>geometry</code> détectera automatiquement quel mode de restitution (DVI, PDF ou PS) est choisi avec <code>VT_EXet</code> fera les réglages associés.

Si le pilote explicitement sélectionné n'est pas associable avec le logiciel utilisé, le pilote par défaut `dvips` sera sélectionné.

5.7 Autres options

Les autres options utiles sont décrites ici.

[†] <code>verbose</code>	affiche les résultats du paramétrage sur le terminal. <code>verbose=false</code> (défaut) les places uniquement dans le fichier journal.
[†] <code>reset</code>	remet les dimensions de la mise en page et autres bascules dans l'état où elles étaient avant le chargement de <code>geometry</code> . Les options données dans <code>geometry.cfg</code> sont également ignorées. Notez que cela ne peut traiter <code>pass</code> et <code>mage</code> avec <code>truedimen</code> . <code>reset=false</code> n'a pas d'effet et n'annule pas les éventuels <code>reset(=true)</code> précédents. Par exemple, quand vous saisissez <pre> \documentclass[landscape]{article} \usepackage[twoside,reset,left=2cm]{geometry} </pre> avec <code>\ExecuteOptions{scale=0.9}</code> dans <code>geometry.cfg</code> , alors, au final, <code>landscape</code> et <code>left=2cm</code> restent actifs et <code>scale=0.9</code> comme <code>twoside</code> sont ignorés.
[†] <code>mag</code>	fixe la valeur de l'agrandissement (<code>\mag</code>) et modifie <code>\hoffset</code> et <code>\voffset</code> en fonction de cet agrandissement. <code>mag=<valeur></code> . Notez que <code><valeur></code> devrait être une valeur entière, avec 1000 désignant la taille normale. Par exemple, <code>mag=1414</code> avec <code>a4paper</code> conduit à un imprimé agrandi tenant sur un format <code>a3paper</code> , qui est 1.414 ($=\sqrt{2}$) fois plus grand que <code>a4paper</code> . L'agrandissement des polices de caractère demande de l'espace disque supplémentaire. Notez que l'option <code>mag</code> doit précéder toute autre options avec les dimensions « réelles » telles que <code>1.5truein</code> ou <code>2truecm</code> et ainsi de suite. Voir aussi l'option <code>truedimen</code> .

[†] <code>truedimen</code>	change toute dimension interne explicite en dimensions <i>vraies</i> , par exemple <code>1cm</code> est changé en <code>1truecm</code> . Cette option s'utilise typiquement avec l'option <code>mag</code> . Notez que ceci n'a pas d'effet pour les dimensions spécifiées extérieurement. Par exemple, quand vous indiquez « <code>mag=1440, margin=10pt, truedimen</code> », les marges ne sont pas « vraies » mais agrandies. Si vous souhaitez définir des marges exactes, vous devriez utiliser quelque chose comme « <code>mag=1440, margin=10truept, truedimen</code> » à la place.
[†] <code>pass</code>	annule toutes les options et calculs de géométrie à l'exception de <code>verbose</code> et <code>showframe</code> . Cette option est insensible à l'ordre et peut être utilisée pour vérifier la mise en page donnée par la classe, d'autres extensions ou des réglages manuels sans <code>geometry</code> .
[†] <code>showframe</code>	affiche des cadres pour la zone de texte et la page ainsi que les lignes pour l'en-tête et le pied de page sur la première page.
[†] <code>showcrop</code>	affiche les traits de coupe à chaque coin de la ✖ zone de mise en page ✖ spécifiée par l'utilisateur.

6 Traitement des options

6.1 Ordre de chargement

S'il existe un fichier `geometry.cfg` quelque part où \TeX peut le trouver, `geometry` le charge en premier. Par exemple, dans `geometry.cfg` vous pouvez écrire `\ExecuteOptions{a4paper}`, ce qui spécifie le format A4 comme format par défaut. De fait, vous pouvez utiliser toutes les options définies dans `geometry` avec `with \ExecuteOptions{}`.

L'ordre de chargement dans le préambule de votre document est le suivant :

1. `geometry.cfg` s'il existe.
2. Options spécifiées avec `\documentclass[options]{...}`.
3. Options spécifiées avec `\usepackage[options]{geometry}`
4. Options spécifiées avec `\geometry{options}`, qui peut être appelé à plusieurs reprises. (L'option `reset` va annuler les options spécifiées dans `\usepackage{geometry}` ou `\geometry`.)

6.2 Ordre des options

La spécification des options `geometry` est indifférente à l'ordre et écrase la précédente pour une même option. Par exemple,

`[left=2cm, right=3cm]` est équivalent à `[right=3cm, left=2cm]`.

Les options appelées de nombreuses fois écrasent les appels précédents de la même option. Par exemple,

`[verbose=true, verbose=false]` se résume à `verbose=false`.

`[hmargin={3cm,2cm}, left=1cm]` est équivalent à `hmargin={1cm,2cm}` où la marge de gauche (ou intérieure) est écrasée par `left=1cm`.

`reset` et `mag` sont des exceptions. L'option `reset` annule toutes les options de géométrie (exceptée `pass`) indiquées avant cette option. Si vous indiquez

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

alors `margin=1cm`, `twoside` et `a5paper` sont retranchés, ce qui ramène ce cas alors à

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

L'option `mag` devrait être indiquée avant les autres options avec une longueur « vraie » telles que `left=1.5truecm`, `width=5truein` et ainsi de suite. La primitive `\mag` peut être indiquée avant que cette extension ne soit appelée.

6.3 Priorité

Il existe plusieurs manières de fixer les dimensions du `corps` : `scale`, `total`, `text` et `lines`. L'extension `geometry` donne une plus haute priorité aux spécifications les plus concrètes. Voici la règle de priorité pour le `corps`.

priorité : basse \longrightarrow élevée

$$\left\{ \begin{array}{c} \text{hscale} \\ \text{vscale} \\ \text{scale} \end{array} \right\} < \left\{ \begin{array}{c} \text{width} \\ \text{height} \\ \text{total} \end{array} \right\} < \left\{ \begin{array}{c} \text{textwidth} \\ \text{textheight} \\ \text{text} \end{array} \right\} < \text{lines}.$$

Par exemple,

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

est équivalent à `\usepackage[textwidth=7in]{geometry}`. Autre exemple :

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

revient à saisir `[lines=30, textwidth=7in]`.

6.4 Valeurs par défaut

Cette section résume le paramétrage par défaut pour l'autocomplétion décrite par la suite.

Le rapport des marges verticales est par défaut 2/3, autrement dit,

$$\text{top} : \text{bottom} = 2 : 3 \quad \text{par défaut.} \quad (6)$$

Pour ce qui est du rapport des marges horizontales, la valeur par défaut est liée au fait que le document est en recto uniquement ou recto-verso,

$$\text{left (inner)} : \text{right (outer)} = \begin{cases} 1 : 1 & \text{par défaut pour recto uniquement,} \\ 2 : 3 & \text{par défaut pour recto-verso.} \end{cases} \quad (7)$$

À l'évidence, le rapport pour les marges horizontales en recto uniquement correspond à un centrage.

L'extension `geometry` a le paramétrage par défaut suivant pour les documents en recto uniquement :

- `scale=0.7` (le `corps` vaut $0,7 \times$ le `papier`) ;
- `marginratio={1:1, 2:3}` (1:1 pour les marges horizontales et 2:3 pour les marges verticales) ;
- `ignoreall` (l'en-tête, le pied de page, les notes marginales sont exclus lors du calcul de la taille du `corps`).

Pour un document recto-verso avec l'option `twoside`, le paramétrage par défaut est le même que celui du recto uniquement à ceci près que le rapport des marges horizontales est fixé à 2:3 également.

Les options additionnelles écrasent les dimensions spécifiées auparavant.

6.5 Autocomplétion

La Figure 7 montre schématiquement comment des motifs de spécification existent et comment résoudre l'ambiguïté des spécifications. Chaque axe montre le nombre de longueurs explicitement spécifiées pour le corps et les marges. $S(m,b)$ représente la spécification avec une paire de nombres (`marge`, `corps`) = (m, b).

Par exemple, la spécification `width=14cm`, `left=3cm` est catégorisée en $S(1,1)$, ce qui est une spécification adéquate. Si vous ajoutez `right=4cm`, elle deviendrait $S(2,1)$, surspécifiée. Si seule `width=14cm` est donnée, elle passe à $S(0,1)$, sous-spécifiée.

L'extension `geometry` dispose d'un mécanisme d'autocomplétion, dans lequel si les paramètres de mise en forme sont sous-spécifiés ou surspécifiés, `geometry` résout l'ambiguïté en utilisant des valeurs par défaut ou d'autres relations. Voici les spécifications et les règles de complétion.

$S(0,0)$ Rien n'est spécifié. L'extension `geometry` précise le `corps` avec la valeur par défaut de `scale` (= 0,7).

Par exemple, `width` est fixée à $0,7 \times \text{layoutwidth}$. Notez que par défaut `layoutwidth` et `layoutheight` seront égaux à `\paperwidth` et `\paperheight` respectivement. Ainsi $S(0,0)$ passe à $S(0,1)$. Voir $S(0,1)$.

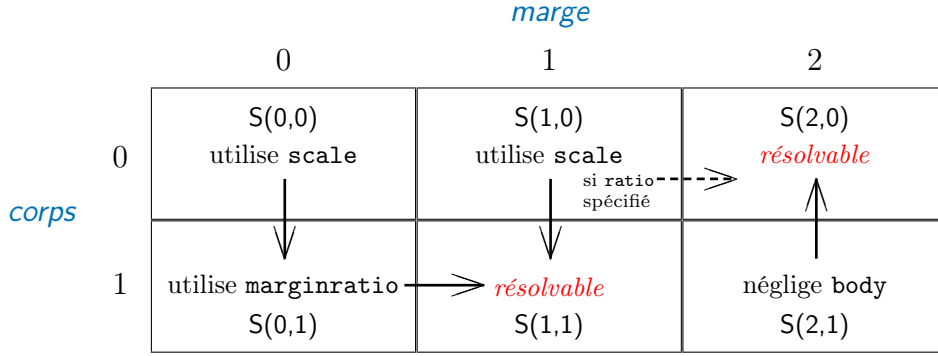


FIGURE 7 – Spécifications S(0,0) à S(2,1) et les règles de complétion (flèches). Les nombres des colonnes et des lignes indiquent le nombre de longueurs explicitement spécifiées pour les marges et le corps respectivement. S(*m*,*b*) résume une spécification avec une paire de nombre (*marge*, *corps*) = (*m*, *c*).

S(0,1) Seul le *corps* est spécifié, comme dans `width=7in, lines=20, body={20cm,24cm}, scale=0.9` et ainsi de suite. Alors `geometry` détermine les marges avec le rapport de marge. Si le rapport de marge n'est pas spécifié, la valeur par défaut est utilisée. Le rapport de marge vertical par défaut est défini comme

$$\text{top} : \text{bottom} = 2 : 3 \quad \text{par défaut.} \quad (8)$$

Tandis qu'avec le rapport de marge horizontal, la valeur par défaut change selon que le document soit en recto uniquement ou recto-verso,

$$\text{left (inner)} : \text{right (outer)} = \begin{cases} 1 : 1 & \text{par défaut en recto uniquement,} \\ 2 : 3 & \text{par défaut en recto-verso.} \end{cases} \quad (9)$$

Par exemple, si `height=22cm` est spécifié pour du papier A4, `geometry` calcule la marge `top` comme suit :

$$\begin{aligned} \text{top} &= (\text{layoutheight} - \text{height}) \times 2/5 \\ &= (29,7 - 22) \times 2/5 = 3,08(\text{cm}) \end{aligned} \quad (10)$$

La marge `top` et la hauteur `height` du corps ayant été déterminées, la spécification verticale passe à S(1,1) et tous les paramètres peuvent être déterminés.

S(1,0) Seule une marge est spécifiée, comme dans `bottom=2cm, left=1in, top=3cm` et ainsi de suite.

- Si le rapport de marge n'est pas spécifié, `geometry` détermine le *corps* avec la valeur par défaut de `scale` (= 0,7). Par exemple, si `top=2.4cm` est spécifié, `geometry` choisit

$$\text{height} = 0,7 \times \text{layoutheight} \quad (= 0,7 \times \backslash\text{paperheight} \text{ par défaut}) \quad (11)$$

et alors S(1,0) passe à S(1,1), avec lequel `bottom` est calculé avec `layoutheight - (height + top)` soit 6,51cm sur format A4 si la taille du gabarit est celle du papier.

- Si le rapport de marge est spécifié, comme dans `hmarginratio={1:2}, vratio={3:4}` et ainsi de suite, `geometry` détermine l'autre marge avec le rapport de marge spécifié. Par exemple, si un ensemble d'options « `top=2.4cm, vratio={3:4}` » est spécifié, `geometry` fixe `bottom` à 3,2cm par le calcul suivant :

$$\text{bottom} = \text{top}/3 \times 4 = 3,2\text{cm} \quad (12)$$

Ainsi S(1,0) passe à S(2,0).

Notez que les versions 4 et précédentes utilisaient le rapport de marge pour déterminer l'autre marge. C'est pourquoi, dans la version 5, avec la même spécification, la mise en page changera par rapport à la version 4. Par exemple, si seule `top=2.4cm` est spécifiée, vous obtenez `bottom=2.4cm` dans les versions 4 et antérieures mais vous obtiendrez `bottom=6.51cm` dans la version 5.

S(2,1) Le *corps* et les deux *marges* sont spécifiés, comme dans `vdivide={1in,8in,1.5in}`, `left=3cm,width=13cm,right=4cm` et ainsi de suite. Comme `geometry` donne la priorité aux *marges* si les dimensions sont surspécifiées, `geometry` redéfinit le *corps*. Par exemple, si vous spécifiez

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

`width` est redéfini à 14cm car la largeur d'un format A4 est de 21cm.

