

Les extensions titlesec, titleps et titletoc *

Javier Bezos[†]

2011-12-15

Table des matières

1. Introduction	1
2. Interface simple	2
2.1. Format, 2.—2.2. Espacement, 2.—2.3. Majuscules, 3.—2.4. Outils, 3.	
3. Interface avancée	3
3.1. Format, 3.—3.2. Espacement, 4.—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.	
4. Notes additionnelles	11
4.1. Numérotations à largeur fixée, 11.—4.2. Versions étoilées, 11.—4.3. Variantes, 12.—4.4. Placement d’un point après un titre de section, 13.	
5. L’extension titleps et le style de page	13
6. L’extension titletoc et la table des matières	13
6.1. Visite guidée de titletoc en 10 minutes, 14.—6.2. Et bien plus, 16.—6.3. Partial TOC’s, 18.—6.4. Partial lists, 18.—6.5. Exemples, 19.—6.6. Insertion d’une figure en table des matières, 19.—6.7. Repérage d’entrées avec des astérisques, 19.	
7. La philosophie de titlesec	20
8. Annexes	20
9.1. Un exemple complet, 23.—9.2. Classes standards, 24.—9.3. Exemple de chapitre, 24.	

1. Introduction

Cette extension est essentiellement un remplacement — partiel ou total — des macros \LaTeX liées au sectionnement — principalement les titres, en-têtes et tables des matières. Le but est d’apporter de nouvelles fonctionnalités non disponibles actuellement sous \LaTeX : si vous souhaitez juste une interface plus accessible que celle d’un \LaTeX standard mais 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 jolies choses.¹

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, avec des titres numérotés ou pas, mesurer la largeur d’un titre, ajouter un nouveau niveau 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 type `\...mark`, et contenant éventuellement ✖ **des marques de haut, de première valeur et de double valeur** ✖. Des marques de haut de page correctement synchronisées avec les titres, sans incompatibilités

*L’extension titlesec est actuellement en version 2.10.0. © 1998–2011 Javier Bezos. L’extension titletoc est actuellement en version 1.6. L’extension titleps est actuellement en version 1.1.0 © 1999–2011 Javier Bezos. Javier Bezos. Tous droits réservés.

[†]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.

avec la 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, incluant celles de l'AMS, et elle fonctionne sans aucune difficulté avec `hyperref`.² Malheureusement, elle n'est pas compatible avec `memoir`, qui propose 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 avec les paramétrages simples prédéfinis (voir section « Interface simple ») ou 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 parce que `\part` est habituellement 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 simple

La façon la plus simple de changer le format se fait par le moyen d'un ensemble d'options de l'extension et par un couple de 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 simple ».

rm sf tt md bf up it sl sc

Option sélectionnant la famille/série/forme correspondante. La valeur par défaut est `bf`.

big medium small tiny

Option sélectionnant la taille des titres. Elle est fixée par défaut à `big`, ce qui donne la taille retenue dans les classes standards. Avec `tiny`, les 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

Option contrôlant l'alignement.

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 implémentent à nouveau 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.

2.3. Majuscules

```
uppercase
```

2.9 Mise en majuscules des titres. Selon la classe, cela peut ne marcher avec `\chapter` et `\part`.

2.4. Outils

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

Changement du format de la numérotation des sections, sous-sections, etc. Une commande `\thetitle` est fournie et correspond respectivement à `\thesection`, `\thesubsection`, etc. La valeur par défaut des classes standards est

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

et vous pouvez simplement ajouter un point après le compteur 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}
```

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 plus plupart des cas, vous souhaitez modifier soit l'espacement, soit le format³. 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 ci-après, et leur format peut être fixé par les commandes décrites ici.

3.1. Format

Un ensemble de formes contrôlant la distribution basique des éléments d'un titre est proposé. Les formes disponibles sont :

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

block compose le titre complet en un bloc (un paragraphe) sans mise en forme additionnelle. Utile pour les titres centrés⁴ 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).

runin Un titre suivi directement du texte sur la même ligne (comme avec `\paragraph` dans les classes standards).

3. 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.

4. La numérotation sera légèrement déplacé vers la gauche si le titre fait plus d'une ligne de long et que la forme `hang` est utilisée, exception faite des cas avec des `\\` explicites.

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 mener à des pages trop peu remplies⁵. 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 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 expliquées ci-après pour `calwidth` 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>}{<sep>}{<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 (ou titres courants).
- `<sep>` 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 `<sep>` 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 qui est le texte du titre⁶. Toutefois, avec l'option d'extension `explicit`, le titre doit être donné explicitement avec `#1` (voir ci-dessous).
- `<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 renvoi vers `\appendixname`. Utilisez-là en lieu et place de `\chaptername` lorsque vous définissez un chapitre.

3.2. Espacement

