

# Les extensions titlesec, titleps et titletoc \*

Javier Bezos<sup>†</sup>

2016-03-21

## Table des matières

<b>1. Introduction</b>	<b>1</b>
<b>2. Interface simplifiée</b>	<b>2</b>
2.1. Format, 2.—2.2. Espacement, 2.—2.3. Majuscules, 3.—2.4. Outils, 3.	
<b>3. Interface avancée</b>	<b>3</b>
3.1. Format, 3.—3.2. Espacement, 5.—3.3. Outils liés à l’espacement, 5.—3.4. Filets, 6.—3.5. Styles de page, 8.—3.6. Sauts de page, 8.—3.7. Autres options de l’extension, 9.—3.8. Paramétrages étendus, 9.—3.9. Création de nouveaux niveaux et changement de classe, 10.	
<b>4. Notes complémentaires</b>	<b>11</b>
4.1. Numérotation à largeur fixe, 11.—4.2. Version étoilée, 12.—4.3. Variantes, 13.—4.4. Placement d’un point après un titre de section, 13.	
<b>5. L’extension titleps et le style de page</b>	<b>13</b>
<b>6. L’extension titletoc et la table des matières</b>	<b>14</b>
6.1. Visite guidée de titletoc en 10 minutes, 14.—6.2. Et bien plus, 16.—6.3. Table des matières partielle, 18.—6.4. Liste partielle, 19.—6.5. Exemples, 19.—6.6. Insertion d’une figure en table des matières, 20.—6.7. Repérage d’entrées avec des astérisques, 20.	
<b>7. La philosophie de titlesec</b>	<b>20</b>
<b>8. Annexes</b>	<b>20</b>
9.1. Un exemple complet, 24.—9.2. Classes standards, 24.—9.3. Exemple de chapitre, 25.	

## 1. Introduction

Cette extension est pour l’essentiel un remplacement — partiel ou total — des commandes  $\LaTeX$  liées au sectionnement, traitant ainsi les titres, en-têtes et tables des matières. Elle cherche à apporter de nouvelles fonctionnalités avancées à  $\LaTeX$  : si vous souhaitez juste une interface plus accessible que celle du  $\LaTeX$  classique sans changer la façon dont  $\LaTeX$  travaille, vous pouvez recourir à fancyhdr de Piet van Oostrum, sectsty de Rowland McDonnell et tocloft de Peter Wilson, avec lesquels vous pouvez faire de belles choses<sup>1</sup>.

Entre autres nouvelles fonctionnalités se trouvent :

- différentes classes et « formes » de titres, avec des outils autorisant une grande variété de formats. Vous pouvez définir des formats différents pour les pages de gauche et de droite, des titres numérotés ou pas. Vous pouvez aussi mesurer la largeur d’un titre, ajouter de nouveaux niveaux de sectionnement, utiliser des graphiques et bien plus encore. L’Annexe montre un bon nombre d’exemples : allez-y tout de suite !
- des en-têtes et pieds de pages définis sans utilisation des commandes de marque (type `\...mark`), contenant au besoin des marques de haut de page, de première valeur *et* de bas de page. Des marques de haut de page correctement synchronisées avec les titres, sans incompatibilité avec la

---

\*L’extension titlesec est actuellement en version 2.10.2. © 1998–2016 Javier Bezos. L’extension titletoc est actuellement en version 1.6. L’extension titleps est actuellement en version 1.1.1 © 1999–2016 Javier Bezos. Tous droits réservés.

<sup>†</sup>Pour des rapports d’erreur, commentaires et suggestions, allez à <http://www.tex-tipografia.com/contact.html>. L’anglais n’étant pas mon point fort, contactez-moi lorsque vous trouvez des erreurs dans ce manuel. D’autres extensions du même auteur : gloss (avec José Luis Díaz), enumitem, accents, tensind, esindex, dotlessi, babeltools.

1. Dans la mesure où les commandes de sectionnement sont réécrites, leur comportement peut être quelque peu différent dans certains cas.

mécanique des flottants. Des éléments décoratifs ajoutés aisément, incluant des environnements d'image.

- de jolies tables des matières de forme libre, avec la possibilité de grouper les entrées de différents niveaux en un paragraphe ou de changer les formats des entrées au milieu d'un document.

Titlesec fonctionne avec les classes standards et de nombreuses autres, telles celles de l'AMS, et interagit sans aucune difficulté avec `hyperref`<sup>2</sup>. Malheureusement, elle n'est pas compatible avec `memoir`, ce dernier proposant ses propres outils avec un sous-ensemble limité des fonctionnalités disponibles dans titlesec.

Comme d'habitude, chargez l'extension de façon classique avec `\usepackage`. Redéfinissez alors les commandes de sectionnement soit avec les paramétrages simples prédéfinis (voir section « Interface simplifiée »), soit avec les commandes mises à disposition si vous souhaitez des formats plus élaborés (voir section « Interface avancée »). Dans ce dernier cas, vous avez uniquement besoin de redéfinir les commandes que vous utiliserez. Les deux méthodes sont disponibles au même moment mais, `\part` étant souvent implémentée de façon non standard, elle reste inchangée par les paramétrages prédéfinis et doit être modifiée avec l'aide de l'« Interface avancée ».

## 2. Interface simplifiée

La façon la plus simple de changer le format se fait avec un ensemble d'options d'extension et par deux commandes. Si les fonctionnalités apportées par cet ensemble d'outils vous donne satisfaction, vous n'avez pas besoin d'aller plus loin dans ce manuel. Lisez uniquement cette section et ignorez les suivantes.

### 2.1. Format

Il y a trois groupes d'options contrôlant les fontes, tailles et alignements. Vous n'avez pas besoin de définir tous ces groupes puisque des valeurs par défaut sont disponibles pour chacun ; cependant, vous devez au moins utiliser une option parmi celles possibles si vous souhaitez utiliser ce « paramétrage simplifié ».

```
rm sf tt md bf up it sl sc
```

Cette option sélectionne la famille/série/forme correspondante. La valeur par défaut est `bf`.

```
big medium small tiny
```

Cette option sélectionne la taille des titres. Elle est fixée par défaut à `big`, ce qui donne la taille retenue dans les classes standards. Avec `tiny`, les niveaux de sectionnements (chapitres exceptés) sont composées à la même taille que le texte. `medium` et `small` sont des présentations intermédiaires entre les deux premières.

```
raggedleft center raggedright
```

Cette option contrôle l'alignement, chaque valeur permettant respectivement l'alignement à droite<sup>3</sup>, le centrage et l'alignement à gauche.

### 2.2. Espacement

```
compact
```

Cette option est indépendante de celles vues ci-dessus et réduit l'espace au-dessus et au-dessous des titres.

2. Cependant, notez bien que les classes AMS réimplémentent les commandes internes d'origine. Ces changements seront perdus ici. La compatibilité avec `hyperref` a été testée avec `dvips`, `dvipdfm` et `pdftex` mais il s'agit d'une fonctionnalité non suivie. Pensez à vérifier que votre version d'`hyperref` est compatible avec titlesec.

3. Le terme anglais est à rapprocher du terme typographique français « drapeau gauche ».

## 2.3. Majuscules

```
uppercase
```

**2.9** Cette option met en majuscules des titres. Selon la classe, cela peut ne pas marcher avec `\chapter` et `\part`.

## 2.4. Outils

```
\titlelabel{<format-de-numérotation>}
```

Cette option change le format de la numérotation des sections, sous-sections, etc. Une commande générique `\thetitle` est fournie et correspond respectivement à `\thesection`, `\thesubsection`, etc. La valeur par défaut des classes standards est :

```
\titlelabel{\thetitle\quad}
```

Vous pouvez ajouter un point après la numérotation avec :

```
\titlelabel{\thetitle.\quad}
```

Ceci a été fait dans ce document.

```
\titleformat*{<commande>}{<format>}
```

Cette commande permet de changer le *<format>* d'une commande de sectionnement, comme par exemple :

```
\titleformat*{\section}{\itshape}
\titleformat*{\subsection}{\raggedleft\bfseries}
```

## 3. Interface avancée

Deux commandes sont mises à disposition pour changer le format des titres. La première est utilisée pour le format « interne », autrement dit la forme, la fonte, la numérotation, etc. La seconde définit le format « externe », autrement dit l'espacement avant et après, l'indentation, etc. Ce principe est pensé pour simplifier les définitions car, dans la plupart des cas, vous souhaitez modifier soit l'espacement, soit le format<sup>4</sup>. Tout ceci redéfinit les commandes de sectionnement existantes mais n'en crée par de *nouvelles*. De nouveaux niveaux de sectionnement peuvent être ajoutés avec `\titleclass`, comme décrit en 3.9, et leur format peut être fixé par les commandes décrites ici.

### 3.1. Format

Des formes contrôlant la distribution basique des éléments d'un titre sont proposées. En voici la liste :

**hang** est la valeur par défaut, avec une numérotation en retrait (comme avec `\section` dans les classes standards).

**block** compose le titre complet en un bloc (un paragraphe) sans mise en forme additionnelle. Utile pour les titres centrés<sup>5</sup> et les mises en forme spéciales (incluant des outils graphiques comme `picture`, `pspicture`, etc.).

**display** place la numérotation dans un paragraphe séparé (comme avec `\chapter` dans les classes standards).

4. L'information est « extraite » des commandes de sectionnement de la classe, excepté pour les chapitres et parties. Les définitions sont supposées utiliser `\@startsection` — si les sections ont été définies sans cette commande, des valeurs arbitraires de format sont proposées et peuvent être modifiées par la suite. Malheureusement, il n'y a pas de manière de récupérer les formats des chapitres ou parties ; un format similaire à celui des classes standards est alors utilisé.