7 Changement de mise en page en cours de document

La version 5 apporte les nouvelles commandes `\newgeometry{...}` et `\restoregeometry` qui vous permettent de changer la mise en page en cours de document. Contrairement à `\geometry` dans le préambule, `\newgeometry` est disponible uniquement après `\begin{document}`. Elle réinitialise toutes les options spécifiées à l'exception des options liées à l'orientation du papier (`landscape`, `portrait`) et les options liées à la taille du papier (telles que `papersize`, `paper=a4paper` et ainsi de suite), qui ne peuvent être changées avec `\newgeometry`.

La commande `\restoregeometry` restaure la mise en page spécifiée dans le préambule (avant `\begin{document}`) avec les options de `\usepackage{geometry}` et `\geometry`.

Notez que `\newgeometry` et `\restoregeometry` insèrent toutes deux un `\clearpage` là où elles sont appelées.

Ci-dessous est présenté un exemple de changement de mise en page en cours de document. La mise en forme M1 indiquant `hmargin=3cm` (les marges `left` et `right` mesurent 3cm) est changée pour la mise en page M2 avec `left=3cm`, `right=1cm` et `bottom=0.1cm`. La mise en page M1 est restaurée avec `\restoregeometry`.

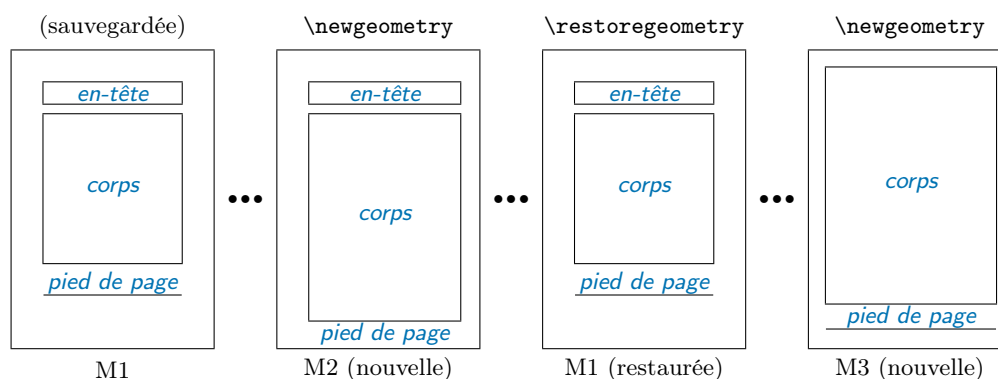
```
\usepackage[hmargin=3cm]{geometry}
\begin{document}
  Mise en page M1

  \newgeometry{left=3cm,right=1cm,bottom=0.1cm}
  Mise en page M2 (nouvelle)

  \restoregeometry
  Mise en page M1 (restaurée)

  \newgeometry{margin=1cm,includefoot}
  Mise en page M3 (nouvelle)

\end{document}
```



Un ensemble de commandes `\savegeometry{<nom>}` et `\loadgeometry{<nom>}` est pratique si vous souhaitez réutiliser à plusieurs reprises différentes mises en page dans votre document. Par exemple,

```
\usepackage[hmargin=3cm]{geometry}
\begin{document}
  M1
\newgeometry{left=3cm,right=1cm,bottom=0.1cm}
\savegeometry{M2}
  M2 (nouvelle, sauvée)
\restoregeometry
  M1 (restaurée)
\newgeometry{margin=1cm,includefoot}
  M3 (nouvelle)
\loadgeometry{M2}
  M2 (chargée)
\end{document}
```

8 Exemples

1. Une mise en page en recto uniquement avec le texte centré sur le papier. Les exemples ci-dessous ont le même résultat car le rapport de marge horizontale est fixé à 1:1 par défaut en recto uniquement.
 - `centering`
 - `marginratio=1:1`
 - `vcentering`
2. Une mise en page recto-verso avec un décalage intérieur pour la reliure fixé à 1cm.
 - `twoside, bindingoffset=1cm`

Dans ce cas, le `textwidth` est plus petit que celui d'un document recto-verso par défaut de $0,7 \times 1\text{cm}$ ($= 0,7\text{cm}$) car la largeur du *corps* est fixée avec `scale=0.7`, ce qui signifie que `width = 0,7 \times layoutwidth` ($= 0,7 \times \text{paperwidth}$ par défaut).
3. Une mise en page avec les marges gauche, droite et haute fixées respectivement à 3cm, 2cm et 2.5in, avec une hauteur de texte mise à 40 lignes et avec une en-tête et un pied de page inclus dans le *corps intégral*. Les deux exemples ci-dessous ont le même effet.
 - `left=3cm, right=2cm, lines=40, top=2.5in, includeheadfoot`
 - `hmargin={3cm,2cm}, tmargin=2.5in, lines=40, includeheadfoot`
4. Une mise en page avec une hauteur du *corps intégral* à 10in, la marge basse à 2cm et une largeur par défaut. La marge haute sera calculée automatiquement. Chaque solution ci-dessous conduit à la même mise en page.
 - `vdivide={*, 10in, 2cm}`
 - `bmargin=2cm, height=10in`
 - `bottom=2cm, textheight=10in`

Notez que les dimensions de l'*en-tête* et du *pied de page* sont exclues de la hauteur `height` du *corps intégral*. Un `includefoot` additionnel inclut `\footskip` dans `totalheight`. C'est pourquoi, dans les deux cas de mise en page ci-dessous, `textheight` dans le premier cas est plus petit que dans le second (avec 10in exactement) avec un écart de `\footskip`. En d'autres termes, `height = textheight + footskip` quand `includefoot=true`.

 - `bmargin=2cm, height=10in, includefoot`
 - `bottom=2cm, textheight=10in, includefoot`
5. Une mise en page avec `textwidth` et `textheight` à 90% de la taille du papier et avec le *corps* centré. Chaque solution ci-dessous aboutit au même résultat tant que `layoutwidth` et `layoutheight` ne sont pas changé par rapport aux valeurs par défaut.
 - `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,*}` (pour les documents en recto uniquement)
 - `margin={0.05\paperwidth,0.05\paperheight}`

Vous pouvez ajouter `heightrounded` pour éviter un message de « *sous-remplissage vertical* » comme

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

Voir la Section 5.3 pour la description détaillée de `heightrounded`.

6. Une mise en page avec la largeur de la note marginale fixée à 3cm et incluse dans la largeur du *corps intégral*. Les exemples suivants donnent le même résultat.

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

7. Une mise en page où le *corps* occupe tout l'espace du papier en format A5 et en orientation paysage. Les exemples suivants conduisent à la même mise en page.

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

8. Une mise en page en format écran appropriée pour les présentations sur ordinateurs et vidéoprojecteurs.

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

9. Une mise en page avec les fontes et les espaces tous redimensionnés du format A4 au A3. Sur le cas ci-dessous, le résultat tient sur un papier A3.

- `a4paper, mag=1414`.

Si vous voulez avoir une mise en page avec des fontes deux fois plus grandes mais pas de changement de la taille du papier, vous pouvez saisir

- `letterpaper, mag=2000, truedimen`.

Vous pouvez ajouter l'option `dvips`, ce qui est pratique pour prévisualiser, avec la taille de papier associée, dans `dviout` ou `xdvi`.

10. Une mise en page avec une première page spécifique et les pages suivantes ramenées à la présentation par défaut existant avant le chargement de `geometry`. Utilisez l'option `pass`, `\newgeometry` et `\restoregeometry`.

```
\documentclass{book}
\usepackage[pass]{geometry}
% 'pass' annule la mise en page de l'extension
% de façon à ce que la mise en page de 'book' soit mémorisée ici.
\begin{document}
\newgeometry{margin=1cm}% change les dimensions de la première page.
Page 1
\restoregeometry % restaure la mise en page initiale de 'book'.
Page 2 et suivantes
\end{document}
```

11. Une mise en page complexe.

```
\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}
```

Essayez de la tester pour voir par vous-même le résultat. :-)

9 Problèmes connus

- Avec `mag` \neq 1000 et `truedimen`, `paperwidth` et `paperheight` indiqué en mode `verbose` sont différents de la véritable taille du PDF final. Le PDF en lui-même est correct cependant.
- Avec `mag` \neq 1000, *aucun* `truedimen` et `hyperref`, `hyperref` devrait être chargé avant `geometry`. Sans cela, la taille du PDF final sera incorrecte.
- Avec l'extension `crop` et `mag` \neq 1000, l'option `center` de `crop` ne fonctionne pas bien.

10 Remerciements

L'auteur remercie pour leurs suggestions, aides et commentaires de nombreuses personnes, dont (par ordre alphabétique des noms) : Jean-Bernard Addor, Frank Bennett, Alexis Dimitriadis, Friedrich Flender,

Adrian Heathcote, Stephan Hennig, Morten Høgholm, Jonathan Kew, James Kilfiger, Yusuke Kuroki, Jean-Marc Lasgouttes, Włodzimierz Macewicz, Frank Mittelbach, Eckhard Neber, Rolf Niepraschk, Hans Fr. Nordhaug, Heiko Oberdiek, Keith Reckdahl, Peter Riocreux, Will Robertson, Pablo Rodriguez, Nico Schlöemer, Perry C. Stearns, Frank Stengel, Plamen Tanovski, Petr Uher, Piet van Oostrum, Vladimir Volovich, Michael Vulis.

11 Implémentation

1 (*package)

Cette extension requiert les extensions suivantes : keyval, ifpdf, ifvtex et ifxetex.

```
2 \RequirePackage{keyval}%
3 \RequirePackage{ifpdf}%
4 \RequirePackage{ifvtex}%
5 \RequirePackage{ifxetex}%
```

Les bascules internes sont déclarées ici.

```
6 \newif\ifGm@verbose
7 \newif\ifGm@landscape
8 \newif\ifGm@swap@papersize
9 \newif\ifGm@includehead
10 \newif\ifGm@includefoot
11 \newif\ifGm@includemp
12 \newif\ifGm@hbody
13 \newif\ifGm@vbody
14 \newif\ifGm@heightrounded
15 \newif\ifGm@showframe
16 \newif\ifGm@showcrop
17 \newif\ifGm@pass
18 \newif\ifGm@resetpaper
19 \newif\ifGm@layout
20 \newif\ifGm@newgm
```

\Gm@cnth Les compteurs pour les motifs de partitionnement horizontal et vertical

```
\Gm@cntv 21 \newcount\Gm@cnth
22 \newcount\Gm@cntv
```

\c@Gm@tempcnt Le compteur est utilisé pour fixer des nombres avec calc.

```
23 \newcount\c@Gm@tempcnt
```

\Gm@bindingoffset Le décalage pour la reluire dans la marge intérieure.

```
24 \newdimen\Gm@bindingoffset
```

\Gm@wd@mp Les longueurs de correction pour \textwidth, \oddsidemargin et \evensidemargin en mode includemp.

```
\Gm@odd@mp 25 \newdimen\Gm@wd@mp
26 \newdimen\Gm@odd@mp
27 \newdimen\Gm@even@mp
```

\Gm@layoutwidth Les dimensions du ✖ gabarit ✖.

```
\Gm@layoutheight 28 \newdimen\Gm@layoutwidth
29 \newdimen\Gm@layoutheight
\Gm@layouthoffset 30 \newdimen\Gm@layouthoffset
31 \newdimen\Gm@layoutvoffset
```

\Gm@dimlist Le token dans lequel peuvent être stockées les dimensions de base de L^AT_EX.

```
32 \newtoks\Gm@dimlist
```

\Gm@warning La commande affichant les messages d'alerte.

```
33 \def\Gm@warning#1{\PackageWarningNoLine{geometry}{#1}}%
```

\ifGm@preamble La commande exécute les options données en argument uniquement si elles sont spécifiées dans le préambule, en tant qu'options de \usepackage et/ou qu'options de \geometry. Sinon, la commande affiche un message d'avertissement et ignore les options indiquées.

```
34 \def\ifGm@preamble#1{%
35   \ifGm@newgm
36     \Gm@warning{‘#1’: not available in ‘\string\newgeometry’; skipped}%
37   \else
38     \expandafter\@firstofone
39   \fi}%
```

`\Gm@Dhratio` Les valeurs par défaut du rapport de marge horizontale et du rapport de marge verticale sont définies.

`\Gm@Dhratio``two` `\Gm@Dhratio``two` représente le rapport de marge horizontale dans une mise en page recto-verso avec les marges gauche et droite interverties sur les pages au verso, ce qui est fixé par `twoside`.

`\Gm@Dvratio`

```

40 \def\Gm@Dhratio{1:1}% = gauche:droite par défaut en recto uniquement
41 \def\Gm@Dhratio{2:3}% = intérieure:extérieure par défaut en recto-verso.
42 \def\Gm@Dvratio{2:3}% = haute:basse par défaut

```

`\Gm@Dhscale` Les valeurs par défaut pour les rapports entre la taille du *corps intégral* et du *papier* sur l'axe horizontal et l'axe vertical sont mises à 0.7.

`\Gm@Dvscale`

```

43 \def\Gm@Dhscale{0.7}%
44 \def\Gm@Dvscale{0.7}%

```

`\Gm@dvi``ps` Les noms des pilotes.

`\Gm@dvi``pdf``fm` `\def\Gm@dvi``ps``{dvi``ps}%`

`\Gm@pdf``tex` `\def\Gm@dvi``pdf``fm``{dvi``pdf``fm}%`

`\Gm@xetex` `\def\Gm@pdf``tex``{pdf``tex}%`

`\Gm@vtex` `\def\Gm@xetex``{xetex}%`

`\def\Gm@vtex``{vtex}%`

`\Gm@true` Les commandes pour `true` et `false`.

`\Gm@false` `\def\Gm@true``{true}%`

`\def\Gm@false``{false}%`

`\Gm@org``pw` Ces commandes gardent intact la taille du papier (medium) original.