```
\titlespacing*{<commande>}{<gauche>}{<avant-sep>}{<après-sep>}{<droite-sep>}
```

La version étoilée supprime l'indentation du paragraphe suivant le titre, sauf avec `drop`, `wrap` et

5. 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

6. 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 le ✖ **leading** ✖ pourrait être faux — trop grand ou trop petit.

runin pour lesquels cette possibilité n’a pas de sens.

- `<gauche>` augmente la marge de gauche, sauf pour les formes `...margin` et `drop` où ce paramètre fixe la largeur du titre, pour `wrap`, la largeur maximale, et dans `runin` l’indentation juste avant le titre. Avec une valeur négative, le titre ✖ **overhangs** ✖⁷.
- `<sep-avant>` est l’espace vertical avant le titre.
- `<sep-après>` 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`⁸.
- Les formes `hang`, `block` et `display` disposent de la possibilité d’augmenter la marge `<sep-droit>` avec cet argument optionnel.

Si vous n’aimez pas saisir les valeurs complètes des espaces, incluant les paramètres `plus` et `moins`, une abréviation `*n` est disponible. Dans l’argument `<sep-avant>` c’est équivalent à `n ex` avec une certaine ✖ **extensibilité** ✖ et une infime ✖ **compressabilité** ✖. Dans le `<sep-après>`, une certaine ✖ **extensibilité** ✖ (plus petite) et aucune ✖ **compressabilité** ✖⁹. 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 changer si vous n’aimez pas les valeurs prédéfinies (ou 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 ainsi génère 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
```

Variantes des commandes `\ragged...`, avec de légères différences. En particulier, les commandes `\ragged...` suppriment les espaces à droite et à gauche disposés par `\titlespacing`¹⁰. `fillast` justifie le paragraphe, à l’exception de sa dernière ligne qui est centrée¹¹. Ces commandes fonctionnent également pour la numérotation associée à `frame`.

`\filinner` et `\filouter` correspondent à `\filleft` ou `\filright` selon la page où elles se trouvent. Du fait du découpage asynchrones 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 « ✖ **Paramétrage étendu** ✖ » ci-dessous.

```
\wordsep
```

L’espace ✖ **inter-mot** ✖ pour la fonte courante.

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

Contournement des paramétrages pour toutes les commandes de sectionnement¹²

7. 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.

8. Voir Goossens, Mittelbach and Samarin : *LT_εX Companion*, Reading, Addison Wesley, 1993, p. 25.

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

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

11. Le nom est un petit peu étrange mais il est court.

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

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

Avec `rigidchapters`, l'espacement pour les titres des chapitres est toujours le même et `sep-après` dans `\titlespacing` ne représente pas l'espace entre le bas du texte du titre et le corps du texte comme décrit plus haut mais depuis le haut du texte du titre, autrement $\langle sep\text{-}avant \rangle + \langle sep\text{-}après \rangle$ est maintenant une distance fixe entre le haut du **✖ corps de page ✔** et le texte principal. La valeur par défaut est `rubberchapters` où $\langle sep\text{-}après \rangle$ désigne la séparation entre le titre et le texte, comme à l'accoutumée. En effet le nom est trompeur car il ne s'applique pas qu'au chapitre par défaut mais à tout titre de classe `top` (les classes sont développées dans la suite du document).

