

*French*L^AT_EX

L^AT_EX POUR LES JURISTES

Documentation dans l'ordre du code,
d'après la méthode de la programmation lettrée

Flora VERN

*French*L^AT_EX

L^AT_EX POUR LES JURISTES

/Commentaire/Documentation dans l'ordre du code,
d'après la méthode */de l'exégèse/de la programmation* lettrée

Flora VERN

Cette documentation a été produite sous X_YL^AT_EX à partir des différents fichiers de l’extension *FrenchL^AW*, qui contiennent tant le code informatique que son explication, dans le même ordre, suivant les principes de la programmation lettrée développés par Donald K_NUTH, créateur du logiciel T_EX.

Le corps du texte est composé avec la police d’écriture *Junicode* de Peter S. BAKER, destinée notamment aux médiévistes, et le code informatique avec *Latin Modern Mono* développée par la L^AT_EX Project Team. Quelques éléments sont composés en sans-serif avec *Concourse* de Matthew BUTTERICK, auteur de plusieurs ouvrages sur la typographie juridique.

La documentation dans son entier constitue un exemple d’utilisation du logiciel qu’elle présente. Elle est rédigée directement à l’intérieur du code informatique qu’elle commente, dans l’ordre de celui-ci. Un petit script sépare ensuite les deux fichiers pour la distribution, en supprimant le double signe pourcent (%) au début des lignes du code pour produire les fichiers de documentation.



© 2009–2024 Flora VERN

FrenchL^AW et sa documentation sont protégés par le droit d’auteur, et soumis à ce titre à la L^AT_EX Project Public Licence, version 1.3c ou ultérieure, sous le statut *author-maintained* : <http://www.latex-project.org/lppl/>

TABLE DES MATIÈRES

Explication d'un document L^AT_EX	1
Paramètres de compilation pour <i>FrenchL^AW</i>	2
Chapitre I. Introduction générale à L ^A T _E X	3
Section I. Pourquoi L ^A T _E X?	3
Section II. Principes de la rédaction sémantique	3
Section III. Création et modification de commandes	3
Section IV. Structure des fichiers dans les projets complexes	3
Section V. Correction des erreurs apparaissant dans le log	3
Chapitre II. Options de la classe <i>FrenchL^AW</i>	4
Section I. Format de la page	4
Section II. Paramètres typographiques	5
Section III. Formats de documents	6
Section IV. Fonctionnalités additionnelles	7
Section V. Fonctionnalités développeur	9
Section VI. Sélection des langues	9
Section VII. Dépendances	10
Chapitre III. Composition typographique	12
Section I. Construction des pages	12
Section II. Choix des polices d'écriture	13
Section III. Choix d'une échelle typographique	14
Section IV. Composition sur la grille typographique	14
Section V. Composition des paragraphes	16
Section VI. Couleurs	17

PARTIE I. STRUCTURE DU DOCUMENT

Chapitre I. Divisions du document	20
Section I. Divisions hiérarchiques	20
Section II. Divisions logiques	23
§ I. Structure d'ensemble de l'ouvrage	23
§ II. Structuration du texte par numéros	24

SOMMAIRE

Chapitre I. Introduction générale à L^AT_EX

Chapitre II. Options de la classe *FrenchL^AT_EX*

Chapitre III. Composition typographique

PARTIE I. STRUCTURE DU DOCUMENT

Chapitre I. Divisions du document

EXPLICATION D'UN DOCUMENT L^AT_EX

Pour créer un document simple, il suffit de mettre le texte suivant dans un fichier `.tex` et le compiler avec L^AT_EX ou X_YL^AT_EX :

```
? \documentclass[options]{frenchlaw}
% Préambule
\begin{document}
% Corps du document
Ceci est un document \TeX.
Il est composé avec la classe \frenchlaw.
\end{document}
```

Les éléments colorés commençant par un backslash (`\`) sont appelés commandes, déclarations ou, de manière générique, macros.

Les éléments entre `{}` parenthèses courbes sont des arguments obligatoires. Dans les documentations, des `<>` chevrons sont utilisés pour indiquer ce que doivent contenir ces parenthèses courbes – selon le contexte, on précisera s’il s’agit d’un texte, d’un nombre, d’une valeur prédéfinie, etc. Dans `\chapter{titre du chapitre}`, par exemple, il faudra remplacer le texte entre chevrons par le titre choisi.

Les arguments placés entre `[]` crochets sont optionnels qui peuvent être omis, ainsi que les crochets. S’ils sont utilisés, les éléments entre `<>` chevrons doivent être remplacés, comme précédemment.


Les éléments en gris sont des `% commentaires`, qui dans cette documentation indiquent généralement la présence d’une explication ou bien la valeur par défaut donnée à une commande. Ces éléments ne sont jamais imprimés dans le document final.

Le fichier `.tex` commence par la déclaration `\documentclass{frenchlaw}`. On appelle *préambule* ce qui se trouve avant `\begin{document}` et *document* ce qui se trouve après celle-ci. Aucun texte ou commande n’est autorisé après `\end{document}`. Le contenu du préambule sera expliqué pour chaque type de document. On peut ajouter, dans le préambule, diverses redéfinitions ou personnalisations qui s’appliqueront uniquement au document concerné.


FrenchLAW est basée sur la classe `memoir` qui propose en outre de nombreuses fonctionnalités utiles pour les documents longs. Ces dernières ne seront pas documentées ici si elles ne présentent pas un intérêt immédiat ou n’ont pas fait l’objet de modifications spéciales. Il peut donc être utile de consulter la documentation, très complète, de la classe de `memoir`. La gestion des langues est assurée par le package `babel` et celle des bibliographies, par `biblatex` dont les documentations pourront aussi être utilement consultées avec profit, tant ces extensions sont complexes.

PARAMÈTRES DE COMPILATION POUR *FrenchLAW*

La classe *FrenchLAW* a été conçue pour automatiser un maximum d'éléments dans la production de documents juridiques. Afin que le document soit compilé correctement, il est nécessaire de donner les instructions suivantes *via* la ligne de commande ou l'interface graphique de l'éditeur utilisé. Dans tous les cas, remplacer `<filename>` par le nom du fichier à compiler ou, dans un éditeur, par la variable permettant d'y accéder. Pour la génération du document L^AT_EX, utiliser :


 `xelatex <filename>.tex`


Lorsque le document comprend des références bibliographiques, celles-ci doivent être compilées, après une première passe, avec :


 `biber <filename>.bcf`

Avec des bibliographies complexes, notamment lorsque certaines entrées contiennent un renvoi vers d'autres entrées (champs `related`, par exemple), il sera nécessaire de compiler à nouveau avec X_YL^AT_EX, puis avec Biber. Le log indique si cette manipulation est nécessaire.

Il est ensuite possible de générer automatiquement les index des matières (*rerum*), des auteurs (*nominum*) et des textes de lois (*locorum*) grâce aux commandes :

 `texindy -L french -M lang/french/utf8-lang -M frenchlaw
 <filename>.idx
texindy -L french -M lang/french/utf8-lang -M frenchlaw
 <filename>-bib.idx
texindy -L french -M lang/french/utf8-lang -M frenchlaw
 <filename>-law.idx`

 Dans l'éventualité où l'on voudrait utiliser des sous-numéros, lors de la rédaction d'un manuel par exemple, et sous réserve des limitations qui s'appliquent à ce procédé, on devrait compiler une nouvelle fois les index avec :

 `makeindex -s frenchlaw <filename>.idx`

Il sera enfin nécessaire de recompiler le document une ou plusieurs fois avec X_YL^AT_EX, jusqu'à ce que les tables des matières, les changements de page et les numéros des notes de bas de page se stabilisent. Cela est indiqué dans le log.

CHAPITRE PREMIER

INTRODUCTION GÉNÉRALE À L^AT_EX

La première question que pourra se poser le juriste est : Pourquoi L^AT_EX ? Ce logiciel, initialement créé pour produire des ouvrages de mathématiques, permet de générer des documents d'une très grande qualité typographique et d'automatiser un grand nombre de tâches chronophages (I).

L^AT_EX est un langage de programmation qui permet de dissocier le fond de la forme, c'est-à-dire de séparer le texte brut de la mise en page. Il permet en cela d'adopter un mode de rédaction sémantique consistant à décrire la fonction de chaque élément du texte plutôt que ce à quoi il doit ressembler (II).