`\Gm@org``ph` `\edef\Gm@org``pw``{\the\paperwidth}%`

`\edef\Gm@org``ph``{\the\paperheight}%`

`\Gm@savelength` Ces commandes sauvegardent la longueur spécifiée dans `\Gm@restore`.

```

54 \def\Gm@savelength#1{%
55   \g@addto@macro\Gm@restore{\expandafter\noexpand\expandafter\csname
56     #1\endcsname\expandafter=\expandafter\the\csname #1\endcsname\relax}}%

```

`\Gm@saveboolean` Ces commandes sauvegardent les booléens spécifiés dans `\Gm@restore`.

```

57 \def\Gm@saveboolean#1{%
58   \csname if#1\endcsname
59   \g@addto@macro\Gm@restore{\expandafter\noexpand\csname #1true\endcsname}%
60   \else
61   \g@addto@macro\Gm@restore{\expandafter\noexpand\csname #1false\endcsname}%
62   \fi}%

```

`\Gm@restore` L'initialisation de `\Gm@restore`.

```

63 \def\Gm@restore{%

```

`\Gm@save` La définition de la commande sauvegardant les longueurs réelles des options de L^AT_EX.

```

64 \def\Gm@save{%
65   \Gm@savelength{\paperwidth}%
66   \Gm@savelength{\paperheight}%
67   \Gm@savelength{\textwidth}%
68   \Gm@savelength{\textheight}%
69   \Gm@savelength{\evensidemargin}%
70   \Gm@savelength{\oddsidemargin}%
71   \Gm@savelength{\topmargin}%
72   \Gm@savelength{\headheight}%
73   \Gm@savelength{\headsep}%
74   \Gm@savelength{\topskip}%
75   \Gm@savelength{\footskip}%
76   \Gm@savelength{\baselineskip}%
77   \Gm@savelength{\marginparwidth}%
78   \Gm@savelength{\marginparsep}%
79   \Gm@savelength{\columnsep}%
80   \Gm@savelength{\hoffset}%
81   \Gm@savelength{\voffset}%
82   \Gm@savelength{\Gm@layoutwidth}%
83   \Gm@savelength{\Gm@layoutheight}%

```

```

84 \Gm@savelength{Gm@layoutoffset}%
85 \Gm@savelength{Gm@layoutvoffset}%
86 \Gm@saveboolean{@twocolumn}%
87 \Gm@saveboolean{@twoside}%
88 \Gm@saveboolean{@mparswitch}%
89 \Gm@saveboolean{@reversemargin}}%

```

\Gm@initnewgm Cette commande initialise les paramètres de mise en page dans `\newgeometry`.

```

90 \def\Gm@initnewgm{%
91   \Gm@passfalse
92   \Gm@swap@papersizefalse
93   \Gm@dimlist={}
94   \Gm@hbodyfalse
95   \Gm@vbodyfalse
96   \Gm@heightroundedfalse
97   \Gm@includeheadfalse
98   \Gm@includefootfalse
99   \Gm@includempfalse
100  \let\Gm@width\@undefined
101  \let\Gm@height\@undefined
102  \let\Gm@textwidth\@undefined
103  \let\Gm@textheight\@undefined
104  \let\Gm@lines\@undefined
105  \let\Gm@hscale\@undefined
106  \let\Gm@vscale\@undefined
107  \let\Gm@hmarginratio\@undefined
108  \let\Gm@vmarginratio\@undefined
109  \let\Gm@lmargin\@undefined
110  \let\Gm@rmargin\@undefined
111  \let\Gm@tmargin\@undefined
112  \let\Gm@bmargin\@undefined
113  \Gm@layoutfalse
114  \Gm@layoutoffset\z@
115  \Gm@layoutvoffset\z@
116  \Gm@bindingoffset\z@}%

```

\Gm@initall Cette initialisation est appelée dès que l'extention est chargée. Elle est également appelée dès que l'option `reset` est spécifiée.

```

117 \def\Gm@initall{%
118   \let\Gm@driver\@empty
119   \let\Gm@truedimen\@empty
120   \let\Gm@paper\@undefined
121   \Gm@resetpaperfalse
122   \Gm@landscapefalse
123   \Gm@verbosefalse
124   \Gm@showframefalse
125   \Gm@showcropfalse
126   \Gm@newgmfalse
127   \Gm@initnewgm}%

```

\Gm@setdriver Cette commande définit le pilote spécifié.

```

128 \def\Gm@setdriver#1{%
129   \expandafter\let\expandafter\Gm@driver\csname Gm@#1\endcsname}%

```

\Gm@unsetdriver Cette commande annule la prise en charge du pilote spécifié si il a été déjà sélectionné.

```

130 \def\Gm@unsetdriver#1{%
131   \expandafter\ifx\csname Gm@#1\endcsname\Gm@driver\let\Gm@driver\@empty\fi}%

```

\Gm@setbool Les commandes pour le traitement des options booléennes.

```

\Gm@setboolrev 132 \def\Gm@setbool{\@dblarg\Gm@@setbool}%
133 \def\Gm@setboolrev{\@dblarg\Gm@@setboolrev}%
134 \def\Gm@@setbool[#1]#2#3{\Gm@doif{#1}{#3}{\csname Gm@#2\Gm@bool\endcsname}}%
135 \def\Gm@@setboolrev[#1]#2#3{\Gm@doifelse{#1}{#3}%
136   {\csname Gm@#2\Gm@false\endcsname}{\csname Gm@#2\Gm@true\endcsname}}%

```

`\Gm@doif` `\Gm@doif` exécute le troisième argument #3 en utilisant la valeur booléenne #2 de l'option #1.
`\Gm@doifelse` `\Gm@doifelse` exécute le troisième argument #3 si l'option booléenne #1 a sa valeur #2 valant `true` et exécute le quatrième argument #4 si cette valeur `false`.

```

137 \def\Gm@doif#1#2#3{%
138   \lowercase{\def\Gm@bool{#2}}%
139   \ifx\Gm@bool\@empty
140     \let\Gm@bool\Gm@true
141   \fi
142   \ifx\Gm@bool\Gm@true
143   \else
144     \ifx\Gm@bool\Gm@false
145     \else
146       \let\Gm@bool\relax
147     \fi
148   \fi
149   \ifx\Gm@bool\relax
150     \Gm@warning{‘#1’ should be set to ‘true’ or ‘false’}%
151   \else
152     #3
153   \fi}%
154 \def\Gm@doifelse#1#2#3#4{%
155   \Gm@doif{#1}{#2}{\ifx\Gm@bool\Gm@true #3\else #4\fi}}%

```

`\Gm@reverse` Cette commande fait une valeur booléenne à la valeur opposée.

```

156 \def\Gm@reverse#1{%
157   \csname ifGm@#1\endcsname
158   \csname Gm@#1false\endcsname\else\csname Gm@#1true\endcsname\fi}%

```

`\Gm@defbylen` Les commandes `\Gm@defbylen` et `\Gm@defbycnt` peuvent être utilisées pour définir des variables
`\Gm@defbycnt` `\Gm@xxxx` respectivement avec une longueur ou avec un compteur, ceci avec l'extension `calc`.

```

159 \def\Gm@defbylen#1#2{%
160   \begingroup\setlength\@tempdima{#2}%
161   \expandafter\xdef\csname Gm@#1\endcsname{\the\@tempdima}\endgroup}%
162 \def\Gm@defbycnt#1#2{%
163   \begingroup\setcounter{Gm@tempcnt}{#2}%
164   \expandafter\xdef\csname Gm@#1\endcsname{\the\value{Gm@tempcnt}}\endgroup}%

```

`\Gm@set@ratio` Cette commande analyse la valeur des options spécifiant les rapports de marge, ce qui est utilisé dans la commande `\Gm@setbyratio`.

```

165 \def\Gm@sep@ratio#1:#2{\@tempcnta=#1\@tempcntb=#2}%

```

`\Gm@setbyratio` Cette commande détermine la dimension spécifiée par #4 en calculant $\#3 \times a/b$, où a et b sont donnés par `\Gm@mratio` avec une valeur $a : b$. Si #1 entre crochet vaut `b`, a et b sont intervertis. Le deuxième argument avec `h` ou `v` précisant l'aspect horizontal ou vertical n'est pas utilisé dans cette commande.

```

166 \def\Gm@setbyratio[#1]#2#3#4{% détermine #4 avec un rapport
167   \expandafter\Gm@sep@ratio\Gm@mratio\relax
168   \if#1b
169     \edef\@tempa{\the\@tempcnta}%
170     \@tempcnta=\@tempcntb
171     \@tempcntb=\@tempa\relax
172   \fi
173   \expandafter\setlength\expandafter\@tempdimb\expandafter
174     {\csname Gm@#3\endcsname}%
175   \ifnum\@tempcntb>\z@
176     \multiply\@tempdimb\@tempcnta
177     \divide\@tempdimb\@tempcntb
178   \fi
179   \expandafter\edef\csname Gm@#4\endcsname{\the\@tempdimb}}%

```

`\Gm@detiv` Cette commande détermine la quatrième longueur (#4) à partir de #1(`layoutwidth` ou `layoutheight`), #2 et #3. Elle est utilisée dans la commande `\Gm@detall`.

```

180 \def\Gm@detiv#1#2#3#4{% détermine #4.
181   \expandafter\setlength\expandafter\@tempdima\expandafter
182     {\csname Gm@layout#1\endcsname}%

```



```

183 \expandafter\setlength\expandafter\@tempdimb\expandafter
184   {\csname Gm@#2\endcsname}%
185 \addtolength\@tempdima{-\@tempdimb}%
186 \expandafter\setlength\expandafter\@tempdimb\expandafter
187   {\csname Gm@#3\endcsname}%
188 \addtolength\@tempdima{-\@tempdimb}%
189 \ifdim\@tempdima<z@
190   \Gm@warning{‘#4’ results in NEGATIVE (\the\@tempdima).%
191   ~~J\@spaces ‘#2’ or ‘#3’ should be shortened in length}%
192 \fi
193 \expandafter\edef\csname Gm@#4\endcsname{\the\@tempdima}}%

```

`\Gm@detiandiii` Cette commande détermine #2 et #3 à partir de #1, l’argument #1 pouvant être `width` ou `height`. Cet argument est développé alors en dimensions du papier et du corps total. Elle est utilisée dans la commande `\Gm@detall`.

```

194 \def\Gm@detiandiii#1#2#3{% détermine #2 et #3.
195   \expandafter\setlength\expandafter\@tempdima\expandafter
196     {\csname Gm@layout#1\endcsname}%
197   \expandafter\setlength\expandafter\@tempdimb\expandafter
198     {\csname Gm@#1\endcsname}%
199   \addtolength\@tempdima{-\@tempdimb}%
200   \ifdim\@tempdima<z@
201     \Gm@warning{‘#2’ and ‘#3’ result in NEGATIVE (\the\@tempdima).%
202     ~~J\@spaces ‘#1’ should be shortened in length}%
203   \fi
204   \ifx\Gm@ratio\@undefined
205     \expandafter\Gm@sep@ratio\Gm@Dmratio\relax
206   \else
207     \expandafter\Gm@sep@ratio\Gm@mratio\relax
208     \ifnum\@tempcntb>z@\else
209       \Gm@warning{margin ratio a:b should be non-zero; default used}%
210       \expandafter\Gm@sep@ratio\Gm@Dmratio\relax
211     \fi
212   \fi
213   \@tempdimb=\@tempdima
214   \advance\@tempcntb\@tempcnta
215   \divide\@tempdima\@tempcntb
216   \multiply\@tempdima\@tempcnta
217   \advance\@tempdimb-\@tempdima
218   \expandafter\edef\csname Gm@#2\endcsname{\the\@tempdima}%
219   \expandafter\edef\csname Gm@#3\endcsname{\the\@tempdimb}}%

```

`\Gm@detall` Cette commande détermine la partition de chaque direction. L’argument #1 vaut `h` or `v`, #2 vaut `width` ou `height`, #3 vaut `lmargin` ou `top`, et #4 vaut `rmargin` or `bottom`.

```

220 \def\Gm@detall#1#2#3#4{%
221   \@tempcnta\z@
222   \if#1h
223     \let\Gm@ratio\Gm@hmarginratio
224     \edef\Gm@Dmratio{\if@twoside\Gm@Dhratio\else\Gm@Dhratio\fi}%
225   \else
226     \let\Gm@ratio\Gm@vmarginratio
227     \edef\Gm@Dmratio{\Gm@Dvratio}%
228   \fi

```

`\@tempcnta` est traité comme un nombre binaire à trois chiffres associés respectivement aux marges `left(top)`, `width(height)` et `right(bottom)` spécifiées par l’utilisateur.

```

229   \if#1h
230     \ifx\Gm@lmargin\@undefined\else\advance\@tempcnta4\relax\fi
231     \ifx\Gm@hbody\advance\@tempcnta2\relax\fi
232     \ifx\Gm@rmargin\@undefined\else\advance\@tempcnta1\relax\fi
233     \Gm@cnth\@tempcnta
234   \else
235     \ifx\Gm@tmargin\@undefined\else\advance\@tempcnta4\relax\fi
236     \ifx\Gm@vbody\advance\@tempcnta2\relax\fi
237     \ifx\Gm@bmargin\@undefined\else\advance\@tempcnta1\relax\fi