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

Si `nobottomtitles` est sélectionné, les titres proches de la marge du bas sont renvoyés en page suivante et la marge sera **✖ ragged ✔**. L'espace minimal requis pour que la marge du bas ne déplace pas le titre est fixée (approximativement) par

```
\renewcommand{\bottomtitlespace}{\langle length \rangle}
```

dont la valeur par défaut est `.2\textheight`. Une simple marge **✖ ragged ✔** 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*` fournit des calculs plus précis mais les titres des formes `margin`, `wrap` ou `drop` peuvent être mal placés. 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 $\langle sep\text{-}après \rangle$ du premier titre est utilisé entre les deux titres. Parfois, ce comportement n'est pas souhaité, en particulier quand l'espace $\langle sep\text{-}avant \rangle$ est plus grand que $\langle sep\text{-}après \rangle$ (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 je présume que personne ne veut un saut de page dans un titre, ceci est le comportement par défaut. Si, pour une raison étrange, vous souhaitez autoriser ces sauts de page dans les titres, utilisez l'option `pageatnewline`, qui est 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 et 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 matériaux « horizontaux ». Elle est pensée principalement pour des filets et des points de conduite¹³ 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 **✖ sous-remplie ✔**. Ici le paramètre optionnel $\langle alignement \rangle$ (`l`, `r` ou `c`) peut aider, de telle sorte que vous puissiez simplement saisir `\titleline[c]{CHAPTER}`¹⁴.

13. Suite de points régulièrement espacés qui se trouvent dans les tables de matières le plus souvent.

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

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) and *⟨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 *⟨before-code⟩* et *⟨label⟩*. Mais vous pouvez tester cette commande ailleurs : il y a de grandes chances qu'elle fonctionne à d'autres endroits.

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 *⟨height⟩* si ce dernier est présent. Par exemple,

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

traces deux filets de différents 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⟩* dans une boîte de sa largeur naturelle. Cette largeur de boîte peut être modifiée avec *⟨largeur⟩* mais la largeur de la première boîte reste inchangée pour 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'étendant au-dessus du texte principal et du titre précèdent la section.

`calwidth` (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`¹⁵.

Du point de vue de \TeX , 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'erreurs¹⁶. Pour être complet, certaines commandes peuvent embrouiller \TeX et interrompre l'analyse du titre. Mais, si vous vous en tenez à du texte, `\\` et `\\[...]` (et il y a peu de chance 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⟩*, *⟨sep⟩*, 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 une **✖ assignation conflictuelle ✖** dans le *⟨code-après⟩* passera mais vous pouvez également passer la commande suivante.

`\iftitlemeasuring{⟨vrai⟩}{⟨faux⟩}`

2.9 Quand le titre est mesuré (première passe), l'argument *⟨vrai⟩* est traité tandis que, quand le titre est réellement composé (seconde passe), 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

15. 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*`.

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

une boîte de largeur `\titlewidth` ; cette boîte sera à son tour placé dans la boîte principal avec l'alignement indiqué. Il n'y a pas d'équivalent `\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 ¹⁷ :

```
\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>\<section>break</code>

En définissant ces commandes avec `\newcommand`, différents saut 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 est accompli 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) 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}{}%
```

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`).

¹⁷. Nommée dans l'éphémère version 2.7 `\titlepagestyle`.

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, par exemple :

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

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

`newparttoc oldparttoc` (options d'extension)

Les parties 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 dans le cas où `\part` a été redéfini).

`cleareempty` (option d'extension)

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 Change le comportement de l'argument optionnel dans les commandes de sectionnement de telle manière qu'elles fixent 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 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.

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 dépendant du contexte¹⁸. Cet argument peut être une liste de paires clé/valeur de la forme :

`<clé>=<valeur>, <clé>=<valeur>, <clé>, <clé>, ...`

Actuellement, seules les pages et les versions non numérotées sont prises en charge, **✗ besides ✗** le nom de la commande de sectionnement. Ainsi, les clés disponibles sont :

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

La forme basique décrite ci-dessus avec le nom de la commande de sectionnement, par exemple

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

est en fait une abréviation pour

¹⁸. L'extension `keyval` est requise pour pouvoir utiliser cette syntaxe.

```
\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 est généralement une base convenable pour parvenir à générer la version non numérotée car la suppression de la numérotation est le seul changement requis dans la plupart des cas. Mais, si vous avez besoin de mises en forme spéciales, la clé `numberless` existe et définit une version alternative des sections sans numéros (autrement dit celles avec un niveau sous `secnumdepth`, dans les parties préliminaires et annexes du document¹⁹ et, bien entendu, la version étoilée). Par exemple

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

```

`<label>` et `<sep>` 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. En effet, « paire » et « impaire » sont des termes bien établis avec L^AT_EX mais souvent source de confusion. Dans une impression en recto uniquement, les pages « impaires » sont également des pages « paires » (cf. `\oddsidemargin`).