Comme tout langage de programmation, L^AT_EX peut être modifié par son utilisateur qui peut, très facilement, créer de nouvelles commandes ou modifier les existantes afin d'adapter son environnement de rédaction à la sémantique recherchée (III).

Il est possible de diviser les gros projets (ouvrages individuels, documents collectifs) en plusieurs fichiers qui seront rassemblés dans un seul document .pdf, la continuité des pages, des références bibliographiques et des index étant assurée (IV).

Le logiciel produit parfois des erreurs ou des avertissements lorsqu'il ne parvient pas à trouver de solution typographiquement correcte, mais parfois aussi lorsque l'utilisateur a introduit des erreurs de syntaxe dans le code, ou lorsque la classe *FrenchL^AT_EX* ou l'une de ses dépendances utilisées contient un bug. Ces erreurs sont signalées dans le log. L'utilisateur peut en résoudre la plupart en suivant des conseils simples (V).

SECTION I. POURQUOI L^AT_EX ?

SECTION II. PRINCIPES DE LA RÉDACTION SÉMANTIQUE

SECTION III. CRÉATION ET MODIFICATION DE COMMANDES

SECTION IV. STRUCTURE DES FICHIERS DANS LES PROJETS COMPLEXES

SECTION V. CORRECTION DES ERREURS APPARAISSANT DANS LE LOG

CHAPITRE II

OPTIONS DE LA CLASSE *FrenchLAW*

La classe *FrenchLAW* (`frenchlaw.cls`) a été conçue avec de nombreuses options qui permettent de paramétrer l'apparence générale et les fonctionnalités du document. La classe charge le package `xkeyval`, qui pourra être utilisé pour développer de nouvelles options. Pour sélectionner plusieurs options, il suffit de les séparer d'une virgule. En cas d'options mutuellement exclusives, seule la dernière sera prise en compte.

SECTION I. FORMAT DE LA PAGE

Les options qui suivent permettent de définir la taille du papier et l'apparence générale de la composition typographique. La plupart sont similaires dans leur principe à leur équivalent de la classe `memoir`, mais leurs effets ont été étendus.

`a4paper`, `a5paper`, `a6paper` par défaut : `a4paper`
Dimensions de la feuille de papier. Contrairement aux options éponymes de la classe `memoir`, la taille du texte, les marges et le bloc de texte sont redéfinis automatiquement par ces options de manière à optimiser la mise en pages pour toutes les tailles de papier. Pour s'adapter à la page, la police d'écriture par défaut sera de `12pt` avec `a4paper`, de `11pt` avec `a5paper` et de `10pt` avec `a6paper`.

{dév.} Il est possible de vérifier quelle option a été chargée et d'en modifier le comportement grâce aux opérateurs : `\if@stockaiv`, `\if@stockav` et `\if@stockavi`.

`oneside`, `twoside` par défaut : `twoside`
Préparer le document pour une impression en recto simple (marges identiques) ou, par défaut, en recto-verso (marges différentes le long de la reliure et sur le bord opposé).

{dév.} Il est possible de vérifier quelle option a été chargée et d'en modifier le comportement grâce aux opérateurs : `\if@oneside` et `\if@twoside`.

`onecolumn`, `twocolumn` par défaut : `onecolumn`
Composition de la page en une ou deux colonnes. Cette option est simplement transmise à la classe `memoir` qui la gère intégralement. La présentation en deux colonnes n'est pas recommandée avec les petites tailles de papier, pour des raisons d'esthétique. Au format A4, il est préférable d'utiliser une police de `10pt`. L'option `twocolumn` désactive automatiquement `marginal`.

{dév.} Il est possible de vérifier quelle option a été chargée et d'en modifier le comportement grâce aux opérateurs : `\if@onecolumn` et `\if@twocolumn`.

landscape par défaut : `false`

Cette option permet de composer la page au format paysage, également dit à l'italienne. Elle est simplement adaptée de la classe `memoir`.

{dév.} Il est possible de modifier le comportement de l'option grâce à l'opérateur `\if@landscape`.

marginalia par défaut : `false`

L'option `marginal` permet d'augmenter la taille de l'une des marges latérales : à droite pour les documents en mode `oneside` et en alternance en mode `twoside`. La marge est alors utilisée pour imprimer notamment les titres des subdivisions secondaires (`\paragraph`, `\subparagraph` et `\para`). Elle est désactivée par l'option `twocolumn` avec laquelle elle est incompatible.

{dév.} Il est possible de modifier le comportement de l'option grâce à l'opérateur `\if@marginalia`.

SECTION II. PARAMÈTRES TYPOGRAPHIQUES

Les options suivantes permettent de configurer la manière dont le texte sera composé : sa taille, sa forme et l'interligne à utiliser. Ces éléments sont essentiels pour la composition de la page et sont donc chargés au niveau des options de classe. Pour modifier la police d'écriture elle-même, voir .

10pt, 11pt, 12pt, 14pt par défaut : `12pt`

Taille de la police utilisée pour le corps du texte. Celle-ci est de `12pt` par défaut avec `a4paper`. Pour les autres tailles de papier, la police est de `11pt` avec `a5paper` et `10pt` avec `a6paper`.

{dév.} Il est possible de vérifier quelle option a été chargée et d'en modifier le comportement grâce aux quatre opérateurs suivants : `\if@tenpoints`, `\if@elevenpoints`, `\if@twelvepoints` et `\if@fourteenpoints`.

classic, modern par défaut : `classic`

modernsans L'option `classic` (par défaut) est utilisée pour composer un document respectant les canons de l'imprimerie traditionnelle : police d'écriture à empattements, paragraphes en retrait. L'option `modern` embrasse les standards contemporains adaptés à un usage mixte imprimé et numérique. Une police sans-serif est utilisée pour contraster la police à empattements, notamment dans les titres ; les paragraphes sont séparés par des sauts de ligne plutôt que par un retrait. L'option `modernsans` reprend les mêmes caractéristiques, mais elle utilise une police sans-serif pour la totalité du texte.

{dév.} Il est possible de modifier le comportement des deux options grâce à l'opérateur `\if@modern` ■

`serif`, `sans-serif` par défaut : `serif`
 Une police d'écriture avec empattements (`serif`) est utilisée pour les documents rédigés, alors qu'une police plus épurée (`sans-serif`) est employée pour la correspondance.

{dév.} Il est possible de modifier le comportement des deux options grâce à l'opérateur `\if@sansserif` ■

`linespacing`=`(simple, default, median, double)` par défaut : `default`
 L'option `linespacing` permet de d'ajuster l'interligne par rapport aux valeurs standard en redéfinissant la valeur de `\baselinestretch`. L'option permet de modifier l'interligne pour l'une des valeurs prédéfinies suivantes :

- Interligne par défaut (ratio de 1,375 ou 12pt/16,5pt)
- Interligne simple (espacement minimal sans chevauchements, ou 12pt/13,8pt)
- Interligne médian (une fois et demie le corps du texte, ou 12pt/18pt)
- Interligne double (soit l'équivalent d'un saut de ligne, ou 12pt/24pt)

{dév.} L'interligne est défini par la macro `\custom@baselinestretch{<value>}`. Pour chaque option, cette définition est contenue dans les macros suivantes : `\set@defaultspacing`, `\set@medianspacing`, `\set@simplespacing` et `\set@doublespacing`. On applique ensuite la valeur de `\custom@baselinestretch` correspondant à l'option sélectionnée. Il est possible de créer une nouvelle option `linespacing`=`(opt-name)` avec le code :

```
\newcommand{\set@<opt-name>spacing}{%
  \def\custom@baselinestretch{<value>}}
```

En outre, la composition du texte en interligne simple étant susceptible requérir des ajustements, l'opérateur conditionnel `\if@simplespacing` a également été créé.

SECTION III. FORMATS DE DOCUMENTS

La classe *FrenchLAW* prévoit différents formats de documents adaptés aux écrits universitaires, à la correspondance, etc. Ceux-ci chargent automatiquement les options nécessaires pour assurer la cohérence typographique du document et son adaptation optimale aux besoins de l'utilisateur. Il est donc, en principe, inutile d'utiliser les options ci-dessus lorsqu'on utilise un modèle, sauf pour en modifier le comportement. Auquel cas, les différentes options doivent être chargées après le modèle.

FrenchLAW est configurée par défaut pour des documents universitaires ou des rapports d'une certaine longueur, pouvant s'apparenter à des monographies. D'autres formats sont disponibles grâce aux options ci-dessous.