```

```

238 \Gm@cntv\@tempcnta
239 \fi
    Cas où le nombre est 000 (=0) lorsque rien n'est fixé (cas par défaut) :
240 \ifcase\@tempcnta
241 \if#1h
242 \Gm@defbylen{width}{\Gm@Dhscale\Gm@layoutwidth}%
243 \else
244 \Gm@defbylen{height}{\Gm@Dvscale\Gm@layoutheight}%
245 \fi
246 \Gm@detiandiii{#2}{#3}{#4}%
    Cas 001 (=1) avec la marge right(bottom) fixée :
247 \or
248 \ifx\Gm@mratio\@undefined
249 \if#1h
250 \Gm@defbylen{width}{\Gm@Dhscale\Gm@layoutwidth}%
251 \else
252 \Gm@defbylen{height}{\Gm@Dvscale\Gm@layoutheight}%
253 \fi
254 \setlength\@tempdimc{\@nameuse{Gm@#4}}%
255 \Gm@detiandiii{#2}{#3}{#4}%
256 \expandafter\let\csname Gm@#2\endcsname\@undefined
257 \Gm@defbylen{#4}{\@tempdimc}%
258 \else
259 \Gm@setbyratio[f]{#1}{#4}{#3}%
260 \fi
261 \Gm@detiv{#2}{#3}{#4}{#2}%
    Cas 010 (=2) avec la dimension centrale width(height) fixée :
262 \or\Gm@detiandiii{#2}{#3}{#4}%
    Cas 011 (=3) avec la dimension centrale width(height) et la marge right(bottom) fixées :
263 \or\Gm@detiv{#2}{#2}{#4}{#3}%
    Cas 100 (=4) avec la marge left(top) fixée :
264 \or
265 \ifx\Gm@mratio\@undefined
266 \if#1h
267 \Gm@defbylen{width}{\Gm@Dhscale\Gm@layoutwidth}%
268 \else
269 \Gm@defbylen{height}{\Gm@Dvscale\Gm@layoutheight}%
270 \fi
271 \setlength\@tempdimc{\@nameuse{Gm@#3}}%
272 \Gm@detiandiii{#2}{#4}{#3}%
273 \expandafter\let\csname Gm@#2\endcsname\@undefined
274 \Gm@defbylen{#3}{\@tempdimc}%
275 \else
276 \Gm@setbyratio[b]{#1}{#3}{#4}%
277 \fi
278 \Gm@detiv{#2}{#3}{#4}{#2}%
    Cas 101 (=5) avec les marges left(top) et right(bottom) fixées :
279 \or\Gm@detiv{#2}{#3}{#4}{#2}%
    Cas 110 (=6) avec la marge left(top) et la dimension centrale width(height) fixées :
280 \or\Gm@detiv{#2}{#2}{#3}{#4}%
    Cas 111 (=7) avec les deux marges et la dimension centrale fixée, cas sur-spécifié :
281 \or\Gm@warning{Over-specification in '#1'-direction.%
282 ~J\@spaces '#2' (\@nameuse{Gm@#2}) is ignored}%
283 \Gm@detiv{#2}{#3}{#4}{#2}%
284 \else\fi}%

```

`\Gm@clean` Cette commande sert à donner à des dimensions non spécifiées la valeur `\@undefined`. Elle sert dans la commande `\geometry`.

```

285 \def\Gm@clean{%
286 \ifnum\Gm@cnth<4\let\Gm@lmargin\@undefined\fi

```

```

287 \ifodd\Gm@cnth\else\let\Gm@rmargin\@undefined\fi
288 \ifnum\Gm@cntv<4\let\Gm@tmargin\@undefined\fi
289 \ifodd\Gm@cntv\else\let\Gm@bmargin\@undefined\fi
290 \ifGm@hbody\else
291   \let\Gm@hscale\@undefined
292   \let\Gm@width\@undefined
293   \let\Gm@textwidth\@undefined
294 \fi
295 \ifGm@vbody\else
296   \let\Gm@vscale\@undefined
297   \let\Gm@height\@undefined
298   \let\Gm@textheight\@undefined
299 \fi
300 }%

```

\Gm@parse@divide Cette commande analyse les options (h,v)divide.

```

301 \def\Gm@parse@divide#1#2#3#4{%
302   \def\Gm@star{*}%
303   \@tempcnta\z@
304   \@for\Gm@tmp:=#1\do{%
305     \expandafter\KV@sp@def\expandafter\Gm@frag\expandafter{\Gm@tmp}%
306     \edef\Gm@value{\Gm@frag}%
307     \ifcase\@tempcnta\relax\edef\Gm@key{#2}%
308       \or\edef\Gm@key{#3}%
309       \else\edef\Gm@key{#4}%
310     \fi
311     \@nameuse{Gm@set\Gm@key false}%
312     \ifx\empty\Gm@value\else
313       \ifx\Gm@star\Gm@value\else
314         \setkeys{Gm}{\Gm@key=\Gm@value}%
315       \fi\fi
316     \advance\@tempcnta\@ne}%
317   \let\Gm@star\relax}%

```

\Gm@branch Cette valeur scinde une valeur en deux valeurs identiques.

```

318 \def\Gm@branch#1#2#3{%
319   \@tempcnta\z@
320   \@for\Gm@tmp:=#1\do{%
321     \KV@sp@def\Gm@frag{\Gm@tmp}%
322     \edef\Gm@value{\Gm@frag}%
323     \ifcase\@tempcnta\relax% cnta == 0
324       \setkeys{Gm}{#2=\Gm@value}%
325     \or% cnta == 1
326       \setkeys{Gm}{#3=\Gm@value}%
327     \else\fi
328     \advance\@tempcnta\@ne}%
329   \ifnum\@tempcnta=\@ne
330     \setkeys{Gm}{#3=\Gm@value}%
331   \fi}%

```

\Gm@magtooffset Cette commande est utilisée pour ajuster les décalages par \mag.

```

332 \def\Gm@magtooffset{%
333   \@tempdima=\mag\Gm@truedimen sp%
334   \@tempdimb=1\Gm@truedimen in%
335   \divide\@tempdimb\@tempdima
336   \multiply\@tempdimb\@m
337   \addtolength{\hoffset}{1\Gm@truedimen in}%
338   \addtolength{\voffset}{1\Gm@truedimen in}%
339   \addtolength{\hoffset}{-\the\@tempdimb}%
340   \addtolength{\voffset}{-\the\@tempdimb}%

```

\Gm@setlength Cette commande stocke les dimensions de base de L^AT_EX, lesquelles sont stockées et fixées par la suite.

```

341 \def\Gm@setlength#1#2{%
342   \let\Gm@len=\relax\let\Gm@td=\relax

```

```

343 \edef\addtolist{\noexpand\Gm@dimlist=%
344 {\the\Gm@dimlist \Gm@len{#1}{#2}}}\addtolist}%

```

\Gm@expandlengths Cette commande traite \Gm@dimlist.

```

345 \def\Gm@expandlengths{%
346 \def\Gm@td{\Gm@truedimen}%
347 \def\Gm@len##1##2{\setlength{##1}{##2}}%
348 \the\Gm@dimlist}%

```

\Gm@setsize Cette commande définit les dimensions paperwidth et paperheight en utilisant la commande \Gm@setlength.

```

349 \def\Gm@setsize#1(#2,#3)#4{%
350 \let\Gm@td\relax
351 \expandafter\Gm@setlength\csname #1width\endcsname{#2\Gm@td #4}%
352 \expandafter\Gm@setlength\csname #1height\endcsname{#3\Gm@td #4}%
353 \ifGm@landscape\Gm@swap@papersizetrue\else\Gm@swap@papersizefalse\fi}%

```

\Gm@setpaper@ifpre Cette commande change le format du papier.

```

354 \def\Gm@setpaper@ifpre#1{%
355 \ifGm@preamble{#1}{\def\Gm@paper{#1}\@nameuse{Gm@#1}{paper}}}%

```

Différents formats de papier sont définis ici.

```

356 \@namedef{Gm@a0paper}#1{\Gm@setsize{#1}(841,1189){mm}}% ISO A0
357 \@namedef{Gm@a1paper}#1{\Gm@setsize{#1}(594,841){mm}}% ISO A1
358 \@namedef{Gm@a2paper}#1{\Gm@setsize{#1}(420,594){mm}}% ISO A2
359 \@namedef{Gm@a3paper}#1{\Gm@setsize{#1}(297,420){mm}}% ISO A3
360 \@namedef{Gm@a4paper}#1{\Gm@setsize{#1}(210,297){mm}}% ISO A4
361 \@namedef{Gm@a5paper}#1{\Gm@setsize{#1}(148,210){mm}}% ISO A5
362 \@namedef{Gm@a6paper}#1{\Gm@setsize{#1}(105,148){mm}}% ISO A6
363 \@namedef{Gm@b0paper}#1{\Gm@setsize{#1}(1000,1414){mm}}% ISO B0
364 \@namedef{Gm@b1paper}#1{\Gm@setsize{#1}(707,1000){mm}}% ISO B1
365 \@namedef{Gm@b2paper}#1{\Gm@setsize{#1}(500,707){mm}}% ISO B2
366 \@namedef{Gm@b3paper}#1{\Gm@setsize{#1}(353,500){mm}}% ISO B3
367 \@namedef{Gm@b4paper}#1{\Gm@setsize{#1}(250,353){mm}}% ISO B4
368 \@namedef{Gm@b5paper}#1{\Gm@setsize{#1}(176,250){mm}}% ISO B5
369 \@namedef{Gm@b6paper}#1{\Gm@setsize{#1}(125,176){mm}}% ISO B6
370 \@namedef{Gm@c0paper}#1{\Gm@setsize{#1}(917,1297){mm}}% ISO C0
371 \@namedef{Gm@c1paper}#1{\Gm@setsize{#1}(648,917){mm}}% ISO C1
372 \@namedef{Gm@c2paper}#1{\Gm@setsize{#1}(458,648){mm}}% ISO C2
373 \@namedef{Gm@c3paper}#1{\Gm@setsize{#1}(324,458){mm}}% ISO C3
374 \@namedef{Gm@c4paper}#1{\Gm@setsize{#1}(229,324){mm}}% ISO C4
375 \@namedef{Gm@c5paper}#1{\Gm@setsize{#1}(162,229){mm}}% ISO C5
376 \@namedef{Gm@c6paper}#1{\Gm@setsize{#1}(114,162){mm}}% ISO C6
377 \@namedef{Gm@b0j}#1{\Gm@setsize{#1}(1030,1456){mm}}% JIS B0
378 \@namedef{Gm@b1j}#1{\Gm@setsize{#1}(728,1030){mm}}% JIS B1
379 \@namedef{Gm@b2j}#1{\Gm@setsize{#1}(515,728){mm}}% JIS B2
380 \@namedef{Gm@b3j}#1{\Gm@setsize{#1}(364,515){mm}}% JIS B3
381 \@namedef{Gm@b4j}#1{\Gm@setsize{#1}(257,364){mm}}% JIS B4
382 \@namedef{Gm@b5j}#1{\Gm@setsize{#1}(182,257){mm}}% JIS B5
383 \@namedef{Gm@b6j}#1{\Gm@setsize{#1}(128,182){mm}}% JIS B6
384 \@namedef{Gm@ansipaper}#1{\Gm@setsize{#1}(8.5,11){in}}%
385 \@namedef{Gm@ansibpaper}#1{\Gm@setsize{#1}(11,17){in}}%
386 \@namedef{Gm@ansicpaper}#1{\Gm@setsize{#1}(17,22){in}}%
387 \@namedef{Gm@ansidpaper}#1{\Gm@setsize{#1}(22,34){in}}%
388 \@namedef{Gm@ansiepaper}#1{\Gm@setsize{#1}(34,44){in}}%
389 \@namedef{Gm@letterpaper}#1{\Gm@setsize{#1}(8.5,11){in}}%
390 \@namedef{Gm@legalpaper}#1{\Gm@setsize{#1}(8.5,14){in}}%
391 \@namedef{Gm@executivepaper}#1{\Gm@setsize{#1}(7.25,10.5){in}}%
392 \@namedef{Gm@screen}#1{\Gm@setsize{#1}(225,180){mm}}%

```

‘paper’ paper prend le nom du format de papier comme sa valeur.

```

393 \define@key{Gm}{paper}{\setkeys{Gm}{#1}}%
394 \let\KV@Gm@papername\KV@Gm@paper

```

‘a[0-6]paper’ Les noms de format de papier disponibles sont les suivants.

‘b[0-6]paper’ 395 \define@key{Gm}{a0paper}[true]{\Gm@setpaper@ifpre{a0paper}}%

‘b[0-6]j’ 396 \define@key{Gm}{a1paper}[true]{\Gm@setpaper@ifpre{a1paper}}%

‘ansi[a-e]paper’ 397 \define@key{Gm}{a2paper}[true]{\Gm@setpaper@ifpre{a2paper}}%

‘letterpaper’ 398 \define@key{Gm}{a3paper}[true]{\Gm@setpaper@ifpre{a3paper}}%

‘legalpaper’ 399 \define@key{Gm}{a4paper}[true]{\Gm@setpaper@ifpre{a4paper}}%

‘executivepaper’ 400 \define@key{Gm}{a5paper}[true]{\Gm@setpaper@ifpre{a5paper}}%

‘screen’ 401 \define@key{Gm}{a6paper}[true]{\Gm@setpaper@ifpre{a6paper}}%

402 \define@key{Gm}{b0paper}[true]{\Gm@setpaper@ifpre{b0paper}}%

403 \define@key{Gm}{b1paper}[true]{\Gm@setpaper@ifpre{b1paper}}%

404 \define@key{Gm}{b2paper}[true]{\Gm@setpaper@ifpre{b2paper}}%

405 \define@key{Gm}{b3paper}[true]{\Gm@setpaper@ifpre{b3paper}}%

406 \define@key{Gm}{b4paper}[true]{\Gm@setpaper@ifpre{b4paper}}%

407 \define@key{Gm}{b5paper}[true]{\Gm@setpaper@ifpre{b5paper}}%

408 \define@key{Gm}{b6paper}[true]{\Gm@setpaper@ifpre{b6paper}}%

409 \define@key{Gm}{c0paper}[true]{\Gm@setpaper@ifpre{c0paper}}%

410 \define@key{Gm}{c1paper}[true]{\Gm@setpaper@ifpre{c1paper}}%

411 \define@key{Gm}{c2paper}[true]{\Gm@setpaper@ifpre{c2paper}}%

412 \define@key{Gm}{c3paper}[true]{\Gm@setpaper@ifpre{c3paper}}%

413 \define@key{Gm}{c4paper}[true]{\Gm@setpaper@ifpre{c4paper}}%

414 \define@key{Gm}{c5paper}[true]{\Gm@setpaper@ifpre{c5paper}}%

415 \define@key{Gm}{c6paper}[true]{\Gm@setpaper@ifpre{c6paper}}%

416 \define@key{Gm}{b0j}[true]{\Gm@setpaper@ifpre{b0j}}%

417 \define@key{Gm}{b1j}[true]{\Gm@setpaper@ifpre{b1j}}%

418 \define@key{Gm}{b2j}[true]{\Gm@setpaper@ifpre{b2j}}%

419 \define@key{Gm}{b3j}[true]{\Gm@setpaper@ifpre{b3j}}%

420 \define@key{Gm}{b4j}[true]{\Gm@setpaper@ifpre{b4j}}%

421 \define@key{Gm}{b5j}[true]{\Gm@setpaper@ifpre{b5j}}%

422 \define@key{Gm}{b6j}[true]{\Gm@setpaper@ifpre{b6j}}%

423 \define@key{Gm}{ansiapaper}[true]{\Gm@setpaper@ifpre{ansiapaper}}%

424 \define@key{Gm}{ansibpaper}[true]{\Gm@setpaper@ifpre{ansibpaper}}%

425 \define@key{Gm}{ansicpaper}[true]{\Gm@setpaper@ifpre{ansicpaper}}%

426 \define@key{Gm}{ansidpaper}[true]{\Gm@setpaper@ifpre{ansidpaper}}%

427 \define@key{Gm}{ansiepaper}[true]{\Gm@setpaper@ifpre{ansiepaper}}%

428 \define@key{Gm}{letterpaper}[true]{\Gm@setpaper@ifpre{letterpaper}}%

429 \define@key{Gm}{legalpaper}[true]{\Gm@setpaper@ifpre{legalpaper}}%

430 \define@key{Gm}{executivepaper}[true]{\Gm@setpaper@ifpre{executivepaper}}%

431 \define@key{Gm}{screen}[true]{\Gm@setpaper@ifpre{screen}}%

‘paperwidth’ La spécification directe du format est aussi possible.

