

Une courte (?) introduction à $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$

par Tobias Oetiker

Hubert Partl, Irene Hyna et Elisabeth Schlegl

traduit en français par Matthieu Herrb

adaptation et compléments par Daniel Flipo

Copyright © 2000-2003 Tobias Oetiker et tous les contributeurs de LShort.

Copyright © 2000-2003 LAAS/CNRS pour la traduction.

Ce document est libre ; vous pouvez le redistribuer ou le modifier selon les termes de la licence publique générale de GNU publiée par la *Free Software Foundation* (version 2 ou tout autre version ultérieure choisie par vous)

Ce document est diffusé en espérant qu'il sera utile, mais *sans aucune garantie*, ni explicite ni implicite, y compris les garanties de commercialisation ou d'adaptation dans un but spécifique. Reportez-vous à la licence publique générale de GNU pour plus de détails.

Copyright © 1998 Tobias Oetiker and all the Contributors to LShort. All rights reserved.

This document is free; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This document is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307, USA.

Avertissement

CE DOCUMENT est une traduction en français de « *The not so short introduction to LaTeX2e* » de Tobias Oetiker par Matthieu Herrb, adaptée pour les *Cahiers GUTenberg* par Daniel Flipo.

Une grande partie de ce document provient d'une introduction autrichienne à L^AT_EX 2.09, écrite en allemand par :

Hubert Partl <partl@mail.boku.ac.at>
Zentraler Informatikdienst der Universität für Bodenkultur, Wien
Irene Hyna <Irene.Hyna@bmf.ac.at>
Bundesministerium für Wissenschaft und Forschung, Wien
Elisabeth Schlegl <no_email>
in Graz

Vous trouverez la version anglaise de Tobias Oetiker sur
CTAN:/info/lshort/english/¹

Si vous êtes intéressé par la version allemande, vous la trouverez sur
CTAN:/info/lshort/german/

Vous trouverez la version officielle française sur
CTAN:/info/lshort/french/

Le document que vous avez entre les mains est disponible, comme les autres *Cahiers GUTenberg*, à l'URL :
<http://www.gutenberg.eu.org/pub/GUTenberg/publications>

Pour la préparation de ce document, l'aide des lecteurs du forum USENET `comp.text.tex` a été sollicitée. De nombreuses personnes ont répondu à l'appel et ont fourni des corrections, des suggestions et du texte pour améliorer ce document. Qu'elles en soient ici remerciées sincèrement.

1. Voir page vi la liste des sites CTAN.

Merci en particulier à :

Rosemary Bailey, Marc Bevand, Friedemann Brauer, Jan Busa,
Markus Brühwiler, Pietro Braione, David Carlisle, José Carlos Santos,
Mike Chapman, Pierre Chardaire, Christopher Chin, Carl Cerecke,
Chris McCormack, Wim van Dam, Jan Dittberner,
Michael John Downes, Matthias Dreier, David Dureisseix, Elliot,
Hans Ehrbar, Daniel Flipo, David Frey, Hans Fugal, Robin Fairbairns,
Jörg Fischer, Erik Frisk, Mic Milic Frederickx, Frank,
Kasper B. Graversen, Arlo Griffiths, Alexandre Guimond, Cyril Goutte,
Greg Gamble, Neil Hammond, Rasmus Borup Hansen, Joseph Hilferty,
Björn Hvittfeldt, Martien Hulsen, Werner Icking, Jakob, Eric Jacoboni,
Alan Jeffrey, Byron Jones, David Jones, Johannes-Maria Kaltenbach,
Michael Koundouros, Andrzej Kawalec, Alain Kessi, Christian Kern,
Jörg Knappen, Kjetil Kjernsmo, Maik Lehradt, Rémi Letot,
Johan Lundberg, Alexander Mai, Martin Maechler,
Aleksandar S Milosevic, Henrik Mitsch, Claus Malten,
Kevin Van Maren, Lenimar Nunes de Andrade, Demerson Andre Polli,
Maksym Polyakov Hubert Partl, John Reffing, Mike Ressler,
Brian Ripley, Young U. Ryu, Bernd Rosenlecher, Chris Rowley,
Risto Saarelma, Hanspeter Schmid, Craig Schlenter, Baron Schwartz,
Christopher Sawtell, Geoffrey Swindale, Boris Tobotras, Josef Tkadlec,
Scott Veirs, Didier Verna, Fabian Wernli, Carl-Gustav Werner,
David Woodhouse, Chris York, Fritz Zaucker, Rick Zaccone,
Mikhail Zotov.

La version française a bénéficié de corrections de lecteurs du forum
`fr.comp.text.tex` et en particulier de

Sébastien Blondeel, Marie-Dominique Cabanne,
Christophe Dousson, Olivier Dupuis, Daniel Flipo, Paul Gaborit,
Thomas Ribo, Philippe Spiesser et Vincent Zoonekynd.

Le traducteur tient également à remercier chaleureusement les auteurs de ce
document de l'avoir mis en domaine public et d'avoir ainsi rendu possible cette
version française.

Introduction

\LaTeX [1]* est un logiciel de composition typographique adapté à la production de documents scientifiques et mathématiques de grande qualité typographique. Il permet également de produire toutes sortes d'autres documents, qu'il s'agisse de simples lettres ou de livres entiers. \LaTeX utilise \TeX [2] comme outil de mise en page.

Cette introduction décrit $\LaTeX 2_{\epsilon}$ et devrait se montrer suffisante pour la plupart des applications de \LaTeX . Pour une description complète du système \LaTeX , reportez-vous à [1, 3].

\LaTeX est disponible pour une vaste gamme de systèmes informatiques, des PC et Mac aux systèmes Unix et VMS. Dans de nombreuses universités, il est installé sur le réseau informatique, prêt à être utilisé. Si vous avez des difficultés d'installation, contactez la personne qui vous a donné cette brochure, elle pourra probablement vous aider. Ce document *n'est pas* un guide d'installation du système \LaTeX . Son but est de vous apprendre à coder vos documents afin qu'ils puissent être traités par \LaTeX .

Cette introduction est composée de cinq chapitres

Le chapitre 1 présente la structure élémentaire d'un document $\LaTeX 2_{\epsilon}$. Il vous renseignera également sur l'histoire de \LaTeX . Après avoir lu ce chapitre, vous devriez avoir une vue générale de ce qu'est \LaTeX et être en mesure d'assimiler ce qui sera présenté dans les chapitres suivants.

Le chapitre 2 entre dans les détails de la mise en page d'un document. Il explique les commandes et les environnements essentiels de \LaTeX . Après avoir lu ce chapitre, vous serez capables de rédiger vos premiers documents.

Le chapitre 3 explique comment coder des formules mathématiques avec \LaTeX . De nombreux exemples sont donnés pour montrer comment utiliser cet atout majeur de \LaTeX . À la fin de ce chapitre, vous trouverez des tableaux qui listent tous les symboles mathématiques disponibles.

Le chapitre 4 explique comment réaliser un index, une liste de références bibliographiques ou l'insertion de figures PostScript. Il présente aussi quelques autres extensions utiles.

Le chapitre 5 explique comment produire des documents hypertextes en PDF à partir de \LaTeX , que ce soit pour l'affichage sur le WEB ou pour préparer des présentations à la *PowerPoint*.

*. Les numéros entre crochets renvoient à la bibliographie page 105 et suivantes.

Le chapitre 6 contient des informations potentiellement dangereuses. Il vous apprend à modifier la mise en page standard produite par \LaTeX et vous permet de transformer les présentations plutôt réussies de \LaTeX en quelque chose d'assez laid.

Il est important de lire ces chapitres dans l'ordre. Ce document n'est d'ailleurs pas si long. Prêtez une attention particulière aux exemples car leur étude attentive est indispensable à la compréhension de l'ensemble.

Si vous avez besoin de récupérer des fichiers relatifs à \LaTeX , utilisez l'une des archives ftp CTAN. En France elles sont sur `ftp.jussieu.fr` dans le répertoire `pub/TeX/CTAN`. Aux États-Unis, il s'agit de `ctan.tug.org`, en Allemagne de `ftp.dante.de` et au Royaume-Uni de `ftp.tex.ac.uk` (répertoire `tex-archive`). Si vous n'êtes pas dans l'un de ces pays, choisissez le site le plus proche de chez vous.

Si vous souhaitez installer \LaTeX sur votre ordinateur, vous trouverez sans doute une version adaptée à votre système sur `CTAN:/systems`.

Si vous avez des suggestions concernant ce qui pourrait être ajouté, supprimé ou modifié dans ce document, contactez soit directement l'auteur de la version originale, soit le traducteur, soit Daniel Flipo.

Les avis des lecteurs débutant en \LaTeX nous intéressent tout particulièrement, nous attendons d'eux qu'ils nous indiquent les points nécessitant des explications plus détaillées.

Tobias Oetiker <oetiker@ee.ethz.ch>
Department of Information Technology and Electrical Engineering
Swiss Federal Institute of Technology, Zürich.

Matthieu Herrb <matthieu@laas.fr>
Laboratoire d'Analyse et d'Architecture des Systèmes
Centre National de la Recherche Scientifique, Toulouse.

Daniel Flipo <Daniel.Flipo@univ-lille1.fr>
Laboratoire de statistique et probabilités
USTL, Villeneuve d'Ascq.

Table des matières

1	Ce qu'il faut savoir	1
1.1	Le nom de la bête	1
1.1.1	TeX	1
1.1.2	L ^A T _E X	1
1.2	Les bases	2
1.2.1	Auteur, éditeur et typographe	2
1.2.2	Choix de la mise en page	3
1.2.3	Avantages et inconvénients	3
1.3	Fichiers source L ^A T _E X	4
1.3.1	Espaces	4
1.3.2	Caractères spéciaux	5
1.3.3	Commandes L ^A T _E X	5
1.3.4	Commentaires	6
1.4	Structure du fichier source	6
1.5	Mise en page du document	8
1.5.1	Classes de documents	8
1.5.2	Extensions	9
1.5.3	Styles de page	10
1.6	Fichiers de la tribu L ^A T _E X	11
1.7	Gros documents	12
2	Mise en page	15
2.1	Structure du document	15
2.2	Sauts de ligne et de page	17
2.2.1	Paragraphe justifiés	17
2.2.2	Césure	18
2.3	Logos et date du jour	19

2.4	Caractères spéciaux et symboles	20
2.4.1	Guillemets	20
2.4.2	Tirets	20
2.4.3	Points de suspension (...)	20
2.4.4	Symbole euro (€)	21
2.4.5	Ligatures	22
2.4.6	Accents et caractères spéciaux	22
2.5	Support multilingue	24
2.5.1	Babel et le français	26
2.5.2	Babel et l'allemand	26
2.6	Espaces entre les mots	27
2.7	Titres, chapitres et sections	28
2.8	Références croisées	30
2.9	Notes de bas de page	30
2.10	Souligner l'importance d'un mot	31
2.11	Environnements	31
2.11.1	Listes, énumérations et descriptions	32
2.11.2	Alignements à gauche, à droite et centrage	32
2.11.3	Citations et vers	33
2.11.4	Impression <i>verbatim</i>	34
2.11.5	Tableaux	35
2.12	Objets flottants	37
3	Formules mathématiques	41
3.1	Généralités	41
3.2	Groupements en mode mathématique	43
3.3	Éléments d'une formule mathématique	44
3.4	Espacement en mode mathématique	48
3.5	Alignements verticaux	49
3.6	Fantômes...	51
3.7	Taille des polices mathématiques	51

3.8	Insertion de texte en mode mathématique	52
3.9	Théorèmes, propositions, etc.	53
3.10	Symboles gras	54
3.11	Liste des symboles mathématiques	55
4	Compléments	63
4.1	Figures PostScript	63
4.2	X _y -pic	65
4.3	Références bibliographiques	68
4.4	Index	68
4.5	En-têtes améliorés	70
4.6	L'extension verbatim	71
4.7	Téléchargement et installation d'extensions	72
4.8	Protection des commandes « fragiles »	73
5	Documents hypertextes en PDF	75
5.1	Documents hypertextes	75
5.2	Documents en PDF pour le WEB	76
5.2.1	Fontes	76
5.2.2	Graphiques	78
5.2.3	Liens hypertextes	78
5.2.4	Problèmes possibles avec les liens	80
5.2.5	Problèmes avec les signets (<i>bookmarks</i>)	81
5.2.6	Fichiers sources compatibles L ^A T _E X et pdfL ^A T _E X	82
5.3	Présentations à l'aide de pdfscreen	83
6	Personnalisation de L^AT_EX	87
6.1	Vos propres commandes, environnements et extensions	87
6.1.1	Nouvelles commandes	88
6.1.2	Nouveaux environnements	89
6.1.3	Votre propre extension	89

6.2	Polices et tailles des caractères	90
6.2.1	Commandes de changement de police	90
6.2.2	Attention danger	93
6.2.3	Un conseil	93
6.3	Espacement	94
6.3.1	Entre les lignes	94
6.3.2	Mise en page d'un paragraphe	94
6.3.3	Espacement horizontal	95
6.3.4	Espacement vertical	97
6.4	Disposition d'une page	97
6.5	Jouons un peu avec les dimensions	100
6.6	Boîtes	100
6.7	Filets	103
	Bibliographie	105
	Index	107

Chapitre 1

Ce qu'il faut savoir

Dans la première partie de ce chapitre vous trouverez une présentation rapide de la philosophie et de l'histoire de $\LaTeX 2_{\epsilon}$. La deuxième partie met l'accent sur les structures fondamentales d'un document \LaTeX . Après avoir lu ce chapitre, vous devriez avoir une idée d'ensemble du fonctionnement de \LaTeX qui vous aidera à mieux comprendre les chapitres suivants.

1.1. Le nom de la bête

1.1.1. \TeX

\TeX est un programme écrit par Donald E. Knuth [2]. Il est conçu pour la composition de textes et d'équations mathématiques.

Knuth a commencé le développement de \TeX en 1977 parce qu'il était frustré par la manière avec laquelle ses articles étaient publiés par l'*American Mathematical Society*. Il avait arrêté de soumettre des articles vers 1974 parce que « le résultat final était trop pénible à regarder ». \TeX , tel que nous l'utilisons aujourd'hui, est sorti en 1982 et a été amélioré au fil des ans. Ces dernières années \TeX a atteint une grande stabilité. Aujourd'hui Knuth affirme qu'il n'y a virtuellement plus de *bug*. Le numéro de version de \TeX tend vers π et est actuellement 3.14159.

\TeX se prononce « Tech », avec un « ch » comme « technique » ou dans le mot écossais « Loch ». En alphabet phonétique cela donne [tex]... Dans un environnement ASCII, \TeX s'écrit TeX.

1.1.2. \LaTeX

\LaTeX est un ensemble de macros qui permettent à un auteur de mettre en page son travail avec la meilleure qualité typographique en utilisant un format professionnel prédéfini. \LaTeX a été écrit par Leslie Lamport [1]. Il utilise \TeX comme outil de mise en page.

En 1994, \LaTeX a été mis à jour par l'équipe $\LaTeX 3$, menée par Frank Mittelbach, afin de réaliser certaines améliorations demandées depuis longtemps et

de fusionner toutes les variantes qui s'étaient développées depuis la sortie de \LaTeX 2.09 il y a quelques années. Pour distinguer cette nouvelle version des précédentes, elle est appelée $\text{\LaTeX} 2_{\varepsilon}$. Ce document est relatif à $\text{\LaTeX} 2_{\varepsilon}$.

\LaTeX se prononce `[latex]`. Si vous voulez faire référence à \LaTeX dans un environnement ASCII, écrivez `LaTeX`. $\text{\LaTeX} 2_{\varepsilon}$ se prononce `[latex døzø]` et s'écrit `LaTeX2e`.

En anglais, cela donne `[latex]` et `[latex tu: i:]`.

1.2. Les bases

1.2.1. Auteur, éditeur et typographe

Pour publier un texte, un auteur confie son manuscrit à une maison d'édition. L'éditeur décide alors de la mise en page du document (largeur des colonnes, polices de caractères, présentation des en-têtes...). L'éditeur note ses instructions sur le manuscrit et le passe à un technicien typographe qui réalise la mise en page en suivant ces instructions.

Un éditeur humain essaye de comprendre ce que l'auteur veut mettre en valeur et décide de la présentation en fonction de son expérience professionnelle et du contenu du manuscrit.

Dans un environnement \LaTeX , celui-ci joue le rôle de l'éditeur et utilise \TeX comme typographe pour la composition. Mais \LaTeX n'est qu'un programme et a donc besoin de plus de directives. L'auteur doit en particulier lui fournir la structure logique de son document. Cette information est insérée dans le texte sous la forme de « commandes \LaTeX ».

Cette approche est totalement différente de l'approche WYSIWYG¹ utilisée par les traitements de texte tels que *Microsoft Word* ou *Corel WordPerfect*. Avec ces programmes, l'auteur définit la mise en page du document de manière interactive pendant la saisie du texte. Tout au long de cette opération, il voit à l'écran à quoi ressemblera le document final une fois imprimé.

Avec \LaTeX , il n'est normalement pas possible de voir le résultat final durant la saisie du texte. Mais celui-ci peut être prévisualisé après traitement du fichier par \LaTeX . Des corrections peuvent alors être apportées avant d'envoyer la version définitive vers l'imprimante.

1. *What you see is what you get* – Ce que vous voyez est ce qui sera imprimé.

1.2.2. Choix de la mise en page

La typographie est un métier (un art?). Les auteurs inexpérimentés font souvent de graves erreurs en considérant que la mise en page est avant tout une question d'esthétique : « si un document est beau, il est bien conçu ». Mais un document doit être lu et non accroché dans une galerie d'art. La lisibilité et la compréhensibilité sont bien plus importantes que le *look*. Par exemple :

- la taille de la police et la numérotation des en-têtes doivent être choisies afin de mettre en évidence la structure des chapitres et des sections ;
- les lignes ne doivent pas être trop longues pour ne pas fatiguer la vue du lecteur, tout en remplissant la page de manière harmonieuse.

Avec un logiciel WYSIWYG, l'auteur produit généralement des documents esthétiquement plaisants (quoi que...) mais très peu ou mal structurés. \LaTeX empêche de telles erreurs de formatage en forçant l'auteur à décrire la structure logique de son document et en choisissant lui-même la mise en page la plus appropriée.

1.2.3. Avantages et inconvénients

Un sujet de discussion qui revient souvent quand des gens du monde WYSIWYG rencontrent des utilisateurs de \LaTeX est le suivant : « les avantages de \LaTeX par rapport à un traitement de texte classique » ou bien le contraire. La meilleure chose à faire quand une telle discussion démarre est de garder son calme, car souvent cela dégénère. Mais parfois on ne peut y échapper. . .