`article` par défaut : `false`
 Cette option est adaptée à la production de textes courts, comme les articles, les notes de

jurisprudence ou les dissertations. Elle charge l'option **oneside**, qui est souvent choisie pour ce type de document.

{dév.} Il est possible de modifier le comportement de l'option grâce à l'opérateur `\if@article`.

didactic, didactic+ par défaut : **false**

L'option **didactic** propose notamment des en-têtes et pages de garde, adaptés aux documents pédagogiques ; elle charge également les options **oneside** et **article**. La variante **didactic+** permet de générer un document adapté à la lecture depuis un pupitre, avec l'option **14pt**, est notamment adapté pour les cours magistraux ou les interventions lors de colloques.

{dév.} Il est possible d'en modifier le comportement grâce à l'opérateur `\if@didactic`.

letter par défaut : **false**

Cette option automatise la mise en page des lettres et documents de correspondance. Elle charge les options **article**, **oneside**, **11pt** et **sans-serif**. Pour accompagner un *curriculum vitae*, il convient de rajouter manuellement l'option **marginal**.

{dév.} Il est possible d'en modifier le comportement grâce à l'opérateur `\if@letter` attaché spécifiquement à cette option, ou à l'opérateur `\if@stationery` qui s'applique à également aux autres documents de correspondance.

vitae par défaut : **false**

Cette option automatise la production de *curriculum vitae*. Tout comme **letter**, elle charge les options **article**, **oneside**, **11pt** et **sans-serif**. Le CV est également composé avec l'option **marginal**, qui permet ici d'inscrire les dates dans la marge.

{dév.} Il est possible d'en modifier le comportement grâce à l'opérateur `\if@vitae` attaché spécifiquement à cette option, ou à l'opérateur `\if@stationery` qui s'applique à également aux autres documents de correspondance.

SECTION IV. FONCTIONNALITÉS ADDITIONNELLES

La classe *FrenchLAW* comporte un certain nombre d'options qui modifient la présentation du document ou certaines de ses fonctionnalités, quel que soit le modèle de base utilisé. Ces options peuvent donc être utilisées en toutes circonstances pour paramétrer la classe (bibliographies, index) ou pour réaliser plusieurs versions d'un même document (brouillons, versions abrégées ou augmentées, etc.).

draft, draft+, final par défaut : **final**

Lorsqu'un document est compilé en mode **draft**, les annotations de l'auteur et les informations de compilation apparaissent sur le .pdf pour faciliter la correction. Cette option a

généralement pour effet de désactiver certains packages. Elle modifie la présentation des pages.

L'option **draft+** produit un document dont le corps est identique à la version définitive. Elle ajoute uniquement des marqueurs dans les marges pour une ultime relecture.

L'option **final**, activée par défaut, produit un document dépourvu de toutes annotations.

{dév.} Il est possible de vérifier quelle option a été chargée et d'en modifier le comportement grâce aux opérateurs suivants : `\ifdraftdoc`, qui est repris à la classe **memoir** pour l'option **draft** et `\if@draft`, spécifique à la classe *FrenchLAW*, qui s'applique aux options **draft** et **draft+**.

short par défaut : **false**

Cette option permet de produire la version courte d'un document de longueur modulable en insérant les balises : `\iflong...\fi`. Sur l'utilisation de cette option, voir . *FrenchLAW* compose par défaut la version longue, qui n'est donc pas présentée comme une option séparée.

{dév.} Il est possible d'en modifier le comportement grâce à l'opérateur `\iflong`. Le choix, peu intuitif en apparence, de cette dénomination tient à la nécessité de pouvoir utiliser l'opérateur conditionnel en cours de document : il n'était pas souhaitable qu'il contienne le signe @, et il est beaucoup plus logique dans le corps du texte d'insérer `\iflong...\fi` pour marquer le texte qui n'apparaîtra que dans la forme longue (par défaut) et sera donc omis si l'option **short** est utilisée.

`color|colour colorall|colourall` par défaut : **false**

Ces options permettent de produire un document en couleurs. On notera que les graphies anglaise et américaine sont acceptées. L'option **colour** ne met en couleurs que les éléments essentiels, alors que **colourall** est plus généreuse. La configuration fine du comportement de ces deux options est expliquée *infra* en p. 17.

precis, **summary** par défaut : **false**

outline, Ces options permettent d'imprimer des éléments supplémentaires dans la table des matières, principalement dans le but de générer un document donnant une vision d'ensemble du manuscrit. Les numéros et les noms des paragraphes sont ajoutés à la table, ainsi que le texte placé dans des environnements spéciaux. Sur ces deux options, voir .

{dév.} Il est possible de vérifier quelle option a été chargée et d'en modifier le comportement grâce aux opérateurs : `\if@outline`, `\if@summary`.

biblatex-style=`(style-name)` par défaut : **frenchlaw**

Cette option permet de configurer le style utilisé par **biblatex** pour composer les bibliographies. Par défaut, le style associé à la classe *FrenchLAW* est utilisé, mais il est possible d'en choisir un autre pour s'adapter au style exigé par certains organismes ou éditeurs.

{dév.} Le nom du style sélectionné est passé dans la macro interne `\set@biblatexstyle`.

index=`<paragraphs, pages>` par défaut : `paragraphs`

Cette option permet de configurer la manière dont les index sont présentés, en choisissant s'ils renvoient aux numéros de paragraphes (par défaut, comme il est d'usage en droit) ou aux numéros de pages.

{dév.} La méthode d'indexation est sélectionnée au moyen des opérateurs `\if@indexparagraphs` et `\if@indexpages`. Ceux-ci sont activés au niveau de l'option par l'intermédiaire des macros `\set@indexparagraphs` et `\set@indexpages`. Le système est construit pour permettre la création de nouvelles méthodes d'indexation, pourvu qu'elles soient construites sur une syntaxe similaire (recours à `\csname`). Ainsi pour une option `<method>` avec le code suivant :

```
\newif\if@index<method>
\newcommand{\set@index<method>}{\@index<method>true}
```

SECTION V. FONCTIONNALITÉS DÉVELOPPEUR

La classe *FrenchLAW* contenant sa propre documentation, elle dispose également de quelques options destinées à composer celle-ci. Ces dernières ne seront probablement d'aucune utilité aux utilisateurs, aux pour documenter un package modifiant cette classe.

developer par défaut : `false`

L'option **developer** est employée pour rédiger cette documentation. Elle ne devrait en principe pas être utile pour la rédaction juridique.

{dév.} Il est possible d'en modifier le comportement grâce à l'opérateur `\if@dev`.

vgrid par défaut : `false`

L'option **vgrid** permet d'imprimer une grille typographique pour vérifier l'alignement vertical des lignes sur la page. Elle a servi à programmer la classe *FrenchLAW*.

{dév.} Il est possible d'en modifier le comportement grâce à l'opérateur `\if@grid`.

SECTION VI. SÉLECTION DES LANGUES

british, english, french par défaut : `french`

Les langues **french**, **british** et **english** (respectivement anglais britannique et américain) sont chargées automatiquement par *FrenchLAW*. L'option permet de désigner la langue principale du document parmi celles-ci, pour le package **babel** et tous les autres packages qui en dépendent.

{dév.} Il est possible de vérifier quelle langue principale a été chargée et de modifier certains comportements en fonction de celle-ci grâce aux opérateurs suivants : `\if@french`, `\if@british` et `\if@english`.

`german`, `italian`, `spanish`

Plusieurs langues peuvent être chargées en supplément, pour être utilisées comme langues secondaires. Seules les langues qui sont mentionnées ici sont prises en charge à ce jour. Il est à noter que les langues `greek` et `latin`, dont l'emploi est fréquent en droit, sont chargées automatiquement et n'ont pas besoin de l'être à nouveau.

SECTION VII. DÉPENDANCES

La classe *FrenchL^AT_EX* requiert le format L^AT_EX_{2 ϵ} . Elle est basée sur la classe `memoir` qu'elle charge juste après lui avoir transmis toutes les options qui en relèvent et désactivé les options incompatibles entre elles. Il est donc possible d'ajouter n'importe quelle option reconnue par `memoir` directement dans les options de classe.