‘paperheight’ 432 \define@key{Gm}{paperwidth}{\ifGm@preamble{paperwidth}{%

‘papersize’ 433 \def\Gm@paper{custom}\Gm@setlength\paperwidth{#1}}%

434 \define@key{Gm}{paperheight}{\ifGm@preamble{paperheight}{%

435 \def\Gm@paper{custom}\Gm@setlength\paperheight{#1}}%

436 \define@key{Gm}{papersize}{\ifGm@preamble{papersize}{%

437 \def\Gm@paper{custom}\Gm@branch{#1}{paperwidth}{paperheight}}%

‘layout’ La spécification du gabarit est aussi possible.

‘layoutwidth’ 438 \define@key{Gm}{layout}{\Gm@layouttrue\@nameuse{Gm#1}{Gm@layout}}%

‘layoutheight’ 439 \let\KV@Gm@layoutname\KV@Gm@layout

‘layoutsizes’ 440 \define@key{Gm}{layoutwidth}{\Gm@layouttrue\Gm@setlength\Gm@layoutwidth{#1}}%

441 \define@key{Gm}{layoutheight}{\Gm@layouttrue\Gm@setlength\Gm@layoutheight{#1}}%

442 \define@key{Gm}{layoutsizes}{\Gm@branch{#1}{layoutwidth}{layoutheight}}%

‘landscape’ Définition de l’orientation du papier.

‘portrait’ 443 \define@key{Gm}{landscape}[true]{\ifGm@preamble{landscape}{%

444 \Gm@doifelse{landscape}{#1}%

445 {\ifGm@landscape\else\Gm@landscapetrue\Gm@reverse{swap@papersize}\fi}%

446 {\ifGm@landscape\Gm@landscapefalse\Gm@reverse{swap@papersize}\fi}}%

447 \define@key{Gm}{portrait}[true]{\ifGm@preamble{portrait}{%

448 \Gm@doifelse{portrait}{#1}%

449 {\ifGm@landscape\Gm@landscapefalse\Gm@reverse{swap@papersize}\fi}%

450 {\ifGm@landscape\else\Gm@landscapetrue\Gm@reverse{swap@papersize}\fi}}%

‘hscale’ Ces options peuvent déterminer la ou les longueurs associées au *corps intégral* en se basant sur des

‘vscale’ rapports (*scales*) entre ces dimensions et le format du papier.

‘scale’

```

451 \define@key{Gm}{hscale}{\Gm@hbodytrue\edef\Gm@hscale{#1}}%
452 \define@key{Gm}{vscale}{\Gm@vbodytrue\edef\Gm@vscale{#1}}%
453 \define@key{Gm}{scale}{\Gm@branch{#1}{hscale}{vscale}}%

```

‘width’ Ces options donnent des dimensions fixées au *corps intégral*. totalwidth et totalheight sont des

‘height’ alias de width et height respectivement.

‘total’

```

454 \define@key{Gm}{width}{\Gm@hbodytrue\Gm@defbylen{width}{#1}}%

```

‘totalwidth’

```

455 \define@key{Gm}{height}{\Gm@vbodytrue\Gm@defbylen{height}{#1}}%

```

‘totalheight’

```

456 \define@key{Gm}{total}{\Gm@branch{#1}{width}{height}}%
457 \let\KV@Gm@totalwidth\KV@Gm@width
458 \let\KV@Gm@totalheight\KV@Gm@height

```

‘textwidth’ Ces options fixent directement les dimensions \textwidth et \textheight. body est un alias de

‘textheight’ text.

‘text’

```

459 \define@key{Gm}{textwidth}{\Gm@hbodytrue\Gm@defbylen{textwidth}{#1}}%

```

‘body’

```

460 \define@key{Gm}{textheight}{\Gm@vbodytrue\Gm@defbylen{textheight}{#1}}%
461 \define@key{Gm}{text}{\Gm@branch{#1}{textwidth}{textheight}}%
462 \let\KV@Gm@body\KV@Gm@text

```

‘lines’ Cette option fixe \textheight avec un nombre de lignes.

```

463 \define@key{Gm}{lines}{\Gm@vbodytrue\Gm@defbycnt{lines}{#1}}%

```

‘includehead’ Ces options impliquent que les dimensions associées sont ✱ parties prenantes du corps ✱.

‘includefoot’

```

464 \define@key{Gm}{includehead}[true]{\Gm@setbool{includehead}{#1}}%

```

‘includeheadfoot’

```

465 \define@key{Gm}{includefoot}[true]{\Gm@setbool{includefoot}{#1}}%

```

‘includemp’

```

466 \define@key{Gm}{includeheadfoot}[true]{\Gm@doifelse{includeheadfoot}{#1}}%

```

‘includeall’

```

467 {\Gm@includeheadtrue\Gm@includefoottrue}%
468 {\Gm@includeheadfalse\Gm@includefootfalse}%
469 \define@key{Gm}{includemp}[true]{\Gm@setbool{includemp}{#1}}%
470 \define@key{Gm}{includeall}[true]{\Gm@doifelse{includeall}{#1}}%
471 {\Gm@includeheadtrue\Gm@includefoottrue\Gm@includemptrue}%
472 {\Gm@includeheadfalse\Gm@includefootfalse\Gm@includempfalse}%

```

‘ignorehead’ Ces options excluent l’en-tête, le pied de page et les notes marginales lors de la détermination du corps.

‘ignorefoot’

```

473 \define@key{Gm}{ignorehead}[true]{%

```

‘ignoreheadfoot’

```

474 \Gm@setboolrev[ignorehead]{includehead}{#1}}%

```

‘ignoremp’

```

475 \define@key{Gm}{ignorefoot}[true]{%

```

‘ignoreall’

```

476 \Gm@setboolrev[ignorefoot]{includefoot}{#1}}%
477 \define@key{Gm}{ignoreheadfoot}[true]{\Gm@doifelse{ignoreheadfoot}{#1}}%
478 {\Gm@includeheadfalse\Gm@includefootfalse}%
479 {\Gm@includeheadtrue\Gm@includefoottrue}}%
480 \define@key{Gm}{ignoremp}[true]{%
481 \Gm@setboolrev[ignoremp]{includemp}{#1}}%
482 \define@key{Gm}{ignoreall}[true]{\Gm@doifelse{ignoreall}{#1}}%
483 {\Gm@includeheadfalse\Gm@includefootfalse\Gm@includempfalse}%
484 {\Gm@includeheadtrue\Gm@includefoottrue\Gm@includemptrue}}%

```

‘heightrounded’ Cette option arrondit \textheight à n fois \baselineskip plus \topskip.

```

485 \define@key{Gm}{heightrounded}[true]{\Gm@setbool{heightrounded}{#1}}%

```

‘hdivide’ Cette option sont utiles pour spécifier la décomposition du papier dans chaque direction.

‘vdivide’

```

486 \define@key{Gm}{hdivide}{\Gm@parse@divide{#1}{lmargin}{width}{rmargin}}%

```

‘divide’

```

487 \define@key{Gm}{vdivide}{\Gm@parse@divide{#1}{tmargin}{height}{bmargin}}%
488 \define@key{Gm}{divide}{\Gm@parse@divide{#1}{lmargin}{width}{rmargin}}%
489 \Gm@parse@divide{#1}{tmargin}{height}{bmargin}}%

```

‘lmargin’ Ces options fixent les marges. left, inner et innermargin sont des alias de lmargin. right, outer

‘rmargin’ et outermargin sont des alias de rmargin. top et bottom sont des alias de tmargin et bmargin

‘tmargin’ respectivement.

‘bmargin’

```

490 \define@key{Gm}{lmargin}{\Gm@defbylen{lmargin}{#1}}%

```

‘left’

```

491 \define@key{Gm}{rmargin}{\Gm@defbylen{rmargin}{#1}}%

```

‘inner’

‘innermargin’

‘right’

‘outer’

‘outermargin’

```

492 \let\KV@Gm@left\KV@Gm@lmargin
493 \let\KV@Gm@inner\KV@Gm@lmargin
494 \let\KV@Gm@innermargin\KV@Gm@lmargin
495 \let\KV@Gm@right\KV@Gm@rmargin
496 \let\KV@Gm@outer\KV@Gm@rmargin
497 \let\KV@Gm@outermargin\KV@Gm@rmargin
498 \define@key{Gm}{tmargin}{\Gm@defbylen{tmargin}{#1}}%
499 \define@key{Gm}{bmargin}{\Gm@defbylen{bmargin}{#1}}%
500 \let\KV@Gm@top\KV@Gm@tmargin
501 \let\KV@Gm@bottom\KV@Gm@bmargin

‘hmargin’ Ces options sont des raccourcis pour définir les marges.
‘vmargin’ 502 \define@key{Gm}{hmargin}{\Gm@branch{#1}{lmargin}{rmargin}}%
‘margin’ 503 \define@key{Gm}{vmargin}{\Gm@branch{#1}{tmargin}{bmargin}}%
504 \define@key{Gm}{margin}{\Gm@branch{#1}{lmargin}{tmargin}}%
505 \Gm@branch{#1}{rmargin}{bmargin}}%

‘hmarginratio’ Ces options spécifient les rapports entre marges.
‘vmarginratio’ 506 \define@key{Gm}{hmarginratio}{\edef\Gm@hmarginratio{#1}}%
‘marginratio’ 507 \define@key{Gm}{vmarginratio}{\edef\Gm@vmarginratio{#1}}%
‘hratio’ 508 \define@key{Gm}{marginratio}{\Gm@branch{#1}{hmarginratio}{vmarginratio}}%
‘vratio’ 509 \let\KV@Gm@hratio\KV@Gm@hmarginratio
‘ratio’ 510 \let\KV@Gm@vratio\KV@Gm@vmarginratio
511 \let\KV@Gm@ratio\KV@Gm@marginratio

‘hcentering’ Quelques raccourcis utiles pour centrer le corps.
‘vcentering’ 512 \define@key{Gm}{hcentering}[true]{\Gm@doifelse{hcentering}{#1}}%
‘centering’ 513 {\def\Gm@hmarginratio{1:1}}}%
514 \define@key{Gm}{vcentering}[true]{\Gm@doifelse{vcentering}{#1}}%
515 {\def\Gm@vmarginratio{1:1}}}%
516 \define@key{Gm}{centering}[true]{\Gm@doifelse{centering}{#1}}%
517 {\def\Gm@hmarginratio{1:1}\def\Gm@vmarginratio{1:1}}}%

✖

‘twoside’ If twoside=true, \@twoside and \@mparswitch is set to true.
518 \define@key{Gm}{twoside}[true]{\Gm@doifelse{twoside}{#1}}%
519 {\@twosidetrue\@mparswitchtrue}{\@twosidefalse\@mparswitchfalse}}%

‘asymmetric’ asymmetric sets \@mparswitchfalse and \@twosidetrue A asymmetric=false has no effect.
520 \define@key{Gm}{asymmetric}[true]{\Gm@doifelse{asymmetric}{#1}}%
521 {\@twosidetrue\@mparswitchfalse}}%

‘bindingoffset’ The macro adds the specified space to the inner margin.
522 \define@key{Gm}{bindingoffset}{\Gm@setlength\Gm@bindingoffset{#1}}%

‘headheight’ The direct settings of en-tête and/or pied de page dimensions.
‘headsep’ 523 \define@key{Gm}{headheight}{\Gm@setlength\headheight{#1}}%
‘footskip’ 524 \define@key{Gm}{headsep}{\Gm@setlength\headsep{#1}}%
‘head’ 525 \define@key{Gm}{footskip}{\Gm@setlength\footskip{#1}}%
‘foot’ 526 \let\KV@Gm@head\KV@Gm@headheight
527 \let\KV@Gm@foot\KV@Gm@footskip

‘nohead’ They are only shorthands to set en-tête and/or pied de page to be Opt.
‘nofoot’ 528 \define@key{Gm}{nohead}[true]{\Gm@doifelse{nohead}{#1}}%
‘noheadfoot’ 529 {\Gm@setlength\headheight\z@\Gm@setlength\headsep\z@}{}}%
530 \define@key{Gm}{nofoot}[true]{\Gm@doifelse{nofoot}{#1}}%
531 {\Gm@setlength\footskip\z@}{}}%
532 \define@key{Gm}{noheadfoot}[true]{\Gm@doifelse{noheadfoot}{#1}}%
533 {\Gm@setlength\headheight\z@\Gm@setlength\headsep\z@\Gm@setlength\footskip\z@}{}}%
534 \z@\Gm@setlength\footskip\z@}{}}%

‘footnotesep’ The option directly sets a native dimension \footnotesep.
535 \define@key{Gm}{footnotesep}{\Gm@setlength\skip\footins{#1}}%

```


‘marginparwidth’ They directly set native dimensions `\marginparwidth` and `\marginparsep`.