Si vous souhaitez créer différentes versions pour les pages paires/impaires *et* avec ou sans numérotation, il est recommandé de définir quatre variantes.

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

3.9. Création de nouveaux niveaux et changement de classe

Quand les formes et autres modifient le comportement des titres en lien avec le texte voisin, 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, s'il y a une table des matières, le format de cette table est laissé à la responsabilité de l'utilisateur.* Il existe trois classes : `page` est comme `\part` (issue de la classe `book`) sur une page séparée, `top` est comme `chapter` en commençant sur une nouvelle page et en plaçant le titre en haut de page, `straight` est pensé pour les titres en milieu de texte²⁰.

```
\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 est directement sous celui du `<commande-du-niveau-supérieur>`. Par exemple :

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

19. respectivement obtenue avec `\frontmatter` et `\backmatter`.

20. 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. Utiliser la classe `straight` à la place. Rappelez-vous que certaines fonctionnalités reposent sur ces classes et que ne change pas `titlesec` par défaut les définitions de `\part` et `\chapter`.

crée un niveau sous celui du chapitre (du code additionnel est montré ici mais vous devez lui ajouter les paramétrages `\titleformat` et `\titlespacing` correspondant)²¹. Si le niveau du chapitre est zéro alors celui du sous-chapitre est 1 ; les niveaux en dessous sont incrémenté 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 et ignorant la classe :

- `\chapter` enregistre la chaîne définie dans `\chaptertitlename` et la prend en compte.
- `\part` n’encapsule 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 tous les définitions antérieures, si elles existent, et retire ainsi la possibilité d’utiliser les options décrites dans « Interface simple ». Alors, vous pouvez utiliser les outils de `titlesec` et construire les compteurs et numérotations correspondantes.

`\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 maniée uniquement lors de la création de commande de sectionnement à partir de rien avec l’aide de `loadonly` et il doit y avoir une unique déclaration de ce type. Après celle-ci, le reste des niveaux est ajouté comme défini ci-dessus.

4. Notes additionnelles

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

4.1. Numérotations à largeur fixée

La commande `\makebox` permet d’utiliser une numérotation à largeur fixée, ce qui fait reposer la marge gauche du titre actuel (pas la numérotation) au même endroit. Par exemple (seul le code important est indiqué) :

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

Voir votre manuel \LaTeX pour plus d’information sur les commandes portant sur les boîtes.

4.2. Versions étoilées

Recourir à la version étoilée dans les commandes de sectionnement est fortement découragé. En lieu et place, vous pouvez utiliser un ensemble de commandes **✖ markup oriented ✖** qui sont facile à définir et 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 incluses 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.
```

21. En ce qui concerne les compteurs, l’extension `remreset` peut être utile.

```

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

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

```

La définition suivante supprime les nombres 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}}

```

(Je trouve ce dernier cas 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 et ils ne sont 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` (voir ci-dessous).

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 ce point à merci ; par exemple, si vous décidez que ces sections doivent être composées en petits caractères (avec `\small`) et que vous réalisez que vous n'aimez pas cela, retirez-le.

Tandis que les commandes d'un L^AT_EX standard 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 avancé » par une astérisque après la numérotation. Le code suivant effectue cette mise en forme :

```

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

```

Pour marquer une section, écrivez :

```

\begin{avance}
\section{...}
...
\end{avance}

```

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 était relativement courant auparavant. La technique de base a été décrite ci-dessus mais en voici un rappel :

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

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
```