Les dépendances principales de la classe *FrenchL^AT_EX*, à l'exclusion de celles qui sont utilisées avec l'option `developer`, sont les suivantes :

<code>memoir</code>	Classe versatile pour les humanités et sciences sociales.
<code>babel</code>	Gestion des langues.
<code>biblatex</code>	Gestion des bibliographies.
<code>booktabs</code>	Gestion des tableaux.
<code>csquotes</code>	Gestion des citations.
<code>environ</code>	Stockage du contenu d'un environnement dans une macro.
<code>etoolbox</code>	Outils de programmation.
<code>fixltxhyph</code>	Corrige un bug affectant la césure de certains mots.
<code>fntcount</code>	Manipulation des nombres.
<code>fontenc</code>	Gestion des types d'encodage de polices par L ^A T _E X.
<code>fontspec</code>	Gestion des polices d'écriture par X _L L ^A T _E X.
<code>graphicx</code>	Inclusion de graphiques.
<code>ifxetex</code>	Tester le moteur utilisé.
<code>ifthen</code>	Tests conditionnels.
<code>inputenc</code>	Gestion des types d'encodage de documents par L ^A T _E X.
<code>marginnote</code>	Jeu de notes marginales flottantes.
<code>metalogo</code>	Personnalisation des logos de la famille T _E X.
<code>microtype</code>	Raffinements typographiques.
<code>multicol</code>	Gestion d'environnements en deux colonnes.
<code>numprint</code>	Formatage des nombres cardinaux.
<code>perpage</code>	Numérotation des notes de bas de page.
<code>ragged2e</code>	Meilleure composition du texte non justifié.
<code>xcolor</code>	Gestion des couleurs.

Options de la classe French^{AW}

<code>xkeyval</code>	Gestion d'options sous la forme $\langle\text{key}\rangle=\langle\text{value}\rangle$.
<code>xspace</code>	Utilisation d'espaces contextuelles dans les macros.
<code>xstring</code>	Manipulation de chaînes de caractères.
<code>xurl</code>	Génération d'adresses URL.
<code>adforn</code>	Police d'écriture pour les fleurons.
<code>ebgaramond</code>	Police d'écriture serif sous L ^A T _E X.
<code>lmodern</code>	Police d'écriture à chasse fixe sous L ^A T _E X.
<code>mathpazo</code>	Police d'écriture pour les mathématiques.
<code>pifont</code>	Police d'écriture pour les symboles.
<code>sourcesanspro</code>	Police d'écriture sans-serif sous L ^A T _E X.

CHAPITRE III

COMPOSITION TYPOGRAPHIQUE

La classe *FrenchLAW* est configurée pour optimiser le confort de lecture, notamment dans le cas des documents A4. Pour cela, elle s’inspire des techniques traditionnelles de conception des pages et implémente, dans la mesure du possible, une composition sur grille verticale. Les différents choix typographiques sont exposés dans ce chapitre, mais ne sont pas destinés à être modifiés, sauf par des mains expertes.

SECTION I. CONSTRUCTION DES PAGES

La page est composée de marges et d’un bloc typographique, sur lequel est composé le corps du texte. Suivant les règles de conception héritée de l’époque médiévale, *FrenchLAW* donne à ce bloc de texte un ratio hauteur / largeur équivalent au nombre d’or, soit 1,618. Pour aérer la page sans en réduire trop le volume, les documents composés avec les options *letter* et *vitae* utilisent un bloc de texte un peu plus fin, qui n’a donc pas exactement les mêmes proportions. Le ratio est alors de 1,715.

La largeur du texte a, elle-même, été choisie de manière à limiter la taille des lignes de manière à ce que chacune puissent accueillir environ 90 caractères (pour une page A4), afin de faciliter la lecture. Ce nombre devrait être plus réduit – autour de 80 caractères —, mais une telle exigence est peu compatible avec la composition sur ce format de papier.

Les marges, quant à elles, sont asymétriques, dans le respect de la tradition médiévale. Pour tous les documents recto-verso (*twoside*), la marge extérieure est de 33 % plus grande que la marge située près de la reliure. Le ratio est volontairement réduit par rapport au canon classique afin de ne pas choquer les lecteurs contemporains, mais permet d’aérer la lecture et d’insérer des annotations imprimées ou manuscrites. La proportion est accentuée en mode *marginal*, dans lequel la marge extérieure (*twoside*) ou la marge de gauche (*oneside*) est deux fois plus grande que l’autre. Les documents en recto simple (*oneside*) conservent des marges symétriques, comme le commande leur usage.

Les différentes options permettant de modifier les marges n’ont aucune incidence sur la largeur ou la hauteur du bloc typographique, garantissant ainsi que les modifications apportées à la mise en pages ne modifient pas le nombre de pages ni la répartition du texte sur celles-ci, à moins que l’on opte pour les options *letter* et *vitae*.

Pour les volumes comportant un grand nombre de pages, les imprimeurs de qualité ajoutent une petite marge supplémentaire dont les dimensions sont fonction du nombre de pages, de l’épaisseur du papier et du type de reliure envisagé. Cette marge de reliure (*binding correction*) peut anticipée sous L^AT_EX. La documentation de la classe *memoir* explique comment modifier la construction de la page, et mentionne à cet effet les reliures.

SECTION II. CHOIX DES POLICES D'ÉCRITURE

Par défaut, la classe *FrenchLAW* utilise les polices d'écriture suivantes, qui sont toutes disponibles dans T_EXlive :

SERIF : EB Garamond (L^AT_EX) et Junicode (X_YL^AT_EX).

SANS-SERIF : Source Sans Pro.

CHASSE-FIXE : Latin Modern Mono.

La police d'écriture ADF Ornaments ([adfor](#)n) est également employée pour les pictogrammes, aux côtés d'une variante de Palatino ([mathpazo](#)) pour les expressions mathématiques. Ces polices d'écriture peuvent être utilisées librement.

Les polices d'écriture sont gérées de manière très différente selon le moteur utilisé. L^AT_EX impose d'utiliser des polices préparées spécialement sous un format spécial. En pratique, cela signifie que l'on est très limité dans le choix des polices, qui sont alors chargées comme des packages. L'utilisation de X_YL^AT_EX est très recommandée pour les documents de langue française, car ce moteur assure une bien meilleure gestion des caractères accentués, des textes multilingues et des polices d'écriture. Il est ainsi possible, avec ce moteur, de choisir n'importe quelle police installée sur l'ordinateur ou dans le même dossier que le document .tex, à condition de tâtonner un peu pour deviner sous quel nom l'invoquer. La documentation du package [fontspec](#) fournit toutes les informations utiles à cet égard. Des exemples sont également donnés dans le fichier `frenchlaw.cnf`, qui permet à l'utilisateur de modifier les paramètres de la classe.

Sans entrer dans la (longue et délicate) question du choix d'une police d'écriture, on notera quelques points essentiels. Sous X_YL^AT_EX, il est recommandé d'installer une police *OpenType* (OTF), car ce format permet d'incorporer dans un seul fichier un grand nombre d'options et de subtilités typographiques, comme des formes alternatives pour certains caractères, des chiffres alignés sur les caractères bas-de-casse (12345) ou de vraies petites-capitales. Le package [fontspec](#) gère parfaitement ces options qui peuvent être activées ou désactivées en cours de document.

Par ailleurs, dans le choix d'une police d'écriture, on pourra dire sans trop exagérer qu'il faut impérativement éviter les polices disponibles par défaut sur l'ordinateur, qui sont soit produites par les marchands de d'ordinateurs, soit de pâles reproductions des dessins de grands typographes dont des versions de bien meilleure facture sont disponibles ailleurs. Il existe un certain nombre d'excellentes polices d'écriture disponibles gratuitement sous licence libre (OFL) – celles que nous avons choisies en font partie –, et un nombre plus important encore de polices d'écriture commerciales de grande qualité (compter au moins une centaine d'euros) produites par des typographes travaillant auprès de fonderies sérieuses. À titre d'exemple, le *Garamond* famélique et terne que l'on trouve sur tous la plupart des ordinateurs est très largement surpassé par d'autres Garamond, tant libres que commerciaux.

SECTION III. CHOIX D'UNE ÉCHELLE TYPOGRAPHIQUE

L'échelle typographique désigne un ensemble de tailles de police d'écriture, lesquelles seront ensuite utilisées dans les documents. On décide ainsi plus exactement de la proportion qu'il faut donner entre les plus petits caractères (pour les notes, par exemple) et les plus grands (dans les titres) pour que la progression soit harmonieuse.