‘marginpar’ 536 `\define@key{Gm}{marginparwidth}{\Gm@setlength\marginparwidth{#1}}%`

‘marginparsep’ 537 `\let\KV@Gm@marginpar\KV@Gm@marginparwidth`
538 `\define@key{Gm}{marginparsep}{\Gm@setlength\marginparsep{#1}}%`

‘nomarginpar’ The macro is a shorthand for `\marginparwidth=0pt` and `\marginparsep=0pt`.

539 `\define@key{Gm}{nomarginpar}[true]{\Gm@doifelse{nomarginpar}{#1}%`
540 `{\Gm@setlength\marginparwidth\z@\Gm@setlength\marginparsep\z@}{}}%`

‘columnsep’ The option sets a native dimension `\columnsep`.

541 `\define@key{Gm}{columnsep}{\Gm@setlength\columnsep{#1}}%`

‘hoffset’ The former two options set native dimensions `\hoffset` and `\voffset`. `offset` can set both of them

‘voffset’ with the same value.

‘offset’ 542 `\define@key{Gm}{hoffset}{\Gm@setlength\hoffset{#1}}%`
543 `\define@key{Gm}{voffset}{\Gm@setlength\voffset{#1}}%`
544 `\define@key{Gm}{offset}{\Gm@branch{#1}{hoffset}{voffset}}%`

‘layouthoffset’

‘layoutvoffset’ 545 `\define@key{Gm}{layouthoffset}{\Gm@setlength\Gm@layouthoffset{#1}}%`

‘layoutoffset’ 546 `\define@key{Gm}{layoutvoffset}{\Gm@setlength\Gm@layoutvoffset{#1}}%`
547 `\define@key{Gm}{layoutoffset}{\Gm@branch{#1}{layouthoffset}{layoutvoffset}}%`

‘twocolumn’ The option sets `\twocolumn` switch.

548 `\define@key{Gm}{twocolumn}[true]{%`
549 `\Gm@doif{twocolumn}{#1}{\csname @twocolumn\Gm@bool\endcsname}}%`

‘onecolumn’ This option has the reverse effect of `twocolumn` option.

550 `\define@key{Gm}{onecolumn}[true]{%`
551 `\Gm@doifelse{onecolumn}{#1}{\@twocolumnfalse}{\@twocolumntrue}}%`

‘reversemp’ The both options set `\reversemargin`.

‘reversemarginpar’ 552 `\define@key{Gm}{reversemp}[true]{%`
553 `\Gm@doif{reversemp}{#1}{\csname @reversemargin\Gm@bool\endcsname}}%`
554 `\define@key{Gm}{reversemarginpar}[true]{%`
555 `\Gm@doif{reversemarginpar}{#1}{\csname @reversemargin\Gm@bool\endcsname}}%`

‘dviver’

556 `\define@key{Gm}{driver}{\ifGm@preamble{driver}{%`
557 `\edef\@tempa{#1}\edef\@auto{auto}\edef\@none{none}%`
558 `\ifx\@tempa\@empty\let\Gm@driver\relax\else`
559 `\ifx\@tempa\@none\let\Gm@driver\relax\else`
560 `\ifx\@tempa\@auto\let\Gm@driver\@empty\else`
561 `\setkeys{Gm}{#1}\fi\fi\fi\let\@auto\relax\let\@none\relax}}%`

‘dvips’ The geometry package supports `dvips`, `dvipdfm`, `pdflatex` and `vtex`. `dvipdfm` works like `dvips`.

‘dvipdfm’ 562 `\define@key{Gm}{dvips}[true]{\ifGm@preamble{dvips}{%`

‘pdftex’ 563 `\Gm@doifelse{dvips}{#1}{\Gm@setdriver{dvips}}{\Gm@unsetdriver{dvips}}}%`

‘xetex’ 564 `\define@key{Gm}{dvipdfm}[true]{\ifGm@preamble{dvipdfm}{%`

‘vtex’ 565 `\Gm@doifelse{dvipdfm}{#1}{\Gm@setdriver{dvipdfm}}{\Gm@unsetdriver{dvipdfm}}}%`
566 `\define@key{Gm}{pdftex}[true]{\ifGm@preamble{pdftex}{%`
567 `\Gm@doifelse{pdftex}{#1}{\Gm@setdriver{pdftex}}{\Gm@unsetdriver{pdftex}}}%`
568 `\define@key{Gm}{xetex}[true]{\ifGm@preamble{xetex}{%`
569 `\Gm@doifelse{xetex}{#1}{\Gm@setdriver{xetex}}{\Gm@unsetdriver{xetex}}}%`
570 `\define@key{Gm}{vtex}[true]{\ifGm@preamble{vtex}{%`
571 `\Gm@doifelse{vtex}{#1}{\Gm@setdriver{vtex}}{\Gm@unsetdriver{vtex}}}%`

‘verbose’ The verbose mode.

572 `\define@key{Gm}{verbose}[true]{\ifGm@preamble{verbose}{\Gm@setbool{verbose}{#1}}%`

‘reset’ The option cancels all the options specified before `reset`, except `pass. mag` ($\neq 1000$) with `truedimen` cannot be also reset.

573 `\define@key{Gm}{reset}[true]{\ifGm@preamble{reset}{%`
574 `\Gm@doifelse{reset}{#1}{\Gm@restore@org\Gm@initall`
575 `\ProcessOptionsKV[c]{Gm}\Gm@setdefaultpaper{}}}%`

‘resetpaper’ If `resetpaper` is set to `true`, the paper size redefined in the package is discarded and the original one is restored. This option may be useful to print nonstandard sized documents with normal printers and papers.

```
576 \define@key{Gm}{resetpaper}[true]{\ifGm@preamble{resetpaper}{%
577   \Gm@setbool{resetpaper}{#1}}}%
```

‘mag’ `mag` is expanded immediately when it is specified. So `reset` can’t reset `mag` when it is set with `truedimen`.

```
578 \define@key{Gm}{mag}{\ifGm@preamble{mag}{\mag=#1}}%
```

‘truedimen’ If `truedimen` is set to `true`, all of the internal explicit dimensions is changed to *true* dimensions, e.g., `1in` is changed to `1truein`.

```
579 \define@key{Gm}{truedimen}[true]{\ifGm@preamble{truedimen}{%
580   \Gm@doifelse{truedimen}{#1}{\let\Gm@truedimen\Gm@true}%
581   {\let\Gm@truedimen\@empty}}}%
```

‘pass’ The option makes all the options specified ineffective except verbose switch.

```
582 \define@key{Gm}{pass}[true]{\ifGm@preamble{pass}{\Gm@setbool{pass}{#1}}}%
```

‘showframe’ The `showframe` option prints page frames to help you understand what the resulting layout is like.

```
583 \define@key{Gm}{showframe}[true]{\Gm@setbool{showframe}{#1}}%
```

‘showcrop’ The `showcrop` option prints crop marks at each corner of the layout area.

```
584 \define@key{Gm}{showcrop}[true]{\Gm@setbool{showcrop}{#1}}%
```

`\Gm@setdefaultpaper` The macro stores paper dimensions. This macro should be called after `\ProcessOptionsKV[c]{Gm}`. If the `landscape` option in `\documentclass` is specified, the class immediately swaps the paper dimensions.

```
585 \def\Gm@setdefaultpaper{%
586   \ifx\Gm@paper\@undefined
587     \Gm@setsize{paper}{\strip@pt\paperwidth,\strip@pt\paperheight}{pt}%
588     \Gm@setsize{Gm@layout}{\strip@pt\paperwidth,\strip@pt\paperheight}{pt}%
589     \Gm@swap@papersizefalse
590   \fi}%
```

`\Gm@adjustpaper` The macro checks if `paperwidth/height` is larger than `0pt`, which is used in `\Gm@process`. The paper dimensions can be swapped when paper orientation is changed over by `landscape` and `portrait` options.

```
591 \def\Gm@adjustpaper{%
592   \ifdim\paperwidth>\p@else
593     \PackageError{geometry}{%
594       \string\paperwidth\space(\the\paperwidth) too short}{%
595       Set a paper type (e.g., ‘a4paper’).}%
596   \fi
597   \ifdim\paperheight>\p@else
598     \PackageError{geometry}{%
599       \string\paperheight\space(\the\paperheight) too short}{%
600       Set a paper type (e.g., ‘a4paper’).}%
601   \fi
602   \ifGm@swap@papersize
603     \setlength\@tempdima{\paperwidth}%
604     \setlength\paperwidth{\paperheight}%
605     \setlength\paperheight{\@tempdima}%
606   \fi
607   \ifGm@layout\else
608     \setlength\Gm@layoutwidth{\paperwidth}%
609     \setlength\Gm@layoutheight{\paperheight}%
610   \fi}%
```

`\Gm@checkmp` The macro checks whether or not the `marginpars` overrun the page.

```
611 \def\Gm@checkmp{%
612   \ifGm@includemp\else
613     \@tempcnta\z@\@tempcntb\@ne
```

```

614 \if@twocolumn
615 \@tempcnta\@ne
616 \else
617 \if@reversemargin
618 \@tempcnta\@ne\@tempcntb\z@
619 \fi
620 \fi
621 \@tempdima\marginparwidth
622 \advance\@tempdima\marginparsep
623 \ifnum\@tempcnta=\@ne
624 \@tempdimc\@tempdima
625 \setlength\@tempdimb{\Gm@lmargin}%
626 \advance\@tempdimc-\@tempdimb
627 \ifdim\@tempdimc>\z@
628 \Gm@warning{The marginal notes overrun the paper edge.^^J
629 \spaces Add \the\@tempdimc\space and more to the left margin}%
630 \fi
631 \fi
632 \ifnum\@tempcntb=\@ne
633 \@tempdimc\@tempdima
634 \setlength\@tempdimb{\Gm@rmargin}%
635 \advance\@tempdimc-\@tempdimb
636 \ifdim\@tempdimc>\z@
637 \Gm@warning{The marginal notes overrun the paper.^^J
638 \spaces Add \the\@tempdimc\space and more to the right margin}%
639 \fi
640 \fi
641 \fi}%

```

\Gm@adjustmp The macro sets marginpar correction when `includemp` is set, which is used in `\Gm@process`. The variables `\Gm@wd@mp`, `\Gm@odd@mp` and `\Gm@even@mp` are set here. Note that `\Gm@even@mp` should be used only for twoside layout.

```

642 \def\Gm@adjustmp{%
643 \ifGm@includemp
644 \@tempdimb\marginparwidth
645 \advance\@tempdimb\marginparsep
646 \Gm@wd@mp\@tempdimb
647 \Gm@odd@mp\z@
648 \Gm@even@mp\z@
649 \if@twocolumn
650 \Gm@wd@mp2\@tempdimb
651 \Gm@odd@mp\@tempdimb
652 \Gm@even@mp\@tempdimb
653 \else
654 \if@reversemargin
655 \Gm@odd@mp\@tempdimb
656 \if@mparswitch\else
657 \Gm@even@mp\@tempdimb
658 \fi
659 \else
660 \if@mparswitch
661 \Gm@even@mp\@tempdimb
662 \fi
663 \fi
664 \fi
665 \fi}%

```

\Gm@adjustbody If the horizontal dimension of *corps* is specified by user, `\Gm@width` is set properly here.

```

666 \def\Gm@adjustbody{
667 \ifGm@hbody
668 \ifx\Gm@width\undefined
669 \ifx\Gm@hscale\undefined
670 \Gm@defbylen{width}{\Gm@Dhscale\Gm@layoutwidth}%
671 \else
672 \Gm@defbylen{width}{\Gm@hscale\Gm@layoutwidth}%

```

```

673     \fi
674 \fi
675 \ifx\Gm@textwidth\@undefined\else
676     \setlength\@tempdima{\Gm@textwidth}%
677     \ifGm@includemp
678         \advance\@tempdima\Gm@wd@mp
679     \fi
680     \edef\Gm@width{\the\@tempdima}%
681 \fi
682 \fi

```

If the vertical dimension of *corps* is specified by user, \Gm@height is set properly here.

```

683 \ifGm@vbody
684     \ifx\Gm@height\@undefined
685         \ifx\Gm@vscale\@undefined
686             \Gm@defbylen{height}{\Gm@Dvscale\Gm@layoutheight}%
687         \else
688             \Gm@defbylen{height}{\Gm@vscale\Gm@layoutheight}%
689         \fi
690     \fi
691 \ifx\Gm@lines\@undefined\else

```

\topskip has to be adjusted so that the formula “ $\text{textheight} = (\text{lines} - 1) \times \text{baselineskip} + \text{topskip}$ ” to be correct even if large font sizes are specified by users. If \topskip is smaller than \ht\strutbox, then \topskip is set to \ht\strutbox.

```

692     \ifdim\topskip<\ht\strutbox
693         \setlength\@tempdima{\topskip}%
694         \setlength\topskip{\ht\strutbox}%
695         \Gm@warning{\noexpand\topskip was changed from \the\@tempdima\space
696             to \the\topskip}%
697     \fi
698     \setlength\@tempdima{\baselineskip}%
699     \multiply\@tempdima\Gm@lines
700     \addtolength\@tempdima{\topskip}%
701     \addtolength\@tempdima{-\baselineskip}%
702     \edef\Gm@textheight{\the\@tempdima}%
703 \fi
704 \ifx\Gm@textheight\@undefined\else
705     \setlength\@tempdima{\Gm@textheight}%
706     \ifGm@includehead
707         \addtolength\@tempdima{\headheight}%
708         \addtolength\@tempdima{\headsep}%
709     \fi
710     \ifGm@includefoot
711         \addtolength\@tempdima{\footskip}%
712     \fi
713     \edef\Gm@height{\the\@tempdima}%
714 \fi
715 \fi}%