5. Si le titre fait plus d'une ligne de long en utilisant la forme `hang`, la numérotation sera légèrement déplacée vers la gauche exception faite des cas avec des `\\` explicites.

**runin** restitue un titre suivi directement du texte sur la même ligne (comme avec `\paragraph` dans les classes standards <sup>6</sup>).

**leftmargin** place le titre dans la marge de gauche. Les titres en bas de page sont déplacés sur la page suivante et ne restent pas isolés dans la marge du bas, ce qui signifie que de longs titres peuvent générer des pages trop peu remplies <sup>7</sup>. Dans ce cas, vous pouvez augmenter l'étirabilité des éléments de la page, utiliser `\raggedbottom` ou utiliser l'option d'extension `nobottomtitles` décrite ci-après. Dans la mesure où le mécanisme retenu est indépendant de celui des paragraphes placés dans les marges, ils peuvent se superposer. Un synonyme obsolète à cette forme est `margin`.

**rightmargin** est similaire à `leftmargin` mais pour la marge de droite.

**drop** dispose le texte autour du titre, uniquement si le premier paragraphe est plus long que le titre (sinon, ils se superposent). Les commentaires associés à `leftmargin` s'appliquent également ici.

**wrap** est assez proche de `drop`. La seule différence est que, tandis que l'espace réservé dans `drop` pour le titre est fixé, il est automatiquement ajusté avec `wrap` à la ligne la plus longue. Les limitations de `calwidth` expliquées en 3.4 s'appliquent également ici.

**frame** Similaire à `display`, mais le titre sera encadré.

Notez bien, cependant, que certaines formes n'ont aucun sens dans le cas des chapitres et des parties.

```
\titleformat{<commande>}[<forme>]{<format>}{<num>}{<sép>}{<code-avant>}{<code-après>}
```

Ici,

- `<commande>` est la commande de sectionnement à redéfinir, autrement dit `\part`, `\chapter`, `\section`, `\subsection`, `\subsubsection`, `\paragraph` ou `\subparagraph`.
- La forme du paragraphe est fixée par `<forme>`, dont les valeurs possibles sont celles décrites ci-dessus.
- `<format>` est le format appliqué à l'ensemble du titre — numérotation et texte. Cet argument peut contenir des éléments verticaux (et horizontaux avec certaines formes) qui sont composés juste après l'espace au-dessus du titre.
- La numérotation est définie dans `<num>`. Vous pouvez le laisser vide s'il n'y a pas de numérotation de sectionnement à ce niveau, mais cela n'est pas recommandé car le nombre n'est pas supprimé dans la table des matières et dans les en-têtes de page.
- `<sép>` est la séparation horizontale entre la numérotation et le corps du titre et doit être une longueur (qui ne doit pas être vide). Cet espace est vertical dans la forme `display` ; dans `frame`, il s'agit de la distance entre le texte et l'encadrement. `<num>` et `<sép>` sont tous deux ignorés dans les versions étoilées des commandes de sectionnement. Si vous utilisez `picture` et ses semblables, mettez ce paramètre à 0 pt.
- `<code-avant>` est un code précédent le corps du titre. La toute dernière commande peut prendre un argument, le texte du titre <sup>8</sup>. Toutefois, avec l'option d'extension `explicit`, le titre doit être donné explicitement avec `#1` (voir 3.7).
- `<code-après>` est le code suivant le corps du titre. Le matériel alors composé est en mode vertical avec `hang`, `block` et `display` tandis qu'il est en mode horizontal avec `runin` et `leftmargin` (2.7 avec ce dernier au début du paragraphe). Sinon, le code est ignoré.

```
\chaptertitlename
```

Elle renvoie par défaut à `\chaptername` sauf dans le cas des annexes où elle renvoie vers `\appendixname`. Utilisez-là en lieu et place de `\chaptername` lorsque vous définissez un chapitre.

6. En fait, pas tout à fait. Le titre est d'abord mis dans une boîte pour éviter des résultats inattendus si, par exemple, il y a une commande `\color` entre le titre et le texte. Malheureusement, du fait d'une optimisation faite par  $\TeX$ , des césures facultatives peuvent être perdues. Je n'ai pas ici de solutions, exceptée celle d'utiliser `luatex`, qui fonctionne ici comme souhaité. Toutefois, si le titre ne contient ni trait d'union ni tiret, ceci ne devrait pas poser de réel problème.

7. Cependant, les flottants placés une ou deux lignes après le titre interfèrent avec la découpe de page utilisée ici et le titre peut parfois rester placé en bas de page.

8. Rappelez-vous que la taille de la fonte peut être changée de façon sécurisée uniquement entre les paragraphes et que les changements dans le texte doivent être faits localement avec un groupe ; sinon l'interlignage pourrait être faux — trop grand ou trop petit.

### 3.2. Espacement

```
\titlespacing*{<commande>}{<gauche-sép>}{<avant-sép>}{<après-sép>}[<droite-sép>]
```

La version étoilée supprime l'indentation du paragraphe suivant le titre, sauf avec `drop`, `wrap` et `runin` pour lesquels cette possibilité n'a pas de sens.

- `<gauche-sép>` augmente la marge de gauche, sauf pour les formes `...margin` et `drop` où ce paramètre fixe la largeur du titre, pour `wrap` où il donne la largeur maximale et pour `runin` où il indique l'indentation juste avant le titre. Avec une valeur négative, le titre se décale dans la marge de gauche<sup>9</sup>.
- `<avant-sép>` est l'espace vertical avant le titre.
- `<après-sép>` est la séparation entre le titre et le texte — verticale avec `hang`, `block`, et `display`, horizontale avec `runin`, `drop`, `wrap` and `...margin`. En retenant une valeur négative, vous pouvez définir un espace réel plus petit que `\parskip`<sup>10</sup>.
- Les formes `hang`, `block` et `display` disposent de la possibilité d'augmenter la marge `<droite-sép>` avec cet argument optionnel.

Si vous n'aimez pas saisir les valeurs complètes des espaces, incluant les paramètres `plus` et `minus`, une abréviation `*n` est disponible. Dans l'argument `<avant-sép>` cette notation équivaut à un espace de `n` ex avec une certaine extensibilité et une infime compressibilité. Dans `<après-sép>`, l'extensibilité est réduite et il n'y a plus de compressibilité<sup>11</sup>. Ainsi vous pouvez écrire :

```
\titlespacing{\section}{0pt}{*4}{*1.5}
```

Les longueurs `\beforetitleunit` et `\aftertitleunit` sont utilisées comme unités dans les paramètres `*` et vous pouvez les modifier (pour de légers changements dans la composition, par exemple).

**Notes.** `\titlespacing` ne fonctionne pas avec `\chapter` et `\part` à moins que vous ne changiez leur format de titre aussi bien par l'utilisation de `\titleformat`, les paramétrages simples ou `\titleclass`. Les arguments de `\titlespacing` doivent être des dimensions ; `\stretch` inclut une commande et génère ainsi une erreur.

### 3.3. Outils liés à l'espacement

Ces commandes sont fournies comme outils pour `\titleformat` et `\titlespacing`.

```
\filright \filcenter \filleft \fillast \filinner \filouter
```

Elles consistent en des variantes des commandes `\ragged...`. En particulier, les commandes `\ragged...` suppriment les espaces à droite et à gauche disposés par `\titlespacing`<sup>12</sup>. `\fillast` justifie le paragraphe, à l'exception de sa dernière ligne qui est centrée. Ces commandes fonctionnent également pour la numérotation associée à la forme `frame`.

`\filinner` et `\filouter` correspondent à `\filleft` ou `\filright` selon la page où elles se trouvent. Du fait du découpage asynchrone des pages par `TEX`, ces commandes peuvent être utilisées uniquement dans `\chapter`. Si vous souhaitez un outil général pour définir différents formats dépendant de la page, voyez 3.8.

```
\wordsep
```

Cette commande donne l'espace inter-mot pour la fonte courante.

9. Ce paramètre n'est pas équivalent à `<indent>` de `@startsection`, qui ne fonctionne pas correctement. Avec une valeur négative dans ce dernier et si `<indent>` est plus grand que la largeur de la numérotation, la première ligne du titre commencera dans la marge extérieure, comme attendu, mais les lignes suivantes ne le feront pas ; pire, les lignes seront raccourcies du côté de la marge droite.

10. Voir Goossens, Mittelbach and Samarin : *L<sup>A</sup>T<sub>E</sub>X Companion*, Reading, Addison Wesley, 1993, p. 25.