En typographie classique, l'échelle est déterminée de manière pratiquement mathématique, en multipliant par deux la taille de police à intervalles réguliers. Cette méthode était paradoxalement surtout utilisée par des typographes qui composaient exclusivement en 10pt ou 12pt, à l'exception de quelques titres, parce qu'il fallait autrement posséder, en quantité suffisante, autant de jeux de caractères que de tailles de polices. En s'émancipant de la règle d'or de la construction des échelles typographiques, la classe *FrenchLAW* cherche tout simplement à obtenir une certaine harmonie visuelle dans l'enchaînement des subdivisions que l'on rencontre en droit.

FrenchLAW est configurée pour composer des textes de 10pt, 11pt, 12pt ou 14pt selon l'option choisie. À cet effet, la classe définit quatre échelles typographiques différentes et l'interlignage correspondant, selon le corps de texte principal. Ces valeurs sont résumées dans le tableau ci-après (TABLEAU 1).

SECTION IV. COMPOSITION SUR LA GRILLE TYPOGRAPHIQUE

La composition typographique traditionnelle est réalisée par alignement sur une grille verticale qui correspondait aux lignes de caractères en plomb. D'une page à l'autre, les lignes commencent exactement au même endroit, de sorte que si l'on regarde les deux côtés de la

Option de classe :	10pt	11pt	12pt	14pt
<code>\miniscule</code>	6 / 8	7 / 8	7 / 8.5	8 / 10
<code>\tiny</code>	7 / 9	8 / 9	8 / 9.5	9 / 11.5
<code>\scriptsize</code>	8 / 12	9 / 13.6	9 / 14.5	10 / 17.5
<code>\footnotesize</code>	8 / 12	9 / 13.6	10 / 14.5	11 / 17.5
<code>\small</code>	9 / 12	10 / 13.6	11 / 14.5	12 / 17.5
<code>\normalsize</code>	10 / 12	11 / 13.6	12 / 14.5	14 / 17.5
<code>\large</code>	12 / 15	13 / 16	14 / 18	17 / 21.5
<code>\Large</code>	15 / 18	16 / 19	17 / 22	20 / 25
<code>\LARGE</code>	18 / 22	19 / 23	20 / 25	25 / 30.5
<code>\huge</code>	22 / 27	23 / 28	25 / 30	30 / 37
<code>\Huge</code>	27 / 32	28 / 34	30 / 37	36 / 45
<code>\HUGE</code>	32 / 37	34 / 42	36 / 45	48 / 60

TABLEAU 1 – Corps de texte et interlignage, exprimés en pt.

feuille par transparence, les lignes du recto et du verso se confondent. Cela permet d'éviter l'effet très désagréable, surtout sur des papiers très fins, consistant à voir transparaître le gris de la page suivante.

Les logiciels de la famille \TeX ont été pensés pour l'écriture mathématique qui suppose des équations, des schémas ou des figures qui rendent assez inharmonieuse la composition classique « sur la grille ». Il fallait, dans ce contexte, permettre un peu de flexibilité dans l'espacement vertical des éléments de la page, et même des lignes du texte, pour ne pas créer trop de blancs. La plupart des moteurs conservent cette particularité.

Une solution aussi radicale n'est toutefois pas requise pour le droit, et *FrenchL^AT_EX* essaie de trouver un équilibre raisonnable entre respecter *pour l'essentiel* le système traditionnel de composition sur la grille, et autoriser des entorses à cette règle chaque fois qu'elle produirait des résultats franchement sous-optimaux. Les éléments qui rompent l'alignement vertical sont, par ailleurs et chaque fois que cela est possible, placés en bas de page pour minimiser la gêne visuelle que cela pourrait causer au lecteur. Ce prélude permet d'expliquer les choix techniques qui ont été faits dans le code.

En plus d'établir une échelle typographique, la classe garantit autant que faire se peut le respect d'une grille verticale, ce qui signifie que l'espacement des lignes doit rester le même – ou un multiple de l'espacement standard – peu importe la taille des caractères. Cela peut se vérifier avec l'option `vgrid`. Toutefois, dans le souci de trouver un équilibre harmonieux, il est également nécessaire de pouvoir violer ce principe : si l'on pense à un titre de chapitre s'étalant sur deux lignes, il sera souvent très laid d'espacer excessivement ces deux lignes dans le seul but de retomber sur la grille verticale.

Dans l'échelle établie ci-dessus, les titres des subdivisions, comme les chapitres ou les sections, seront composés dans un corps plus gros, ce qui rompra l'alignement vertical sur la grille : la hauteur des caractères et de leur interlignage n'est pas un multiple de la hauteur des lignes composées en corps normal. *FrenchL^AT_EX* s'applique ensuite à atténuer cet effet. La solution qui a été retenue ici consiste à associer à chaque taille de caractères un interligne confortable et agréable à lire, tout en prévoyant deux mécanismes pour forcer l'alignement sur la grille, avec plus ou moins de subtilité.

Le premier mécanisme utilisé consiste à ajouter un espace vertical qui affecte seulement la première ligne de texte dans un corps donné. Pour les titres de chapitres ou de sections, par exemple, cela est en principe suffisant – pourvu, du moins, que les auteurs sachent rester brefs. Comme seule la première ligne est affectée, un titre trop long restera agréable à lire, mais rompra l'alignement vertical sur la grille, ce qui est sans doute un moindre délit typographique.

Cela est réalisé en stockant dans une macro l'ajustement nécessaire à chaque corps de texte pour retomber sur la grille : ce dernier pourra ensuite être ajouté au compte-gouttes, comme un espace vertical, de manière à n'affecter que la première ligne du texte composé dans ce corps.

`{dév.}` Les dimensions internes ainsi créées, et dont la valeur dépend du corps de texte choisi en option de classe, peuvent être utilisées par les développeurs :

`\addto@miniscule` pour le corps `\miniscule`;

`\addto@tiny` pour le corps `\tiny`;

`\addto@large` pour le corps `\large` ;
`\addto@Large` pour le corps `\Large` ;
`\addto@LARGE` pour le corps `\LARGE` ;
`\addto@huge` pour le corps `\huge` ;
`\addto@Huge` pour le corps `\Huge` ;
`\addto@HUGE` pour le corps `\HUGE`.

On notera que les corps `\scriptsize`, `\footnotesize` et `\small` ne sont pas concernés par ces ajustements : leur interligne est calqué sur celui du corps standard, de manière à ce que les légendes ou les notes de bas de page ne rompent pas systématiquement l'alignement vertical. Cela est d'autant moins gênant que ce sont des tailles de texte assez proches du corps normal. On remarquera toutefois que les notes de bas de page ont ainsi un aspect plus aéré que le corps du texte, ce qui en facilite la lecture.

Avec ce système, la grille d'alignement vertical est rompue : 1° lorsque le texte d'un titre de subdivision hiérarchique s'étend sur plusieurs lignes ; 2° lorsque deux divisions se suivent sans texte intermédiaire, ce qui se produit par exemple lorsqu'un `\chapter` est immédiatement suivi d'une nouvelle `\section`. La rupture de la grille verticale permet toutefois d'éviter le résultat disgracieux que l'on obtiendrait en imposant à toutes les tailles de texte le même interlignage.

Le second mécanisme est volontairement dépourvu de toute subtilité : il impose un strict respect de la grille typographique, quel que soit le nombre de lignes. Il consiste simplement à donner à toutes les tailles de texte un interlignage multiple de celui retenu pour le corps principal. Son utilisation est, en principe, réservée aux développeurs.

{dév.} Ce mécanisme a été créé dans l'idée qu'il pourrait être employé localement, à l'intérieur d'un groupe (`\begingroup ... \endgroup` ou `{...}`) au sein duquel il est important de préserver la grille verticale, par exemple pour une page de garde. Pour cette raison, il utilise volontairement la commande interne `\normalize@fontsize`, qui s'utilise comme suit :

☞ `{\normalize@fontsize Texte à aligner sur la grille}% ou :`
`\bgroup\normalize@fontsize Texte à aligner sur la grille\egroup`

La composition sur une grille typographique interdit à L^AT_EX d'insérer des espaces verticaux flexibles (*glue*) pour adapter le contenu à la taille de la page. La commande `\raggedbottom` est donc utilisée dans la classe *FrenchL^AW* pour éviter les messages d'erreur liés à l'existence de pages incomplètement remplies (`Underfull \vbox`). Ce système donne de bons résultats, en particulier si l'on insère des notes de bas de page (`\footnote`) qui sont, quant à elles, propulsées en bas de page et comblent les vides résiduels.