Cependant, vous pourriez préférer ici 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 les styles de pages (en-têtes et pieds de page). Une interface de plus 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, est ainsi :

```
\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 \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 l'extension `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, et vous avez alors accès à de nouvelles fonctionnalités. La mise en forme

des paragraphes et les 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.

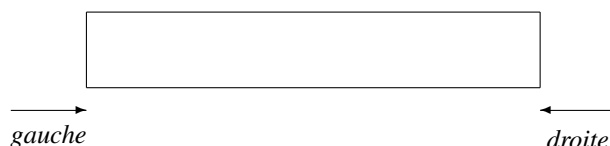
Le comportement des entrées définies avec `titletoc` est différent à certains points de celui de celles créées avec les commandes standards. En particulier :

- les pages ne sont jamais coupées entre 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 coupure est autorisée. Enfin, si la première entrée est d'un niveau inférieur à la seconde, la possible coupure 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.

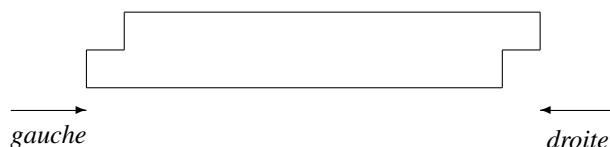
Je souhaite ici souligner qu'aucun essai pour manipuler les tables des matières ne peut être complet car les commandes standards de \LaTeX écrivent directement des commandes de mises en forme qui ne peuvent être retranchées. C'est tout particulièrement le cas pour les listes de figures et de tables et pour la commande `\part`²²

6.1. Visite guidée de titletoc en 10 minutes

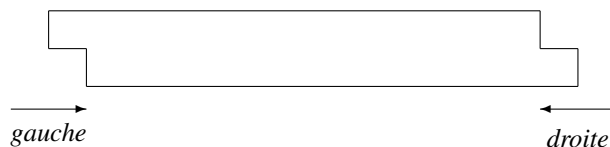
Les entrées de table de matière sont traitées comme des zones rectangulaires où le texte et un probable élément de remplissage seront composés. Dessinons une telle zone (les lignes ne seront pas ici affichées) :



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 êtes autorisé à 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 :



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 tout à la 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 l'espace au début et y place la numérotation ;

²² Mais certains de ces problèmes sont résolus par `titlesec`.

- `\contentspage` crée l'espace à la fin de largeur $\langle droite \rangle$ et y écrit le numéro de page aligné à droite.

Maintenant, nous allons voir trois commandes basiques :

```
\dottedcontents{ $\langle section \rangle$ }[ $\langle gauche \rangle$ ]{ $\langle code-dessus \rangle$ }
{ $\langle largeur-num \rangle$ }{ $\langle largeur-motif \rangle$ }
```

Here :

- $\langle section \rangle$ est le nom de la **section** 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).
- $\langle code-dessus \rangle$ est le code mettant en forme globalement l'entrée. Des éléments 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`.
- $\langle gauche \rangle$, 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.
- $\langle largeur-num \rangle$ est la largeur de l'espace créé pour la numérotation comme décrit ci-dessus.
- $\langle largeur-motif \rangle$ est la largeur de la boîte contenant le caractère qui va servir de motif au remplissage de la ligne, comme décrit ci-dessous.

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{ $\langle section \rangle$ }[ $\langle gauche \rangle$ ]{ $\langle code-dessus \rangle$ }
{ $\langle format-entrée-numérotée \rangle$ }{ $\langle format-entrée-non-numérotée \rangle$ }
{ $\langle format-remplissage-page \rangle$ }[ $\langle code-dessous \rangle$ ]
```

Ici $\langle section \rangle$, $\langle gauche \rangle$ and $\langle code-dessus \rangle$ fonctionnent comme ci-dessus et

- $\langle format-entrée-numérotée \rangle$ 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 ;
- $\langle format-entrée-non-numérotée \rangle$ est comme le précédent argument... en l'absence de numérotation ;
- $\langle format-remplissage-page \rangle$ 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 matières²³. Vous pourriez préférer un `\hspace` à la place.
- et, enfin, $\langle code-dessous \rangle$ 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, listes des tables et des listes à n'importe quel moment. Le nouveau format est écrit dans le fichier toc et nécessite donc deux compilations pour être observable.

```
\contentsmargin{ $\langle droite \rangle$ }
```

La valeur fixée est utilisée dans toutes les sections. Si vous vous demandez pourquoi, la réponse est plutôt simple : dans la plupart des cas, la marge $\langle droite \rangle$ sera constante. Cependant, vous avez le droit de la changer localement dans les arguments $\langle code-avant \rangle$. 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 :

23. Pour les T_EXniciens, `\xleaders` devient ici `\leaders`.

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

Comparez cette définition avec celle donnée plus haut et vous comprendrez 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. De façon assez étonnante, \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` dans la syntaxe complète est la suivante :

```
\contentsmargin[⟨correction⟩]{⟨droite⟩}
```

La longueur de `⟨correction⟩` est ajoutée à celle de `⟨right⟩` dans toutes les lignes sauf la dernière, 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 la commande `\contentsmargin` pour y ajouter une `⟨correction⟩` de 6 pt²⁴. Contrairement aux outils \LaTeX standards, la commande `\titlerule*` a été développée pour que la `⟨correction⟩` ait la valeur la plus petite 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.