11. Ils correspondent à `n` fois `lex` plus `.3ex` minus `.06ex` et `lex` plus `.1ex` respectivement.

12. Rappelez-vous que l'extension `ragged2e` définit quelques commandes additionnelles pour l'alignement comme `\justifying`.

```
indentafter noindentafter (options d'extension)
```

Ces options contournent le réglage du retrait du premier paragraphe pour toutes les commandes de sectionnement<sup>13 14</sup>.

```
rigidchapters rubberchapters (options d'extension)
```

Avec `rigidchapters`, l'espacement pour les titres des chapitres est toujours le même et *⟨sép-après⟩* dans `\titlespacing` ne représente plus l'espace entre le bas du texte du titre et le texte principal comme décrit plus haut mais depuis le *haut* du texte du titre, autrement dit *⟨sép-avant⟩* + *⟨sép-après⟩* est maintenant une distance fixe entre le haut du corps de la page et le texte principal. La valeur par défaut est `rubberchapters` où *⟨sép-après⟩* désigne la séparation entre le titre et le texte, comme à l'accoutumée. Notez que le nom est trompeur : ces options ne s'appliquent pas qu'aux chapitres mais à tout titre de classe `top` (les classes sont développées en 3.9).

```
bottomtitles nobottomtitles nobottomtitles* (options d'extension)
```

Si `nobottomtitles` est sélectionné, les titres proches de la marge du bas sont renvoyés en page suivante, agrandissant localement la taille de la marge de bas de page. L'espace minimal requis pour que cet effet soit ignoré est fixé (approximativement) par

```
\renewcommand{\bottomtitlespace}{⟨longueur⟩}
```

dont la valeur par défaut est `.2\textheight`. Une marge du bas agrandie (et donc irrégulière de page en page) sur la page précédente est obtenue en mettant une valeur de 0 pt. `bottomtitles` est l'option par défaut, qui donne à `\bottomtitlespace` une valeur négative.

L'option `nobottomtitles*` propose des calculs plus précis mais les titres de forme `margin`, `wrap` ou `drop` peuvent être mal disposés. En temps normal, vous devriez utiliser la version étoilée.

```
aftersep largestsep (options d'extension)
```

Par défaut, lorsqu'il y a deux titres consécutifs, l'espace *⟨sép-après⟩* du premier titre est utilisé entre les deux titres. Parfois, ce comportement n'est pas souhaité, en particulier quand l'espace *⟨sép-avant⟩* est plus grand que *⟨sép-après⟩* (sinon le comportement par défaut semble préférable). Avec `largestsep`, le plus grand espace est utilisé. La valeur par défaut est `aftersep`.

```
\\ \\*
pageatnewline (option d'extension)
```

**2.6** Dans les versions 2.6 et suivantes, `\\` ne permet pas de saut de page et devient un équivalent de `\\*`. Comme personne, *a priori*, ne veut un saut de page dans un titre, il s'agit du comportement par défaut. Si, pour une raison étrange, vous souhaitez autoriser ces sauts de page dans les titres, utilisez l'option `pageatnewline`, fournie pour des questions de rétrocompatibilité.

### 3.4. Filets

L'extension inclut des outils pour aider à l'ajout de filets et autres éléments au-dessus comme au-dessous du titre. Les marges des titres pouvant être modifiées, ces commandes prennent en compte les paramètres locaux pour placer les filets proprement. Elles tiennent également compte de l'espace utilisé dans les titres marginaux.

```
\titleline[⟨alignement⟩]{⟨matériel horizontal⟩}
\titlerule[⟨hauteur⟩]
\titlerule*[⟨largeur⟩]{⟨texte⟩}
```

La commande `\titleline` permet l'insertion d'une ligne, qui peut contenir du texte et d'autres

13. Anciennement nommées `indentfirst` et `nonindentfirst`, maintenant obsolètes.

14. L'option `french` de l'extension `babel` dans son réglage de base rend ces options inefficaces en imposant l'indentation.

matériels horizontaux. Elle est pensée principalement pour des filets et des points de conduite<sup>15</sup> mais elle peut servir également pour d'autres besoins. La ligne a une largeur fixée et doit être remplie, autrement dit `\titleline{CHAPITRE}` produit une boîte trop peu remplie et donc mal espacée. Ici le paramètre optionnel *alignement* (l, r ou c) peut aider en vous permettant de saisir `\titleline[c]{CHAPITRE}`<sup>16</sup>.

L'utilisation de `\titleline` à des endroits où un matériel vertical n'est pas attendu peut conduire à des anomalies. En d'autres termes, vous pouvez l'utiliser dans les arguments *format* (toujours) et *code-après*, que ce soit dans les formes `hang`, `display` et `block`. Pour la forme `display`, vous pouvez la placer au tout début des arguments *code-avant* et *num*. Mais vous pouvez tester cette commande à d'autres endroits : il y a de grandes chances qu'elle y fonctionne.

La commande `\titlerule`, qui est incluse automatiquement dans `\titleline` si nécessaire, peut être utilisée pour construire des filets et des motifs de remplissage. La version non étoilée trace des filets épais de 0.4 pt ou de *hauteur* si ce dernier est présent. Par exemple,

```
\titlerule[.8pt]%
\vspace{1pt}%
\titlerule
```

trace deux filets de différentes épaisseurs avec une séparation de 1 pt.

La version étoilée fait une variante des points de conduite en remplaçant les points par le motif répété du *texte* placé dans une boîte de sa largeur naturelle. Cette largeur de boîte peut être modifiée avec *largeur* mais la première boîte conserve sa largeur naturelle afin que le *texte* soit aligné avec les bords gauche et droit de l'espace à remplir.

Par exemple, avec

```
\titleformat{\section}[leftmargin]
{\titlerule*[1pc]{.}}%
\vspace{1ex}%
\bfseries}
{... définitions à la suite
```

des points de conduite s'étendent au-dessus du titre de la section et du texte.

calcwidth (option d'extension)
--------------------------------

La forme `wrap` a la capacité de mesurer les lignes du titre pour mettre en forme le paragraphe. Cette capacité peut être étendue à trois autres formes — `display`, `block` et `hang` — avec cette option d'extension. La longueur de la plus longue ligne est restituée dans `\titlewidth`<sup>17</sup>.

Du point de vue de T<sub>E</sub>X, toute boîte est considérée comme du matériel composable. Si une boîte a été élargie avec de l'espace ou, inversement, si une boîte avec du texte a été réduite, la valeur de `\titlewidth` peut être faussée (du point de vue d'un humain). La forme `hang`, par exemple, recourt en interne à de telles boîtes mais, dans ce cas, ce comportement est souhaité quand le titre est aligné à droite ; sinon la forme `block` produit de meilleurs résultats. En d'autres termes, l'utilisation de boîtes dont la largeur a été manipulée peut être source d'erreurs<sup>18</sup>. Pour être complet, certaines commandes peuvent embrouiller T<sub>E</sub>X et interrompre l'analyse du titre. Mais, si vous vous en tenez à du texte, `\\` et `\\[...]` (et il y a peu de chances que vous puissiez avoir besoin d'autre chose), vous ne devriez pas rencontrer de problèmes.

Un autre point important est que les paramètres *code-avant*, *num*, *sép*, et *titre* (mais pas *code-après*) sont évalués deux fois au niveau local ; si vous augmentez un compteur *globalement*, vous l'augmenterez alors deux fois. Dans la plupart des cas, placer cette augmentation dans le *code-après* passera mais vous pouvez également utiliser la commande suivante.

15. Suite de points régulièrement espacés qui se trouvent généralement dans les tables de matières.

16. La valeur par défaut est le paramètre `s` de la commande `\makebox`.

17. Il existe deux paramètres plus fins `\titlewidthfirst` and `\titlewidthlast` qui restituent la longueur de la première et de la dernière ligne respectivement. Il n'y a pas de fonctions dédiées pour s'en servir mais vous pouvez assigner leurs valeurs à `\titlewidth` et utiliser alors `\titleline*`.

18. Ce qui inclut les lignes justifiées dont l'espace inter-mot a été augmenté.

```
\iftitlemeasuring{<vrai>}{<faux>}
```

**2.9** Quand le titre est mesuré (première lecture), l'argument *<vrai>* est traité tandis que, quand le titre est réellement composé (seconde lecture), l'argument *<faux>* est traité.

```
\titleline*{<alignement>}{<matériel horizontal>}
```

Une variante de `\titleline` à utiliser uniquement avec `calwidth`. Le texte sera d'abord placé dans une boîte de largeur `\titlewidth` ; cette boîte sera à son tour placée dans la boîte principale avec l'alignement indiqué. Il n'y a pas d'équivalent à `\titlerule` et ceci explique pourquoi vous devez le placer explicitement dans un `\titleline*` si vous voulez que `\titlewidth` soit pris en compte :

```
\titleline*[c]{\titlerule[.8pc]}
```

### 3.5. Styles de page

**2.8** Vous pouvez assigner un style de page à des niveaux ayant la classe `top` ou `page` (voir 3.9, page 10), aussi bien qu'au chapitre de base avec la commande suivante <sup>19</sup> :

```
\assignpagestyle{<commande>}{<style de page>}
```

Par exemple, pour supprimer le numéro de page avec les chapitres :

```
\assignpagestyle{\chapter}{empty}
```

### 3.6. Sauts de page

<code>\sectionbreak</code>	<code>\subsectionbreak</code>	<code>\subsubsectionbreak</code>
<code>\paragraphbreak</code>	<code>\subparagraphbreak</code>	<code>\&lt;section&gt;break</code>

En définissant ces commandes avec `\newcommand`, différents sauts de page peuvent être appliqués à différents niveaux hiérarchiques. Dans celles non définies, une pénalité avec une valeur interne fournie par la classe est retenue (typiquement `-300`). Par exemple,

```
\newcommand{\sectionbreak}{\clearpage}
```

fait que les sections commencent sur une nouvelle page. Dans certaines mises en page, l'espace au-dessus du titre est préservé même si la section commence sur une nouvelle page ; ceci s'obtient avec :

```
\newcommand{\sectionbreak}{%
  \addpenalty{-300}%
  \vspace*{0pt}}
```

**2.6** `\<section>break` est également disponible avec la classe `top`. Les valeurs admises sont `\cleardoublepage` (valeur par défaut avec `openright`) et `\clearpage` (valeur par défaut avec `openany`). Ainsi, vous pouvez annuler l'effet de `openright` en définissant `\chapterbreak` comme `\clearpage`, du moment que sa classe a été changée en `top` (dans cet exemple, les parties conserveront le paramétrage `openright`).

```
\chaptertolists
```

**2.6** Si cette commande est définie, l'espace blanc écrit d'habitude dans les listes (que sont la liste des figures et la liste des tables) pour isoler les éléments d'un même chapitre est remplacé par le code de cette commande. Si vous ne voulez pas de l'espace quand un chapitre commence, définissez-le comme vide, autrement dit,

```
\newcommand{\chaptertolists}{}%
```

<sup>19</sup>. Nommée dans l'éphémère version 2.7 `\titlepagestyle`.