Voici donc quelques arguments. Les principaux avantages de \LaTeX par rapport à un traitement de texte traditionnel sont :

- mise en page professionnelle qui donne aux documents l'air de sortir de l'atelier d'un imprimeur ;
- la composition des formules mathématiques se fait de manière pratique ;
- il suffit de connaître quelques commandes de base pour décrire la structure logique du document. Il n'est pas nécessaire de se préoccuper de la mise en page ;
- des structures complexes telles que des notes de bas de page, des renvois, la table des matières ou les références bibliographiques sont produites facilement ;
- il existe des extensions gratuites pour la plupart des tâches qui ne sont pas gérées directement par \LaTeX . Par exemple pour inclure des figures POSTSCRIPT ou pour formater une bibliographie selon un standard précis. La majorité de ces extensions sont décrites dans *The \LaTeX Companion* [3] (en anglais) et dans [4, 5] (en français) ;
- \LaTeX encourage les auteurs à écrire des documents bien structurés, parce que c'est ainsi qu'il fonctionne (en décrivant la structure) ;

- T_EX, l’outil de formatage de L^AT_EX 2_ε, est réellement portable et gratuit. Ainsi il est disponible sur quasiment toutes les machines existantes.

L^AT_EX a également quelques inconvénients ; il est difficile pour moi d’en trouver, mais d’autres vous en citeront des centaines :

- L^AT_EX ne fonctionne pas bien pour ceux qui ont vendu leur âme ;
- bien que quelques paramètres des mises en page prédéfinies puissent être personnalisés, la mise au point d’une présentation entièrement nouvelle est difficile et demande beaucoup de temps² ;
- écrire des documents mal organisés et mal structurés est très difficile.

1.3. Fichiers source L^AT_EX

L’entrée de L^AT_EX est un fichier texte ASCII. Vous pouvez le créer avec l’éditeur de texte de votre choix. Il contient le texte de votre document ainsi que les commandes qui vont permettre à L^AT_EX de le mettre en page. On appelle ce fichier *fichier source*.

1.3.1. Espaces

Les caractères d’espacement, tels que les blancs ou les tabulations, sont traités de manière unique comme « espace » par L^AT_EX. Plusieurs blancs *consécutifs* sont considérés comme *une seule* espace³. Les espaces en début d’une ligne sont en général ignorés et un seul retour à la ligne est traité comme une espace.

Une ligne vide entre deux lignes de texte marque la fin d’un paragraphe. *Plusieurs* lignes vides sont considérées comme *une seule* ligne vide. Le texte ci-dessous est un exemple. Sur la gauche se trouve le contenu du fichier source et à droite le résultat formaté.

Saisir un ou plusieurs
espaces entre les mots
n’a pas d’influence.

Une ligne vide commence
un nouveau paragraphe.

Saisir un ou plusieurs espaces entre les
mots n’a pas d’influence.

Une ligne vide commence un nouveau pa-
ragraphe.

2. La rumeur dit que c’est l’un des points qui devrait être améliorés dans la future version L^AT_EX3.

3. En langage typographique, *espace* est un mot féminin. *NdT*.

1.3.2. Caractères spéciaux

Les symboles suivants sont des caractères réservés qui, soit ont une signification spéciale dans \LaTeX , soit ne sont pas disponibles dans toutes les polices. Si vous les saisissez directement dans votre texte, ils ne seront pas imprimés et forceront \LaTeX à faire des choses que vous n'avez pas voulues.

\$ % ^ & _ { } ~ \

Comme vous le voyez ci-dessous, certains de ces caractères peuvent être utilisés dans vos documents en les préfixant par un antislash (barre inversée `\`) :

`\# \ $ \% \^{} \& _ \{ \} \~{} \`

\$ % ^ & _ { } ~

L'antislash `\` ne peut pas être saisi en ajoutant un second antislash (`\\`) car cette séquence est utilisée pour indiquer les coupures de ligne ; on utilise la commande `\textbackslash` pour produire le caractère `\`.

1.3.3. Commandes \LaTeX

Les commandes \LaTeX utilisent l'un des deux formats suivants :

- soit elles commencent par un antislash `\` et ont un nom composé uniquement de lettres. Le nom de commande est terminé par une espace, un chiffre ou tout autre caractère qui n'est pas une lettre ;
- soit elles sont composées d'un antislash et d'un caractère spécial (non-lettre) exactement (`\$`, `\^`, `\!` par exemple).

Elles sont sensibles à la casse des caractères (majuscules ou minuscules) c.-à-d. que `\AB`, `\Ab` et `\ab` sont trois commandes différentes. \LaTeX ignore les espaces après les commandes. Si vous souhaitez obtenir un blanc après une commande, il faut ou bien insérer `{ }` suivi d'un blanc ou bien utiliser une commande d'espacement spécifique de \LaTeX . La séquence `{ }` empêche \LaTeX d'ignorer les blancs après une commande.

J'ai lu que Knuth classe les gens qui utilisent `\TeX{}` en `\TeX{}`nciens et en `\TeX` perts.`\`
Nous sommes le `\today`.

J'ai lu que Knuth classe les gens qui utilisent \TeX en \TeX nciens et en \TeX perts.
Nous sommes le 12 janvier 2003.

Certaines commandes sont suivies de paramètres obligatoires (saisis entre accolades `{ }`) ou optionnels (saisis entre crochets `[]`). Voici quelques exemples

de commandes à paramètres, ne cherchez pas à les comprendre, elles seront expliquées plus loin.

```
\textsl{Penchez}-vous !\newline
S'il vous plaît, passez à la
ligne ici.\linebreak[4] Merci !
\rule[-2mm]{10mm}{4mm} dessine un
rectangle.
```

Penchez-vous!
 S'il vous plaît, passez à la ligne ici.
 Merci! ██████ dessine un rectangle.

1.3.4. Commentaires

Quand \LaTeX rencontre un caractère % dans le fichier source, il ignore le reste de la ligne en cours. C'est utile pour ajouter des notes qui n'apparaîtront pas dans la version imprimée. Le caractère % peut également être utilisé en fin de ligne pour supprimer l'espace introduit par le passage à la ligne.

```
% Démonstration :
Ceci est un % mauvais
exemple : anticonstitu%
tionnellement
```

Ceci est un exemple : anticonstitutionnel-
 lement

Pour créer des commentaires plus longs sans avoir à placer un % devant chaque ligne, on peut utiliser l'environnement `comment` fourni par l'extension `verbatim` (voir page 71).

1.4. Structure du fichier source

Quand \LaTeX analyse un fichier source, il s'attend à y trouver une certaine structure. C'est pourquoi chaque fichier source doit commencer par la commande :

```
\documentclass{...}
```

Elle indique quel type de document vous voulez écrire. Après cela vous pouvez insérer des commandes qui vont influencer le style du document ou vous pouvez charger des extensions qui ajoutent de nouvelles fonctions au système \LaTeX . Pour charger une extension, utilisez la commande :

```
\usepackage{...}
```

Quand tout le travail de préparation est fait ⁴, vous pouvez commencer le corps du texte avec la commande :

```
\begin{document}
```

4. La partie entre `\documentclass` et `\begin{document}` est appelée la *préambule*.

Maintenant vous pouvez saisir votre texte et y insérer des commandes \LaTeX . À la fin de votre document, utilisez la commande

```
\end{document}
```

pour dire à \LaTeX qu'il en a fini. Tout ce qui suivra dans le fichier source sera ignoré.

La figure 1.1 montre (à gauche) le contenu d'un document $\LaTeX 2_{\epsilon}$ minimal et (à droite) un exemple de document en français. Un fichier source plus complet est présenté en figure 1.2.

<pre>\documentclass{article}</pre>	<pre>\documentclass{article}</pre>
<pre>\begin{document}</pre>	<pre>\usepackage[latin1]{inputenc}</pre>
<pre>Small is beautiful.</pre>	<pre>\usepackage[T1]{fontenc}</pre>
<pre>\end{document}</pre>	<pre>\usepackage[frenchb]{babel}</pre>
	<pre>\begin{document}</pre>
	<pre>Où le caïd a-t-il passé la fête</pre>
	<pre>de Noël ? À Évian ? Ça alors !</pre>
	<pre>\end{document}</pre>

FIGURE 1.1 – Un fichier \LaTeX minimal en anglais et un en français

```
\documentclass[a4paper,11pt]{article}
\usepackage[T1]{fontenc}
\usepackage[english,frenchb]{babel}
\author{P.~Tar}
\title{Le Minimalisme}
\begin{document}
\maketitle
\tableofcontents
\section{Début}
\A écrire\dots
\section{Suite et fin}
On verra plus tard.
\end{document}
```

FIGURE 1.2 – Exemple d'article de revue plus réaliste

1.5. Mise en page du document

1.5.1. Classes de documents

La première information dont L^AT_EX a besoin est le type du document à produire. Ce type est spécifié par la commande `\documentclass`.

```
\documentclass[options]{classe}
```

TABLE 1.1 – Classes de documents

article	pour des articles dans des revues scientifiques, des présentations, des rapports courts, des documentations, des invitations. . .
report	pour des rapports plus longs contenant plusieurs chapitres, des petits livres, des thèses. . .
book	pour des vrais livres.
slides	pour des transparents. Cette classe utilise de grands caractères sans serif. Voir également la classe seminar (plus riche) ou l'extension pdfscreen (cf. page 83) pour les présentations à la <i>PowerPoint</i> .

Ici *classe* indique le type de document à créer. Le tableau 1.1 donne la liste des classes de documents présentées dans cette introduction. Il en existe bien d'autres : des variantes des classes **article** et **book** (par exemple **amsart**, **amsbook**, **smfart**, **smfbook**), les classes **letter** et **lettre** conçues pour les courriers, etc.

Le paramètre *options* permet de modifier le comportement de la classe de document. Les noms des options sont séparés par des virgules. Les principales options disponibles sont présentées dans le tableau 1.2.

Exemple : un fichier source pour un document L^AT_EX pourrait commencer par la ligne

```
\documentclass[11pt,twoside,a4paper]{article}
```

elle informe L^AT_EX qu'il doit composer ce document comme un *article* avec une taille de caractère de base de *onze points* et qu'il devra produire une mise en page pour une impression *double face* sur du papier au format *A4* (sans l'option **a4paper**, le format de papier serait américain : 8,5 × 11 pouces, soit 216 × 280 mm).

TABLE 1.2 – Options de classes de document

<code>10pt, 11pt, 12pt</code>	définit la taille de la police principale du document. Si aucune option n'est présente, la taille par défaut est de <code>10pt</code> .
<code>a4paper, letterpaper, ...</code>	définit la taille du papier. Le papier par défaut est <code>letterpaper</code> , le format standard américain. Les autres valeurs possibles sont : <code>a5paper, b5paper, executivepaper</code> et <code>legalpaper</code> .
<code>fleqn</code>	aligne les formules mathématiques à gauche au lieu de les centrer.
<code>leqno</code>	place la numérotation des formules à gauche plutôt qu'à droite.
<code>titlepage, notitlepage</code>	indique si une nouvelle page doit être commencée après le titre du document ou non. La classe <code>article</code> continue par défaut sur la même page contrairement aux classes <code>report</code> et <code>book</code> .
<code>twocolumn</code>	demande à <code>L^AT_EX</code> de présenter le texte sur deux colonnes.
<code>twoside, oneside</code>	indique si la sortie se fera en recto-verso ou en recto simple. Par défaut, les classes <code>article</code> et <code>report</code> sont en recto alors que la classe <code>book</code> est en recto-verso.
<code>openright, openany</code>	fait commencer un chapitre sur une page de droite ou indifféremment sur une page droite ou gauche. Cette option n'a pas de sens avec la classe <code>article</code> qui ne connaît pas la notion de chapitre. Par défaut, la classe <code>book</code> place les débuts de chapitre toujours sur une page de droite, c'est ce qui la différencie de la classe <code>report</code> .

1.5.2. Extensions

`LATEX` est un ensemble modulaire dont les possibilités peuvent être complétées si nécessaire en faisant appel à des extensions. Si vous voulez par exemple inclure des graphiques, du texte en couleur ou le code d'un programme dans votre document, il vous faudra charger des extensions en utilisant la commande

```
\usepackage[options]{extension}
```

Extension est le nom de l'extension et *options* une liste de mots-clés qui déclenchent certaines fonctions de l'extension. Certaines extensions font partie de la distribution standard de `LATEX 2ε` (reportez-vous au tableau 1.3), d'autres sont distribuées à part.

The L^AT_EX Companion [3] est la principale source d'information sur `LATEX 2ε`. Ce livre contient la description de certaines d'extensions ainsi que les indications nécessaires pour écrire vos propres extensions à `LATEX 2ε`.

TABLE 1.3 – Quelques extensions fournies avec L^AT_EX

<code>doc</code>	permet de documenter des programmes pour L ^A T _E X. Décrité dans <code>doc.dtx</code> ^a et dans <i>The L^AT_EX Companion</i> [3].
<code>exscale</code>	fournit des versions de taille paramétrable des polices mathématiques étendues. Décrité dans <code>ltxscale.dtx</code> .
<code>fontenc</code>	spécifie le codage des polices de caractères utilisées. Décrité dans <code>ltoutenc.dtx</code> .
<code>ifthen</code>	fournit des commandes de la forme ‘if...then do...otherwise do...’ Décrité dans <code>ifthen.dtx</code> , dans <i>The L^AT_EX Companion</i> [3] et dans [4].
<code>latexsym</code>	permet l’utilisation de la police des symboles L ^A T _E X. Décrité dans <code>latexsym.dtx</code> , dans <i>The L^AT_EX Companion</i> [3] et dans [4].
<code>makeidx</code>	fournit des commandes pour réaliser un index. Décrité dans ce document, section 4.4 et dans <i>The L^AT_EX Companion</i> [3] ou dans [4].
<code>syntonly</code>	analyse un document sans le formater. Décrité dans <code>syntonly.dtx</code> et dans <i>The L^AT_EX Companion</i> [3]. Utile pour une vérification rapide de la syntaxe.
<code>inputenc</code>	permet de spécifier le codage des caractères utilisé dans le fichier source, parmi ISO Latin-1, ISO Latin-2, 437/850 IBM code pages, Apple Macintosh, Next, ANSI-Windows ou un codage défini par l’utilisateur. Décrité dans <code>inputenc.dtx</code> .

^a. Ce fichier devrait être installé sur votre système et vous devriez être capable de le formater avec la commande `latex doc.dtx`. Il en est de même pour les autres fichiers cités dans ce tableau.

1.5.3. Styles de page

L^AT_EX propose trois combinaisons d’en-têtes et de pieds de page, appelées styles de page et définies par le paramètre *style* de la commande :

`\pagestyle{style}`

Le tableau 1.4 donne la liste des styles prédéfinis.

TABLE 1.4 – Les styles de page de L^AT_EX

plain	imprime le numéro de page au milieu du pied de page. C'est le style par défaut.
headings	imprime le titre du chapitre courant et le numéro de page dans l'en-tête de chaque page et laisse le pied de page vide. C'est à peu près le style utilisé dans ce document.
empty	laisse l'en-tête et le pied de page vides.

On peut changer le style de la page en cours grâce à la commande

```
\thispagestyle{style}
```

Au chapitre 4, page 70, vous apprendrez comment créer vos propres en-têtes et pieds de pages.

1.6. Fichiers de la tribu L^AT_EX

L'utilisateur de L^AT_EX est amené à côtoyer un grand nombre de fichiers aux suffixes divers. Chaque suffixe renseigne sur le rôle du fichier, il est donc utile d'en connaître la signification. Voici les suffixes les plus courants :

- .tex** fichier source T_EX ou L^AT_EX ;
- .sty** fichier contenant des commandes, que l'on charge dans le préambule d'un document grâce à une commande `\usepackage` ;
- .cls** désigne un fichier de *classe* contenant la description d'un type de document, par exemple `article.cls`, `book.cls`, etc. ;
- .fd** fichier contenant des définitions pour les polices de caractères ;
- .dtx** fichier contenant du code L^AT_EX (commandes) documenté ; le lancement de L^AT_EX sur un fichier `.dtx` en extrait la documentation (voir section 4.7, page 72) ;
- .ins** fichier permettant d'installer le contenu du fichier `.dtx` de même nom (lancer L^AT_EX sur un fichier `.ins` crée un ou plusieurs fichiers de suffixes `.sty`, `.fd`, etc., directement utilisables par L^AT_EX).

Les fichiers suivants sont produits par L^AT_EX à partir du fichier source (de suffixe `.tex`) :

- .dvi** signifie *DeVice Independent*, c'est le fichier que l'on visualise à l'écran et qui servira à l'impression (par `dvips` par exemple) ;

-
- .log** fichier contenant le compte-rendu de la compilation du fichier source (avec les messages d'erreur éventuels...),
 - .toc** contient le matériel nécessaire à la production de la table des matières, si celle-ci a été demandée;
 - .lof** contient la liste numérotée des figures, si elle a été demandée;
 - .lot** contient la liste numérotée des tableaux, si elle a été demandée;
 - .aux** contient diverses informations utiles à \LaTeX , en particulier ce qui est nécessaire au fonctionnement des références croisées;
 - .idx** fichier produit seulement si un index est demandé, il doit être traité par `makeindex` (voir section 4.4 page 68);
 - .ind** fichier produit par `makeindex` à partir du `.idx`, il contient l'index prêt à être inclus dans le document;
 - .ilg** fichier contenant le compte-rendu du travail de `makeindex`.

1.7. Gros documents

Lorsque l'on travaille sur de gros documents, il peut être pratique de couper le fichier source en plusieurs morceaux. \LaTeX a deux commandes qui vous permettent de gérer plusieurs fichiers sources.

`\include{nom de fichier}`

Vous pouvez utiliser cette commande dans le corps de votre document pour insérer le contenu d'un autre fichier source, obligatoirement suffixé par `.tex`, le suffixe sera ajouté automatiquement. Remarquez que \LaTeX ajoutera un saut de page avant l'insertion du fichier *nom de fichier* et qu'aucune commande `\include` n'est admise dans un fichier lui-même chargé par un `\include`.

La seconde commande peut être utilisée dans le préambule. Elle permet de dire à \LaTeX de n'inclure que certains des fichiers désignés par les commandes `\include`.

`\includeonly{nom de fichier,nom de fichier,...}`

Si une telle commande figure dans le préambule d'un document, seules les commandes `\include` dont les fichiers sont cités en paramètre de la commande `\includeonly` seront exécutées. Attention, il ne doit pas y avoir d'espace entre le nom de fichier et la virgule qui le suit.

La commande `\include` insère un saut de page avant de commencer le formatage du texte inclus. Ceci est utile lorsqu'on utilise `\includeonly`, parce qu'ainsi les sauts de pages ne bougeront pas, même si certains morceaux ne

sont pas inclus. Parfois ce comportement n'est pas souhaitable. Dans ce cas, vous pouvez utiliser la commande :

```
\input{nom de fichier}
```

qui insère simplement le fichier indiqué sans aucun traitement sophistiqué (le suffixe `.tex` est ajouté automatiquement).

Il est possible de demander à L^AT_EX de simplement vérifier la syntaxe d'un document, sans produire de fichier `.dvi` pour gagner du temps, en utilisant l'extension `syntonly` :

```
\usepackage{syntonly}  
\syntonly
```

La vérification terminée, il suffit de mettre ces deux lignes (ou simplement la seconde) en commentaire en plaçant un `%` en tête de ligne.

Chapitre 2

Mise en page

Après la lecture du chapitre précédent vous connaissez maintenant les éléments de base qui constituent un document \LaTeX . Dans ce chapitre, nous allons compléter vos connaissances afin de vous rendre capables de créer de vrais documents.