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

Comme décrit plus haut, mais avec différents `⟨format⟩`s. 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

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

document demande :

```
\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>*.

✖ Let us explain how the optional arguments works. First note the number of them determines their meaning—since there should be a separator between entries this one is always present ; on the other hand, *<begin>* is rarely used and hence it has the lowest “preference.” The simplest case is when the titles are of the same level ; in this case the *<separator>* and the *<end>* parameters (which default to empty) are inserted between consecutive entries and at the end of the block, respectively. *<before-code>* is executed just once at the very beginning of the block and its declarations are local to the whole set of entries.

Now suppose we want to group entries of two levels ; in this case a nesting principle applies. To fix ideas, we will use section and subsection. When a subsection entry begins after a section one, *<before-code>* is executed and *<begin>* of subsection is inserted, which should contain text format only. Then subsections are added inserting separators as explained above. When a section arrives, the ending punctuation of subsection and the separator of section is added (except if the block is finished by a subsection, where the ending of section is added instead). We said “after a section” because a subsection never begins a block.²⁵ The subsection entries are nested inside the section ones, and declarations are again local.

An example will illustrate that.

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

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

produces something similar to :

The first section • The second one • The third one (A subsection in it, 1. Another, 2) • *A fourth section* (A subsection in it, 1. Another, 2).

`\contentsuse{<name>}{<ext>}`

Makes titletoc aware of the existence of a contents file with *<ext>* extension. Mainly, it makes sure the command `\contentsfinish` is added at the end of the corresponding contents (and which must be added at the end of tocs made by hand). The package performs

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

`leftlabels rightlabels` (Package options)

These package options set how the labels are aligned in `\contentslabel`. Default is `rightlabels`. With `leftlabels` the default *<format>* for `\contentslabel` becomes `\thecontentstlabel\enspace`.

`dotinlabels` (Package option)

With this package option, a dot is added after the label in `\contentslabel`.

25. In rare cases that could be necessary, yet.

6.3. Partial TOC's

```
\startcontents[⟨name⟩]
```

At the point where this command is used, a partial toc begins (note the document doesn't require a `\tableofcontents` for partial tocs to work). The `⟨name⟩` argument allows different sets of tocs and it defaults to default. These sets may be intermingled, but usually will be nested. For example, you may want two kinds of partial tocs : by part and by chapter (besides the full toc, of course). When a part begins, write `\startcontents[parts]`, and when a chapter `\startcontents[chapters]`. This way a new toc is started at each part and chapter.²⁶

```
\stopcontents[⟨name⟩]
\resumecontents[⟨name⟩]
```

Stops the partial toc of `⟨name⟩` kind, which may be resumed. Since partial contents are stopped by `\startcontents` if necessary, those macros will not be used very often.

```
\printcontents[⟨name⟩][⟨prefix⟩][⟨start-level⟩][⟨toc-code⟩]
```

Print the current partial toc of `⟨name⟩` kind. The format of the main toc entries are used, except if there is a `⟨prefix⟩`. In such a case, the format of `⟨prefix⟩⟨level⟩` is used, provided it is defined. For example, if prefix is 1 and the format of `1section` is defined, then this definition will be used ; otherwise, the format is that of `section`. The `⟨start-level⟩` parameter sets the top level of the tocs—for a part toc it would be 0 (chapter), for a chapter toc 1 (section), and so on. Finally, `⟨toc-code⟩` is local code for the current toc ; it may be used to change the `tocdepth` value or `\contentsmargin`, for instance.

A simple usage might look like (provided you are using `titlesec` as well) :

```
\titleformat{\chapter}[display]
{...}{...}{...} % Your definitions come here
[\vspace*{4pc}%
\startcontents
\printcontents{1}{1}{\setcounter{tocdepth}{2}}]

\titlecontents*{1section}[0pt]
{\small\itshape}{}{}
{}[ \textbullet\ ]{.}
```

The included entries are those in levels 1 to 2 inclusive (i.e., 1 and 2).

6.4. Partial lists 2.6

You may want to create partial LOFs and LOTs. The syntax is similar to that of partial TOCs and what was said for them can be applied here. The commands are :