Cette commande n'est pas un outil général pour contrôler l'espacement dans les listes et n'est valable que dans les titres de classe `top`. Aussi, elle ne fonctionnera pas avec les chapitres standards sauf si vous changez leur classe (d'un autre côté, `\...tolists` peut être utilisé dans tout titre dans la classe est `top`).

### 3.7. Autres options de l'extension

`explicit` (option d'extension)

**2.7** Avec cette option, le titre n'est pas implicite à la fin de *code-avant* mais doit être indiqué explicitement avec `#1` comme dans l'exemple suivant :

```
\titleformat{\section}
{..}
{\thesection}{..}{#1.}
```

(Comparez ceci avec l'exemple de la section 4.4.)

`newparttoc oldparttoc` (options d'extension)

Les parties des classes standards écrivent leur numéro en table des matières de façon non standard. Vous pouvez changer cela avec `newparttoc` de façon à ce que `titletoc` ou une extension similaire puisse manipuler cette entrée (ceci fonctionne seulement dans le cas où `\part` a été redéfini).

`cleareempty` (option d'extension)

Cette option modifie le comportement de `\cleardoublepage` de sorte que le style de page `empty` soit utilisé dans les pages vides.

`toctitles` (option d'extension)

**2.6** Cette option change le comportement de l'argument optionnel dans les commandes de sectionnement de sorte qu'elles déterminent uniquement le titre en en-tête et non l'entrée en table des matières qui sera donc basée sur le titre complet.

`newlinetospace` (option d'extension)

**2.6** Cette option remplace chaque occurrence de `\\` ou `\\*` dans les titres par une espace dans les en-têtes et les entrées en table des matières. De cette façon, vous n'avez pas à répéter le titre juste pour retrancher une commande de mise en forme.

`notocpart*` (option d'extension)

**2.10.1** Il y a un moment de cela (en l'an 2000), j'ai décidé pour certaines raisons que la commande `\part*` se comporterait comme dans les classes de l'AMS et qu'elle devrait donc avoir une entrée en table de matière. C'est pour le moins assez curieux et il s'avère que personne ne s'en est plaint jusqu'à présent ! Par ailleurs, imposer le comportement attendu plus de 15 ans plus tard ne semble pas être une bonne idée. Un nouveau style de page/partie est en cours de constitution mais, pour le moment, cette option restaure le comportement standard.

### 3.8. Paramétrages étendus

Le premier argument de `\titleformat` et de `\titlespacing` dispose d'une syntaxe étendue qui permet de paramétrer différentes mises en forme en fonction du contexte<sup>20</sup>. Cet argument peut prendre la forme d'une liste de paires clé/valeur :

20. L'extension `keyval` est requise pour utiliser cette syntaxe.

$\langle clé \rangle = \langle valeur \rangle, \langle clé \rangle = \langle valeur \rangle, \langle clé \rangle, \langle clé \rangle, \dots$

Actuellement, en plus du nom de la commande de sectionnement, seules la parité des pages et la présence ou pas de numérotation sont prises en charge pour définir le contexte. Ainsi, les clés disponibles sont :

- name. Les valeurs possibles sont `\chapter`, `\section`, etc.
- page. Les valeurs possibles sont `odd` (impaire) ou `even` (paire).
- numberless. Une clé sans valeur. Elle ne sert que si vous souhaitez disposer de variantes numérotées (sans cette clé) et non numérotées (avec `numberless`).

De fait, la forme basique décrite plus haut avec le nom de la commande de sectionnement. Par exemple :

```
\titleformat{\section} ...
```

est en fait une abréviation pour :

```
\titleformat{name=\section} ...
```

Supposons que nous souhaitons une mise en page avec des titres dans la marge extérieure. Nous pourrions définir quelque chose comme :

```
\titleformat{name=\section,page=even}[leftmargin]
{\filleft\scshape}{\thesection}{.5em}{}
```

```
\titleformat{name=\section,page=odd}[rightmargin]
{\filright\scshape}{\thesection}{.5em}{}
```

Dans la mesure où l'information de page est écrite dans le fichier `aux`, il faut deux compilations au minimum pour obtenir le résultat souhaité.

La version numérotée constitue généralement une bonne base pour parvenir à générer la version non numérotée car la suppression de la numérotation est le seul changement requis dans la plupart des cas. Si vous avez besoin de mises en forme spéciales, la clé `numberless` définit une version alternative des sections sans numéros (autrement dit celles avec un niveau inférieur à `secnumdepth`, dans les parties préliminaires et annexes du document<sup>21</sup> et, bien entendu, les versions étoilées). Par exemple :

```
\titleformat{name=\section}{...% La définition normale suit
\titleformat{name=\section,numberless}{...% La définition non numérotée suit
```

$\langle num \rangle$  et  $\langle sép \rangle$  ne sont pas ignorés dans les variantes avec `numberless`.

Ces clés sont disponibles à la fois pour `\titleformat` et `\titlespacing`. En utilisant `page` dans l'une d'elles (ou les deux), les pages paires et impaires seront mises en forme différemment. Accessoirement, « paire » et « impaire » sont des termes bien établis avec  $\text{\LaTeX}$  mais parfois source de confusion. Dans une impression en recto uniquement, les pages « impaires » sont également des pages « paires » (cf. `\oddsidemargin`).

Si vous souhaitez créer différentes mises en forme pour les pages paires/impaires qu'elles soient avec ou sans numérotation, il est recommandé de définir quatre variantes.

Si vous retirez la clé `page` d'une définition, vous devez supprimer le fichier `.aux`.

### 3.9. Création de nouveaux niveaux et changement de classe

Alors que les formes et autres modifient le comportement des titres en lien avec le texte environnant, les classes de titre autorisent le changement de leur comportement générique. Avec l'aide des classes, vous pouvez insérer, par exemple, un nouveau niveau `subchapter` entre `chapter` et `section`, ou créer un nouveau plan de votre cru. *Faire un plan cohérent et définir tous les éléments liés comme les compteurs, les commandes, les formats, l'espacement et le format de l'éventuelle table des matières, le format de cette table est laissé à la responsabilité de l'utilisateur.* Il existe trois classes :

- `page` travaille sur une page séparée, comme `\part` (de la classe `book`) ;
- `top` commence sur une nouvelle page en plaçant le titre en haut de page, comme `\chapter` ;
- `straight` est pensé pour les titres en milieu de texte<sup>22</sup>.

21. Souvent traitées avec les commandes `\frontmatter` et `\backmatter`.

22. Il existe une autre classe nommée `part` qui sert à émuler le `\part` de la classe `article` mais vous ne devriez pas l'utiliser du tout. Utilisez la classe `straight` à la place. Rappelez-vous que certaines fonctionnalités reposent sur ces classes et que `titlesec` ne change pas par défaut les définitions de `\part` et `\chapter`.

```
\titleclass{<nom>}{<classe>}
\titleclass{<nom>}{<classe>}[<commande-du-niveau-supérieur>]
```

Si vous n'utilisez pas l'argument optionnel, vous changez juste la *<classe>* de *<nom>*. Par exemple :

```
\titleclass{\part}{straight}
```

attribue la classe `straight` à `part`.

Quand la seconde forme est utilisée, le numéro du niveau suit directement celui de la *<commande-du-niveau-supérieur>*. Par exemple :

```
\titleclass{\subchapter}{straight}[\chapter]
\newcounter{subchapter}
\renewcommand{\thesubchapter}{\Alph{subchapter}}
```

crée un niveau sous celui du chapitre (du code additionnel est proposé ici mais vous devez lui ajouter les paramétrages `\titleformat` et `\titlespacing` souhaités)<sup>23</sup>. Si le niveau du chapitre est 0 alors celui du sous-chapitre est 1 et les niveaux en-dessous sont incrémentés de 1 (la section devient 2, la sous-section 3 et ainsi de suite).

Il existe deux commandes de sectionnement qui exécutent quelques actions supplémentaires selon leur nom en ignorant la classe :

- `\chapter` récupère la chaîne définie dans `\chaptertitlename` et la prend en compte ;
- `\part` ne place pas la numérotation de l'entrée en table des matières sauf si vous utilisez l'option `newparttoc`.

`loadonly` (option d'extension)

Supposons que vous voulez créer vos commandes de sectionnement à partir de rien. Cette option d'extension ignore toutes les définitions antérieures, si elles existent, et retire ainsi la possibilité d'utiliser les options décrites dans la section 2. Vous pouvez alors utiliser les outils de `titlesec` et construire les compteurs et numérotations associées.

```
\titleclass{<nom>}[<numéro-de-niveau-initial>]{<classe>}
```

Ici, le titre *<nom>* est considéré comme le plus haut niveau, avec un numéro *<numéro-de-niveau-initial>* (typiquement 0 ou -1). Cette commande devrait être retenue uniquement lors de la création de commandes de sectionnement à partir de rien avec l'aide de `loadonly` et il devrait y avoir une unique déclaration de ce type. Après ceci, le reste des niveaux est ajouté comme indiqué ci-dessus.