2.1. Structure du document

La principale raison d'être d'un texte (à l'exception de certains textes de la littérature contemporaine) est de diffuser des idées, de l'information ou de la connaissance au lecteur. Celui-ci comprendra mieux le texte si ces idées sont bien structurées et il ressentira d'autant mieux cette structure si la typographie utilisée reflète la structure logique et sémantique du contenu.

Ce qui distingue \LaTeX d'autres logiciels de traitement de texte comme *Word*, c'est qu'il suffit d'indiquer à \LaTeX la structure logique et sémantique d'un texte. Il en déduit la forme typographique en fonction des « règles » définies dans la classe de document et dans les différents fichiers de style.

L'élément de texte le plus important pour \LaTeX (et en typographie) est le paragraphe. Le paragraphe est la forme typographique qui contient une pensée cohérente ou qui développe une idée. Vous allez apprendre dans les pages suivantes la différence entre un retour à la ligne (obtenu avec la commande \backslash) et un changement de paragraphe (obtenu en laissant une ligne vide dans le document source). Une nouvelle réflexion doit débiter sur un nouveau paragraphe, mais si vous poursuivez une réflexion déjà entamée, un simple retour à la ligne suffit.

En général, on sous-estime complètement l'importance du découpage en paragraphes. Certains ignorent même la signification d'un changement de paragraphe ou bien, notamment avec \LaTeX , coupent des paragraphes sans le savoir. Cette erreur est particulièrement fréquente lorsque des équations sont présentes au milieu du texte. Étudiez les exemples suivants et essayez de comprendre pourquoi des lignes vides (changements de paragraphe) sont parfois utilisées avant et après l'équation et parfois non. Si vous ne comprenez pas suffisamment les commandes utilisées, lisez d'abord la suite du chapitre puis revenez à cette section.

```

% Exemple 1
... lorsqu'Einstein introduit sa formule
\begin{equation}
  e = m \cdot c^2 \ ; \ ;
\end{equation}
qui est en même temps la formule la plus connue et la
moins comprise de la physique.

% Exemple 2
... d'où vient la loi des courants de Kirchhoff :
\begin{equation}
  \sum_{k=1}^n I_k = 0 \ ; \ .
\end{equation}

La loi de tensions de Kirchhoff s'en déduit...

% Exemple 3
\dots{} qui a plusieurs avantages.

\begin{equation}
  I_D = I_F - I_R
\end{equation}
est le c\oe{}ur d'un modèle de transistor très
différent...

```

L'unité de texte immédiatement inférieure est la phrase. Dans les documents américains, l'espace après le point terminant une phrase est plus grande que celle qui suit un point après une abréviation. Cette règle n'a pas cours en typographie française. En général \LaTeX détermine automatiquement la bonne largeur des espaces en fonction de la langue utilisée, à condition bien sûr que celle-ci lui soit déclarée. Il arrive que des corrections soient nécessaires, vous apprendrez dans la suite comment les faire.

La structure du texte s'étend même aux morceaux d'une phrase. Les règles grammaticales de chaque langue gèrent la ponctuation de manière très précise. Dans la plupart des langues, la virgule représente une courte respiration dans le flux du langage. Si vous ne savez pas trop où placer une virgule, lisez la phrase à voix haute en respirant à chaque virgule. Si cela ne sonne pas naturellement à certains endroits, supprimez la virgule. Au contraire, si vous ressentez le besoin de respirer (ou de marquer une courte pause), insérez un virgule à cet endroit.

Enfin, les paragraphes d'un texte sont en général regroupés dans des structures de niveau supérieur, sections, chapitres, etc. En \LaTeX , on utilise des commandes telles que `\section{Structure du document}` ou

`\chapter{Mise en page}` qui effectuent automatiquement la mise en page des titres de sections et de chapitres et enregistrent dans les fichiers `.aux`, `.toc`, etc. les informations qui permettront de faire référence à ces parties du document et de construire une table des matières.

2.2. Sauts de ligne et de page

2.2.1. Paragraphes justifiés

Bien souvent les livres sont composés de lignes qui ont toutes la même longueur ; on dit qu'elles sont justifiées à droite. \LaTeX insère des retours à la ligne et des espacements entre les mots de manière à optimiser la présentation de l'ensemble d'un paragraphe. En cas de besoin, il coupe les mots qui ne tiennent pas en entier sur une ligne. La présentation exacte d'un paragraphe dépend de la classe de document. Normalement la première ligne d'un paragraphe est en retrait par rapport à la marge gauche et il n'y a pas d'espace vertical supplémentaire entre deux paragraphes (cf. section 6.3.2).

Dans certains cas particuliers, il peut être nécessaire de demander à \LaTeX de couper une ligne :

`\` ou `\newline`

commence une nouvelle ligne sans commencer un nouveau paragraphe.

`\`*

empêche un saut de page après le saut de ligne demandé.

`\newpage`

provoque un saut de page.

`\linebreak[n]`, `\nolinebreak[n]`, `\pagebreak[n]` et `\nopagebreak[n]`

font ce que suggère leur nom anglais. Ces commandes permettent à l'auteur de paramétrer leur action par l'intermédiaire du paramètre optionnel n . Il peut prendre une valeur entre zéro et quatre. En donnant à n une valeur inférieure à quatre, vous laissez à \LaTeX la possibilité de ne pas tenir compte de votre commande si cela devait rendre le résultat réellement laid. Ne confondez pas

ces commandes —*break* avec les commandes *new*—. Même lorsque vous utilisez une commande —*break*, L^AT_EX essaye de justifier le bord droit du texte ou d’ajuster la longueur totale de la page, comme expliqué plus loin. Si vous voulez réellement commencer une « nouvelle » ligne, utilisez la commande *new*—correspondante.

Pour chaque paragraphe, L^AT_EX recherche activement la meilleure répartition du texte en lignes le même longueur. S’il ne trouve pas de solution pour couper les lignes de manière conforme à ses normes de qualité, il laisse dépasser un bout de texte sur la marge droite du paragraphe. L^AT_EX émet alors le message d’erreur *overfull hbox*¹. Cela se produit surtout quand L^AT_EX ne trouve pas de point de césure dans un mot. En utilisant alors la commande `\sloppy`, vous pouvez demander à L^AT_EX d’être moins exigeant. Il ne produira plus de lignes trop longues mais ajoutera de l’espace entre les mots du paragraphe, même si ceux-ci finissent par être trop espacés selon ses critères. Dans ce cas le message *underfull hbox*² est produit. Souvent, malgré tout, le résultat est acceptable. La commande `\fussy` agit dans l’autre sens, au cas où vous voudriez voir L^AT_EX revenir à ses exigences normales.

2.2.2. Césure

L^AT_EX coupe les mots en bout de ligne si nécessaire. Il n’y a normalement pas à intervenir manuellement. Lorsqu’on constate de mauvaises césures, c’est presque toujours qu’on a oublié de déclarer la langue de travail (par défaut L^AT_EX travaille en anglais), on verra comment le faire à la section 2.5. Si les mots contenant des caractères accentués ne sont pas coupés, il s’agit vraisemblablement d’un problème de codage de sortie, voir page 25.

Pour informer T_EX d’éventuelles exceptions, on peut utiliser la commande :

```
\hyphenation{liste de mots}
```

qui impose de ne couper les mots cités en argument qu’aux endroits indiqués par « - ». Lorsque cette commande est placée dans le préambule, elle affecte uniquement la langue par défaut, c’est-à-dire l’anglais. Placée dans le corps du document, elle n’a d’effet que sur la langue active à cet endroit du document (voir section 2.5). La casse des caractères n’est pas prise en compte : l’exemple ci-dessous empêche toute césure de « FORTRAN », « Fortran » ou « fortran ». Il autorise la coupure de « antidémocratique » et de « antédiluvien » (avec

1. Débordement horizontal.
2. Boîte horizontale pas assez pleine.

ou sans majuscules, mais au *au singulier*, le pluriel et le féminin ne sont pas affectés) uniquement après le préfixe anti ou anté.

```
\hyphenation{FORTRAN}
\hyphenation{Anti-démocratique anté-diluvien}
```

La commande `\hyphenation{liste de mots}` a un effet *global* sur toutes les occurrences des mots de la liste. Si l'exception ne concerne qu'une occurrence d'un mot on utilise la commande `\-` qui insère un point de césure potentiel dans un mot. Ces positions deviennent alors les *seuls* points de césure possibles pour cette occurrence du mot.

```
I think this is: su\per\cal\-%
i\frag\i\lis\tic\ex\pi\-%
al\i\do\cious
```

```
I think this is: supercalifragilisticxpiali-
docious
```

Normalement, en français, on ne coupe pas la dernière syllabe d'un mot si elle est muette, mais il arrive qu'on soit obligé de le faire, par exemple si on travaille sur des textes étroits (cas de colonnes multiples).

Exemple: on pourra coder `ils ex\pri\ment` pour autoriser *exceptionnellement* le rejet à la ligne suivante de la syllabe muette `ment`.

Plusieurs mots peuvent être maintenus ensemble sur une ligne avec la commande :

```
\mbox{texte}
```

Elle a pour effet d'interdire toute coupure de ligne dans *texte*.

```
Mon numéro de téléphone va
changer. \A partir du 5 octobre,
ce sera le 0561 336 330.
```

```
Mon numéro de téléphone va
changer. \A partir du 5 octobre,
ce sera le \mbox{0561 336 330}.
```

```
Mon numéro de téléphone va changer. À
partir du 5 octobre, ce sera le 0561 336
330.
```

```
Mon numéro de téléphone va chan-
ger. À partir du 5 octobre, ce sera le
0561 336 330.
```

2.3. Logos et date du jour

Certains lecteurs se demandent peut-être comment sont produits les logos $\text{T}_{\text{E}}\text{X}$, $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, etc. Le tableau ci-dessous donne la réponse et introduit également une commande bien pratique pour afficher la date.

Commande	Résultat	Description
<code>\today</code>	12 janvier 2003	Date du jour dans la langue courante
<code>\TeX</code>	\TeX	Logo TeX
<code>\LaTeX</code>	\LaTeX	Logo LaTeX
<code>\LaTeXe</code>	$\LaTeX_{2\epsilon}$	Version actuelle de \LaTeX

2.4. Caractères spéciaux et symboles

2.4.1. Guillemets

Pour insérer des guillemets n'utilisez pas le caractère " comme sur une machine à écrire. En typographie, il y a des guillemets ouvrants et fermants spécifiques. Avec \LaTeX , en anglais utilisez deux ‘ (accent grave) pour les guillemets ouvrants et deux ’ (apostrophe) pour les guillemets fermants. En français, avec l'extension `babel/frenchb`, utilisez `\og` et `\fg`³.

`‘‘Please press the ‘x’ key.’’`

`\og Appuyez sur la touche ‘x’.\fg`

“Please press the ‘x’ key.”

« Appuyez sur la touche ‘x’. »

2.4.2. Tirets

\LaTeX connaît quatre types de tirets. Trois d'entre eux sont obtenus en juxtaposant un nombre variable de tirets simples. Le quatrième n'est pas réellement un tiret, il s'agit du signe mathématique moins.

`belle-fille, c.-à-d.\l`
`Il parle --- en vain ---`
`du passé.\l Oui --- ou non ?`
`0, 1 et -1.\l`
`See pages~13--67.`

belle-fille, c.-à-d.
 Il parle — en vain — du passé.
 Oui — ou non ? 0, 1 et -1.
 See pages 13-67.

2.4.3. Points de suspension (...)

Sur une machine à écrire, une virgule ou un point occupent la même largeur que les autres lettres. En typographie professionnelle, le point occupe très peu

³. ou les symboles « et » si votre clavier en dispose, ceci nécessite une petite adaptation : voir le fichier `frenchb.cfg`.

de place et il est placé tout près du caractère qui le précède. Il n'est donc pas possible d'utiliser trois points consécutifs pour créer des points de suspension. À la place on utilise la commande spécifique :

```
\dots
```

```
Non, pas comme ça...
mais ainsi :\
New York, Tokyo, Budapest\dots
```

```
Non, pas comme ça... mais ainsi :
New York, Tokyo, Budapest...
```

2.4.4. Symbole euro (€)

Le symbole monétaire € n'est malheureusement pas disponible en codage T1, le recours à une police de caractères complémentaire et à une extension est indispensable. Il y a plusieurs choix possibles.

L'extension `eurosym` fournit le symbole officiel (sans serif), quelle que soit la police utilisée pour le texte. L'euro se code `\euro`. Si on ajoute l'option `gen` à l'appel de `eurosym`, le symbole s'adapte au type de la police courante (romain, sans serif, ou *typewriter*, voir page 91).

```
\usepackage[gen]{eurosym}
```

Si les fontes « Eurofonts » d'Adobe sont installées sur votre système (elles sont disponibles gratuitement sur <ftp://ftp.adobe.com/pub/adobe/type/win/all>), vous pourrez faire appel à l'une ou l'autre des extensions `europs` (symbole euro adapté à la police courante, l'euro se code `\EUR`) ou `eurosans` (euro officiel indépendant de la police courante, l'euro se code `\euro`).

L'extension `marvosym` fournit de nombreux symboles, dont l'euro, qui se code `\EUR` (son aspect est indépendant de la police courante). L'inconvénient de cette solution par rapport aux deux précédentes est l'absence de variantes grasse et italique pour le symbole euro.

L'extension `eurofont` permet, grâce à de multiples options, d'avoir accès à pratiquement tous les symboles disponibles pour l'euro (voir la documentation dans le fichier `eurofont.dtx`). Utilisée sans option, cette extension fait appel aux fontes « Eurofonts » d'Adobe et produit les mêmes résultats que `europs` mais l'euro se code `\euro`.

Enfin, précisons qu'une somme en euros s'écrit en français 12,45 € et se code `12,45~\euro` ou `12,45~\EUR` selon l'extension utilisée (avec espace insécable) mais jamais 12.45 € ni 12€45 !

TABLE 2.1 – Différents symboles pour l’euro

extension	codage	romain	sans-serif	chasse fixe
eurosym	<code>\euro</code>	€	€	€
[gen]eurosym	<code>\euro</code>	€	€	€
europs	<code>\EUR</code>	€	€	€
eurosans	<code>\euro</code>	€	€	€
marvosym	<code>\EUR</code>	€	€	€

2.4.5. Ligatures

Certaines séquences de lettres ne sont pas composées simplement en juxtaposant les différentes lettres les unes à la suite des autres, mais en utilisant des symboles spéciaux.

`ff fi fl ffi...` à la place de `ff fi fl ffi...`

Ces ligatures peuvent être désactivées en insérant un `\mbox{}` entre les lettres en question. Cela peut s’avérer utile pour certains mots composés anglais ou allemands.

Not shelfful
but shelf\mbox{}ful\
Wachstube oder Wachs\mbox{}tube

Not shelfful but shelfful Wachstube oder Wachstube

En allemand, la ligature « st », présente dans certaines fontes, est légitime si on parle du corps de garde (Wach-stube) mais doit être évitée si on parle du tube de cire (Wachs-tube).

2.4.6. Accents et caractères spéciaux

L^AT_EX permet l’utilisation d’accents et de caractères spéciaux issus de nombreuses langues. Le tableau 2.2 montre tous les accents que l’on peut ajouter à la lettre o (ils s’appliquent naturellement aux autres lettres) et certains autres caractères spéciaux.

Pour placer un accent sur un i ou un j, il faut supprimer leur point. Ceci s’obtient en tapant `\i` et `\j`.

TABLE 2.2 – Accents et caractères spéciaux

ò	\'o	ó	\'o	ô	\^o	õ	\~o
ō	\=o	ô	\.o	ö	\"o	ç	\c c
ö	\u o	ö	\v o	ö	\H o	ø	\c o
ø	\d o	ø	\b o	ö	\t oo		
œ	\oe	Œ	\OE	ø	\o	Ø	\O
å	\aa	Å	\AA	æ	\ae	Æ	\AE
ı	\i	ı	\j	ı	ı	ı	ı
ı	\l	ı	\L				

```
H\^otel, na\"i ve, \'el\'eve,\
sm\o rrebr\o d, !'Se\~norita!,\
Sch\"onbrunner Schlo\ss{ }
Stra\ss e
```

```
Hôtel, naïve, élève,
smørrebrød, ¡Señorita!,
Schönbrunner Schloß Straße
```

Cependant, les systèmes informatiques modernes vous permettent de saisir directement les caractères accentués ou les symboles spécifiques d'une langue. Pour les majuscules (Ç, À, É) le codage `\c C`, `\'A`, `\'E` peut être utile. Depuis la version de décembre 1994, L^AT_EX 2_ε sait gérer les caractères accentués grâce à l'extension `inputenc` :

```
\usepackage[codage]{inputenc}  codage est latin1 sous Unix ou Windows,
                               applemac sur Mac...
```

En utilisant cette extension, il faut prendre garde au fait que les systèmes informatiques n'utilisent pas tous le même codage des caractères spéciaux. Le caractère *é* par exemple, est codé 142 sur Mac et 233 sur PC Windows ou sous Unix (codage latin-1). D'autres personnes risquent donc de ne pas pouvoir lire votre document source, parce que leur système utilise un codage différent mais ceci ne les empêchera cependant pas de le compiler, la sortie sera correcte !

Lorsqu'un texte source L^AT_EX doit être transmis à un autre site, il est recommandé de convertir les caractères accentués et spéciaux en séquences du type `\'e` préalablement au transfert. Il existe de nombreux utilitaires pour faire la conversion automatique dans les deux sens : `recode` sous Unix, `Tower of Babel` sur Macintosh, etc.

2.5. Support multilingue

Pour composer des documents dans des langues autres que l'anglais, \LaTeX doit pouvoir s'adapter aux règles typographiques et aux règles de césure propres à chaque langue. Il y a plusieurs domaines pour lesquels il faut configurer \LaTeX pour chaque langue :

1. Toutes les chaînes de caractères générées automatiquement⁴ doivent être traduites.
2. \LaTeX doit connaître les règles de césure de la nouvelle langue. Définir les règles de césure utilisées par \LaTeX est une tâche assez complexe, qui impose la construction de formats spécifiques.
3. Certaines règles typographiques changent en fonction de la langue ou de la région géographique.

Ces adaptations à la langue sont faites automatiquement par l'extension `babel` ou par des extensions spécifiques (telles que `french` pour le français, développée par Bernard Gaulle [13]). Il ne sera question, dans ce fascicule, que de la francisation faite par `babel` ; pour l'utilisation de `french`, on se reportera à [13].

Vous devez déclarer comme options de `babel` la liste de toutes les langues⁵ qui seront utilisées dans le document avec la commande :

```
\usepackage[langue1, ... ,languen]{babel}
```

après la commande `\documentclass`. La *dernière* langue de la liste sera la langue active au début du document. Les *langues* disponibles sur votre système sont affichées par `babel` au début du fichier `.log`, par exemple :

```
Babel <v3.7h> and hyphenation patterns for english, french,
german, loaded.
```

Pour le français, le nom de l'option à utiliser avec `babel` est `frenchb` ou `français` et non `french`⁶, par exemple :

```
\usepackage[english,frenchb]{babel}
```

Lorsqu'on utilise `babel`, la commutation de langue peut se faire par l'une des commandes suivantes :

4. « Table des matières », « Liste des figures »...