```

\Gm@process The main macro processing the specified dimensions is defined.

```

716 \def\Gm@process{%

```

If pass is set, the original dimensions and switches are restored and process is ended here.

```

717 \ifGm@pass
718     \Gm@restore@org
719 \else
720     \Gm@@process
721 \fi}%

```

The main processing macro.

```

722 \def\Gm@@process{%
723     \Gm@expandlengths
724     \Gm@adjustpaper
725     \addtolength\Gm@layoutwidth{-\Gm@bindingoffset}%
726     \Gm@adjustmp
727     \Gm@adjustbody

```

```

728 \Gm@detall{h}{width}{lmargin}{rmargin}%
729 \Gm@detall{v}{height}{tmargin}{bmargin}%

```

The real dimensions are set properly according to the result of the auto-completion calculation.

```

730 \setlength\textwidth{\Gm@width}%
731 \setlength\textheight{\Gm@height}%
732 \setlength\topmargin{\Gm@tmargin}%
733 \setlength\oddsidemargin{\Gm@lmargin}%
734 \addtolength\oddsidemargin{-1\Gm@truedimen in}%

```

If `includemp` is set to true, `\textwidth` and `\oddsidemargin` are adjusted.

```

735 \ifGm@includemp
736     \advance\textwidth-\Gm@wd@mp
737     \advance\oddsidemargin\Gm@odd@mp
738 \fi

```

Determining `\evensidemargin`. In the twoside page layout, the right margin value `\Gm@rmargin` is used. If the marginal note width is included, `\evensidemargin` should be corrected by `\Gm@even@mp`.

```

739 \if@mparswitch
740     \setlength\evensidemargin{\Gm@rmargin}%
741     \addtolength\evensidemargin{-1\Gm@truedimen in}%
742     \ifGm@includemp
743         \advance\evensidemargin\Gm@even@mp
744     \fi
745 \else
746     \evensidemargin\oddsidemargin
747 \fi

```

The `bindingoffset` correction for `\oddsidemargin`.

```

748 \advance\oddsidemargin\Gm@bindingoffset
749 \addtolength\topmargin{-1\Gm@truedimen in}%

```

If the head of the page is included in *corps intégral*, `\headheight` and `\headsep` are removed from `\textheight`, otherwise from `\topmargin`.

```

750 \ifGm@includehead
751     \addtolength\textheight{-\headheight}%
752     \addtolength\textheight{-\headsep}%
753 \else
754     \addtolength\topmargin{-\headheight}%
755     \addtolength\topmargin{-\headsep}%
756 \fi

```

If the foot of the page is included in *corps intégral*, `\footskip` is removed from `\textheight`.

```

757 \ifGm@includefoot
758     \addtolength\textheight{-\footskip}%
759 \fi

```

If `heightrounded` is set, `\textheight` is rounded.

```

760 \ifGm@heightrounded
761     \setlength\@tempdima{\textheight}%
762     \addtolength\@tempdima{-\topskip}%
763     \@tempcnta\@tempdima
764     \@tempcntb\baselineskip
765     \divide\@tempcnta\@tempcntb
766     \setlength\@tempdimb{\baselineskip}%
767     \multiply\@tempdimb\@tempcnta
768     \advance\@tempdima-\@tempdimb
769     \multiply\@tempdima\tw@
770     \ifdim\@tempdima>\baselineskip
771         \addtolength\@tempdimb{\baselineskip}%
772     \fi
773     \addtolength\@tempdimb{\topskip}%
774     \textheight\@tempdimb
775 \fi

```

The paper width is set back by adding `\Gm@bindingoffset`.

```

776 \advance\oddsidemargin\Gm@layouthoffset%
777 \advance\evensidemargin\Gm@layouthoffset%

```

```

778 \advance\topmargin\Gm@layoutvoffset%
779 \addtolength\Gm@layoutwidth{\Gm@bindingoffset}%
780 }% end of \Gm@@process

```

\Gm@detectdriver The macro checks the typeset environment and changes the driver option if necessary. To make the engine detection more robust, the macro is rewritten with packages ifpdf, ifvtex and ifxetex.

```

781 \def\Gm@detectdriver{%
  If the driver option is not specified explicitly, then driver auto-detection works.

782 \ifx\Gm@driver\@empty
783 \typeout{*geometry* driver: auto-detecting}%
  \ifpdf is defined in ifpdf package in ‘oberdiek’ bundle.
784 \ifpdf
785 \Gm@setdriver{pdftex}%
786 \else
787 \Gm@setdriver{dvips}%
788 \fi

  \ifvtex is defined in ifvtex package in ‘oberdiek’ bundle.
789 \ifvtex
790 \Gm@setdriver{vtex}%
791 \fi

  \ifxetex is defined in ifxetex package written by Will Robertson.
792 \ifxetex
793 \Gm@setdriver{xetex}
794 \fi

  When the driver option is set by the user, check if it is valid or not.
795 \else
796 \ifx\Gm@driver\Gm@xetex %%
797 \ifxetex\else
798 \Gm@warning{Wrong driver setting: ‘xetex’; trying ‘pdftex’ driver}%
799 \Gm@setdriver{pdftex}
800 \fi
801 \fi
802 \ifx\Gm@driver\Gm@vtex
803 \ifvtex\else
804 \Gm@warning{Wrong driver setting: ‘vtex’; trying ‘dvips’ driver}%
805 \Gm@setdriver{dvips}%
806 \fi
807 \fi
808 \fi
809 \ifx\Gm@driver\relax
810 \typeout{*geometry* detected driver: <none>}%
811 \else
812 \typeout{*geometry* detected driver: \Gm@driver}%
813 \fi}%

```

\Gm@showparams Prints the resulted parameters and dimensions to STDOUT if verbose is true. \Gm@width and \Gm@height are expanded to get the real size.

```

814 \def\Gm@showparams#1{%
815 \ifGm@verbose\expandafter\typeout\else\expandafter\wlog\fi
816 {\Gm@logcontent{#1}}}%
817 \def\Gm@showdim#1{* \string#1=\the#1^^J}%
818 \def\Gm@showbool#1{\@nameuse{ifGm@#1}#1\space\fi}%

```

\Gm@logcontent The content of geometry parameters and native dimensions for the page layout.

```

819 \def\Gm@logcontent#1{%
820 *geometry* verbose mode - [ #1 ] result:^^J%
821 \ifGm@pass * pass: disregarded the geometry package!^^J%
822 \else
823 * driver: \if\Gm@driver<none>\else\Gm@driver\fi^^J%
824 * paper: \ifx\Gm@paper\@undefined<default>\else\Gm@paper\fi^^J%
825 * layout: \ifGm@layout<custom>\else<same size as paper>\fi^^J%
826 \ifGm@layout

```

```

827 * layout(width,height): (\the\Gm@layoutwidth,\the\Gm@layoutheight)^^J%
828 \fi
829 * layoutoffset:(h,v)=(\the\Gm@layoutoffset,\the\Gm@layoutvoffset)^^J%
830 \@ifundefined{Gm@lines}{* lines: \Gm@lines^^J}%
831 \@ifundefined{Gm@hmarginratio}{* hratio: \Gm@hmarginratio^^J}%
832 \@ifundefined{Gm@vmarginratio}{* vratio: \Gm@vmarginratio^^J}%
833 \ifdim\Gm@bindingoffset=\z@else
834 * bindingoffset: \the\Gm@bindingoffset^^J\fi
835 * modes: %
836 \Gm@showbool{landscape}%
837 \Gm@showbool{includehead}%
838 \Gm@showbool{includefoot}%
839 \Gm@showbool{includemp}%
840 \if@twoside twoside\space\fi%
841 \if@mparswitchelse\if@twoside asymmetric\space\fi\fi%
842 \Gm@showbool{heightrounded}%
843 \ifx\Gm@truedimen\emptyelse truedimen\space\fi%
844 \Gm@showbool{showframe}%
845 \Gm@showbool{showcrop}%
846 ^^J%
847 * h-part:(L,W,R)=(\Gm@lmargin, \Gm@width, \Gm@rmargin)^^J%
848 * v-part:(T,H,B)=(\Gm@tmargin, \Gm@height, \Gm@bmargin)^^J%
849 \fi
850 \Gm@showdim{\paperwidth}%
851 \Gm@showdim{\paperheight}%
852 \Gm@showdim{\textwidth}%
853 \Gm@showdim{\textheight}%
854 \Gm@showdim{\oddsidemargin}%
855 \Gm@showdim{\evensidemargin}%
856 \Gm@showdim{\topmargin}%
857 \Gm@showdim{\headheight}%
858 \Gm@showdim{\headsep}%
859 \Gm@showdim{\topskip}%
860 \Gm@showdim{\footskip}%
861 \Gm@showdim{\marginparwidth}%
862 \Gm@showdim{\marginparsep}%
863 \Gm@showdim{\columnsep}%
864 * \string\skip\string\footins=\the\skip\footins^^J%
865 \Gm@showdim{\hoffset}%
866 \Gm@showdim{\voffset}%
867 \Gm@showdim{\mag}%
868 * \string@twocolumn\if@twocolumn trueelse false\fi^^J%
869 * \string@twoside\if@twoside trueelse false\fi^^J%
870 * \string@mparswitch\if@mparswitch trueelse false\fi^^J%
871 * \string@reversemargin\if@reversemargin trueelse false\fi^^J%
872 * (1in=72.27pt=25.4mm, 1cm=28.453pt)^^J%

```

Macros for the page frames and cropmarks.

```

873 \def\Gm@cropmark(#1,#2,#3,#4){%
874 \begin{picture}(0,0)
875 \setlength\unitlength{1truemm}%
876 \linethickness{0.25pt}%
877 \put(#3,0){\line(#1,0){17}}%
878 \put(0,#4){\line(0,#2){17}}%
879 \end{picture}}%
880 \providecommand*{\vb@xt@{\vbox to}%
881 \def\Gm@vrule{\vrule width 0.2pt height\textheight depth\z@}%
882 \def\Gm@hrule{\hrule height 0.2pt depth\z@ width\textwidth}%
883 \def\Gm@hruled{\hrule height\z@ depth0.2pt width\textwidth}%
884 \newcommand*{\Gm@vrules@mpi}{%
885 \hb@xt@{\tempdima}{\llap{\Gm@vrule}\ignorespaces
886 \hskip \textwidth\Gm@vrule\hskip \marginparsep
887 \llap{\Gm@vrule}\hfil\Gm@vrule}}%
888 \newcommand*{\Gm@vrules@mpii}{%
889 \hb@xt@{\tempdima}{\hskip-\marginparwidth\hskip-\marginparsep

```



```

890 \llap{\Gm@vrule}\ignorespaces
891 \hskip \marginparwidth\rlap{\Gm@vrule}\hskip \marginparsep
892 \llap{\Gm@vrule}\hskip\textwidth\rlap{\Gm@vrule}\hss}}%
893 \newcommand*{\Gm@pageframes}{%
894 \vb@xt@\z{%
895 \ifGm@showcrop
896 \vb@xt@\z{\vskip-1\Gm@truedimen in\vskip\Gm@layoutvoffset%
897 \hb@xt@\z{\hskip-1\Gm@truedimen in\hskip\Gm@layouthoffset%
898 \vb@xt@\Gm@layoutheight{%
899 \let\protect\relax
900 \hb@xt@\Gm@layoutwidth{\Gm@cropmark(-1,1,-3,3)\hfil\Gm@cropmark(1,1,3,3)}}%
901 \vfil
902 \hb@xt@\Gm@layoutwidth{\Gm@cropmark(-1,-1,-3,-3)\hfil\Gm@cropmark(1,-1,3,-3)}}%
903 \hss}}%
904 \vss}}%
905 \fi%
906 \ifGm@showframe
907 \if@twoside
908 \ifodd\count\z@
909 \let\@themargin\oddsidemargin
910 \else
911 \let\@themargin\evensidemargin
912 \fi
913 \fi
914 \moveright\@themargin%
915 \vb@xt@\z{%
916 \vskip\topmargin\vb@xt@\z{\vss\Gm@hrule}%
917 \vskip\headheight\vb@xt@\z{\vss\Gm@hruled}%
918 \vskip\headsep\vb@xt@\z{\vss\Gm@hrule}%
919 \@tempdima\textwidth
920 \advance\@tempdima by \marginparsep
921 \advance\@tempdima by \marginparwidth
922 \if@mparswitch
923 \ifodd\count\z@
924 \Gm@vrules@mpi
925 \else
926 \Gm@vrules@mpii
927 \fi
928 \else
929 \Gm@vrules@mpi
930 \fi
931 \vb@xt@\z{\vss\Gm@hrule}%
932 \vskip\footskip\vb@xt@\z{\vss\Gm@hruled}%
933 \vss}}%
934 \fi%
935 }}}%

```

\ProcessOptionsKV This macro can process class and package options using ‘key=value’ scheme. Only class options are processed with an optional argument ‘c’, package options with ‘p’, and both of them by default.

```

936 \def\ProcessOptionsKV{\@ifnextchar[%]
937 {\@ProcessOptionsKV}{\@ProcessOptionsKV[]}}%
938 \def\@ProcessOptionsKV[#1]#2{%
939 \let\@tempa\@empty
940 \@tempcnta\z@
941 \if#1p\@tempcnta\@ne\else\if#1c\@tempcnta\tw\fi\fi
942 \ifodd\@tempcnta
943 \edef\@tempa{\@optionlist{\@currname.\@current}}%
944 \else
945 \@for\CurrentOption:=\@classoptionslist\do{%
946 \@ifundefined{KV@#2@\CurrentOption}%
947 {}{\edef\@tempa{\@tempa,\CurrentOption,}}}%
948 \ifnum\@tempcnta=\z@
949 \edef\@tempa{\@tempa,\@optionlist{\@currname.\@current}}%
950 \fi
951 \fi