SECTION V. COMPOSITION DES PARAGRAPHES

L^AT_EX permet de limiter ou d'interdire la survenance de certaines imperfections typographiques grâce à un système de pénalités. *FrenchL^AW* a été paramétrée pour donner de bons résultats sans générer trop d'avertissements. Il s'agit, en particulier, de pénaliser les

lignes veuves et orphelines, esseulées respectivement en début ou en fin de page, ainsi que les coupures de mots sur des lignes consécutives ou en fin de page. Pour modifier les paramètres retenus, les utilisateurs pourront se référer à la documentation générale de L^AT_EX.

En principe, le texte est composé en alinéas, de sorte qu’aucun espacement vertical n’est ajouté entre les paragraphes. Par exception, pour tous les documents de correspondance (options **letter** et **vitae**) et ceux rédigés selon les codes modernes, notamment pour écran (option **modern**), les paragraphes sont séparés par un saut de ligne plutôt que par un alinéa.

Il sera parfois utile de modifier ces paramètres directement dans le préambule en modifiant les longueurs de `\parskip` et de `\parindent` :

```
% Pour un texte en alinéas :
\setlength{\parskip}{0pt}% Pas d'espace entre les paragraphes
\setlength{\parindent}{1.5em}% Première ligne en alinéa
% Pour un texte avec des sauts de lignes :
\setlength{\parskip}{\baselineskip}% Espace entre les paragraphes
\setlength{\parindent}{0em}% Pas de mise en alinéa
```

Enfin, en typographie soignée, on évite de placer à la fin d’un paragraphe une ligne comportant seulement quelques lettres ou, inversement, une ligne occupant toute la largeur du texte, susceptible d’entraîner une confusion entre les paragraphes. Pour cela, *FrenchL^AW* impose à L^AT_EX de ne pas terminer un paragraphe par un ou des mots mesurant moins de deux fois la largeur de l’alinéa (ce qui évite les lignes trop courtes), et de toujours laisser un espace mesurant environ deux caractères libres en fin de paragraphe (ce qui évite de confondre les paragraphes).

{dév.} Il est possible de modifier ce comportement en redéfinissant les dimensions minimale et maximale de la glue `\parfillskip`, qui est paramétrée comme suit :

```
\setlength{\parfillskip}{2ex plus \dimexpr\linewidth -3em}
```

SECTION VI. COULEURS

La classe *FrenchL^AW* permet de colorer automatiquement certains éléments du document grâce à l’option **colour**. Celle-ci fonctionne, en principe, avec des jeux de couleurs prédéfinis, mais l’utilisateur peut, d’une part, choisir ses propres couleurs pour les appliquer et, d’autre part, modifier la couleur de tel ou tel élément en particulier.

{dév.} Un jeu de couleurs peut être créé avec la commande `\setcolourscheme`, qui permet d’associer un nom à deux couleurs : la première sera considérée comme principale ; l’autre est une couleur de contraste, en principe plus claire.

```
\setcolourscheme{<name>}{<hex colour code>}{<hex colour code>}
```

FrenchLAW définit deux jeux de couleurs : *red*, basé sur *Crimson Red*, et *blue*, qui utilise *Steel Blue*. La commande `\setcolours{<name>}` a pour effet de définir, selon les préférences de l'utilisateur, la couleur principale du document (`\maincolour`), ainsi que la couleur secondaire de celui-ci (`\subcolour`) :

`\setcolours{red}` donne : `\maincolour` `\subcolour`
`\setcolours{blue}` donne : `\maincolour` `\subcolour`

Les différents éléments susceptibles d'être mis en couleurs par la classe sans autre forme de modification de la part de l'utilisateur sont indiqués dans la suite de la documentation. Pour mémoire, toutefois, une liste de toutes les commandes permettant de configurer la couleur de ces éléments est présentée dans le TABLEAU 2). De manière générale – et ce principe connaît des variations dans les différents styles proposés – l'option *colour*, qui correspond en réalité à *colour=some*, se contente de mettre en couleurs quelques éléments peu susceptibles de gêner la lecture, dans le seul but d'attirer légèrement le regard vers tel ou tel élément, à la manière d'une discrète enluminure. L'option *colour=all* permet de composer en couleurs un plus grand nombre d'éléments du texte, généralement en conjonction avec l'option *modern*. L'imprimerie traditionnelle utilise, en effet, très peu les couleurs – plus chères – qui se sont démocratisées dans les ouvrages en même temps que leur usage sur les écrans. Ces choix en sont la traduction.

<code>\bookcolour</code>	<code>\partcolour</code>	<code>\headingcolour</code>
<code>\subheadingcolour</code>	<code>\chaptercolour</code>	<code>\sectioncolour</code>
<code>\subsectioncolour</code>	<code>\subsubsectioncolour</code>	<code>\paragraphcolour</code>
<code>\subparagraphcolour</code>	<code>\paracolour</code>	<code>\paranumcolour</code>
<code>\subparacolour</code>	<code>\subparanumcolour</code>	<code>\marginaliacolour</code>
<code>\vitaecolour</code>		

TABLEAU 2 – Macros utilisées pour la mise en couleurs.

PREMIÈRE PARTIE

STRUCTURE DU DOCUMENT

CHAPITRE PREMIER

DIVISIONS DU DOCUMENT

Une fois passé le préambule, il convient de structurer le document au moyen de subdivisions hiérarchiques (parties, chapitres, sections, etc.) ou tout simplement logiques (pages liminaires, numéros de paragraphes, annexes). \LaTeX permet d’automatiser les deux processus en utilisant des compteurs (*counters*) incrémentés à chaque changement de subdivision, ce qui permet d’obtenir une numérotation continue sans effort. On distinguera les divisions hiérarchiques classiques des divisions seulement logiques, souvent propres aux juristes.

SECTION I. DIVISIONS HIÉRARCHIQUES

La division du document en chapitres, sections, paragraphes, et autres subdivisions institue une hiérarchie entre divers éléments ; la section se trouve ainsi dans le chapitre, qui se trouve dans le titre, lui-même situé dans la partie.

Ces subdivisions hiérarchiques existent par défaut sous \LaTeX , mais sont en nombre insuffisant pour les juristes qui les multiplient volontiers. *FrenchLAW* commence par créer toutes les commandes nécessaires pour utiliser des titres et sous-titres dans le document et définir, par la même occasion, leur positionnement dans la hiérarchie des titres – étape nécessaire en vue de la création ultérieure de tables des matières.

La classe *FrenchLAW* définit ensuite le style de chaque niveau de subdivision en utilisant les méthodes de la classe `memoir`. Le style `frenchlaw-classic` est créé et appliqué par défaut. Il est adapté à tous les documents composés dans un style classique, avec des polices d’écriture à empattements, comme les thèses et certaines monographies classiques. Un autre style de chapitre, dénommé `frenchlaw-modern` est utilisé avec l’option `modern` ; il utilise une police sans-serif pour les titres. La plupart des niveaux de subdivision sont composés en petites ou grandes capitales, en jouant sur le contraste entre les deux casses pour un même corps.

La composition des nombreuses subdivisions correspondant aux sections, sous-sections et paragraphes ne dépend pas du système de styles, mais exclusivement des options choisies au début du document. Ces commandes, héritées des classes standard, sont beaucoup plus faciles à modifier par l’utilisateur. Dans la configuration proposée par *FrenchLAW*, on notera en particulier de subtiles différences dans l’utilisation des petites-capitales entre les documents classiques (avec empattements), qui conservent la casse des intitulés de sections, et l’option `modern` qui compose les titres de sections entièrement en petites-capitales. L’option `marginal`, quant à elle insère les titres de plusieurs niveaux de subdivision directement dans la marge, comme dans certains ouvrages anciens. L’option `vitae` bénéficie également d’une attention particulière : les sections y sont indiquées par un trait coloré placé dans la marge.

Il est possible de créer de nouveaux styles de chapitres (et de subdivisions en général) en utilisant les commandes de la classe `memoir`, à la documentation de laquelle le lecteur est renvoyé.