5. Y compris l'anglais bien que ce soit la langue « maternelle » de \LaTeX .

6. L'option `french` est ambiguë : elle peut charger, selon les installations, soit `frenchb`, soit l'extension `french` de Bernard Gaulle, soit une version allégée de celle-ci `frenchle` !

-
- `\selectlanguage{french}`⁷ *texte*
la suite du texte sera en français jusqu'à nouvel ordre ;
 - `\begin{otherlanguage}{german}`
texte
`\end{otherlanguage}`
pour passer *localement* en allemand et revenir à la langue initiale à la sortie de l'environnement ;
 - `\foreignlanguage{english}{texte}` pour une courte citation en anglais, dans un texte écrit dans une autre langue.

L'appel à l'extension `babel` n'est toutefois pas suffisant lorsqu'on compose dans des langues comme le français ou l'allemand, qui comportent des caractères accentués ou spéciaux (é, ä, ç, ß, etc.). En effet la possibilité de couper les mots contenant des accents en bout de ligne dépend également du *codage de sortie* utilisé, à ne pas confondre avec le *codage d'entrée*, défini par l'extension `inputenc`, dont il a été question page 23. Le codage de sortie se choisit grâce à l'extension `fontenc`.

La police originale de \TeX , Computer Modern, qui correspond au codage de sortie appelé `OT1`, ne contient que les 128 caractères du jeu de caractères ASCII. Pour produire un caractère accentué, \TeX combine un caractère normal avec un accent. Bien que le résultat obtenu ainsi soit (presque) parfait, cette approche empêche la coupure automatique des mots contenant des accents⁸.

Heureusement, la plupart des installations modernes de \TeX disposent de la police EC. Cette police (à 256 caractères, codage `T1`) a la même apparence que la police Computer Modern, mais elle contient également les caractères accentués pour la plupart des langues européennes, rendant ainsi possible la coupure des mots contenant des caractères accentués. L'extension `fontenc` permet de remplacer la police Computer Modern par la police EC, ceci se fait en ajoutant dans le préambule :

```
\usepackage[T1]{fontenc}
```

Il convient d'utiliser systématiquement cette commande lorsque le texte composé contient des caractères accentués ou spéciaux (voir figure 1.1 page 7).

Pour certaines langues `babel` définit également de nouvelles commandes qui simplifient la saisie des caractères spéciaux, les deux sous-sections suivantes donnent quelques indications pour le français et l'allemand.

7. Dans le cours du document, le français se sélectionne indifféremment par `french`, `frenchb`, ou `francais` (sans cédille).

8. Sauf si on dispose du moteur \TeX modifié `Mi \TeX` et que l'on utilise l'extension `mltex`.

2.5.1. Babel et le français

La table 2.3 présente les commandes introduites par le module `frenchb` de `babel`.

TABLE 2.3 – Commandes de saisie en français.

<code>\og guillemets \fg{}</code>	« guillemets »
<code>M\up{me}, D\up{r}</code>	M ^{me} , D ^r
<code>1\ier{} , 1\iere{} , 1\ieres{}</code>	1 ^{er} , 1 ^{re} , 1 ^{res}
<code>2\ieme{} , 4\iemes{}</code>	2 ^e 4 ^{es}
<code>\No 1, \no 2</code>	N ^o 1, n ^o 2
<code>20~\degres C, 45\degres</code>	20 °C, 45°
<code>\bsc{M. Durand}</code>	M. DURAND
<code>\nombre{1234,56789}</code>	1 234,567 89

La langue courante est déterminée au début du document par la dernière option de `babel` et ensuite par les commandes de changement de langue présentées page 25.

Le passage en français provoque la traduction des chaînes de caractères générées automatiquement (`\chapter` affiche « Chapitre » au lieu de « Chapter », `\today` affiche la date en français, etc.), impose les règles de césure françaises et modifie la mise en page des listes (voir page 32) pour la rendre plus conforme aux usages français⁹.

2.5.2. Babel et l'allemand

Lorsque `german` a été sélectionné dans les options de `babel`, le choix de l'allemand comme langue courante introduit un certain nombre de commandes (voir table 2.4) facilitant la saisie des caractères spécifiques à cette langue lorsqu'ils ne sont pas directement accessibles au clavier.

9. Il est toutefois possible de revenir aux réglages standards de L^AT_EX pour tout le document, en ajoutant la commande `\StandardLayout` dans le préambule, ou bien d'imposer une présentation à la française de toutes les listes, quelle que soit la langue, par `\FrenchLayout`.

TABLE 2.4 – Caractères spéciaux pour l’allemand.

"a	ä	"o	ö	"u	ü
"‘	”	"’	“		
"<	«	">	»		
\dq	"	"s	ß		

Bien sûr, comme c’est le cas pour le français, dans les portions de texte où l’allemand a été choisi comme langue courante, les césures allemandes sont activées et les chaînes de caractères générées par `\chapter`, `\today`, etc. sont traduites en allemand.

2.6. Espaces entre les mots

Pour obtenir une marge droite alignée, \LaTeX insère des espaces plus ou moins larges entre les mots.

Un tilde « `~` » produit une espace interdisant tout saut de ligne (dit espace *insécable*). `~` est à utiliser pour éviter les coupures indésirables : on code `M.~Dupont`, cf. `~Fig.~5`, etc.

La commande « `\,` » permet d’insérer une espace fine insécable (demie espace mot), utile pour améliorer la lisibilité des grands nombres.

`1\,234\,567` ou `1~234~567`

1 234 567 ou 1 234 567

Terminons par une subtilité de la typographie américaine : après la ponctuation finale d’une phrase, les règles de la typographie américaine veulent que l’on insère une espace plus large. Mais si un point suit une lettre majuscule, \LaTeX considère qu’il s’agit d’une abréviation et insère alors une espace normale. La commande `\@` avant un point indique que celui-ci termine une phrase, même lorsqu’il suit une lettre majuscule.

`I like PHP. Do you?\`
`I like PHP\@. Do you?`

I like PHP. Do you? I like PHP. Do you?
--

L'ajout d'espace supplémentaire à la fin d'une phrase peut être supprimé par la commande :

```
\frenchspacing
```

qui est active par défaut avec l'extension `babel/frenchb`. Dans ce cas, la commande `\@` n'est pas nécessaire.

2.7. Titres, chapitres et sections

Pour aider le lecteur à suivre votre pensée, vous souhaitez séparer vos documents en chapitres, sections ou sous-sections. \LaTeX utilise pour cela des commandes qui prennent en argument le titre de chaque élément. C'est à vous de les utiliser dans l'ordre.

Dans la classe de document `article`, les commandes de sectionnement suivantes sont disponibles :

```
\part{...}
\section{...}
\subsection{...}
\subsubsection{...}
\paragraph{...}
\subparagraph{...}
\appendix
```

Les classes `report` et `book` offrent un niveau supplémentaire de sectionnement, le chapitre : la commande

```
\chapter{...}
```

s'intercale entre `\part` et `\section`¹⁰.

L'espacement entre les sections, la numérotation et le choix de la police et de la taille des titres sont gérés automatiquement par \LaTeX .

Deux commandes de sectionnement ont un comportement spécial :

- la commande `\part` ne change pas la numérotation des chapitres ;
- la commande `\appendix` ne prend pas d'argument. Elle bascule simplement la numérotation des chapitres¹¹ en lettres.

10. Puisque la classe `article` ne connaît pas les chapitres, il est facile par exemple de regrouper des articles en tant que chapitres d'un livre en remplaçant le `\title` de chaque article par `\chapter`.

11. Pour la classe `article`, elle change la numérotation des sections.

L^AT_EX peut créer la table des matières en récupérant la liste des titres et de leur numéro de page d'une exécution précédente (fichier `.toc`). La commande :

```
\tableofcontents
```

imprime la table des matières. Un document doit être traité (on dit aussi « compilé ») deux fois par L^AT_EX pour avoir une table des matières correcte. Dans certains cas, un troisième passage est même nécessaire, L^AT_EX vous en informe le cas échéant.

Toutes les commandes citées ci-dessus existent dans une forme « étoilée » obtenue en ajoutant une étoile `*` au nom de la commande. Ces commandes produisent des titres de section qui n'apparaissent pas dans la table des matières et qui ne sont pas numérotés. On peut ainsi remplacer la commande `\section{Introduction}` par `\section*{Introduction}`.

Par défaut, les titres de section apparaissent dans la table des matières exactement comme ils sont dans le texte. Parfois il n'est pas possible de faire tenir un titre trop long dans la table des matières. On peut donner un titre spécifique pour la table des matières en argument optionnel avant le titre principal :

```
\chapter[Le LAAS du CNRS]{Le Laboratoire
d'Analyse et d'Architecture
des Systèmes du Centre National
de la Recherche Scientifique}
```

Le titre du document est obtenu par la commande :

```
\maketitle
```

Les éléments de ce titre sont définis par les commandes

```
\title{...}, \author{...} et éventuellement \date{...}
```

qui doivent être appelées avant `\maketitle`. Dans l'argument de la commande `\author`, vous pouvez citer plusieurs auteurs en séparant leurs noms par des commandes `\and`.

Vous trouverez un exemple des commandes citées ci-dessus en figure 1.2, page 7.

En plus des commandes de sectionnement expliquées ci-dessus, L^AT_EX 2_ε a introduit trois nouvelles commandes destinées à être utilisées avec la classe `book` :

`\frontmatter` doit être la première commande après le `\begin{document}`, elle introduit le préambule du document. Les numéros de pages sont alors en romain (i, ii, iii, etc.). En général on ne numérote pas les parties d'un

préambule, ceci se fait en utilisant les variantes étoilées des commandes de sectionnement comme `\chapter*{Preface}`.
`\mainmatter` se place juste avant le début du premier (vrai) chapitre du livre, la numérotation des pages se fait alors en chiffres arabes et le compteur de pages est remis à 1.
`\appendix` indique le début des appendices, les numéros des chapitres sont alors remplacés par des lettres majuscules (A, B, etc.).
`\backmatter` se place juste avant la bibliographie et les index.

2.8. Références croisées

Dans les livres, rapports ou articles, on trouve souvent des références croisées vers des figures, des tableaux ou des passages particuliers du texte. \LaTeX dispose des commandes suivantes pour faire des références croisées :

`\label{marque}, \ref{marque} et \pageref{marque}`

où *marque* est un identificateur choisi par l'utilisateur. \LaTeX remplace `\ref` par le numéro de la section, de la sous-section, de la figure, du tableau, ou du théorème où la commande `\label` correspondante a été placée. L'utilisation de références croisées rend nécessaire de compiler deux fois le document : à la première compilation les numéros correspondant aux étiquettes `\label{}` sont inscrits dans le fichier `.aux` et, à la compilation suivante, `\ref{}` et `\pageref{}` peuvent imprimer ces numéros.

Une référence à cette section `\label{ma-section}` ressemble à :
`\log voir section~\ref{ma-section},`
`page~\pageref{ma-section}.\fg`

Une référence à cette section ressemble à :
 « voir section 2.8, page 30. »

2.9. Notes de bas de page

La commande :

`\footnote{texte}`

imprime une note en bas de la page en cours.

Les notes de bas de page%
`\footnote{Ceci est une note`
`de bas de page.}`
 sont très prisées par les
 utilisateurs de `\LaTeX`.

Les notes de bas de page^a sont très pri-
 sées par les utilisateurs de `\LaTeX`.

a. Ceci est une note de bas de page.

2.10. Souligner l'importance d'un mot

Dans un manuscrit réalisé sur une machine à écrire, les mots importants sont soulignés, ce qui peut se faire en `\LaTeX` en utilisant la commande `\underline{texte}`. Dans un ouvrage imprimé, on préfère les *mettre en valeur*¹². La commande de mise en valeur est :

`\emph{texte}`

Son argument est le texte à mettre en valeur. En général, la police *italique* est utilisée pour la mise en valeur, sauf si le texte est déjà en italique, auquel cas `\LaTeX` utilise une police romaine (droite) comme le veulent les usages typographiques.

`\emph{Pour \emph{insister}`
`dans un passage déjà`
`mis en valeur, \LaTeX{}`
`utilise une police droite.}`

*Pour insister dans un passage déjà mis
 en valeur, \LaTeX utilise une police droite.*

2.11. Environnements

Pour composer du texte dans des contextes spécifiques, `\LaTeX` définit des environnements différents pour divers types de mise en page :

`\begin{nom} texte \end{nom}`

nom est le nom de l'environnement. Les environnements peuvent être emboîtés .

`\begin{aaa}... \begin{bbb}... \end{bbb}... \end{aaa}`

Dans les sections suivantes tous les environnements importants sont présentés.

^{12.} *Emphasize* en anglais.

2.11.1. Listes, énumérations et descriptions

L'environnement `itemize` est utilisé pour des listes simples, `enumerate` est utilisé pour des énumérations (listes numérotées) et `description` est utilisé pour des descriptions.

```
\begin{enumerate}
\item Il est possible d'emboîter
les environnements à sa guise :
\begin{itemize}
\item mais cela peut ne pas
être très beau,
\item ni facile à suivre.
\end{itemize}
\item Souvenez-vous :
\begin{description}
\item[clarté :] les faits ne
vont pas devenir plus sensés
parce qu'ils sont dans une liste,
\item[synthèse :] cependant une
liste peut très bien
résumer des faits.
\end{description}
\end{enumerate}
```

- | |
|---|
| <ol style="list-style-type: none"> 1. Il est possible d'emboîter les environnements à sa guise : <ul style="list-style-type: none"> • mais cela peut ne pas être très beau, • ni facile à suivre. 2. Souvenez-vous : <ul style="list-style-type: none"> clarté : les faits ne vont pas devenir plus sensés parce qu'ils sont dans une liste, synthèse : cependant une liste peut très bien résumer des faits. |
|---|

Notez que l'extension `babel/frenchb` adapte la présentation des listes simples aux usages français :

```
Une liste simple française :
\begin{itemize}
\item voici un élément ;
\item puis un autre.
\end{itemize}
```

<p>Une liste simple française :</p> <ul style="list-style-type: none"> – voici un élément ; – puis un autre.
--

```
An english list:
\begin{itemize}
\item one item
\item an other one
\end{itemize}
```

<p>An english list:</p> <ul style="list-style-type: none"> • one item • an other one
--

2.11.2. Alignements à gauche, à droite et centrage

Les environnements `flushleft` et `flushright` produisent des textes alignés à gauche ou à droite. L'environnement `center` produit un texte centré. Si vous

n'utilisez pas la commande `\` pour indiquer les sauts de ligne, ceux-ci sont calculés automatiquement par \LaTeX .

```
\begin{flushleft}
Ce texte est\
aligné à gauche. \LaTeX{
n'essaye pas d'aligner
la marge droite.
\end{flushleft}
```

Ce texte est aligné à gauche. \LaTeX n'essaye pas d'aligner la marge droite.

```
\begin{flushright}
Ce texte est\
aligné à droite. \LaTeX{
n'essaye pas d'aligner
la marge gauche.
\end{flushright}
```

Ce texte est aligné à droite. \LaTeX n'essaye pas d'aligner la marge gauche.

```
\begin{center}
Au centre de la terre.
\end{center}
```

Au centre de la terre.

2.11.3. Citations et vers

L'environnement `quote` est utile pour les citations, les phrases importantes ou les exemples.

```
Une règle typographique
simple pour la longueur
des lignes :
\begin{quote}
Une ligne ne devrait pas comporter
plus de 66~caractères.
```

```
C'est pourquoi les pages
composées par \LaTeX{ ont des
marges importantes.
\end{quote}
Cela explique pourquoi les
journaux utilisent souvent
plusieurs colonnes.
```

Une règle typographique simple pour la longueur des lignes :

Une ligne ne devrait pas comporter plus de 66 caractères.

C'est pourquoi les pages composées par \LaTeX ont des marges importantes.

Cela explique pourquoi les journaux utilisent souvent plusieurs colonnes.

Il existe deux autres environnements comparables : `quotation` et `verse`. L'environnement `quotation` est similaire à `quote` mais il ajoute un retrait en début

de chaque paragraphe. L'environnement `verse` est utilisé pour la poésie, là où les retours à la ligne sont importants. Les vers sont séparés par des commandes `\\` et les strophes par une ligne vide.

```
Voici le début d'une
fugue de Boris Vian :
\begin{flushleft}
\begin{verse}
Les gens qui n'ont plus
rien à faire\\
Se suivent dans la rue comme\\
Des wagons de chemin de fer.

Fer fer fer\\
Fer fer fer\\
Fer quoi faire\\
Fer coiffeur.\\
\end{verse}
\end{flushleft}
```

Voici le début d'une fugue de Boris Vian :

```
Les gens qui n'ont plus rien
à faire
Se suivent dans la rue
comme
Des wagons de chemin de
fer.

Fer fer fer
Fer fer fer
Fer quoi faire
Fer coiffeur.
```

2.11.4. Impression *verbatim*

Tout texte inclus entre `\begin{verbatim}` et `\end{verbatim}` est imprimé tel quel, comme s'il avait été tapé à la machine, avec tous les retours à la ligne et les espaces, les commandes `LATEX` ne sont pas exécutées mais leur nom est imprimé.

L'environnement `verbatim` produit toujours un nouveau paragraphe. Pour obtenir du texte *verbatim* à l'intérieur d'un paragraphe, on utilise la commande

```
\verb+texte+
```

Le caractère `+` est seulement un exemple de caractère séparateur. Vous pouvez utiliser n'importe quel caractère, sauf les lettres, `*` ou l'espace. La plupart des exemples de commandes `LATEX` dans ce document sont réalisés avec cette commande.

```
La commande \verb|\dots| \dots
\begin{verbatim}
10 PRINT "HELLO WORLD ";
20 GOTO 10
\end{verbatim}
```

La commande `\dots` ...

```
10 PRINT "HELLO WORLD ";
20 GOTO 10
```

```
\begin{verbatim*}
La version étoilée de
l'environnement verbatim
met les espaces en
évidence
\end{verbatim*}
```

```
La version étoilée de
l'environnement verbatim
met les espaces en
évidence
```

La commande `\verb` peut également être utilisée avec une étoile :

```
\verb*|comme ceci :-)|
```

```
comme ceci :-)|
```

L'environnement `verbatim` et la commande `\verb` ne peuvent être utilisés à l'intérieur d'autres commandes comme `\footnote{}`.

2.11.5. Tableaux

L'environnement `tabular` permet de réaliser des tableaux avec ou sans lignes de séparation horizontales ou verticales. \LaTeX ajuste automatiquement la largeur des colonnes.

L'argument *description du tableau* de la commande :

```
\begin{tabular}[pos]{description du tableau}
```

définit le format des colonnes du tableau. Utilisez un `l` pour une colonne alignée à gauche, `r` pour une colonne alignée à droite et `c` pour une colonne centrée. `p{largeur}` permet de réaliser une colonne de largeur donnée, justifiée à droite s'étendant éventuellement sur plusieurs lignes et enfin `|` permet d'obtenir un filet vertical. Le paramètre optionnel *pos* permet de préciser l'alignement du tableau par rapport à la ligne courante : `t`, `b` et `c` positionnent respectivement le haut (*top*), le bas (*bottom*) ou le milieu (*center*) du tableau sur la ligne courante.