## 4. Notes complémentaires

Cette partie décrit brièvement quelques commandes  $\text{\LaTeX}$ , utiles lors de la définition de commandes de sectionnement.

### 4.1. Numérotation à largeur fixe

La commande `\makebox` permet de gérer une numérotation à largeur fixe, en plaçant la marge gauche du titre actuel (pas la numérotation) toujours au même endroit. Par exemple, en se limitant au seul le code important :

```
\titleformat{\section}
{..}
{\makebox[2em]{\thesection}}{..}{..}
```

N'hésitez pas à consulter votre documentation de  $\text{\LaTeX}$  pour plus d'informations sur les commandes portant sur les boîtes.

23. En ce qui concerne les compteurs, l'extension `remreset` peut être utile.

## 4.2. Version étoilée

Recourir à la version étoilée dans les commandes de sectionnement est très peu recommandé. En lieu et place, vous pouvez utiliser des commandes plus explicites, simples à définir et à modifier si nécessaire. Ainsi, vous pouvez tester différentes mises en page avant de choisir l'une d'entre elles.

Tout d'abord, rappelez-vous que si vous écrivez

```
\setcounter{secnumdepth}{0}
```

les sections ne seront pas numérotées mais seront insérées dans la table des matières et les en-têtes.

Maintenant, supposons que vous voulez des sections avec un contenu spécial ; par exemple, une section (ou plus) avec des exercices. Nous utiliserons un environnement nommé `exercices` tel que :

```
\section{A section}
Texte d'une section normale.
```

```
\begin{exercices}
\section{Exercices A}
Des exercices
```

```
\section{Exercices B}
Des exercices
\end{exercices}
```

La définition suivante supprime les numérotations mais pas les entrées en table des matières et pas les impacts sur l'en-tête.

```
\newenvironment{exercices}
{ \setcounter{secnumdepth}{0} }
{ \setcounter{secnumdepth}{2} }
```

La suivante ajoute une entrée en table des matières mais ne modifie pas l'en-tête.

```
\newenvironment{exercices}
{ \setcounter{secnumdepth}{0}%
  \renewcommand\sectionmark[1]{} }
{ \setcounter{secnumdepth}{2} }
```

La suivante met à jour l'en-tête mais ne crée pas d'entrée en table des matières.

```
\newenvironment{exercices}
{ \setcounter{secnumdepth}{0}%
  \addtocontents{toc}{\protect\setcounter{tocdepth}{0}\ignorespaces} }
{ \setcounter{secnumdepth}{2}%
  \addtocontents{toc}{\protect\setcounter{tocdepth}{2}\ignorespaces} }
```

(Ce dernier cas reste sans doute un peu étrange dans notre exemple ; le premier et le deuxième cas ont bien plus de sens. Par ailleurs, les `\ignorespaces` ne sont pas très importants, ne devenant nécessaires que si vous avez des espaces non souhaités dans la table des matières.)

Ceci fonctionne avec les classes standards mais, si vous utilisez `fancyhdr` ou `titlesec` pour définir les en-têtes, il faudra avoir recours à des mécaniques plus avancées pour supprimer les numéros de section. Dans `titlesec`, ceci s'obtient avec `\ifthesection`.

Comme vous pouvez le voir, il n'y a pas de `\addcontentsline`, de `\markboth` ou de `\section*`, juste de la structure logique. Bien entendu, vous pouvez changer tout ceci à merci ; par exemple, si vous décidez que ces sections doivent être composées en petits caractères, ajoutez un `\small`, et si vous réalisez que vous n'aimez pas cela, retirez-le.

Tandis que les commandes standards de  $\text{\LaTeX}$  sont plus faciles et directes pour les cas simples, je pense que la méthode proposée ci-dessus est de loin préférable pour les longs documents.

### 4.3. Variantes

Supposons que vous souhaitez marquer des sections comme « sujet complexe » par un astérisque après la numérotation. Le code suivant effectue cette mise en forme :

```
\newcommand{\secmark}{}
\newenvironment{complexe}
  {\renewcommand{\secmark}{*}}
  {}
\titleformat{\section}
  {...}
  {\thesection\secmark\quad}{...}{...}
```

Pour marquer une section, écrivez :

```
\begin{complexe}
\section{...}
...
\end{complexe}
```

Ceci marque les sections mais pas les sous-sections. Si vous souhaitez un peu de redondance en marquant les sous-sections également, vous devez les définir de façon similaire.

### 4.4. Placement d'un point après un titre de section

De nos jours, ce style n'est plus en usage mais il a été relativement courant auparavant. La technique de base a été décrite ci-dessus mais en voici un rappel :

```
\newcommand{\pointapres}[1]{#1.}
\titleformat{\section}
  {...}
  {\thesection}{...}{...}\pointapres}
```

Si vous avez à combiner ce point avec une commande (tel un soulignement), vous pouvez saisir :

```
\newcommand{\periodafter}[2]{#1{#2.}}
\titleformat{\section}
  {...}
  {\thesection}{...}{...}\periodafter{\ul}} % \ul de l'extension soul
```

Bien entendu, vous pourriez aussi travailler avec l'option d'extension explicit.

## 5. L'extension titleps et le style de page

L'extension titleps fournit des outils pour configurer en une étape le style de page (en-tête et pied de page). Une interface de haut niveau est ici utilisée : la mécanique des marques est cachée et il n'y a pas besoin de gérer les commandes `\leftmark` et `\rightmark`. Il suffit d'utiliser une commande ou une variable déclarée comme une « marque » pour que la valeur voulue soit restituée, autrement dit, celles utilisées lors la marque est émise (soit par une commande de sectionnement soit explicitement par `\chaptermark`, `\sectionmark`, etc). Un simple exemple, dont le sens devrait être évident :

```
\newpagestyle{main}{
  \sethead[\thepage][\chaptertitle][(\thesection] % paire
  {\thesection)}{\sectiontitle}{\thepage}} % impaire
\pagestyle{main}
```

Les autres fonctionnalités sont :

- des marques de début de page (*top mark*) compatibles avec les flottants (contrairement au `\topmark` standard, qui ne fonctionne pas correctement dans  $\text{\LaTeX}$ );
- l'accès aux marques de début de page ou de fin de page (*bottom mark*) ainsi qu'à la première marque (*first mark*) dans un unique en-tête ou pied de page (par exemple les numéros des premières et dernières sections);
- des marques pour plus de deux niveaux de sectionnement;

- des filets d'en-tête et de pied de page simples (ou bien moins simples) ;
- des lignes d'en-tête et de pied de page pour les pages avec des flottants ;
- des lignes d'en-tête et de pied de page des flottants spécifiques (une sorte de `\thispagestyle` pour flottants) ;
- de multiples ensembles de marques (désignés sous les termes *lots de marques* et *marques supplémentaires*).

L'extension peut être utilisée sans titlesec mais vous en tirerez le meilleur en usage combiné. Pour la charger seule, saisissez le traditionnel `\usepackage{titleps}`. Pour la charger avec titlesec, passez par :

```
\usepackage[pagestyles]{titlesec}
```

Vous pouvez lire `titleps.pdf` (ou éditer `titleps.tex`) pour toute information complémentaire.

## 6. L'extension titletoc et la table des matières

Cette extension accompagne titlesec et gère les entrées en table des matières. Cependant elle constitue une extension indépendante et peut être utilisée seule. Sa philosophie est similaire à celle de titlesec — au lieu de manipuler les commandes définies par la version standard et ses classes, elle propose de nouvelles commandes avec lesquelles vous pouvez établir le format des entrées de table de façon générique. Ceci implique que vous avez juste à apprendre deux nouvelles commandes basiques et deux outils, ni plus ni moins, pour accéder à de nouvelles fonctionnalités. La mise en forme des paragraphes et des fontes associées sont traitées avec des commandes comme `\`, `\makebox`, `\large`, `\itshape` et ainsi de suite. Les entrées en elles-mêmes ne sont pas mises en forme de quelque manière que ce soit car elles sont souvent de forme libre.

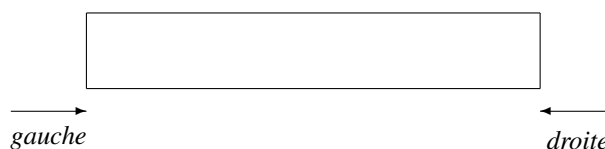
Le comportement des entrées définies avec titletoc diffère sur certains points de celui des entrées définies avec les commandes standards. En particulier :

- la découpe des pages ne se fait jamais entre deux entrées si la première est d'un rang plus élevé sur la seconde, par exemple entre une section et une sous-section. Si les deux entrées sont de même niveau, la découpe est autorisée. Enfin, si la première entrée est d'un niveau inférieur à la seconde, la découpe potentielle est alors considérée comme pertinente ;
- Les symboles remplaçant les points dans des points de conduite ne sont pas centrés mais alignés à droite. Ceci est généralement plus pratique.

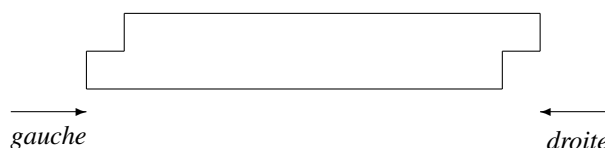
Je souhaite ici souligner qu'aucun essai pour manipuler les tables des matières ne peut être complet car les commandes standards de L<sup>A</sup>T<sub>E</sub>X écrivent directement des commandes de mises en forme qui ne peuvent être retranchées. C'est tout particulièrement le cas pour les listes de figures et de tables et pour la commande `\part`<sup>24</sup>.