Les subdivisions disponibles dans la classe `FrenchLAW` sont : `\book`, `\part`, `\heading`, `\subheading`, `\chapter`, `\section`, `\subsection`, `\subsubsection`, `\paragraph`, `\subparagraph` ■

Les commandes correspondantes peuvent être employées comme suit :

```
\book{<title>}% Livre
\part{<title>}% Partie
\heading[short-title]{title}% Titre
\subheading[short-title]{title}% Sous-titre
\chapter[toc-title][head-title]{title}% Chapitre
\section[toc-title][head-title]{title}% Section
\subsection[toc-title][head-title]{title}% Sous-section
\subsubsection[toc-title][head-title]{title}% Sous-sous-section
\paragraph[toc-title][head-title]{title}% Paragraphe
\subparagraph[toc-title][head-title]{title}% Sous-paragraphe
```

L'argument obligatoire `{<title>}` contient le titre de la subdivision dont il s'agit. Le premier argument optionnel `[<toc-title>]` permet d'indiquer un titre différent – par exemple, plus court – à utiliser dans la table des matières. Quant au second argument optionnel, `[<head-title>]`, il permet s'il y a lieu, de préciser un titre différent à utiliser dans l'en-tête de la page. Pour les commandes `\heading` et `\subheading`, l'argument `[<short-title>]` est utilisé à la fois dans la table des matières et dans les en-têtes.

Certaines subdivisions existent également sous une forme étoilée, qui n'est alors pas numérotée ni (en principe) intégrée automatiquement à la table des matières. La classe `FrenchLAW` modifie toutefois ce comportement pour ajouter automatiquement à la table des matières les chapitres créés avec `\chapter*`, afin d'épargner à l'utilisateur la nécessité de réaliser manuellement cet ajout à chaque fois. Les commandes étoilées sont les suivantes :

```
\chapter*[<short-title>]{<title>}% Chapitre
\section*{<title>}% Section
\subsection*{<title>}% Sous-section
\subsubsection*{<title>}% Sous-sous-section
\paragraph*{<title>}% Paragraphe
\subparagraph*{<title>}% Sous-paragraphe
```

`FrenchLAW` crée, en outre, la commande `\fakechapter`, qui mérite une explication. Elle permet de créer un chapitre non numéroté, mais elle comporte deux particularités : 1) il est possible d'insérer un texte en lieu et place du numéro de chapitre (par exemple « Chapitre liminaire » ou « Avant-propos », suivis d'un titre plus substantiel) ; 2) ce chapitre est renseigné dans la table des matières comme s'il s'agissait d'une section. Cette commande est particulièrement utile pour les chapitres non numérotés que l'on trouve en tout début ou en fin d'ouvrage, par exemple une préface, une liste d'abréviations, une bibliographie ou

un index. Lorsque plusieurs chapitres non numérotés se suivent – ce qui est fréquent –, la commande `\fakechapter` produit un résultat visuellement plus harmonieux dans la table des matières.

🔗 `\fakechapter[⟨short-title⟩]{⟨title⟩}`

Une fois insérées dans le document, les différentes subdivisions hiérarchiques sont numérotées différemment, selon que l'on se trouve dans un document long ou que l'on utilise l'option `article`. Le tableau suivant indique le niveau de chaque subdivision dans la table des matières, qui pourra être utile aux développeurs, ainsi que son mode de numérotation en fonction de l'option sélectionnée (TABLEAU 3).

{dév.} D'une manière générale, le passage au chapitre suivant réinitialise la numérotation de toutes les subdivisions inférieures au chapitre, et ainsi de suite pour toutes les subdivisions hiérarchiques. La classe `memoir` permet de paramétrer facilement cela grâce aux commandes `\counterwithin` et `\counterwithout` repris à la classe `memoir`.

La classe `memoir` permet, par ailleurs, grâce à la commande `\maxsecnumdepth` de choisir jusqu'à quel niveau les subdivisions du document doivent porter une numérotation apparente sans avoir à modifier le reste du code. Ce paramètre est configuré différemment selon les options sélectionnées, et s'utilise comme suit :

🔗 `\maxsecnumdepth{⟨counter⟩}` % Par défaut: subparagraph

Les subdivisions hiérarchiques seront composées en couleurs avec la combinaison des options `modern` et `colourall`. Certains styles peuvent comporter des variations plus complexes. Il est possible de modifier la couleur de certaines subdivisions seulement – par

Subdivision	Niv.	Monographies, thèses	Articles
<code>\book</code>	-2	Livre premier, Livre II, etc.	
<code>\part</code>	-1	Première partie, Deuxième partie, etc.	
<code>\heading</code>	-1	Titre premier, Titre II, etc.	
<code>\subheading</code>	-1	Sous-titre premier, Sous-titre II, etc.	
<code>\chapter</code>	0	Chapitre premier, Chapitre II, etc.	
<code>\section</code>	1	Section I, Section II, etc.	I, II, etc.
<code>\subsection</code>	2	§ I, § II, etc.	A, B, etc.
<code>\subsubsection</code>	3	A, B, etc.	1, 2, etc.
<code>\paragraph</code>	4	1, 2, etc.	a, b, etc.
<code>\subparagraph</code>	5	a, b, etc.	i, ii, etc.

TABLEAU 3 – Numérotation des différentes subdivisions.

exemple, si l'on veut repasser en noir en-deçà d'un certain niveau – grâce aux commandes `\Xcolour`, où X peut être remplacé par le nom de la subdivision en question. On pourra ainsi redéfinir, par exemple, la commande pour les chapitres, grâce au code suivant :

```
❧ \renewcommand{\chaptercolour}{\color{black}} % Chapitres toujours en noir.
```

Avec l'option `marginalia`, qui crée de grandes marges pour y insérer les titres des paragraphes, le trait placé sous le titre – qui ne risque pas de gêner la lecture – sera mis en couleurs simplement avec l'option `colour`. Ce comportement peut également être modifié en redéfinissant la commande `\marginaliacolour`.

SECTION II. DIVISIONS LOGIQUES

Outre les divisions hiérarchiques qui structurent le *texte*, il existe également – et particulièrement dans l'écriture juridique – des divisions logiques qui structurent le *document* en son entier. Dans une monographie, on distinguera, par exemple, les pages liminaires qui sont souvent numérotées en chiffres romains, ou encore les sections finales dont les chapitres ne sont généralement pas numérotés. En droit, il est également d'usage de numéroté les paragraphes tout au long de l'ouvrage, pour structurer le passage d'une idée à une autre, sans pourtant créer de hiérarchie entre les différents numéros. La pratique est identique à celle consistant à numéroté les clauses d'un contrat ou les paragraphes d'une décision de justice.

§ I. STRUCTURE D'ENSEMBLE DE L'OUVRAGE

Quant aux divisions logiques que l'on trouve dans les monographies et les thèses, elles sont intégralement reprises à la classe `memoir` qui les définit comme suit :

```
❧ \frontmatter % Pages liminaires  
  \mainmatter % Corps de l'ouvrage  
  \appendix % Annexes de l'ouvrage  
  \backmatter % Pages terminales
```

La déclaration `\frontmatter` délimite la partie liminaire de l'ouvrage qui comporte généralement une préface ou un avant propos, un sommaire, une table d'abréviations, etc. Ces chapitres ou sections ne sont pas numérotés et les pages sont généralement numérotées en chiffres romains minuscules. Dans certains types d'ouvrages, comme les mémoires et les thèses, la numérotation doit être continue, ce qui peut être obtenu avec la version étoilée `\frontmatter*`. Dans tous les cas, cette déclaration est incluse juste après les pages de garde dans tous les types de documents configurés par *FrenchLAW*.

La déclaration `\mainmatter` met fin aux dérogations instaurées pour les pages liminaires ; les subdivisions hiérarchiques sont désormais numérotées normalement et la numérotation des pages, qui est remise à zéro, se fait en chiffres arabes. Pour les mémoires

et thèses, *FrenchL^AT_EX* fait en sorte que la numérotation des pages ne soit pas affectée par l'usage de cette commande. La déclaration est également incluse juste après le titre des documents autres que les monographies.

Les annexes, s'il y en a, doivent être intégrées au corps de l'ouvrage grâce à la déclaration `\appendix`, qui modifie la dénomination et la numérotation des chapitres, lesquels deviendront par exemple une « Annexe A ». Il est également possible de faire précéder ces annexes d'une page reprenant le style d'une partie (part) grâce à la commande `\appendixpage`.

Enfin, le `\backmatter` contient toutes les sections terminales de l'ouvrage, parmi lesquelles on compte, le plus souvent, les bibliographies, les index et la table des matières. Ces sections ne sont pas numérotées.