À l'intérieur de l'environnement `tabular`, le caractère `&` est le séparateur de colonnes, `\\` commence une nouvelle ligne et `\hline` insère un filet horizontal.

```
\begin{tabular}{|r|l|}
\hline
7C0 & hexadécimal \\
3700 & octal \\
11111000000 & binaire \\
\hline
1984 & décimal \\
\hline
\end{tabular}
```

7C0	hexadécimal
3700	octal
11111000000	binaire
1984	décimal

```
\begin{tabular}{|p{4.7cm}|}
\hline
Bienvenue dans cette
case de 4,7~cm de large.\\
Merci de votre visite.\\
\hline
\end{tabular}
```

Bienvenue dans cette case de 4,7 cm de large. Merci de votre visite.
--

La construction `@{...}` permet d'imposer le séparateur de colonnes. Cette commande supprime l'espace inter-colonnes et le remplace par ce qui est indiqué entre les crochets. Une utilisation courante de cette commande est présentée plus loin comme solution au problème de l'alignement des nombres décimaux. Une autre utilisation classique consiste à supprimer l'espace inter-colonnes avec `@{}`.

```
\begin{tabular}{@{} 1 @{}}
\hline
sans espace\\
\hline
\end{tabular}
```

sans espace

```
\begin{tabular}{|1}
\hline
avec espaces\\
\hline
\end{tabular}
```

avec espaces

La commande `\multicolumn` qui prend trois arguments, permet de réaliser des lignes spéciales dans un tableau ; on peut l'utiliser pour placer un titre commun à plusieurs colonnes :

```
\begin{tabular}{|1|1|}
\hline
\multicolumn{2}{|c|}{\textbf{Nom}}\\
\hline
Dupont & Jules \\
\hline
\end{tabular}
```

Nom	
Dupont	Jules

S'il n'y a pas de commande prévue¹³ pour aligner les nombres sur le point décimal (ou la virgule si on respecte les règles françaises) nous pouvons « tricher » et réaliser cet alignement en utilisant deux colonnes : la première alignée

13. Si les extensions de l'ensemble `tools` sont installées sur votre système, jetez un œil sur l'extension `dcolumn` faite pour résoudre ce problème.

à droite contient la partie entière et la seconde alignée à gauche contient la partie décimale. La commande `\@{,}` dans la description du tableau remplace l'espace inter-colonnes par une simple virgule, donnant l'impression d'une seule colonne alignée sur le séparateur décimal. N'oubliez pas de remplacer dans votre tableau le point ou la virgule par un séparateur de colonnes (`&`)! Un titre peut être placé au-dessus de cette colonne en utilisant la commande `\multicolumn`.

```
\begin{tabular}{c r @{,} l}
expression & & \\
\multicolumn{2}{c}{Valeur} \\
\hline
 $\pi$  & 3,1416 & \\
 $\pi^\pi$  & 36,46 & \\
 $(\pi^\pi)^\pi$  & 80662,7 & \\
\end{tabular}
```

expression	Valeur
π	3,1416
π^π	36,46
$(\pi^\pi)^\pi$	80662,7

\LaTeX traite le contenu d'un environnement `tabular` comme une boîte indivisible, en particulier il ne peut y avoir de coupure de page. Pour réaliser de longs tableaux s'étendant sur plusieurs pages il faut avoir recours aux extensions `supertabular` ou `longtabular`.

2.12. Objets flottants

De nos jours, la plupart des publications contiennent un nombre important de figures et de tableaux. Ces éléments nécessitent un traitement particulier car ils ne peuvent être coupés lors d'un changement de page. Commencer une nouvelle page à chaque fois qu'une figure ou un tableau ne tient pas dans la page en cours laisserait de nombreuses pages à moitié vides, ce qui n'est pas acceptable.

La solution est de laisser « flotter » les figures et les tableaux qui ne tiennent pas dans la page en cours, vers une page suivante et de compléter la page avec le texte qui suit l'objet « flottant ». \LaTeX fournit deux environnements pour les objets flottants adaptés respectivement aux figures (`figure`) et aux tableaux (`table`). Pour faire le meilleur usage de ces deux environnements, il est important de comprendre comment \LaTeX traite ces objets flottants de manière interne; sinon, vous risquez de ne jamais parvenir à obtenir de \LaTeX qu'il les place à un endroit acceptable à vos yeux.

Commençons par regarder les commandes que \LaTeX propose pour les objets flottants : tout objet inclus dans un environnement `figure` ou `table` est traité

comme un objet flottant. Les deux environnements flottants ont un paramètre optionnel :

```
\begin{figure}[placement] ou \begin{table}[placement]
```

appelé *placement*. Ce paramètre permet de dire à \LaTeX où vous autorisez l'objet à flotter. Un *placement* est composé d'une chaîne de caractères représentant des *placements possibles*. Reportez-vous au tableau 2.5.

TABLE 2.5 – Placements possibles

Caractère	Emplacement pour l'objet flottant. . .
h	<i>here</i> , ici, à l'emplacement dans le texte où la commande se trouve. Utile pour les petits objets.
t	<i>top</i> , en haut d'une page
b	<i>bottom</i> , en bas d'une page
p	<i>page</i> , sur une page à part ne contenant que des objets flottants.
!	ici, sans prendre en compte les paramètres internes ^a qui pourraient empêcher ce placement.

a. Tels que le nombre maximum d'objets flottants sur une page

Un tableau peut commencer par exemple par la ligne suivante :

```
\begin{table}[!htp]
```

L'emplacement `[!htp]` permet à \LaTeX de placer le tableau soit sur place (**h**), soit en haut de la page suivante (**t**), soit enfin sur une page à part (**p**) et tout cela même si les règles internes de \LaTeX ne sont pas toutes respectées (**!**). Si aucun placement n'est indiqué, les classes standard utilisent `[tbp]` par défaut. \LaTeX essaie les options proposées dans l'ordre **h**, **t**, **b**, **p**, quel que soit l'ordre choisi par l'auteur du document, `[!hb]` est équivalent à `[!bh]`.

\LaTeX place tous les objets flottants qu'il rencontre dans l'ordre, en suivant les indications fournies par l'auteur. Si un objet ne peut être placé sur la page en cours, il est placé soit dans la file des figures soit dans la file des tableaux¹⁴. Quand une nouvelle page est entamée, \LaTeX essaie d'abord de voir si les objets en tête des deux files pourraient être placés sur une page spéciale, à part. Si cela n'est pas possible, les objets en tête des deux files sont traités comme s'ils venaient d'être trouvés dans le texte : \LaTeX essaie de les mettre aux emplacements encore disponibles. Tous les nouveaux objets flottants rencontrés dans la suite du texte sont ajoutés à la queue des files. \LaTeX respecte scrupuleusement

14. Il s'agit de files FIFO (*First In, First Out*) : premier arrivé, premier servi.

l'ordre d'apparition des objets flottants. C'est pourquoi un objet flottant qui ne peut être placé dans le texte repousse également tous les suivants.

D'où la règle :

Si \LaTeX ne place pas les objets flottants comme vous le souhaitez, c'est souvent à cause d'un seul objet trop grand qui bouche l'une des deux files d'objets flottants.

Essayer d'imposer à \LaTeX un emplacement particulier en utilisant l'option `[h]` pour un flottant est une idée à *proscrire*, les versions modernes de \LaTeX changent d'ailleurs automatiquement l'option `[h]` en `[ht]`.

Voici quelques éléments supplémentaires qu'il est bon de connaître sur les environnements `table` et `figure`.

Avec la commande :

```
\caption{texte de la légende}
```

vous définissez une légende pour l'objet. Un numéro (incrémenté automatiquement) et le mot « Figure » ou « Table » sont ajoutés par \LaTeX .

Les deux commandes :

```
\listoffigures et \listoftables
```

fonctionnent de la même manière que la commande `\tableofcontents` ; elles impriment respectivement la liste des figures et des tableaux. Dans ces listes, la légende est reprise en entier. Si vous désirez utiliser des légendes longues, vous pouvez en donner une version courte entre crochets qui sera utilisée pour la table :

```
\caption[courte]{LLLLLooooooooonnnnnngggggguuuueee}
```

Avec `\label` et `\ref` vous pouvez faire référence à votre objet à l'intérieur de votre texte.

L'exemple suivant dessine un carré centré horizontalement¹⁵ et l'insère dans le document. Vous pouvez utiliser cette commande pour réserver de la place pour une illustration que vous allez coller sur le document terminé.

```
La figure~\ref{blanche} est un exemple de Pop-Art.
\begin{figure}[!hbp]
\centering
\framebox[5cm]{\rule{0cm}{5cm}}
\caption{Cinq centimètres sur cinq} \label{blanche}
\end{figure}
```

15. Les commandes `\framebox` et `\rule` seront présentées respectivement pages 102 et 103.

Dans l'exemple ci-dessus¹⁶ \LaTeX va s'acharner (!) à placer la figure là où se trouve la commande (**h**) dans le texte. S'il n'y arrive pas, il essaiera de la placer en bas (**b**) de la page. Enfin s'il ne peut la placer sur la page courante, il essaiera de créer une page à part avec d'autres objets flottants. S'il n'y a pas suffisamment de tableaux en attente pour remplir une page spécifique, \LaTeX continue et, au début de la page suivante, réessayera de placer la figure comme si elle venait d'apparaître dans le texte.

Dans certains cas il peut s'avérer nécessaire d'utiliser la commande :

`\clearpage` ou même `\cleardoublepage`

Elle ordonne à \LaTeX de placer tous les objets en attente immédiatement puis de commencer une nouvelle page. `\cleardoublepage` commence une nouvelle page de droite.

Vous apprendrez à la section 4.1 comment inclure une figure `POSTSCRIPT` dans vos documents.

16. En supposant que la file des figures soit vide.

Chapitre 3

Formules mathématiques

Vous êtes prêts ! Dans ce chapitre nous allons aborder l'atout majeur de $\text{T}_{\text{E}}\text{X}$: la composition de formules mathématiques. De nombreuses fonctionnalités présentées dans ce chapitre font partie de l'extension $\text{AMS-}\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ développée par l'AMS¹. Il est vivement recommandé de charger systématiquement cette extension pour composer des mathématiques, ceci se fait en ajoutant dans le préambule `\usepackage{amsmath}`. Mais attention, ce chapitre ne fait que décrire les commandes de base. Ce qui est expliqué ici devrait être suffisant pour la majorité des utilisateurs ; toutefois, si vous n'y trouvez pas de réponse à votre problème de mise en forme d'une équation mathématique complexe, consultez la documentation de l'extension `amsmath` (voir [11]), les nombreux environnements proposés (`split`, `multline`, etc.) devraient vous fournir une solution.

3.1. Généralités

$\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ dispose de deux modes spécifiques pour la mise en page de formules mathématiques. À l'intérieur d'un paragraphe, les formules sont saisies entre `\(` et `\)`, entre `$` et `$` ou entre `\begin{math}` et `\end{math}`.

Ajoutez `a` au carré et `b` au carré pour obtenir `c` au carré. Ou, en utilisant une approche plus matheuse : `$c^2=a^2+b^2$`

Ajoutez a au carré et b au carré pour obtenir c au carré. Ou, en utilisant une approche plus matheuse : $c^2 = a^2 + b^2$

`100~m3` d'eau.
`J'\heartsuit \LaTeX !`

100 m³ d'eau. J'♥ $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$!

Il vaut mieux composer les équations ou les formules plus importantes « *hors-texte* », c'est-à-dire sur des lignes à part. Pour cela, on les inclut entre `\[` et `\]` ou entre `\begin{displaymath}` et `\end{displaymath}`². On obtient

1. *American Mathematical Society* = Société américaine de mathématiques.

2. L'utilisation de `$$... $$` pour encadrer les formules hors-texte est à proscrire en $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$.

ainsi des formules non numérotées. Si vous voulez qu'elles soient numérotées automatiquement par L^AT_EX, utilisez l'environnement `equation`.

Ajoutez `a` au carré
et `b` au carré pour obtenir
`c` au carré. Ou, en
utilisant une approche plus
matheuse :
`\[c^2=a^2+b^2. \]`
Le mot de la fin.

Ajoutez a au carré et b au carré pour obtenir c au carré. Ou, en utilisant une approche plus matheuse :

$$c^2 = a^2 + b^2.$$

Le mot de la fin.

Avec `\label` et `\ref` (ou `\eqref` de `amsmath` pour ajouter les parenthèses automatiquement), vous pouvez faire référence à une équation.

`\begin{equation} \label{eq:eps}`
`\epsilon > 0`
`\end{equation}`
L'équation[^](`\ref{eq:eps}`)
nous donne\dots{ } \\
L'équation[^](`\eqref{eq:eps}`) aussi.

$$\epsilon > 0 \quad (3.1)$$

L'équation (3.1) nous donne...
L'équation (3.1) aussi.

Remarquez que les expressions mathématiques sont formatées différemment selon qu'elles sont composées « en ligne » ou « hors texte ». Comparez

`\lim_{n \to \infty}`
`\sum_{k=1}^n \frac{1}{k^2}`
`= \frac{\pi^2}{6}`

$$\lim_{n \rightarrow \infty} \sum_{k=1}^n \frac{1}{k^2} = \frac{\pi^2}{6}$$

et

`\begin{displaymath}`
`\lim_{n \to \infty}`
`\sum_{k=1}^n \frac{1}{k^2}`
`= \frac{\pi^2}{6}`
`\end{displaymath}`

$$\lim_{n \rightarrow \infty} \sum_{k=1}^n \frac{1}{k^2} = \frac{\pi^2}{6}$$

Il y a des différences entre le mode *mathématique* et le mode *texte*. Par exemple, en mode *mathématique* :

1. La plupart des espaces et des retours à la ligne n'ont aucun effet³. La taille des espaces est fixée automatiquement en fonction de règles internes ou spécifiée à l'aide de commandes spécifiques telles que `\,`, `\quad` ou `\qquad`.

3. Dans l'exemple ci-dessous, l'espace après `\forall` est indispensable, mais les codages `$ab+c=d$` et `$a b + c = d$` donnent le même résultat : $ab + c = d$.

2. La notion de paragraphe n'existe pas.
3. Chaque lettre est considérée comme un nom de variable et traitée comme telle. Pour insérer du texte normal (police et espacement standard) dans une formule, il faut utiliser la commande `\textrm{...}` (voir aussi section 3.7 page 51).

```
\begin{equation}
\forall x \in \mathbf{R}:
\quad x^2 \geq 0
\end{equation}
```

$$\forall x \in \mathbf{R} : \quad x^2 \geq 0 \quad (3.2)$$

```
\begin{equation}
x^2 \geq 0 \quad \text{pour tout }
x \in \mathbf{R}
\end{equation}
```

$$x^2 \geq 0 \quad \text{pour tout } x \in \mathbf{R} \quad (3.3)$$

Une mode récente et contestable pousse à utiliser, pour désigner les ensembles de nombres (entiers, réels, etc.), la police « blackboard bold » (littéralement « gras de tableau noir ») qui simule le gras par le doublement des barres verticales ; on y accède par la commande `\mathbb` de l'extension `amsmath`. L'exemple précédent devient :

```
\begin{displaymath}
x^2 \geq 0 \quad \text{pour tout } x \in \mathbb{R}
\end{displaymath}
```

$$x^2 \geq 0 \quad \text{pour tout } x \in \mathbb{R}$$

3.2. Groupements en mode mathématique

La plupart des commandes du mode mathématique ne s'appliquent qu'au caractère suivant. Pour qu'une commande s'applique à un ensemble de caractères, il faut les grouper en utilisant des accolades : `{...}`.

```
\begin{equation}
a^x + y \neq a^{x+y}
\end{equation}
```

$$a^x + y \neq a^{x+y} \quad (3.4)$$

3.3. Éléments d'une formule mathématique

Dans cette section nous allons découvrir les commandes les plus importantes du mode mathématique. Une liste de tous les symboles disponibles, est donnée à la section 3.11, page 55.

Les lettres **grecques minuscules** sont saisies de la manière suivante : `\alpha`, `\beta`, `\gamma`, etc. Les lettres **grecques majuscules**⁴ sont saisies ainsi : `\Gamma`, `\Delta`, etc.

`\lambda, \xi, \pi, \mu, \Phi, \Omega`

$\lambda, \xi, \pi, \mu, \Phi, \Omega$

Les **indices et exposants** sont obtenus en utilisant les caractères `_` et `^`.

`a_{1} \quad x^2 \quad`
`$e^{-\alpha t}$ \quad`
`a^3_{ij} \quad`
`$e^{x^2} \neq (e^x)^2$`

$a_1 \quad x^2 \quad e^{-\alpha t} \quad a_{ij}^3$
 $e^{x^2} \neq (e^x)^2$

La **racine carrée** est produite par la commande `\sqrt` qui admet un argument optionnel utilisé pour les racines n -ièmes : `\sqrt[n]`. La taille du symbole racine est calculée par L^AT_EX. Pour obtenir le symbole seul, utilisez `\surd`.

`\sqrt{x} \quad`
`$\sqrt{x^2+\sqrt{y}}$ \quad`
`\quad $\sqrt[3]{2}$ \quad`
`$\surd[x^2 + y^2]$`

$\sqrt{x} \quad \sqrt{x^2 + \sqrt{y}} \quad \sqrt[3]{2}$
 $\surd[x^2 + y^2]$

Un trait de **fraction** est produit par la commande :

`\frac{numérateur}{dénominateur}`

La forme utilisant une barre oblique ($1/2$) est souvent préférable pour les petits éléments.

`$1\frac{1}{2}$ \quad`
`\begin{displaymath}`
`\frac{x^2}{k+1}`
`x^{\frac{2}{k+1}}`
`x^{1/2}`
`\end{displaymath}`

$1\frac{1}{2}$ hours
 $\frac{x^2}{k+1} \quad x^{\frac{2}{k+1}} \quad x^{1/2}$

4. Il n'y a pas de Alpha majuscule dans L^AT_EX 2_ε parce que c'est le même caractère que le A romain. Lorsque le nouveau codage mathématique sera en place, cela changera.

Les commandes `\overline` et `\underline` ajoutent un **trait horizontal** au-dessus ou au-dessous d'une expression.

`\overline{m+n}`

$$\overline{m+n}$$

Les commandes `\overbrace` et `\underbrace` ajoutent une grande **accolade horizontale** au-dessus ou au-dessous d'une expression.

`\underbrace{ a+b+\cdots+z }_{26}`

$$\underbrace{a+b+\cdots+z}_{26}$$

Pour saisir les accents mathématiques tels que les flèches ou les tildes, vous pouvez utiliser les commandes du tableau 3.1 page 55. Les chapeaux et les tildes larges, couvrant plusieurs caractères, s'obtiennent par les commandes `\widetilde` et `\widehat`. L'accent ' produit un prime.

`\begin{displaymath}`
`y=x^2\quad y'=2x\quad y''=2`
`\end{displaymath}`

$$y = x^2 \quad y' = 2x \quad y'' = 2$$

Les **vecteurs** se distinguent en général⁵ par une flèche ajoutée au-dessus de leur nom. La flèche s'obtient par la commande `\vec`. Les commandes `\overrightarrow` et `\overleftarrow` produisent des flèches de longueur adaptée à leur argument, elles sont utilisées pour coder le vecteur AB par exemple.