### 6.1. Visite guidée de titletoc en 10 minutes

Les entrées de table de matière sont traitées comme des zones rectangulaires où le texte et un probable élément de remplissage seront composés. Dessinons une telle zone (les lignes ne seront pas ici affichées) :

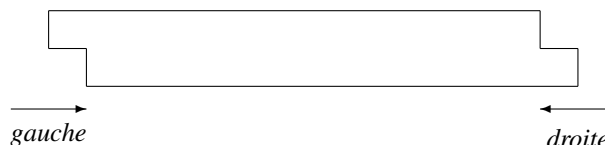


L'espace entre la marge gauche de la page et le bord gauche de notre zone sera nommé `<gauche>` ; de manière similaire se définit `<droite>`. Vous pouvez modifier le début de la première ligne et la fin de la dernière ligne. Par exemple, en augmentant leur taille avec `\hspace*{0.5cm}`, cette zone devient :



24. Mais certains de ces problèmes sont résolus par titlesec.

Et en retranchant de l'espace à ces deux endroits avec `\hspace*{-0.5cm}`, la zone ressemble alors à ceci :



Si vous avez déjà vu des tables de matières, ce qui suit devrait vous être familier — la numérotation en tout début et le numéro de page en toute fin :

```
3.2 Ceci est un exemple montant une entrée de table des matières
    respectant ce principe . . . . . 4
```

Tout ce dont vous avez besoin est de mettre ces éléments dans les bonnes positions. Si vous avez réservé l'espace avec `\hspace*{-0.5cm}`, mettez simplement une boîte de 0,5 cm de large contenant la numérotation ou le numéro de page pour que cet espace soit récupéré ; cette présentation est si courante que deux commandes sont données pour la générer pour vous :

- `\contentslabel{⟨longueur⟩}` crée un espace au début et y place la numérotation ;
- `\contentspage` crée l'espace à la fin de largeur `⟨droite⟩` et y écrit le numéro de page aligné à droite.

Maintenant, nous allons voir trois commandes basiques :

```
\dottedcontents{⟨nom⟩}[⟨gauche⟩]{⟨code-dessus⟩}
                {⟨largeur-num⟩}{⟨largeur-motif⟩}
```

Ici :

- `⟨nom⟩` est le nom de la commande de sectionnement sans l'antislash : `part`, `chapter`, `section`, etc. `figure` et `table` sont autorisés également (l'antislash est omis car nous nous occupons du concept et non des commandes de sectionnement elles-mêmes. Par ailleurs, `figure` et `table` sont des environnements).
- `⟨code-dessus⟩` est le code mettant en forme globalement l'entrée. Les matériels verticaux sont autorisés. À ce niveau, la valeur de `\thecontentslabel` (voir ci-dessous) est connue, ce qui permet de prendre des décisions basées sur sa valeur (avec l'aide de l'extension `ifthen`). Vous pouvez utiliser les commandes de `titlesec` `\filleft`, `\filright`, `\filcenter` et `\fillast`.
- `⟨gauche⟩`, même s'il est mis entre crochet, est actuellement obligatoire et fixe l'espacement à gauche à partir de la marge gauche de la page.
- `⟨largeur-num⟩` est la largeur de l'espace créé pour la numérotation comme décrit ci-dessus.
- `⟨largeur-motif⟩` est la largeur de la boîte contenant le caractère qui va servir de motif au remplissage de la ligne, comme décrit ci-dessus.

Les définitions des entrées pour les sections et sous-sections dans la classe `book` sont à peu près équivalentes à :

```
\contentsmargin{2.55em}
\dottedcontents{section}[3.8em]{}{2.3em}{1pc}
\dottedcontents{subsection}[6.1em]{}{3.2em}{1pc}
```

```
\titlecontents{⟨nom⟩}[⟨gauche⟩]{⟨code-dessus⟩}
                {⟨format-entrée-numérotée⟩}{⟨format-entrée-non-numérotée⟩}
                {⟨format-remplissage-page⟩}[⟨code-dessous⟩]
```

Ici `⟨nom⟩`, `⟨gauche⟩` and `⟨code-dessus⟩` fonctionnent comme ci-dessus. Puis :

- `⟨format-entrée-numérotée⟩` est en mode horizontal et va être utilisée juste avant le titre de l'entrée. Comme dans `\titleformat`, la dernière commande ici peut prendre comme argument le titre ;
- `⟨format-entrée-non-numérotée⟩` est comme le précédent argument en l'absence de numérotation ;
- `⟨format-remplissage-page⟩` se comprend d'elle-même. Les éléments de remplissage sont créés avec la commande `\titlerule` qui est partagée entre cette extension et `titlesec`. Cependant,

utilisée dans ce contexte, son comportement change quelque peu pour convenir aux besoins des points de conduite de la table des matières<sup>25</sup>. Vous pourriez préférer un `\hspace` à la place.

— et, enfin, `<code-dessous>` est le code suivant l'entrée pour, par exemple, un espace vertical.

Lors de la définition des entrées, utilisez `\addvspace` si vous voulez ajouter un espace vertical et `\\*` au lieu de `\\` pour des retours à la ligne.

Cette commande peut être utilisée en cours de document pour changer le format des entrées des tables des matières, des listes des tables et des listes des figures à n'importe quel moment. Le nouveau format est écrit dans le fichier toc et nécessite donc deux compilations pour être observable.

```
\contentsmargin{<droite>}
```

La valeur fixée est utilisée dans tous les sectionnements. Si vous vous demandez pourquoi, la réponse est plutôt simple : dans la plupart des cas, la marge `<droite>` sera constante. Cependant, vous avez le droit de la changer localement dans les arguments `<code-avant>`. Notez d'ailleurs que l'espace par défaut des classes standards ne laisse pas de place pour afficher des numéros de page dépassant 100 en gras et vous pourriez avoir besoin de fixer une largeur de marge plus importante avec cette commande.

La mise en forme des entrées de section de la classe book est similaire (mais pas égale) à :

```
\titlecontents{section}
    [3.8em] % autrement dit, 1.5em (chapitre) + 2.3em
    {}
    {\contentslabel{2.3em}}
    {\hspace*{-2.3em}}
    {\titlerule*[1pc]{.}\contentspage}
```

La comparaison de cette définition avec celle donnée plus haut permet de comprendre comment `\dottedcontents` est défini.

Bien les classes standards se servent d'unités dépendant de la taille des caractères (principalement l'em), il est recommandé de se servir d'unités absolues (cm, pt, etc.) pour garantir le bon alignement des entrées.

## 6.2. Et bien plus

Des règles typographiques strictes indiquent que les lignes de texte pleines ne peuvent aller plus loin que le dernier point d'un ensemble de points de conduite ; idéalement, elles devraient être alignées avec ce point. Étonnamment,  $\TeX$  ne dispose pas d'un outil pour faire ceci automatiquement — lorsque vous remplissez une boîte de points de conduite, ils peuvent être centrés dans la boîte avec `\cleaders`, justifiés avec `\xleaders` ou alignés avec le bord extérieur de la boîte qui les contient avec `\leaders` mais il n'y a pas de façon de les aligner avec la marge « courante ».

Aussi, la seule façon d'obtenir un bon rendu est de le faire à la main. Pour cela, vous pouvez utiliser l'argument optionnel dans la commande `\contentsmargin` dont la syntaxe complète est la suivante :

```
\contentsmargin[<correction>]{<droite>}
```

La longueur de `<correction>` est ajoutée à celle de `<droite>` dans toutes les lignes sauf la dernière, là où les points de conduite sont placés. Par exemple, si les lignes de texte font 6 pt de plus que celles du dernier point, vous devriez utiliser la commande `\contentsmargin` pour y ajouter une `<correction>` de 6 pt<sup>26</sup>. Contrairement aux outils  $\LaTeX$  standards, la commande `\titlerule*` a été développée pour que la `<correction>` ait la plus petite valeur possible.

```
\thecontentslabel \thecontentspage
```

Ces commandes contiennent le texte de la numérotation et de la page sans autre mise en forme que celle générée par la classe.

25. Pour les  $\TeX$ niens, `\xleaders` devient ici `\leaders`.

26. Pour vous aider, de nombreux prévisualisateurs de dvi permettent d'obtenir les coordonnées de la localisation pointée.



```
\contentslabel[⟨format⟩]{⟨espace⟩}
\contentspage[⟨format⟩]
```

Ces commandes reprennent le principe de celles décrites ci-dessus, mais avec un *⟨format⟩* différent. Les valeurs par défaut sont juste `\thecontentslabel` et `\thecontentspage`, respectivement.

```
\contentspush{⟨texte⟩}
```

Affiche le *⟨texte⟩* et augmente *⟨gauche⟩* de la largeur du *⟨texte⟩*. Ceci est similaire à la forme `hang` de `titlesec`.

```
\titlecontents*{⟨section⟩}[⟨gauche⟩]{⟨code-dessus⟩}
    {⟨format-entrée-numérotée⟩}{⟨format-entrée-non-numérotée⟩}
    {⟨format-remplissage-page⟩}[⟨séparateur⟩]
    ou ...{⟨format-remplissage-page⟩}[⟨séparateur⟩][⟨fin⟩]
    ou ...{⟨format-remplissage-page⟩}[⟨début⟩][⟨séparateur⟩][⟨fin⟩]
```

Cette version étoilée regroupe les entrées dans un unique paragraphe. L'argument *⟨séparateur⟩* est le séparateur entre les entrées et il y a un argument optionnel pour une ponctuation finale. Par exemple, ce document utilise :

```
\titlecontents*{subsection}[1.5em]
    {\small}
    {\thecontentslabel. }
    {}
    {, \thecontentspage}
    [---][.]
```

dont le résultat est affiché dans la table des matières en tout début de ce document. Notez que la mise en forme du paragraphe doit être écrite dans l'argument *⟨code-dessus⟩*.