Dans l'optique de la rédaction d'une monographie ou d'une thèse, il faudra donc anticiper ces divisions logiques, puis insérer les divisions hiérarchiques au sein de cette structure au moment de la rédaction. Ces éléments pourront être approfondis en lisant la documentation de la classe `memoir`.

§ II. STRUCTURATION DU TEXTE PAR NUMÉROS

Dans l'écriture juridique, le propos est généralement structuré en numérotant les paragraphes afin de pouvoir s'y référer ensuite, mais aussi dans un style plus ancien, pour offrir au lecteur une table des matières augmentée d'un précis composé de tous ces numéros de paragraphes et de leurs intitulés.

Il n'est pas évident de trouver une dénomination adéquate à ces numéros : les lettres accentuées ne sont pas autorisées dans les noms de commandes (*exit* donc la dénomination évidente), l'abréviation `\no` est déjà utilisée dans des extensions vitales pour composer l'abréviation *n^o*, alors que `\paragraph` désigne, dans toutes les versions de L^AT_EX, un niveau de division hiérarchique. Les numéros « de paragraphes » utilisés par les juristes ne sont, à la vérité, pas des *alinéas* non plus, dans la mesure où un numéro peut contenir plusieurs paragraphes typographiques pourvu qu'ils participent d'une même idée ou démonstration.

Vu la fréquence et la quantité de numéros que l'on utilise, ne serait-ce que dans un article de doctrine, il fallait, en outre, une commande courte et facile à mémoriser. Le choix qui fut arrêté, à l'usage, est celui de la commande `\para`, qui renvoie adéquatement à l'idée de paragraphe coulé. La commande s'emploie comme suit :

☞ `\para % ou :`
`\para[<Intitulé du numéro>]`

La commande `\para`, utilisée en début de paragraphe, permet de passer au numéro suivant. La première fois que la commande est utilisée dans le document, elle imprimera donc le chiffre 1 et incrémentera le compteur `para`, de sorte que l'emploi suivant de la commande imprimera le chiffre 2, etc. Cela signifie donc que, si l'on ajoute un numéro en cours de rédaction, tous les numéros seront modifiés en conséquence. La commande peut recevoir un argument contenant l'intitulé du paragraphe. Il s'agit ici d'un argument *optionnel*, placé entre [crochets], parce que certains numéros, comme les annonces de plan, n'ont pas forcément besoin de titre. Il s'agit toutefois d'un argument qu'il est vivement recommandé

de remplir chaque fois que cela fait sens, car il permettra notamment de faire référence au bon paragraphe en dépit des évolutions de la numérotation.

Une seconde commande, dénommée `\subpara`, utilise la même syntaxe et permet de créer un sous-numéro de paragraphe. Si l'on se trouve dans le `\para` numéro 37, la première commande `\subpara` créera un 37-1 ; la seconde produira le 37-2, et ainsi de suite jusqu'à la prochaine commande `\para` qui remettra le compteur `subpara` à zéro.

🐼 `\subpara` % ou:
`\subpara[⟨Intitulé du numéro⟩]`

En principe, le titre du paragraphe coulé est composé à côté du numéro de celui-ci. Il est parfois souhaitable que ce ne soit pas le cas. La commande `\noparanames` permet de désactiver l'impression des titres des paragraphes, alors que `\paranames` permet de la réactiver.

La mise en forme des numéros de paragraphes est réalisée au moyen de styles prédéfinis, auxquels l'utilisateur pourra ajouter sa griffe. La commande `\makeparastyle` permet de définir un nouveau style de paragraphe, alors que la commande `\setparastyle`

Pour définir et sélectionner un nouveau style de numéros, il suffit de redéfinir, à l'intérieur de `\makeparastyle`, tout les paramètres que l'on souhaite modifier parmi la liste suivante. Les sous-numéros utilisent exactement la même syntaxe, à laquelle on ajoute seulement le préfixe sub (`...subpara...`). Dans l'exemple ci-dessous, les paramètres les plus susceptibles d'être modifiés sont indiqués par `⟨...⟩` :

🐼 `\makeparastyle{⟨nom du style⟩}{ % Création du style`
`% Pour la commande \para`
`% Mode de numérotation (ici, chiffres arabes):`
`\renewcommand{\thepara}{\arabic{para}}`
`% Ponctuation et espace après le titre associé au numéro:`
`\renewcommand{\parasep}{⟨...⟩}`
`% Ponctuation et espace après le numéro:`
`\renewcommand{\paranumsep}{⟨...⟩}`
`% Couleur du titre:`
`\renewcommand{\paracolour}{\if@colourall\maincolour\fi}`
`% Couleur du numéro:`
`\renewcommand{\paranumcolour}{\paracolour}`
`% Style du titre (ajouter par exemple \bfseries ou \scshape):`
`\renewcommand{\paratitlestyle}[1]{⟨...⟩#1}`
`% Style du numéro (idem):`
`\renewcommand{\paranumstyle}[1]{⟨...⟩#1}`
`}`

Pour sélectionner un style de paragraphe, il est nécessaire de l'indiquer au moyen de la commande `\makeparastyle`. Plusieurs styles sont déjà définis par *FrenchL^AT_EX*. Le style

default, chargé automatiquement, compose, sans originalité, les numéros et titres de paragraphes en petites capitales (**classic**) ou en gras (**modern**). Le style textbook compose les numéros de paragraphe en gros dans la marge.

☞ `\parastyle{<nom du style>} % Sélection du style`

Le style default, chargé automatiquement, compose très classiquement les numéros de paragraphe et sous-paragraphe en gras. Le titre coulé des paragraphes est composé, selon les autres options choisies, en petites-capitales si le corps du texte utilise une police avec serif (**classic**), en gras pour les compositions sans-serif (**modern**) ou en italiques dans les marges (**marginalia**).

Le style textbook utilise un design un peu plus élaboré, dans un but à la fois ludique et esthétique. Dans une inspiration stylisée des enluminures médiévales, il compose les numéros de paragraphes en couleur (avec les options par défaut, dans la marge et dans un corps bien supérieur à celui du texte. Ce style est aussi particulièrement adapté à l'utilisation de couleurs.

Les numéros de paragraphe peuvent être intégrés à la table des matières pour former un « précis ». Cette fonctionnalité pourra être utilisée pour la table des matières de certains ouvrages, mais aussi, par exemple, pendant la phase d'édition et de relecture d'un ouvrage, pour en percevoir plus clairement la structure. Elle peut être invoquée par l'option de classe `precis`. Les commandes `\precis` et `\noprecis` permettent également de produire le même résultat.

{dév.} Le compteur `precis` contient le numéro du paragraphe dans la subdivision courante.

Les commandes permettant de composer les numéros de paragraphes dans la table des matières peuvent également être insérées dans un style de paragraphe. Elles sont documentées dans le chapitre sur les tables et sommaires (??).

1. TEST. Chaque paragraphe et sous-paragraphe est automatiquement associé à un « label », qui permettra d'y faire référence ailleurs dans le texte. La commande `\label{<nom>}`, utilisée en interne, permet de créer une balise dans le texte et d'enregistrer à quel numéro de page ou dans quel paragraphe elle se trouve. La commande `\ref{<nom>}` permet à l'utilisateur d'imprimer le numéro de paragraphe sous lequel se trouve la balise, alors que `\pageref{<nom>}` renverra au numéro de page en question (sur ces questions, voir). Les labels associés aux paragraphes et sous-paragraphes ont toujours la forme `para:<titre du paragraphe>` ou, si aucun titre n'est donné, `para:<numéro du paragraphe>` (le numéro risque alors de changer si de nouveaux paragraphes sont ajoutées). La syntaxe pour renvoyer à un numéro de paragraphe est, classiquement, la suivante : (numéro : .1, 26)

% Paragraphe sans titre

`\para`

% Paragraphes avec un titre

`\para[Second paragraphe]`

`\subpara[Sous-paragraphe]`

Divisions du document

```
\para[Troisième paragraphe]
%%
% Référence au premier paragraphe (sans titre):
\ref{para:1}% Imprime: "1"
% Référence au troisième paragraphe (avec titre):
\ref{para:Troisième paragraphe}
% Référence au sous-paragraphe:
\ref{para:Sous-paragraphe}% Imprime: 2-1 (para 2, sous-para 1)
```

{dév.} Les commandes `\para` et `\subpara`, décrites plus haut, gèrent la composition des numéros et titres de (sous-)paragraphe, la création des labels correspondants, ainsi que l'insertion éventuelle dans la table des matières en mode `precis`.