`\begin{displaymath}`
`\vec a\quad\overrightarrow{AB}`
`\end{displaymath}`

$$\vec{a} \quad \overrightarrow{AB}$$

Habituellement, on omet le signe de multiplication. Lorsque sa présence facilite la lecture d'une formule, on utilise l'une des commandes `\cdot` ou `\times`, à ne pas confondre avec le point ou la lettre x .

`\begin{displaymath}`
`v = {\sigma}_1 \cdot {\sigma}_2`
`{\tau}_1 \cdot {\tau}_2`
`\text{ sur } E \times F.`
`\end{displaymath}`

$$v = \sigma_1 \cdot \sigma_2 \tau_1 \cdot \tau_2 \text{ sur } E \times F.$$

5. Certains auteurs préfèrent les noter en caractères gras, voir page 54.

Les noms des fonctions telles que sinus doivent être imprimés en caractères romains et non en italique comme les variables. Le codage `\sin x` produirait « *sinx* » ! L^AT_EX fournit donc les commandes suivantes pour les fonctions les plus utilisées :

<code>\arccos</code>	<code>\cos</code>	<code>\csc</code>	<code>\exp</code>	<code>\ker</code>	<code>\limsup</code>	<code>\min</code>
<code>\arcsin</code>	<code>\cosh</code>	<code>\deg</code>	<code>\gcd</code>	<code>\lg</code>	<code>\ln</code>	<code>\Pr</code>
<code>\arctan</code>	<code>\cot</code>	<code>\det</code>	<code>\hom</code>	<code>\lim</code>	<code>\log</code>	<code>\sec</code>
<code>\arg</code>	<code>\coth</code>	<code>\dim</code>	<code>\inf</code>	<code>\liminf</code>	<code>\max</code>	<code>\sin</code>
<code>\sinh</code>	<code>\sup</code>	<code>\tan</code>	<code>\tanh</code>			

```
\[\lim_{x \rightarrow 0}
\frac{\sin x}{x}=1\]
```

$$\lim_{x \rightarrow 0} \frac{\sin x}{x} = 1$$

Pour la fonction modulo, il y a deux commandes possibles : `\bmod` pour l'opérateur binaire et `\pmod` pour l'opérateur unaire :

```
$a\bmod b$\
$x\equiv a \pmod{b}$
```

$$a \bmod b$$

$$x \equiv a \pmod{b}$$

Pour imprimer des coefficients binomiaux (à l'américaine) ou d'autres structures semblables, on peut utiliser la commande `\binom` de l'extension `amsmath` :

```
\begin{displaymath}
\binom{n}{k}\quad \mathrm{C}_n^k
\end{displaymath}
```

$$\binom{n}{k} \quad C_n^k$$

La commande `\stackrel` permet de superposer des symboles, elle place son premier argument en taille réduite au dessus du second :

```
\begin{displaymath}
X \stackrel{!}{=} 1
\end{displaymath}
```

$$X \stackrel{!}{=} 1$$

Les **intégrales** sont produites par la commande `\int`, les **sommes** par la commande `\sum`, les produits par la commande `\prod`. Les limites inférieures et supérieures sont obtenues grâce à `_` et `^` comme les indices et exposants.

```

\begin{displaymath}
\sum_{i=1}^n \quad \quad \quad
\int_0^{\frac{\pi}{2}} \quad \quad \quad
\prod_k
\end{displaymath}

```

$$\sum_{i=1}^n \quad \int_0^{\frac{\pi}{2}} \quad \prod_k$$

Pour superposer des indices, l'extension `amsmath` propose la commande `\substack` et l'environnement `subarray` qui permet d'aligner les indices à gauche au lieu de les centrer.

```

\begin{displaymath}
\sum_{\substack{0 < i < n \\ 1 < j < m}}
P(i, j) =
\sum_{\begin{subarray}{l} i \in I \\ 1 < j < m \end{subarray}}
Q(i, j)
\end{displaymath}

```

$$\sum_{\substack{0 < i < n \\ 1 < j < m}} P(i, j) = \sum_{\substack{i \in I \\ 1 < j < m}} Q(i, j)$$

Pour les **crochets et les autres délimiteurs**, il existe toutes sortes de symboles en `TEX`, par exemple [< || ↓. Les parenthèses et les crochets sont obtenus avec les caractères correspondants, les accolades avec `\{` et tous les autres délimiteurs sont obtenus par des commandes spéciales (par exemple `\updownarrow`). Le tableau 3.8, page 57 dresse la liste de tous les délimiteurs disponibles.

```

\begin{displaymath}
\{a, b, c\} \neq \{a, b, c\}
\end{displaymath}

```

$$a, b, c \neq \{a, b, c\}$$

Si vous ajoutez la commande `\left` avant un délimiteur ouvrant ou `\right` avant un délimiteur fermant, `TEX` adapte automatiquement la taille des délimiteurs à leur contenu. Remarquez qu'il est nécessaire de fermer chaque délimiteur ouvrant (`\left`) avec un délimiteur fermant (`\right`). Si vous ne voulez pas de délimiteur fermant, la commande `\right.` produit un délimiteur invisible!

```

\begin{displaymath}
1 + \left( \frac{1}{1-x^2} \right)^3
\end{displaymath}

```

$$1 + \left(\frac{1}{1-x^2} \right)^3$$

Remarquez que le ‘d’ de l’élément différentiel est traditionnellement imprimé en caractère romain par la commande `\ud`.

AMS- \LaTeX propose un ensemble de commandes pour ajuster finement l’espace entre les signes intégrale : `\iint`, `\iiint`, `\iiiint` et `\idotsint`. Avec l’extension `amsmath`, l’exemple précédent peut se coder de cette manière :

```
\newcommand{\ud}{\mathrm{d}}
\begin{displaymath}
\iint_{D} \, \, \ud x \, \, \ud y
\end{displaymath}
```

$$\iint_D dx dy$$

Reportez-vous au document `testmath.tex` distribué avec AMS- \LaTeX ou au chapitre 9 de [4] pour plus de détails.

3.5. Alignements verticaux

Pour imprimer des **matrices**, utilisez l’environnement `array`. Il fonctionne de manière similaire à l’environnement `tabular`. La commande `\` est utilisée pour séparer les lignes et `&` pour séparer les colonnes.

```
\begin{displaymath}
\mathbf{X} =
\left( \begin{array}{ccc}
x_{11} & x_{12} & \dots \\
x_{21} & x_{22} & \dots \\
\vdots & \vdots & \ddots
\end{array} \right)
\end{displaymath}
```

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \dots \\ x_{21} & x_{22} & \dots \\ \vdots & \vdots & \ddots \end{pmatrix}$$

L’environnement `array` peut également être utilisé pour imprimer des expressions qui ont un délimiteur invisible obtenu par la commande `\right.` :

```
\begin{displaymath}
y = \left\{ \begin{array}{ll}
a & \text{\texttrm{si } $d > c$,} \\
b+x & \text{\texttrm{le matin,}} \\
l & \text{\texttrm{la journée.}}
\end{array} \right.
\end{displaymath}
```

$$y = \begin{cases} a & \text{si } d > c, \\ b+x & \text{le matin,} \\ l & \text{la journée.} \end{cases}$$

L’environnement `array` permet, comme l’environnement `tabular`, d’insérer des filets horizontaux ou verticaux :

```

\begin{displaymath}
\left(\begin{array}{c|c}
1 & 2 \\ \hline
3 & 4
\end{array}\right)
\end{displaymath}

```

$$\left(\begin{array}{c|c} 1 & 2 \\ \hline 3 & 4 \end{array}\right)$$

Pour les systèmes d'équations ou les formules qui prennent plusieurs lignes, l'extension `amsmath` propose les environnements `align`, `multline` et `split`.

L'environnement `align` et sa variante `align*` permettent d'aligner verticalement plusieurs équations. Avec `align` chaque ligne est numérotée, alors que la variante `align*` ne produit aucun numéro. L'alignement se fait sur le caractère suivant le `&`.

```

\begin{align}
f(x) &= \cos x & \\
f'(x) &= -\sin x & \\
\int_0^x f(y) dy &= \sin x & \notag
\end{align}

```

$$\begin{aligned}
f(x) &= \cos x & (3.5) \\
f'(x) &= -\sin x & (3.6) \\
\int_0^x f(y) dy &= \sin x
\end{aligned}$$

Si on souhaite numéroté certaines lignes du système et pas d'autres, on utilise `align` et on ajoute `\notag` à la fin de chaque ligne à ne pas numéroté (voir ci-dessus).

Les **équations longues** ne sont pas découpées automatiquement en morceaux harmonieux. L'auteur doit indiquer où les couper et comment décaler la suite. Avec l'environnement `multline` ou sa variante non numérotée `multline*`, on indique uniquement les coupures de lignes par `\\`, la première ligne est poussée à gauche, la dernière est poussée à droite, les autres sont en général centrées.

```

\begin{multline}
\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \\
+ \frac{x^7}{7!} + \dots + \\
+ \frac{(-x)^{2n+1}}{(2n+1)!} \\
+ \dots
\end{multline}

```

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots + \frac{(-x)^{2n+1}}{(2n+1)!} + \dots \quad (3.7)$$

L'environnement `split` permet de contrôler les alignements verticaux grâce à des tabulations `&` (une seule par ligne), mais il ne peut être utilisé, contraire-

ment à `align` et `multline`, qu'à l'intérieur d'un environnement mathématique comme `equation` (numéroté), `displaymath` (non numéroté) ou `align` (avec ou sans `*`) :

```
\begin{equation}
\begin{split}
\sin x = x & - \frac{x^3}{3!} + \frac{x^5}{5!} - \\
& + \frac{x^7}{7!} + \cdots + \\
& - \frac{(-x)^{2n+1}}{(2n+1)!} \\
& + \cdots
\end{split}
\end{equation}
```

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \cdots + \frac{(-x)^{2n+1}}{(2n+1)!} + \cdots \quad (3.8)$$

Ces exemples ne présentent qu'une toute petite partie des possibilités offertes par l'extension `amsmath` en matière de présentation d'équations; pour plus d'information consultez [11], [4] ou encore [3].

3.6. Fantômes...

Les fantômes sont invisibles mais permettent des ajustements intéressants en \LaTeX .

Il arrive que \LaTeX en fasse un peu trop dans des alignements verticaux d'indices ou d'exposants. La commande `\phantom` réserve de l'espace pour des caractères invisibles et permet ainsi un positionnement fin des éléments d'une formule :

```
\begin{displaymath}
{}^{12}_{6}C_{\phantom{1}6}\text{trm}{C} \\
\quad \quad \quad \text{à comparer à} \quad \quad \quad \\
{}^{12}_{6}C_{6}\text{trm}{C}
\end{displaymath}
```

$${}^{12}_6C \quad \text{à comparer à} \quad {}^{12}_6C$$

3.7. Taille des polices mathématiques

En mode mathématique \TeX choisit la taille de la police en fonction du contexte. Les exposants, par exemple, sont imprimés avec une police plus petite.

Malgré tout, il peut être nécessaire d'indiquer à \LaTeX la taille exacte. En mode mathématique, la taille de la police est déterminée par les quatre commandes :

`\displaystyle (123)`, `\textstyle (123)`, `\scriptstyle (123)` and `\scriptscriptstyle (123)`.

Le choix du style influe également sur la façon dont sont affichés les indices et les exposants des sommes et des limites.

```
\[\mathrm{c}(X,Y)=
\frac{\displaystyle
\sum_{i=1}^n(x_i-\overline{x})
(y_i-\overline{y})}
{\displaystyle\biggl[
\sum_{i=1}^n(x_i-\overline{x})^2
\sum_{i=1}^n(y_i-\overline{y})^2
\biggr]^{1/2}}
\]
```

$$c(X, Y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\left[\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (y_i - \bar{y})^2 \right]^{1/2}}$$

Dans cet exemple, on a besoin de crochets plus petits que ceux qui seraient produits par les commandes standard `\left[` et `\right]`.

3.8. Insertion de texte en mode mathématique

La commande `\mathrm` permet d'insérer du texte en caractères romains tout en restant en mode *mathématique*. La taille du texte est ajustée automatiquement, mais on est limité à un mot et les accents sont interdits. La commande `\textrm` qui fait passer en mode *texte* permet l'utilisation des lettres accentuées et respecte les espaces. L'extension `amsmath` fournit une commande `\text` également très pratique pour insérer du texte dans une équation.

Dans l'exemple suivant, le texte entourant le système étant penché, le contenu de `\textrm` et `\text` l'est aussi, mais le contenu de `\mathrm` reste en romain.

```
\slshape
\begin{align*}
\mathrm{d}x &= \mathrm{d}y \\
\textrm{sur } U, \\
f(x) &= 0 \text{ si } x \in V. \\
\end{align*}
```

$$\begin{aligned} dx &= dy \text{ sur } U, \\ f(x) &= 0 \text{ si } x \in V. \end{aligned}$$

3.9. Théorèmes, propositions, etc.

Dans les documents mathématiques, les énoncés des lemmes, des définitions, des axiomes et d'autres structures similaires nécessitent une présentation adaptée. L^AT_EX propose à cet effet la commande :

```
\newtheorem{nom}[compteur]{texte}[section]
```

L'argument *nom* est un mot-clef utilisé pour identifier le type d'énoncé. L'argument *texte* définit le nom réel qui sera utilisé pour l'énoncé.

Les arguments entre crochets sont optionnels. Ils servent à indiquer la numérotation à utiliser pour l'énoncé. Avec *compteur* vous indiquez le *nom* d'un énoncé déjà déclaré. Le nouvel énoncé sera alors numéroté avec le même compteur. Avec *section* vous indiquez quel niveau de sectionnement remettra à zéro le compteur utilisé par votre énoncé, ce qui permet de numérotter les énoncés par chapitre, par section, etc.

Après avoir exécuté `\newtheorem` dans le préambule de votre document, vous pourrez utiliser la commande suivante :

```
\begin{nom}[texte]
Ceci est mon premier théorème
\end{nom}
```

Voilà pour la théorie. Les exemples qui suivent devraient faire mieux comprendre l'utilisation de la commande `\newtheorem`.

```
% définitions dans le préambule
\newtheorem{loi}{Loi}
\newtheorem{decret}[loi]{Décret}
%dans le document
\begin{loi} \label{chef}
Le chef a raison.
\end{loi}
\begin{decret}[important]
Le chef a toujours raison.
\end{decret}
\begin{loi}
Si le chef a tort, voir
la loi~\ref{chef}.
\end{loi}
```

Loi 1 *Le chef a raison.*

Décret 2 (important) *Le chef a toujours raison.*

Loi 3 *Si le chef a tort, voir la loi 1.*

L'énoncé « décret » a été défini pour utiliser le même compteur que le l'énoncé « loi ». Il apparaît après la « loi » portant le numéro 1, il porte donc le numéro

suisant, soit 2. L'argument entre crochets de l'environnement « décret » permet de spécifier un titre pour les énoncés de ce type.

```
\newtheorem{prop}{Murphy}[section]
\begin{prop} Tout ce qui peut
aller mal, ira mal.\end{prop}
```

Murphy 3.9.1 *Tout ce qui peut aller mal, ira mal.*

Dans cet exemple, les énoncés de type `prop` sont numérotés par section. On aurait pu utiliser un autre niveau tel que `chapter` ou `subsection`.

3.10. Symboles gras

Il est relativement difficile d'obtenir des symboles gras avec \LaTeX ; cela est sans doute fait exprès car les typographes amateurs ont tendance à abuser du gras. La commande `\mathbf` permet d'obtenir des caractères gras, mais ce sont des caractères romains (droits), alors que les caractères mathématiques sont normalement en italique.

Les commandes `\mathversion{bold}` et `\mathversion{normal}`, utilisables *en mode texte uniquement*, permettent de changer le style par défaut utilisé par le mode mathématique.

```
\mathversion{bold}
\begin{displaymath}
\mu, M
\end{displaymath}
\mathversion{normal}
\begin{displaymath}
\mu, M \quad
\mathbf{\mu}, \mathbf{M}
\end{displaymath}
```

μ, M

$\mu, M \quad \mu, M$

Notez que dans la première ligne, la virgule est également en gras, ce qui n'est peut-être pas souhaité.

L'extension `amsbsy` (chargée automatiquement par `amsmath`), ainsi que l'extension `bm` permettent d'utiliser le gras seulement pour certains éléments d'une formule. Elles fournissent pour cela la commande `\boldsymbol`.

```
\begin{displaymath}
\mu, M \quad
\boldsymbol{\mu}, \boldsymbol{M}
\end{displaymath}
```

$\mu, M \quad \mu, M$

3.11. Liste des symboles mathématiques

Dans les tableaux suivants, vous trouverez tous les symboles accessibles en mode *mathématique*.

Pour utiliser des symboles présents dans les tables 3.12 à 3.16⁷, l'extension `amssymb` doit être chargée dans le préambule du document et les polices mathématiques de l'AMS doivent être présentes sur votre système. Si les extensions et les polices de l'AMS ne sont pas installées sur votre système, vous pouvez les récupérer sur `CTAN:/macros/latex/required/amslatex`

TABLE 3.1 – Accents en mode mathématique

\hat{a}	<code>\hat{a}</code>	\check{a}	<code>\check{a}</code>	\tilde{a}	<code>\tilde{a}</code>	\acute{a}	<code>\acute{a}</code>
\grave{a}	<code>\grave{a}</code>	\dot{a}	<code>\dot{a}</code>	\ddot{a}	<code>\ddot{a}</code>	\breve{a}	<code>\breve{a}</code>
\bar{a}	<code>\bar{a}</code>	\vec{a}	<code>\vec{a}</code>	\widehat{A}	<code>\widehat{A}</code>	\widetilde{A}	<code>\widetilde{A}</code>

TABLE 3.2 – Alphabet grec minuscule

α	<code>\alpha</code>	θ	<code>\theta</code>	o	<code>o</code>	v	<code>\upsilon</code>
β	<code>\beta</code>	ϑ	<code>\vartheta</code>	π	<code>\pi</code>	ϕ	<code>\phi</code>
γ	<code>\gamma</code>	ι	<code>\iota</code>	ϖ	<code>\varpi</code>	φ	<code>\varphi</code>
δ	<code>\delta</code>	κ	<code>\kappa</code>	ρ	<code>\rho</code>	χ	<code>\chi</code>
ϵ	<code>\epsilon</code>	λ	<code>\lambda</code>	ϱ	<code>\varrho</code>	ψ	<code>\psi</code>
ε	<code>\varepsilon</code>	μ	<code>\mu</code>	σ	<code>\sigma</code>	ω	<code>\omega</code>
ζ	<code>\zeta</code>	ν	<code>\nu</code>	ς	<code>\varsigma</code>		
η	<code>\eta</code>	ξ	<code>\xi</code>	τ	<code>\tau</code>		

TABLE 3.3 – Alphabet grec majuscule

Γ	<code>\Gamma</code>	Λ	<code>\Lambda</code>	Σ	<code>\Sigma</code>	Ψ	<code>\Psi</code>
Δ	<code>\Delta</code>	Ξ	<code>\Xi</code>	Υ	<code>\Upsilon</code>	Ω	<code>\Omega</code>
Θ	<code>\Theta</code>	Π	<code>\Pi</code>	Φ	<code>\Phi</code>		

7. Ces tables sont dérivées du fichier `symbols.tex` de David Carlisle et modifiées selon les suggestions de Josef Tkadlec

TABLE 3.4 – Relations binaires

Vous pouvez produire la négation de ces relations en préfixant ces commandes par `\not`.