```

```

952 \edef\@tempa{\noexpand\setkeys{#2}{\@tempa}}%
953 \@tempa
954 \AtEndOfPackage{\let\@unprocessedoptions\relax}}%

955 \def\Gm@setkeys{\setkeys{Gm}}%

\Gm@processconf \ExecuteOptions is replaced with \Gm@setkey to make it possible to deal with 'key=value' as
its argument.

956 \def\Gm@processconfig{%
957 \let\Gm@origExecuteOptions\ExecuteOptions
958 \let\ExecuteOptions\Gm@setkeys
959 \InputIfFileExists{geometry.cfg}{-}{-}
960 \let\ExecuteOptions\Gm@origExecuteOptions}%

The original page layout before loading geometry is saved here. \Gm@restore@org is defined here
for reset option.

961 \Gm@save
962 \edef\Gm@restore@org{\Gm@restore}%
963 \Gm@initall

Processing config file.

964 \Gm@processconfig

The optional arguments to \documentclass are processed here.

965 \ProcessOptionsKV[c]{Gm}%

Paper dimensions given by class default are stored.

966 \Gm@setdefaultpaper

The optional arguments to \usepackage are processed here.

967 \ProcessOptionsKV[p]{Gm}%

Actual settings and calculation for layout dimensions are processed.

968 \Gm@process

\AtBeginDocument The processes for verbose, showframe and drivers are added to \AtBeginDocument. \Gm@restore@org
is redefined here with the paper size specified in the preamble for \newgeometry to use it. This should
be done before magnifying the paper size with \mag because the layout calculation would be affected
by changing the paper size.

969 \AtBeginDocument{%
970 \Gm@savelength{paperwidth}%
971 \Gm@savelength{paperheight}%
972 \edef\Gm@restore@org{\Gm@restore}%

The original paper size is used if resetpaper.

973 \ifGm@resetpaper
974 \edef\Gm@pw{\Gm@orgpw}%
975 \edef\Gm@ph{\Gm@orgph}%
976 \else
977 \edef\Gm@pw{\the\paperwidth}%
978 \edef\Gm@ph{\the\paperheight}%
979 \fi

If pass is not set, the paper size is multiplied according to the specified mag.

980 \ifGm@pass\else
981 \ifnum\mag=\@m\else
982 \Gm@magtooffset
983 \divide\paperwidth\@m
984 \multiply\paperwidth\the\mag
985 \divide\paperheight\@m
986 \multiply\paperheight\the\mag
987 \fi
988 \fi

Checking the driver options.

989 \Gm@detectdriver

```

If `xetex` and `\pdfpagewidth` is defined, `\pdfpagewidth` and `\pdfpageheight` would be set.

```

990 \ifx\Gm@driver\Gm@xetex
991   \ifundefined{pdfpagewidth}{\setlength{pdfpagewidth}{\Gm@pw}%
992     \setlength{pdfpageheight}{\Gm@ph}}%
993   \ifnum\mag=\@m\else
994     \ifx\Gm@truedimen\Gm@true
995       \setlength{paperwidth}{\Gm@pw}%
996       \setlength{paperheight}{\Gm@ph}%
997     \fi
998   \fi
999 \fi
1000 \fi

```

If `pdftex` is set to `true`, pdf-commands are set properly. To avoid `pdftex` magnification problem, `\pdfhorigin` and `\pdfvorigin` are adjusted for `\mag`.

```

1001 \ifx\Gm@driver\Gm@pdftex
1002   \ifundefined{pdfpagewidth}{\setlength{pdfpagewidth}{\Gm@pw}%
1003     \setlength{pdfpageheight}{\Gm@ph}}%
1004   \ifnum\mag=\@m\else
1005     \@tempdima=\mag sp%
1006     \ifundefined{pdfhorigin}{\divide\pdfhorigin\@tempdima
1007       \multiply\pdfhorigin\@m
1008       \divide\pdfvorigin\@tempdima
1009       \multiply\pdfvorigin\@m}%
1010     \ifx\Gm@truedimen\Gm@true
1011       \setlength{paperwidth}{\Gm@pw}%
1012       \setlength{paperheight}{\Gm@ph}%
1013     \fi
1014   \fi
1015 \fi
1016 \fi
1017 \fi

```

With `VTeX` environment, `VTeX` variables are set here.

```

1018 \ifx\Gm@driver\Gm@vtex
1019   \ifundefined{mediawidth}{\setlength{mediawidth}{\paperwidth}
1020     \setlength{mediaheight}{\paperheight}}%
1021   \ifvtexdvi
1022     \AtBeginDvi{\special{papersize=\the\paperwidth,\the\paperheight}}%
1023   \fi
1024 \fi
1025 \fi

```

If `dvips` or `dvipdfm` is specified, paper size is embedded in dvi file with `\special`. For `dvips`, a landscape correction is added because a landscape document converted by `dvips` is upside-down in PostScript viewers.

```

1026 \ifx\Gm@driver\Gm@dvips
1027   \AtBeginDvi{\special{papersize=\the\paperwidth,\the\paperheight}}%
1028   \ifx\Gm@driver\Gm@dvips\ifGm@landscape
1029     \AtBeginDvi{\special{! /landscape true store}}%
1030   \fi\fi

```

If `dvipdfm` is specified and `atbegshi` package in ‘`oberdiek`’ bundle is loaded, `\AtBeginShipoutFirst` is used instead of `\AtBeginDvi` for compatibility with `hyperref` and `dvipdfm` program.

```

1031 \else\ifx\Gm@driver\Gm@dvipdfm
1032   \ifcase\ifx\AtBeginShipoutFirst\relax\@ne\else
1033     \ifx\AtBeginShipoutFirst\@undefined\@ne\else\z@\fi\fi
1034     \AtBeginShipoutFirst{\special{papersize=\the\paperwidth,\the\paperheight}}%
1035   \or
1036     \AtBeginDvi{\special{papersize=\the\paperwidth,\the\paperheight}}%
1037   \fi
1038 \fi\fi

```

Page frames are shipped out when `showframe=true`, cropmarks for `showcrop=true` on each page.

The `atbegshi` package is used for overloading `\shipout`.

```

1039 \@tempswafalse
1040 \ifGm@showframe

```

```

1041 \@tempwattrue
1042 \else\ifGm@showcrop
1043 \@tempwattrue
1044 \fi\fi
1045 \if@tempwa
1046 \RequirePackage{atbegshi}%
1047 \AtBeginShipout{\setbox\AtBeginShipoutBox=\vbox{%
1048 \baselineskip\z@skip\lineskip\z@skip\lineskiplimit\z@
1049 \Gm@pageframes\box\AtBeginShipoutBox}}%
1050 \fi

```

The layout dimensions for `\restoregeometry` are saved at the end of the `\AtBeginDocument`.

```

1051 \Gm@save
1052 \edef\Gm@restore@pkg{\Gm@restore}%

```

The package checks whether or not the marginpars overrun the page, if `verbose` and unless `pass`.

```

1053 \ifGm@verbose\ifGm@pass\else\Gm@checkmp\fi\fi

```

`\Gm@showparams` puts the resulting parameters and dimensions into the log file. With `verbose`, they are shown on the terminal as well.

```

1054 \Gm@showparams{preamble}%

```

The following lines free the memories no longer needed.

```

1055 \let\Gm@pw\relax
1056 \let\Gm@ph\relax
1057 }% end of \AtBeginDocument

```

`\geometry` The macro `\geometry` can be called multiple times in the preamble (before `\begin{document}`).

```

1058 \newcommand{\geometry}[1]{%
1059 \Gm@clean
1060 \setkeys{Gm}{#1}%
1061 \Gm@process}%
1062 \@onlypreamble\geometry

```

`\Gm@changelayout` The macro, which can be called from `\newgeometry`, `\restoregeometry` and `\loadgeometry`, changes the layout in the middle of the document.

```

1063 \DeclareRobustCommand\Gm@changelayout{%
1064 \setlength{\@colht}{\textheight}
1065 \setlength{\@colroom}{\textheight}%
1066 \setlength{\vsize}{\textheight}
1067 \setlength{\columnwidth}{\textwidth}%
1068 \if@twocolumn%
1069 \advance\columnwidth-\columnsep
1070 \divide\columnwidth\tw@%
1071 \@firstcolumntrue%
1072 \fi%
1073 \setlength{\hsize}{\columnwidth}%
1074 \setlength{\linewidth}{\hsize}}%

```

`\newgeometry` The macro `\newgeometry`, which changes the layout, can be used only in the document. It would reset the options specified in the preamble except for paper size options and `\mag`.

```

1075 \newcommand{\newgeometry}[1]{%
1076 \clearpage
1077 \Gm@restore@org
1078 \Gm@initnewgm
1079 \Gm@newgmtrue
1080 \setkeys{Gm}{#1}%
1081 \Gm@newgmfalse
1082 \Gm@process
1083 \ifnum\mag=\@m\else\Gm@magtooffset\fi
1084 \Gm@changelayout
1085 \Gm@showparams{newgeometry}}%

```

`\restoregeometry` The macro restores the resulting layout specified in the preamble, namely the first-page layout right after `\begin{document}`.

```

1086 \newcommand{\restoregeometry}{%

```

```

1087 \clearpage
1088 \Gm@restore@pkg
1089 \Gm@changelayout}%

```

\savegeometry The macro saves the layout with the name specified with the argument. The saved layout can be loaded with `\loadgeometry{<name>}`.

```

1090 \newcommand*{\savegeometry}[1]{%
1091   \Gm@save
1092   \expandafter\edef\csname Gm@restore@@#1\endcsname{\Gm@restore}}%

```

\loadgeometry The macro loads the layout saved with `\savegeometry{<name>}`. If the name is not found, the macro would warn it and do nothing for the layout.

```

1093 \newcommand*{\loadgeometry}[1]{%
1094   \clearpage
1095   \ifundefined{Gm@restore@@#1}{%
1096     \PackageError{geometry}{%
1097       \string\loadgeometry : name ‘#1’ undefined}{%
1098       The name ‘#1’ should be predefined with \string\savegeometry}%
1099   }{\@nameuse{Gm@restore@@#1}%
1100   \Gm@changelayout}%
1101 \</package>

```

✖

12 Fichier de configuration

Dans le fichier de configuration `geometry.cfg`, il est possible d'utiliser `\ExecuteOptions` pour définir les options par défaut du site ou de l'utilisateur.

```

1102 <*config>
1103 %<<SAVE_INTACT
1104
1105 % Retirer le symbole de commentaire (%) de la ligne suivante et éditez-la
1106 % pour fixer les options par défaut.
1107 %\ExecuteOptions{a4paper}
1108
1109 %SAVE_INTACT
1110 </config>

```

13 Fichier d'exemple

✖Here is a sample document for the geometry package.

```

1111 <*samples>
1112 %<<SAVE_INTACT
1113 \documentclass[12pt]{article}% uses letterpaper by default
1114 % \documentclass[12pt,a4paper]{article}% for A4 paper
1115 %-----
1116 % Edit and uncomment one of the settings below
1117 %-----
1118 % \usepackage{geometry}
1119 % \usepackage[centering]{geometry}
1120 % \usepackage[width=10cm,vscale=.7]{geometry}
1121 % \usepackage[margin=1cm, papersize={12cm,19cm}, resetpaper]{geometry}
1122 % \usepackage[margin=1cm,includeheadfoot]{geometry}
1123 % \usepackage[margin=1cm,includeheadfoot,includemp]{geometry}
1124 % \usepackage[margin=1cm,bindingoffset=1cm,twoside]{geometry}
1125 % \usepackage[hmarginratio=2:1, vmargin=2cm]{geometry}
1126 % \usepackage[hscale=0.5,twoside]{geometry}
1127 % \usepackage[hscale=0.5,asymmetric]{geometry}
1128 % \usepackage[hscale=0.5,heightrounded]{geometry}
1129 % \usepackage[left=1cm,right=4cm,top=2cm,includefoot]{geometry}
1130 % \usepackage[lines=20,left=2cm,right=6cm,top=2cm,twoside]{geometry}
1131 % \usepackage[width=15cm, marginparwidth=3cm, includemp]{geometry}

```

```

1132 % \usepackage[hdivide={1cm,,2cm}, vdivide={3cm,8in,}, nohead]{geometry}
1133 % \usepackage[headsep=20pt, head=40pt,foot=20pt,includeheadfoot]{geometry}
1134 % \usepackage[text={6in,8in}, top=2cm, left=2cm]{geometry}
1135 % \usepackage[centering,includemp,twoside,landscape]{geometry}
1136 % \usepackage[mag=1414,margin=2cm]{geometry}
1137 % \usepackage[mag=1414,margin=2truecm,truedimen]{geometry}
1138 % \usepackage[a5paper, landscape, twocolumn, twoside,
1139 %   left=2cm, hmarginratio=2:1, includemp, marginparwidth=43pt,
1140 %   bottom=1cm, foot=.7cm, includefoot, textheight=11cm, heightrounded,
1141 %   columnsep=1cm,verbose]{geometry}
1142 %-----
1143 % No need to change below
1144 %-----
1145 \geometry{verbose,showframe}% the options appended.
1146 \usepackage{lipsum}% for dummy text of 150 paragraphs
1147 \newcommand\mynote{\marginpar[\raggedright
1148 A sample margin note in the left side.]}%
1149 {\raggedright A sample margin note.}}%
1150 \newcommand\myfootnote{\footnote{This is a sample footnote text.}}
1151 \begin{document}
1152 \lipsum[1-2]\mynote\lipsum[3-4]\mynote
1153 \lipsum[5-11]\mynote\lipsum[12]\myfootnote
1154 \lipsum[13-22]\mynote\lipsum[23-32]
1155 \end{document}
1156 %SAVE_INTACT
1157 </samples>
1158 %\tradfin

```