Expliquons comment fonctionnent les arguments optionnels. Notez d'une part que leur nombre détermine leur sens — comme il doit y avoir un séparateur entre les entrées, celui est toujours présent ; d'autre part, *⟨début⟩* est rarement utilisé et est défavorisé en n'apparaissant que dans la version ayant le plus d'argument. Le cas le plus simple est celui où les titres ont le même niveau ; dans ce cas, les paramètres *⟨séparateur⟩* et *⟨fin⟩* (dont les valeurs par défaut sont des vides) sont respectivement insérés entre des entrées consécutives et à la fin du bloc. *⟨code-avant⟩* est exécuté une seule fois au tout début du bloc et ses déclarations sont locales à l'ensemble des entrées.

Maintenant supposons que nous souhaitons grouper les entrées de deux niveaux ; dans ce cas un principe d'imbrication s'applique. Pour détailler ce point, nous illustrerons avec une section et une sous-section. Quand une entrée de sous-section commence après une section, *⟨code-avant⟩* est exécuté et *⟨début⟩* (de sous-section), qui contient de la mise en forme de texte uniquement, est inséré. Les sous-sections sont ajoutées en insérant des séparateurs comme expliqué ci-dessus. Quand une section arrive, la ponctuation finale de la sous-section et le séparateur de la section sont ajoutés (à moins que le bloc ne s'achève par une sous-section, auquel cas seule la fin de la section est ajoutée). Nous avons dit qu'« une sous-section commence après une section » car une sous-section ne commence jamais un bloc<sup>27</sup>. Les entrées de sous-section sont regroupées au sein de chaque entrée de section et les déclarations sont à nouveau locales.

Un exemple va illustrer ceci :

```
\titlecontents*{section}[0pt]
    {\small\itshape}{}{}
    {}[ \textbullet\ ]{.}

\titlecontents*{subsection}[0pt]
    {\upshape}{}{}
    {, \thecontentspage}[ ( ) [. ] ]
```

27. Dans de rares cas, cela pourrait être nécessaire, cependant.

Ce code produit quelque chose similaire à :

*La première section • La deuxième • La troisième* (Une sous-section, 1. Une autre, 2) • *La quatrième* (Une sous-section, 1. Une autre, 2).

```
\contentsuse{<nom>}{<extension>}
```

Cette commande rend titletoc sensible à l'existence d'un fichier de table de matières (ou d'un fichier de liste) avec l'extension *<extension>*. Elle a pour rôle principal de s'assurer que la commande `\contentsfinish` est ajoutée à la fin de la table ou liste ciblée (et qui doit être ajoutée à la fin des tables des matières faites manuellement). L'extension exécute

```
\contentsuse{figure}{lof}
\contentsuse{table}{lot}
```

```
leftlabels rightlabels (options d'extension)
```

Ces options d'extension définissent l'alignement des numérotations dans `\contentslabel`. La valeur par défaut est `rightlabels`. Avec `leftlabels` le *<format>* par défaut pour `\contentslabel` devient `\thecontentstlabel\enspace`.

```
dotinlabels (option d'extension)
```

Avec cette option d'extension, un point est ajouté après la numérotation dans `\contentslabel`.

### 6.3. Table des matières partielle

```
\startcontents[<nom>]
```

À l'endroit où cette commande est utilisée, une table des matières partielle commence (notez que le document ne nécessite pas la présence d'un `\tableofcontents` pour que les tables de matières partielles fonctionnent). L'argument *<nom>* autorise différents ensembles de tables de matières et sa valeur par défaut est `default`. Ces ensembles peuvent être mélangés mais sont d'habitude séparés. Par exemple, vous pouvez souhaiter deux types de tables de matières partielles : par partie et par chapitre (en plus de la table générale complète, bien entendu). Quand une partie commence, écrivez `\startcontents[parts]` et, quand un chapitre commence, `\startcontents[chapters]`. De cette manière, une nouvelle table est entamée pour chaque partie et chapitre<sup>28</sup>.

```
\stopcontents[<nom>]
\resumecontents[<nom>]
```

La première commande arrête la table partielle de type *<nom>*, la seconde permet de la prolonger à nouveau. Comme les tables partielles peuvent être interrompues avec `\startcontents` si nécessaire, ces commandes ne seront pas souvent utilisées.

```
\printcontents[<nom>]{<préfixe>}{<niveau-début>}{<code-table>}
```

Cette commande affiche la table des matières actuelle de type *<nom>*. Le format des entrées de la table principale est utilisé, sauf si il y a un *<préfixe>*. Dans ce cas, le format de *<préfixe><niveau>* est utilisé s'il existe. Par exemple, si le préfixe est `l` et que le format `lsection` est défini, alors la définition sera utilisée ; sinon le format est celui de `section`. L'argument *<niveau-début>* fixe le niveau le plus haut des tables de matières — pour une table de matières d'une partie ce devrait être `0` (chapitre), pour une table de matières d'un chapitre `1` (section) et ainsi de suite. Enfin, *<code-table>* est un code local pour la table locale ; il peut être utilisé pour changer la valeur de `tocdepth` ou `\contentsmargin`, par exemple.

Une utilisation simple pourrait ressembler à ce qui suit (en supposant que vous utilisez aussi titlesec) :

<sup>28</sup>. Toutes les tables partielles sont stockées dans un unique fichier d'extension `.ptc`.

```

\titleformat{\chapter}[display]
{...}{...}{...} % Vos définitions sont ici
[\vspace*{4pc}%
\startcontents
\printcontents{1}{1}{\setcounter{tocdepth}{2}}]

\titlecontents*{lsection}[0pt]
{\small\itshape}{}{}
{}[ \textbullet\ ]{.}

```

Les entrées ici traitées sont celles de niveau 1 et 2.

#### 6.4. Liste partielle 2.6

Vous pouvez créer des tables partielles de figures et de tables. La syntaxe est ici similaire à celle des tables des matières et ce qui a été dit à leur propos peut être appliqué également ici. Les commandes sont :

```

\startlist[⟨nom⟩]{⟨liste⟩}
\stoplist[⟨nom⟩]{⟨liste⟩}
\resumelist[⟨nom⟩]{⟨liste⟩}
\printlist[⟨nom⟩]{⟨liste⟩}{⟨préfixe⟩}{⟨code-table⟩}

```

Ici, la *⟨liste⟩* est soit *lof* pour les figures, soit *lot* pour les tables. Notez également que *\printlist* n'a pas d'argument *⟨début-niveau⟩* parce que les tables de figures et de tables n'ont pas de niveau. Actuellement, seuls ces deux types de tables sont gérés mais dans une prochaine version le support d'autres tables sera ajouté. Malheureusement, de nombreuses classes placent des commandes de mise en forme dans ces listes (plus exactement, *\addvspace* dans les chapitres) ; je ne suis pas sûr de savoir comment retrancher ces commandes sans en retirer d'autres qui peuvent être souhaitées. En attendant, une astuce revient à retirer ces espaces en redéfinissant *\addvspace* dans *⟨code-table⟩* avec *\renewcommand\addvspace[1]{}*.

#### 6.5. Exemples

```

\titlecontents{chapter}
[0pt]
{\addvspace{1pc}%
\itshape}%
{\contentsmargin{0pt}%
\bfseries
\makebox[0pt][r]{\huge\thecontentslabel\enspace}%
\large}
{\contentsmargin{0pt}%
\large}
{\quad\thepage}
[\addvspace{.5pc}]

```

Le numéro de chapitre est placé dans la marge de gauche (à proximité du titre), dans une fonte plus grande que celle du titre. Si le chapitre n'a pas de numérotation (parce qu'il s'agit, par exemple, d'une préface ou d'une bibliographie), il n'est pas mis en gras. Le numéro de page suit le titre sans motif de remplissage mais avec un espace d'un em. Toute l'entrée est en italique.

```

\titlecontents{chapter}
[3pc]
{\addvspace{1.5pc}%
\filcenter}
{CHAPITRE \thecontentslabel\*.2pc}%
\huge}
{\huge}
{} % Autrement dit, sans numéro de page.
[\addvspace{.5pc}]

```

Le titre du chapitre est centré avec le mot « CHAPITRE » et la numérotation associée placés juste au dessus. Il n'y a pas de numéro de page.

## 6.6. Insertion d'une figure en table des matières

La commande `\addtocontents` est toujours disponible et vous pouvez l'utiliser pour exécuter des opérations particulières, comme insérer une figure juste avant ou après une entrée. Hélas, les arguments fragiles ne sont pas autorisés et écrire un code complexe peut devenir un enfer. L'astuce est de définir une commande qui effectue les opérations souhaitées et de l'utiliser en la faisant précéder de la commande `\protect`.