$<$	<code><</code>	$>$	<code>></code>	$=$	<code>=</code>
\leq	<code>\leq</code> ou <code>\le</code>	\geq	<code>\geq</code> ou <code>\ge</code>	\equiv	<code>\equiv</code>
\ll	<code>\ll</code>	\gg	<code>\gg</code>	\doteq	<code>\doteq</code>
\prec	<code>\prec</code>	\succ	<code>\succ</code>	\sim	<code>\sim</code>
\preceq	<code>\preceq</code>	\succeq	<code>\succeq</code>	\simeq	<code>\simeq</code>
\subset	<code>\subset</code>	\supset	<code>\supset</code>	\approx	<code>\approx</code>
\subseteq	<code>\subseteq</code>	\supseteq	<code>\supseteq</code>	\cong	<code>\cong</code>
\sqsubset^a	<code>\sqsubset^a</code>	\sqsupset^a	<code>\sqsupset^a</code>	\Join^a	<code>\Join^a</code>
\sqsubseteq	<code>\sqsubseteq</code>	\sqsupseteq	<code>\sqsupseteq</code>	\bowtie	<code>\bowtie</code>
\in	<code>\in</code>	\ni	<code>\ni</code> , <code>\owns</code>	\propto	<code>\propto</code>
\vdash	<code>\vdash</code>	\dashv	<code>\dashv</code>	\models	<code>\models</code>
\mid	<code>\mid</code>	\parallel	<code>\parallel</code>	\perp	<code>\perp</code>
\smile	<code>\smile</code>	\frown	<code>\frown</code>	\asymp	<code>\asymp</code>
$:$	<code>:</code>	\notin	<code>\notin</code>	\neq	<code>\neq</code> ou <code>\ne</code>

TABLE 3.5 – Opérateurs binaires

$+$	<code>+</code>	$-$	<code>-</code>	\triangleleft	<code>\triangleleft</code>
\pm	<code>\pm</code>	\mp	<code>\mp</code>	\triangleangleright	<code>\triangleangleright</code>
\cdot	<code>\cdot</code>	\div	<code>\div</code>	\star	<code>\star</code>
\times	<code>\times</code>	\setminus	<code>\setminus</code>	\ast	<code>\ast</code>
\cup	<code>\cup</code>	\cap	<code>\cap</code>	\circ	<code>\circ</code>
\sqcup	<code>\sqcup</code>	\sqcap	<code>\sqcap</code>	\bullet	<code>\bullet</code>
\vee	<code>\vee</code> , <code>\lou</code>	\wedge	<code>\wedge</code> , <code>\land</code>	\diamond	<code>\diamond</code>
\oplus	<code>\oplus</code>	\ominus	<code>\ominus</code>	\uplus	<code>\uplus</code>
\odot	<code>\odot</code>	\oslash	<code>\oslash</code>	\amalg	<code>\amalg</code>
\otimes	<code>\otimes</code>	\bigcirc	<code>\bigcirc</code>	\dagger	<code>\dagger</code>
\triangle	<code>\bigtriangleup</code>	∇	<code>\bigtriangledown</code>	\ddagger	<code>\ddagger</code>
\triangleleft	<code>\lhd^a</code>	\triangleright	<code>\rhd^a</code>	\wr	<code>\wr</code>
\triangleleft	<code>\unlhd^a</code>	\triangleright	<code>\unrhd^a</code>		

^aUtilisez l'extension `latexsym` pour avoir accès à ces symboles

TABLE 3.6 – Opérateurs n-aires

\sum	<code>\sum</code>	\bigcup	<code>\bigcup</code>	\bigvee	<code>\bigvee</code>	\bigoplus	<code>\bigoplus</code>
\prod	<code>\prod</code>	\bigcap	<code>\bigcap</code>	\bigwedge	<code>\bigwedge</code>	\bigotimes	<code>\bigotimes</code>
\coprod	<code>\coprod</code>	\bigsqcup	<code>\bigsqcup</code>			\bigodot	<code>\bigodot</code>
\int	<code>\int</code>	\oint	<code>\oint</code>			\biguplus	<code>\biguplus</code>

TABLE 3.7 – Flèches

\leftarrow	<code>\leftarrow</code> ou <code>\gets</code>	\longleftarrow	<code>\longleftarrow</code>	\uparrow	<code>\uparrow</code>
\rightarrow	<code>\rightarrow</code> ou <code>\to</code>	\longrightarrow	<code>\longrightarrow</code>	\downarrow	<code>\downarrow</code>
\leftrightarrow	<code>\leftrightarrow</code>	\longleftrightarrow	<code>\longleftrightarrow</code>	\updownarrow	<code>\updownarrow</code>
\Leftarrow	<code>\Leftarrow</code>	\Lleftarrow	<code>\Lleftarrow</code>	\Uparrow	<code>\Uparrow</code>
\Rightarrow	<code>\Rightarrow</code>	\Rrightarrow	<code>\Rrightarrow</code>	\Downarrow	<code>\Downarrow</code>
\Leftrightarrow	<code>\Leftrightarrow</code>	\Leftrightarrow	<code>\Leftrightarrow</code>	\Updownarrow	<code>\Updownarrow</code>
\mapsto	<code>\mapsto</code>	\longmapsto	<code>\longmapsto</code>	\nearrow	<code>\nearrow</code>
\hookrightarrow	<code>\hookrightarrow</code>	\hookrightarrow	<code>\hookrightarrow</code>	\searrow	<code>\searrow</code>
\leftharpoonup	<code>\leftharpoonup</code>	\rightharpoonup	<code>\rightharpoonup</code>	\swarrow	<code>\swarrow</code>
\leftharpoondown	<code>\leftharpoondown</code>	\rightharpoondown	<code>\rightharpoondown</code>	\nwarrow	<code>\nwarrow</code>
\rightleftharpoons	<code>\rightleftharpoons</code>	\Leftrightarrow	<code>\iff</code> (plus d'espace)	\leadsto	<code>\leadsto</code> ^a

^aUtilisez l'extension `latexsym` pour obtenir ces symboles

TABLE 3.8 – Délimiteurs

$($	<code>(</code>	$)$	<code>)</code>	\uparrow	<code>\uparrow</code>	\Uparrow	<code>\Uparrow</code>
$[$	<code>[</code> ou <code>\lbrack</code>	$]$	<code>] ou \rbrack</code>	\downarrow	<code>\downarrow</code>	\Downarrow	<code>\Downarrow</code>
$\{$	<code>\{</code> ou <code>\lbrace</code>	$\}$	<code>\} ou \rbrace</code>	\updownarrow	<code>\updownarrow</code>	\Updownarrow	<code>\Updownarrow</code>
\langle	<code>\langle</code>	\rangle	<code>\rangle</code>	$ $	<code> </code> ou <code>\vert</code>	$\ $	<code>\ </code> ou <code>\Vert</code>
\lfloor	<code>\lfloor</code>	\rfloor	<code>\rfloor</code>	\lceil	<code>\lceil</code>	\rceil	<code>\rceil</code>
$/$	<code>/</code>	\backslash	<code>\backslash</code>				

TABLE 3.9 – Grands délimiteurs

$\left($	<code>\lgroup</code>	$\right)$	<code>\rgroup</code>	$\left[$	<code>\lmoustache</code>	$\right]$	<code>\rmoustache</code>
\uparrow	<code>\arrowvert</code>	\uparrow	<code>\Arrowvert</code>	\uparrow	<code>\bracevert</code>	\uparrow	

TABLE 3.10 – Symboles divers

\dots	<code>\dots</code>	\cdots	<code>\cdots</code>	\vdots	<code>\vdots</code>	\ddots	<code>\ddots</code>
\hbar	<code>\hbar</code>	\imath	<code>\imath</code>	\jmath	<code>\jmath</code>	ℓ	<code>\ell</code>
\Re	<code>\Re</code>	\Im	<code>\Im</code>	\aleph	<code>\aleph</code>	\wp	<code>\wp</code>
\forall	<code>\forall</code>	\exists	<code>\exists</code>	\mho	<code>\mho</code> ^a	∂	<code>\partial</code>
$'$	<code>'</code>	$'$	<code>\prime</code>	\emptyset	<code>\emptyset</code>	∞	<code>\infty</code>
∇	<code>\nabla</code>	\triangle	<code>\triangle</code>	\square	<code>\Box</code> ^a	\diamond	<code>\Diamond</code> ^a
\perp	<code>\bot</code>	\top	<code>\top</code>	\sphericalangle	<code>\angle</code>	\surd	<code>\surd</code>
\diamondsuit	<code>\diamondsuit</code>	\heartsuit	<code>\heartsuit</code>	\clubsuit	<code>\clubsuit</code>	\spadesuit	<code>\spadesuit</code>
\neg	<code>\neg</code> ou <code>\lnot</code>	\flat	<code>\flat</code>	\natural	<code>\natural</code>	\sharp	<code>\sharp</code>

^aUtilisez l'extension `latexsym` pour obtenir ces symboles

TABLE 3.11 – Symboles non-mathématiques

Ces symboles peuvent également être utilisés en mode *texte*.

\dagger	<code>\dag</code>	\S	<code>\S</code>	\copyright	<code>\copyright</code>	\textregistered	<code>\textregistered</code>
\ddagger	<code>\ddag</code>	\pounds	<code>\P</code>	\pounds	<code>\pounds</code>	$\%$	<code>\%</code>

TABLE 3.12 – Délimiteurs de l'AMS

\ulcorner	<code>\ulcorner</code>	\urcorner	<code>\urcorner</code>	\llcorner	<code>\llcorner</code>	\lrcorner	<code>\lrcorner</code>
\lvert	<code>\lvert</code>	\rvert	<code>\rvert</code>	\lVert	<code>\lVert</code>	\rVert	<code>\rVert</code>

TABLE 3.13 – Caractères grecs et hébreux de l'AMS

\digamma	<code>\digamma</code>	\varkappa	<code>\varkappa</code>	\beth	<code>\beth</code>	\daleth	<code>\daleth</code>	\gimel	<code>\gimel</code>
------------	-----------------------	-------------	------------------------	---------	--------------------	-----------	----------------------	----------	---------------------

TABLE 3.14 – Relations binaires de l'AMS

\triangleleft	<code>\lessdot</code>	\triangleright	<code>\gtrdot</code>	\doteq	<code>\doteqdot</code> ou <code>\Doteq</code>
\leqslant	<code>\leqslant</code>	\geqslant	<code>\geqslant</code>	$\dot{=}$	<code>\risingdotseq</code>
\leqslantless	<code>\leqslantless</code>	\leqslantgtr	<code>\leqslantgtr</code>	\fallingdotseq	<code>\fallingdotseq</code>
\leqq	<code>\leqq</code>	\geqq	<code>\geqq</code>	\equiv	<code>\eqcirc</code>
\lll ou \llless	<code>\lll</code> ou <code>\llless</code>	\ggg ou \gggtr	<code>\ggg</code> ou <code>\gggtr</code>	\circ	<code>\circeq</code>
\lesssim	<code>\lesssim</code>	\gtrsim	<code>\gtrsim</code>	\triangleq	<code>\triangleq</code>
\lessapprox	<code>\lessapprox</code>	\gtrapprox	<code>\gtrapprox</code>	\bumpeq	<code>\bumpeq</code>
\lessgtr	<code>\lessgtr</code>	\gtrless	<code>\gtrless</code>	\Bumpeq	<code>\Bumpeq</code>
\lesseqgtr	<code>\lesseqgtr</code>	\gtreqless	<code>\gtreqless</code>	\sim	<code>\thicksim</code>
\lesseqqgtr	<code>\lesseqqgtr</code>	\gtreqqless	<code>\gtreqqless</code>	\approx	<code>\thickapprox</code>
\preccurlyeq	<code>\preccurlyeq</code>	\succcurlyeq	<code>\succcurlyeq</code>	\approx	<code>\approxeq</code>
\curlyeqprec	<code>\curlyeqprec</code>	\curlyeqsucc	<code>\curlyeqsucc</code>	\smile	<code>\backsim</code>
\precsim	<code>\precsim</code>	\succsim	<code>\succsim</code>	\subseteq	<code>\backsimeq</code>
\precapprox	<code>\precapprox</code>	\succapprox	<code>\succapprox</code>	\vDash	<code>\vDash</code>
\subseteqeq	<code>\subseteqeq</code>	\supseteqeq	<code>\supseteqeq</code>	\Vdash	<code>\Vdash</code>
\Subset	<code>\Subset</code>	\Supset	<code>\Supset</code>	\Vvdash	<code>\Vvdash</code>
\sqsubset	<code>\sqsubset</code>	\sqsupset	<code>\sqsupset</code>	ε	<code>\backepsilon</code>
\therefore	<code>\therefore</code>	\because	<code>\because</code>	\propto	<code>\varpropto</code>
\shortmid	<code>\shortmid</code>	\shortparallel	<code>\shortparallel</code>	\between	<code>\between</code>
\smallsmile	<code>\smallsmile</code>	\smallfrown	<code>\smallfrown</code>	\pitchfork	<code>\pitchfork</code>
\vartriangleleft	<code>\vartriangleleft</code>	\vartriangleright	<code>\vartriangleright</code>	\blacktriangleleft	<code>\blacktriangleleft</code>
\trianglelefteq	<code>\trianglelefteq</code>	\trianglerighteq	<code>\trianglerighteq</code>	\blacktriangleright	<code>\blacktriangleright</code>

TABLE 3.15 – Flèches de l'AMS

\dashleftarrow	<code>\dashleftarrow</code>	\dashrightarrow	<code>\dashrightarrow</code>	\multimap	<code>\multimap</code>
\leftleftarrows	<code>\leftleftarrows</code>	\rightrightarrows	<code>\rightrightarrows</code>	\Uparrow	<code>\upuparrows</code>
\leftrightarrows	<code>\leftrightarrows</code>	\rightleftarrows	<code>\rightleftarrows</code>	\Downarrow	<code>\downdownarrows</code>
\Lleftarrow	<code>\Lleftarrow</code>	\Rrightarrow	<code>\Rrightarrow</code>	\Uparrow	<code>\upharpoonleft</code>
\twoheadleftarrow	<code>\twoheadleftarrow</code>	\twoheadrightarrow	<code>\twoheadrightarrow</code>	\Uparrow	<code>\upharpoonright</code>
\leftarrowtail	<code>\leftarrowtail</code>	\rightarrowtail	<code>\rightarrowtail</code>	\Downarrow	<code>\downharpoonleft</code>
\leftrightharpoons	<code>\leftrightharpoons</code>	\rightleftharpoons	<code>\rightleftharpoons</code>	\Downarrow	<code>\downharpoonright</code>
\Lsh	<code>\Lsh</code>	\Rsh	<code>\Rsh</code>	\rightsquigarrow	<code>\rightsquigarrow</code>
\looparrowleft	<code>\looparrowleft</code>	\looparrowright	<code>\looparrowright</code>	\rightsquigarrow	<code>\leftrightsquigarrow</code>
\curvearrowleft	<code>\curvearrowleft</code>	\curvearrowright	<code>\curvearrowright</code>		
\circlearrowleft	<code>\circlearrowleft</code>	\circlearrowright	<code>\circlearrowright</code>		

TABLE 3.16 – Négations des relations binaires et des flèches de l'AMS

\nless	\ngtr	\nvarsubsetneqq
\lneq	\gneq	\nvarsupsetneqq
\nleq	\ngeq	\nsubsetneqq
\nleqslant	\ngeqslant	\nsupsetneqq
\lneqq	\gneqq	\nmid
\lvertneqq	\gvertneqq	\nparallel
\nleqq	\ngeqq	\nshortmid
\lnsim	\gnsim	\nshortparallel
\lnapprox	\gnapprox	\nsim
\nprec	\nsucc	\ncong
\npreceq	\nsucceq	\nvdash
\nprecneqq	\nsuccneqq	\nvDash
\nprecnsim	\succnsim	\nVDash
\nprecnapprox	\succnapprox	\nVDash
\nsubsetneq	\nsupsetneq	\ntriangleleft
\nvarsubsetneq	\nvarsupsetneq	\ntriangleright
\nsubseteq	\nsupseteq	\ntrianglelefteq
\nsubseteqq	\nsupseteqq	\ntrianglerighteq
\nleftarrow	\nrightarrow	\nleftrightarrow
\nLeftarrow	\nRightarrow	\nLeftrightarrow

TABLE 3.17 – Opérateurs binaires de l'AMS

\dotplus	\centerdot	\intercal
\ltimes	\rtimes	\divideontimes
\Cup ou \doublecup	\Cap ou \doublecap	\smallsetminus
\veebar	\barwedge	\doublebarwedge
\boxplus	\boxminus	\circleddash
\boxtimes	\boxdot	\circledcirc
\leftthreetimes	\rightthreetimes	\circledast
\curlyvee	\curlywedge	

TABLE 3.18 – Symboles divers de l'AMS

\hbar	<code>\hbar</code>	\hbar	<code>\hslash</code>	\mathbb{k}	<code>\Bbbk</code>
\square	<code>\square</code>	\blacksquare	<code>\blacksquare</code>	\textcircled{S}	<code>\circledS</code>
\triangle	<code>\vartriangle</code>	\blacktriangle	<code>\blacktriangle</code>	\complement	<code>\complement</code>
∇	<code>\triangledown</code>	\blacktriangledown	<code>\blacktriangledown</code>	\Game	<code>\Game</code>
\diamond	<code>\lozenge</code>	\blacklozenge	<code>\blacklozenge</code>	\bigstar	<code>\bigstar</code>
\sphericalangle	<code>\angle</code>	\sphericalangle	<code>\measuredangle</code>	\sphericalangle	<code>\sphericalangle</code>
\diagup	<code>\diagup</code>	\diagdown	<code>\diagdown</code>	\backprime	<code>\backprime</code>
\nexists	<code>\nexists</code>	\Finv	<code>\Finv</code>	\varnothing	<code>\varnothing</code>
\eth	<code>\eth</code>	\mho	<code>\mho</code>		

TABLE 3.19 – Polices mathématiques

Exemple	Commande	Extension à utiliser
ABCdef	<code>\mathrm{ABCdef}</code>	
ABCdef	<code>\mathit{ABCdef}</code>	
\mathnormal{ABCdef}	<code>\mathnormal{ABCdef}</code>	
ABCdef	<code>\mathsf{ABCdef}</code>	
\mathtt{ABCdef}	<code>\mathtt{ABCdef}</code>	
\mathcal{ABCDEF}	<code>\mathcal{ABCDEF}</code>	
\mathcal{ABCDEF}	<code>\mathcal{ABCDEF}</code>	eucal
\mathscr{ABCDEF}	<code>\mathscr{ABCDEF}</code>	mathrsfs
\mathfrak{ABCdef}	<code>\mathfrak{ABCdef}</code>	eufrak
\mathbb{ABCDEF}	<code>\mathbb{ABCDEF}</code>	amsfonts ou amssymb

Chapitre 4

Compléments

En plus des possibilités présentées dans les chapitres précédents, \LaTeX permet d'insérer des graphiques dans un document, de gérer la couleur, de réaliser un index, une liste de références bibliographiques et bien d'autres choses. Des descriptions beaucoup plus complètes de ces possibilités se trouvent dans le *LaTeX Manual* [1], dans *The LaTeX Companion* [3], dans [4], etc.