```
\startlist[⟨name⟩][⟨list⟩]
\stoplist[⟨name⟩][⟨list⟩]
\resumelist[⟨name⟩][⟨list⟩]
\printlist[⟨name⟩][⟨list⟩][⟨prefix⟩][⟨toc-code⟩]
```

Here `⟨list⟩` is either `lof` or `lot`. Note as well `\printlist` does not have the `⟨start-level⟩` argument, because figures and tables have not levels. Currently, only those two float lists are supported, but in a future release support for more kinds of float lists will be added. Unfortunately, many classes write some formatting commands to these lists (more precisely, `\addvspaces` in chapters) ; I'm still not sure how to remove these commands without removing as well others which can be wanted, but for the time being a quick trick to remove these spaces is to redefine `\addvspace` in the `⟨toc-code⟩` with `\renewcommand\addvspace[1]{}.`

²⁶. All partial tocs are stored in a single file with extension `.ptc`.

6.5. Examples

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

The chapter number is out at the edge of the page margin, in a font larger than the font of the title. If the chapter lacks of number (because, say, it is the preface or the bibliography) it is not boldfaced. The page number follows the title without fillers, but after an em-space.

```
\titlecontents{chapter}
    {3pc}
    {\addvspace{1.5pc}%
     \filcenter}
    {CHAPTER \thecontentslabel\*[.2pc]%
     \huge}
    {\huge}
    {} % That is, without page number
    [\addvspace{.5pc}]
```

The chapter title is centered with the chapter label on top of it. There is no page number.

6.6. Insertion d'une figure en table des matières

The `\addtocontents` command is still available and you may use it to perform special operation, like inserting a figure just before or after of an entry. Sadly, fragile arguments are not allowed and writing complex code could be a mess. The trick is to define a command to perform the required operations which in turn is written with `\protect`.

Let's suppose we want to insert a figure before an entry.

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

```

makes the dirty work.

In the place where a figure is inserted write :

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

✖

6.7. Repérage d'entrées avec des astérisques

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

```
\newcommand{\secmark}{}
\newcommand{\marktoc}[1]{\renewcommand{\secmark}{#1}}
\newenvironment{advanced}
  {\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 qui aime la présentation classique et souhaite 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 fait non plus 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}%
   \titleline*[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 réajusté. 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 spaces out letters
\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 par de la commande `\chapter` et illustre ce point en utilisant `\section` avec quelques rares changements.

CHAPTER 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}
```



CHAPTER 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]
```

9.1. Un exemple complet

Voici maintenant un exemple complet d'une structure de titres.

```
\documentclass[twoside]{report}
\usepackage[sf,sl,outermarks]{titlesec}

% \chapter, \subsection...: pas de code additionnel

\titleformat{\section}
  {\LARGE\sffamily\slshape}
  {\thesection}{1em}{}
\titlespacing{\section}
  {-6pc}{3.5ex plus .1ex minus .2ex}{1.5ex minus .1ex}

\titleformat{\paragraph}[leftmargin]
  {\sffamily\slshape\filright}
  {}{}{}
\titlespacing{\paragraph}
  {5pc}{1.5ex minus .1 ex}{1pc}

% 5+1=6, autrement dit, la marge négative de gauche dans
% la section

\widenedhead{6pc}{0pc}

\renewpagestyle{plain}{}

\newpagestyle{special}[\small\sffamily]{
  \headrule
  \sethead[\usepage][\textsl{\chaptertitle}][
    {}{\textsl{\chaptertitle}}{\usepage}]

\newpagestyle{main}[\small\sffamily]{
  \headrule
```

```

\sethead[\usepage][\textsl{\thechapter. \chaptertitle}][
  {}{\textsl{\thesection. \sectiontitle}}{\usepage}]

\pagestyle{special}

\begin{document}

---Table des matières

\pagestyle{main}

---Corps du document

\pagestyle{special}

---Index
\end{document}

```

9.2. Classes standards

Voici, pour votre information, comment peuvent être définies les commandes des classes standards.

```

\titleformat{\chapter}[display]
  {\normalfont\huge\bfseries}{\chaptertitlename\ \thechapter}{20pt}{\Huge}
\titleformat{\section}
  {\normalfont\Large\bfseries}{\thesection}{1em}{}
\titleformat{\subsection}
  {\normalfont\large\bfseries}{\thesubsection}{1em}{}
\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 \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 exaces à utiliser dépendent de la zone de texte, de la classe, de `\unitlength`, du format du papier, etc.)