Supposons que nous voulons insérer une figure avant une entrée.

```
\newcommand{\figureintoc}[1]{
  \begin{figure}
    \includegraphics{#1}%
  \end{figure}}
```

fait le travail de fond. Là où est insérée la figure, indiquez alors :

```
\addtocontents{\protect\figureintoc{myfig}}
```

## 6.7. Repérage d'entrées avec des astérisques

Retrouvons maintenant un problème expliqué en relation avec `titlesec` : placer un astérisque sur les sections abordant un « sujet complexe » auquel nous ajoutons le besoin de faire figurer l'astérisque dans la table des matières.

```
\newcommand{\secmark}{}
\newcommand{\marktoc}[1]{\renewcommand{\secmark}{#1}}
\newenvironment{complexe}
  {\renewcommand{\secmark}{*}%
   \addtocontents{toc}{\protect\marktoc{*}}}
  {\addtocontents{toc}{\protect\marktoc{}}}
\titleformat{\section}
  {...}
  {\thesection\secmark}{...}{...}
\titlecontents{section}[...]{...}
  {\contentslabel{\thecontentslabel\secmark}{1.5pc}}{...}{...}
```

## 7. La philosophie de titlesec

Une fois que vous avez lu cette documentation, il devrait être clair que cette extension n'est pas destinée à l'utilisateur occasionnel aimant la présentation classique et souhaitant juste y apporter quelques modifications simples. Il s'agit ici d'un outil pour le typographe sérieux qui a une idée claire de la présentation souhaitée et qui n'avait auparavant pas les moyens pour l'obtenir. Rien n'est d'ailleurs fait ici pour améliorer votre goût en matière de mise en forme des sectionnements.

## 8. Annexes

Les exemples suivants servent d'illustrations. Dans cette partie, `\parskip` est fixé à 0 pt.

■

**9** Ceci est un exemple de la commande de section définie ci-dessous et, je dirais même plus, ceci est un exemple de la commande de section définie ci-dessous

```
\titleformat{\section}[block]
  {\normalfont\bfseries\filcenter}{\fbox{\itshape\thesection}}{1em}{}
```

■

## SECTION 10

## Un titre encadré

```
\titleformat{\section}[frame]
{\normalfont}
{\filright
\footnotesize
\enspace SECTION \thesection\enspace}
{8pt}
{\Large\bfseries\filcenter}
```

■

### 11. Un titre avec filet horizontal

```
\titleformat{\section}
{\titlerule
\vspace{.8ex}%
\normalfont\itshape}
{\thesection.}{.5em}{}
```

■

## 12

### Un autre titre avec filet horizontal

```
\titleformat{\section}[block]
{\normalfont\sffamily}
{\thesection}{.5em}{\titlerule\[\.8ex]\bfseries}
```

■

.....

13 La largeur du filet ci-dessus est celle de la plus longue  
ligne de ce titre augmentée de deux picas

.....

14 Celui-ci est plus court

```
\titleformat{\section}[block]
{\filcenter\large
\addtolength{\titlewidth}{2pc}%
\titeline*[c]{\titlerule* [.6pc]{\tiny\textbullet}}}%
\addvspace{6pt}%
\normalfont\sffamily}
{\thesection}{1em}{}
```

```
\titlespacing{\section}
{5pc}{*2}{*2}{5pc}
```

■

## SECTION 15

Ceci est un exemple de la commande de section définie ci-dessous et, je dirais même plus, ceci est un exemple de la commande de section définie ci-dessous. Répétons-le : ceci est un exemple de la commande de section définie ci-dessous et, je dirais même plus, ceci est un exemple de la commande de section définie ci-dessous

```
\titleformat{\section}[display]
{\normalfont\fillast}
```

```
{\scshape section \oldstylenums{\thesection}}
{1ex minus .1ex}
{\small}
\titlespacing{\section}
{3pc}{*3}{*2}{3pc}
```

■

CETTE PARTIE EST LE TITRE LUI-MÊME et cette partie est le corps du texte associé...

```
\titleformat{\section}[runin]
{\normalfont\scshape}
{}{0pt}{}
\titlespacing{\section}
{\parindent}{*2}{\wordsep}
```

■

#### 16. Un exemple simple d'une forme « wrap » sur une section

Suivi d'un peu de texte pour montrer le résultat. Suivi d'un peu de texte pour montrer le résultat. Suivi d'un peu de texte pour montrer le résultat. Suivi d'un peu de texte pour montrer le résultat. Suivi d'un peu de texte pour montrer le résultat. Suivi d'un peu de texte pour montrer le résultat.

#### 17. Et un autre

Notez comment le texte enveloppe (en anglais, « wrap ») le titre et que l'espace qui lui est réservé est automatiquement réajustée. Et il est suivi par un peu de texte pour illustrer le propos. Et il est suivi par un peu de texte pour illustrer le propos.

```
\titleformat{\section}[wrap]
{\normalfont\fontseries{b}\selectfont\filright}
{\thesection.}{.5em}{}
\titlespacing{\section}
{12pc}{1.5ex plus .1ex minus .2ex}{1pc}
```

■

**§ 18. Un titre attaché à l'ancienne.**—Bien sûr, vous pourriez préférer juste un point après le titre. Dans ce cas, l'argument optionnel devrait être [. ] et l'espace attenante devrait avoir une valeur assez forte (1em, par exemple).

```
\titleformat{\section}[runin]
{\normalfont\bfseries}
{\S\ thesection.}{.5em}{}[.---]
\titlespacing{\section}
{\parindent}{1.5ex plus .1ex minus .2ex}{0pt}
```

■

#### Exemple de section marginale

Suivi d'un peu de texte pour montrer le résultat. Mais ne vous arrêtez pas de lire car l'exemple qui suit illustre comment tirer parti d'autres extensions. La dernière commande dans le dernier argument peut prendre un argument, à savoir le titre sans autre commande additionnelle incluse. Nous donnons ici juste le code mais vous pouvez l'essayer. Donc, avec l'extension soul, vous pourriez écrire

```
\newcommand{\secformat}[1]{\MakeLowercase{\so{#1}}}% \so écarte les lettres
\titleformat{\section}[block]
{\normalfont\scshape\filcenter}
{\thesection}
{1em}
{\secformat}
```

Le titre marginal ci-dessus a été défini avec le code suivant :

```
\titleformat{\section}[leftmargin]
{\normalfont
\titlerule*{.6em}{\bfseries.}%
\vspace{6pt}%
\sffamily\bfseries\filleft}
{\thesection}{.5em}{}
\titlespacing{\section}
{4pc}{1.5ex plus .1ex minus .2ex}{1pc}
```

■

Les exemples suivants sont pensés pour des chapitres. Cependant, cette documentation ne dispose pas de la commande `\chapter` et illustre ce point en utilisant `\section` avec quelques rares changements.

---

## CHAPITRE 19

---

### Le titre

```
\titleformat{\chapter}[display]
{\normalfont\Large\filcenter\sffamily}
{\titlerule[1pt]%
\vspace{1pt}%
\titlerule
\vspace{1pc}%
\LARGE\MakeUppercase{\chaptertitlename} \thechapter}
{1pc}
{\titlerule
\vspace{1pc}%
\Huge}
```

■

## CHAPITRE XX

---

### Le titre

---

```
\renewcommand{\thechapter}{\Roman{chapter}}
\titleformat{\chapter}[display]
{\bfseries\Large}
{\filleft\MakeUppercase{\chaptertitlename} \Huge\thechapter}
{4ex}
{\titlerule
\vspace{2ex}%
\filright}
[\vspace{2ex}%
\titlerule]
```





```

\titleformat{\subsubsection}
  {\normalfont\normalsize\bfseries}{\thesubsubsection}{1em}{}
\titleformat{\paragraph}[runin]
  {\normalfont\normalsize\bfseries}{\theparagraph}{1em}{}
\titleformat{\subparagraph}[runin]
  {\normalfont\normalsize\bfseries}{\thesubparagraph}{1em}{}

\titlespacing*{\chapter}      {0pt}{50pt}{40pt}
\titlespacing*{\section}      {0pt}{3.5ex plus 1ex minus .2ex}{2.3ex plus .2ex}
\titlespacing*{\subsection}   {0pt}{3.25ex plus 1ex minus .2ex}{1.5ex plus .2ex}
\titlespacing*{\subsubsection}{0pt}{3.25ex plus 1ex minus .2ex}{1.5ex plus .2ex}
\titlespacing*{\paragraph}    {0pt}{3.25ex plus 1ex minus .2ex}{1em}
\titlespacing*{\subparagraph} {\parindent}{3.25ex plus 1ex minus .2ex}{1em}

```

### 9.3. Exemple de chapitre

Un dernier exemple montre comment tirer parti de l'environnement `picture` pour des formats de sectionnement moins courants. Même avec les outils simples fournis par la version standard de  $\text{\LaTeX}$ , vous pouvez créer des titres marquants. Mais vous pouvez en faire d'encore plus élaborés avec, par exemple, `pspicture` (de l'extension `PSTricks`) ou en incorporant des graphiques créés avec l'aide de programmes externes.

```

\usepackage[dvips]{color}
\usepackage[rigidchapters,explicit]{titlesec}

\DeclareFixedFont{\chapterfont}{T1}{phv}{bx}{n}{11cm}

\titlespacing{\chapter}{0pt}{0pt}{210pt}
% La plupart des titres ont un peu de profondeur. L'espace total
% est un peu plus large que la boîte contenant l'image.

\titleformat{\chapter}[block]
  {\begin{picture}(330,200)}
  {\put(450,80){%
    \makebox(0,0)[rb]{%
      \chapterfont\textcolor[named]{SkyBlue}{\thechapter}}}}
  {\put(0,230){%
    \makebox(0,0)[lb]{%
      \Huge\sffamily\underline{Chapitre \thechapter}}}}
  {0pt}
  {\put(0,190){\parbox[t]{300pt}{%
    \Huge\sffamily\filright#1}}}
  [\end{picture}]

```

(Les valeurs exactes à utiliser dépendent de la zone de texte, de la classe, de `\unitlength`, du format du papier, etc.)