4.1. Figures PostScript

Avec les environnements `figure` et `table`, \LaTeX fournit les mécanismes de base pour travailler avec des objets tels que des images ou des graphiques.

Il existe également plusieurs possibilités pour produire des graphiques avec des commandes \LaTeX ou des extensions de \LaTeX . Malheureusement, la plupart des utilisateurs trouvent ces commandes difficiles à mettre en œuvre. Pour plus d'informations à leur sujet, reportez-vous à *The LaTeX Companion* [3] ou à [4].

Un moyen bien plus simple d'inclure des graphiques dans un document est de les produire à l'aide d'un logiciel spécialisé¹ puis d'inclure le résultat dans le document. Ici encore \LaTeX offre de nombreuses solutions. Dans cette introduction, seule l'utilisation de graphiques PostScript encapsulé (EPS) sera traitée. En effet, c'est la solution la plus simple et la plus utilisée. Pour pouvoir utiliser des graphiques au format EPS, il faut disposer d'une imprimante acceptant les documents PostScript ou utiliser le logiciel GHOSTSCRIPT, disponible sur `CTAN:/support/ghostscript`, voir page vi la liste des sites CTAN.

Un ensemble de commandes bien adaptées à l'insertion de graphiques est fourni par l'extension `graphicx`, développée par D. P. Carlisle. Elle fait partie d'un ensemble d'extensions appelé `graphics`².

En supposant maintenant que vous travaillez sur un système où l'extension `graphicx` est installée et qui dispose d'une imprimante PostScript ou de GHOSTSCRIPT, voici la marche à suivre pour inclure une figure dans un document :

1. Exportez la figure de votre logiciel graphique au format EPS.

1. Tel que XFig, idraw, CorelDraw!, FreeHand, Gnuplot...

2. `CTAN:/macros/latex/required/graphics`, voir page vi la liste des sites CTAN.

2. Chargez l'extension `graphicx` dans le préambule de votre fichier source avec :

```
\usepackage[pilote]{graphicx}
```

L'argument optionnel *pilote* est le nom du programme de conversion « DVI vers PostScript » à utiliser. Le programme le plus répandu est `dvips`, c'est le pilote utilisé par défaut. Cette information est nécessaire car il n'y a pas de standard pour l'insertion de figures en \TeX ; celle-ci est réalisée par le pilote au moment de l'impression. Connaître le nom du pilote permet à l'extension `graphicx` d'inclure les bonnes commandes dans le fichier `.dvi` pour réaliser cette insertion.

3. Utilisez la commande :

```
\includegraphics[clef=valeur, ... ]{fichier}
```

pour insérer *fichier*³ dans votre document. Le paramètre optionnel est une liste de paires de *clefs* et de *valeurs* séparées par des virgules. Les *clefs* permettent de modifier la largeur, la hauteur, ou l'angle de rotation de la figure. Le tableau 4.1 présente les clefs les plus importantes.

TABLE 4.1 – Clefs pour l'extension `graphicx`

<code>width</code>	définit la largeur de la figure
<code>height</code>	définit la hauteur de la figure
<code>angle</code>	(en degrés) tourne la figure dans le sens des aiguilles d'une montre

L'exemple suivant illustre l'utilisation de `\includegraphics`. Rappelons que les figures sont en général traitées comme des objets flottants (voir section 2.12 page 37). La commande `\centering` centre la figure horizontalement, comme le ferait un environnement `\begin{center} ... \end{center}` mais sans ajouter d'espace vertical avant et après.

3. Il est recommandé de choisir `.eps` (plutôt que `.ps`) comme suffixe pour les capsules PostScript et dans ce cas d'omettre le suffixe `.eps` lors de l'appel de `\includegraphics` : il sera ajouté automatiquement.

```

\begin{figure}
\centering
\includegraphics[angle=90, width=10cm]{test}
\end{figure}

```

Cette commande inclut la figure stockée dans le fichier `test.eps`. La figure est *d'abord* tournée de 90 degrés, sa taille est ensuite ajustée pour que sa largeur finale soit de 10 cm (les proportions largeur/hauteur sont conservées).

Pour plus d'informations, reportez vous à [9] et [15].

4.2. X_Y-pic

L'extension `xy` permet de dessiner des diagrammes; pour faire appel à elle, il suffit d'ajouter dans le préambule du document :

```
\usepackage[options]{xy}
```

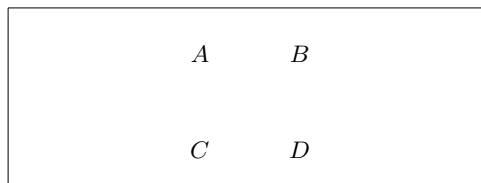
`options` est la liste des options à charger. Il est recommandé de choisir l'option `all`, qui donne accès à la totalité des options.

Les diagrammes produits par X_Y-pic sont placés sur une grille matricielle composée de cellules :

```

\begin{displaymath}
\xymatrix{A & B \\
          C & D }
\end{displaymath}

```

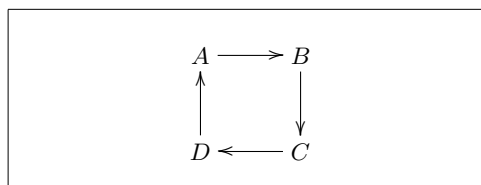


La commande `\xymatrix` ne peut être utilisée qu'en mode mathématique. Dans l'exemple ci-dessus il y a deux lignes et deux colonnes (4 cellules). Pour en faire un diagramme, il suffit d'ajouter des flèches avec la commande `\ar`.

```

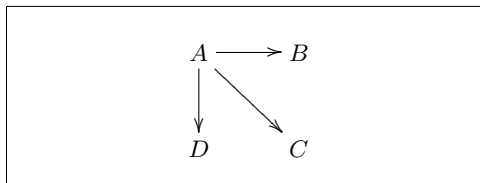
\begin{displaymath}
\xymatrix{ A \ar[r] & B \ar[d] \\
          D \ar[u] & C \ar[l] }
\end{displaymath}

```



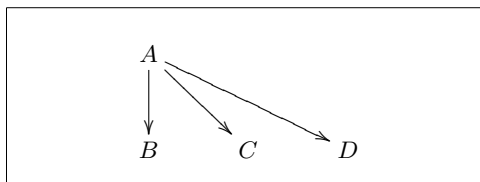
La commande `\ar` se place dans la cellule d'où part la flèche, les arguments indiquent la direction de la flèche (up, down, right and left).

```
\begin{displaymath}
\xymatrix{
  A \ar[d] \ar[dr] \ar[r] & B \\
  D & C }
\end{displaymath}
```



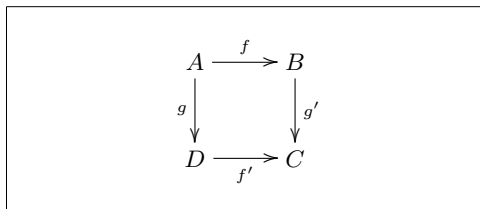
Pour produire des flèches diagonales, il suffit d'ajouter un ou plusieurs arguments de direction :

```
\begin{displaymath}
\xymatrix{
  A \ar[d] \ar[dr] \ar[dr] & & \\
  B & & C & D }
\end{displaymath}
```



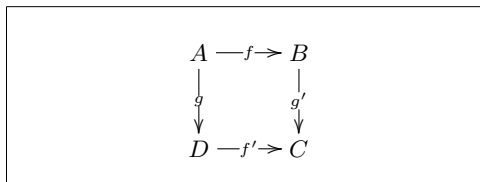
Il est possible d'ajouter des noms sur les flèches grâce aux indices et exposants (nous sommes en mode mathématique) :

```
\begin{displaymath}
\xymatrix{
  A \ar[r]^f \ar[d]_g & & \\
  B \ar[d]^{g'} & & C \\
  D \ar[r]_{f'} & & C }
\end{displaymath}
```



Les exposants placent le texte au dessus de la flèche, les indices en dessous. La barre verticale positionne le texte *sur* la flèche :

```
\begin{displaymath}
\xymatrix{
  A \ar[r]|f \ar[d]|g & & \\
  B \ar[d]|{g'} & & C \\
  D \ar[r]|{f'} & & C }
\end{displaymath}
```



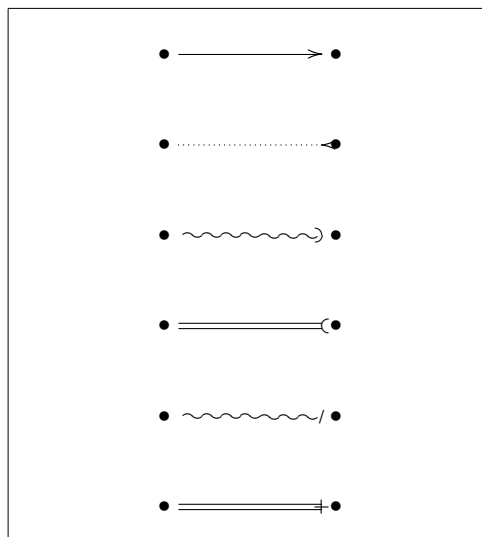
Pour ajouter un petit cercle au milieu de la flèche sous le texte, on utilise la commande `\ar[...]|hole`.

L'exemple suivant montre quelques types de flèches disponibles :

```

\begin{displaymath}
\begin{matrix}
\bullet \longrightarrow \bullet \\
\bullet \cdots \longleftarrow \bullet \\
\bullet \rightsquigarrow \bullet \\
\bullet \curvearrowright \bullet \\
\bullet \equiv \bullet \\
\bullet \rightsquigarrow / \bullet \\
\bullet \equiv \# \bullet
\end{matrix}
\end{displaymath}

```

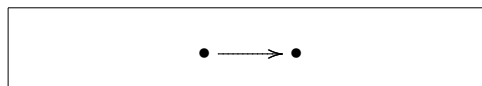


Comparez les deux diagrammes suivants :

```

\begin{displaymath}
\begin{matrix}
\bullet \longrightarrow \bullet \\
\bullet \curvearrowright \bullet
\end{matrix}
\end{displaymath}

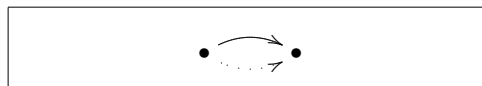
```



```

\begin{displaymath}
\begin{matrix}
\bullet \curvearrowright \bullet \\
\bullet \rightsquigarrow / \bullet
\end{matrix}
\end{displaymath}

```



Les arguments $/\sim/$ et $/_ /$ incurvent la flèche respectivement vers le haut ou vers le bas. L'extension Xy-pic offre beaucoup d'autres possibilités non décrites dans cette brève présentation, consulter la documentation qui accompagne cette extension pour les découvrir.

4.3. Références bibliographiques

L'environnement `thebibliography` permet de produire une liste de références bibliographiques. Chaque référence commence par

```
\bibitem[label]{marque}
```

La *marque* est utilisée pour citer la référence dans le document :

```
\cite{marque}
```

Si l'option *label* n'est pas utilisée, la numérotation des références est produite automatiquement. Le paramètre qui suit `\begin{thebibliography}` définit la largeur du décrochement utilisé pour placer ces numéros. Dans l'exemple suivant, `{99}` signifie qu'aucun des numéros servant de référence ne dépasse 99.

```
Yves~Perrousseau~\cite{YP}
recommande\ldots

{\small%\footnotesize
\begin{thebibliography}{99}
\bibitem{YP} Yves~Perrousseau.
\emph{Manuel de typographie
française élémentaire}.
Ateliers Perrousseau, 1995.
\end{thebibliography}
}
```

Yves Perrousseau [1] recommande . . .

Bibliographie

[1] Yves Perrousseau. *Manuel de typographie française élémentaire*. Ateliers Perrousseau, 1995.

Pour des projets plus importants, il est recommandé d'utiliser l'outil BiB \TeX . Celui-ci est fourni avec la plupart des installations de \TeX . Il permet de maintenir une base de données de références bibliographiques et d'en extraire la liste des références citées dans votre document. La génération des listes de références par BiB \TeX utilise un mécanisme de feuilles de style qui permettent de réaliser tous les types de présentations habituellement demandés.

4.4. Index

L'index est un élément fort utile dans de nombreux ouvrages. \LaTeX et le programme associé `makeindex`⁴ permettent de créer des index assez facilement.

4. sur les systèmes qui ne supportent pas les noms de fichiers de plus de huit caractères, ce programme s'appelle `makeidx`.

Dans cette introduction, seules les commandes élémentaires de gestion d'un index sont présentées. Pour une description plus détaillée, reportez-vous à *The L^AT_EX Companion* [3] ou à [4].

Pour utiliser cette fonctionnalité, l'extension `makeidx` doit être chargée dans le préambule avec :

```
\usepackage{makeidx}
```

La création de l'index doit être activée par la commande :

```
\makeindex
```

placée dans le préambule.

Le contenu de l'index est défini par une série de commandes :

```
\index{clef}
```

où *clef* est un mot-clef de l'index. Vous insérez des commandes `\index` aux endroits du texte que vous voulez voir référencés par l'index. Le tableau 4.2 explique la syntaxe de l'argument *clef* avec plusieurs exemples.

TABLE 4.2 – Exemples de clefs d'index

Exemple	Résultat	Commentaires
<code>\index{hello}</code>	hello, 1	Entrée normale
<code>\index{hello!Peter}</code>	Peter, 3	Sous-entrée de 'hello'
<code>\index{Sam@\textsl{Sam}}</code>	<i>Sam</i> , 2	Entrée formatée
<code>\index{Lin@\textbf{Lin}}</code>	Lin , 7	<i>idem</i>
<code>\index{Jenny textbf}</code>	Jenny, 3	Numéro de page formaté
<code>\index{Joe textit}</code>	Joe, <i>5</i>	<i>idem</i>
<code>\index{eolienne@éolienne}</code>	éolienne, 4	clef accentuée

Quand le fichier source est traité par L^AT_EX, chaque commande `\index` crée une entrée adaptée contenant le numéro de la page en cours dans le fichier qui porte le même nom de base que le fichier source, mais avec le suffixe `.idx`. Ce fichier est ensuite traité par le programme `makeindex`.

```
makeindex nom de fichier
```

Le programme `makeindex` crée un index trié dans le fichier `.ind`. Ensuite, à la prochaine compilation du fichier source, le contenu du fichier `.ind` sera inclus à l'endroit où \LaTeX rencontrera la commande :

```
\printindex
```

L'extension `showidx` permet de visualiser les entrées de l'index dans la marge gauche du texte. Cela permet la relecture et la mise au point de l'index.

Le programme `makeindex` standard ne traite malheureusement pas correctement les caractères accentués dans les clefs : il les place systématiquement en tête de l'ordre alphabétique. Pour obtenir un classement correct des clés contenant des caractères accentués (le « é » doit être classé comme un « e »), on peut utiliser le caractère `@` : la dernière ligne de la table 4.2 montre comment procéder pour faire classer « éolienne » correctement.

4.5. En-têtes améliorés

L'extension `fancyhdr`⁵, développée par P. van Oostrum, offre quelques commandes simples permettant de personnaliser les en-têtes et les pieds de page. Les hauts de page du présent document sont réalisés grâce à cette extension.

Voyons comment fonctionne la mise à jour des noms de la section ou du chapitre en cours en vue de leur affichage dans les en-têtes et les pieds de page. \LaTeX réalise l'opération en deux étapes. Dans la définition des en-têtes et pieds de page les commandes `\leftmark` et `\rightmark` sont utilisées pour désigner respectivement les noms de la section et du chapitre courants. La valeur de ces commandes est redéfinie chaque fois qu'un nouveau chapitre ou qu'une nouvelle section commence.

Pour plus de souplesse, les commandes de sectionnement ne redéfinissent pas directement `\leftmark` et `\rightmark`. Elles appellent les commandes `\chaptermark`, `\sectionmark` et `\subsectionmark` qui sont chargées de redéfinir `\leftmark` ou `\rightmark`, selon la présentation désirée.

Ainsi, si vous voulez modifier la présentation du nom du chapitre courant dans l'en-tête, vous devrez redéfinir la commande `\chaptermark`.

La figure 4.1 montre un exemple de configuration de l'extension `fancyhdr` qui se rapproche de la présentation utilisée pour ce document. La documentation

⁵. disponible sur CTAN : `/macros/latex/contrib/supported/fancyhdr/`, voir page vi la liste des sites CTAN.

```

\documentclass{book}
\usepackage{fancyhdr}
\pagestyle{fancy}
% Ceci permet d'avoir les noms de chapitre et de section
% en minuscules
\renewcommand{\chaptermark}[1]{\markboth{#1}{}}
\renewcommand{\sectionmark}[1]{\markright{\thesection\ #1}}
\fancyhf{} % supprime les en-têtes et pieds prédéfinis
\fancyhead[LE,RO]{\bfseries\thepage}% Left Even, Right Odd
\fancyhead[LO]{\bfseries\rightmark} % Left Odd
\fancyhead[RE]{\bfseries\leftmark} % Right Even
\renewcommand{\headrulewidth}{0.5pt}% filet en haut de page
\addtolength{\headheight}{0.5pt} % espace pour le filet
\renewcommand{\footrulewidth}{0pt} % pas de filet en bas
\fancypagestyle{plain}{%          pages de têtes de chapitre
  \fancyhead{}%                  % supprime l'en-tête
  \renewcommand{\headrulewidth}{0pt} % et le filet
}

```

FIGURE 4.1 – Exemple de configuration de l'extension fancyhdr

complète de cette extension se trouve à l'adresse mentionnée dans la note de bas de page.

4.6. L'extension verbatim

Plus haut dans ce document, vous avez appris à utiliser l'*environnement verbatim*. Dans cette section vous allez découvrir l'*extension verbatim*. L'extension `verbatim` fournit une nouvelle implémentation de l'environnement du même nom qui corrige certaines de ses limitations et ajoute de nouvelles fonctionnalités. L'extension `verbatim` propose la commande :

```
\verbatiminput{filename}
```

qui permet d'inclure un fichier ASCII brut dans votre document, comme s'il se trouvait à l'intérieur d'un environnement `verbatim`.

L'extension `verbatim` définit aussi l'environnement `comment`; tout le texte contenu dans cet environnement est ignoré, exactement comme si un % était placé devant chaque ligne.

Voici un autre exemple
`\begin{comment}`
 limité mais significatif
`\end{comment}`
 de commentaires.

Voici un autre exemple de commentaires.

Une application intéressante est l'impression ou non de solutions d'exercices⁶ : on place celles-ci dans un environnement `comment` (elles sont donc cachées) et on redéfinit l'environnement `comment` lorsqu'on veut les imprimer⁷.

```
\renewenvironment{comment}{\textit{Solution :}\slshape}{}

```

Puisque l'extension `verbatim` fait partie de l'ensemble « `tools` », elle devrait être installée sur la plupart des systèmes. Pour en savoir plus sur cette extension, reportez-vous à [10].

4.7. Téléchargement et installation d'extensions

La plupart des installations \LaTeX fournissent en standard un grand nombre d'extensions, mais il arrive que justement celle dont on aurait besoin manque, ou qu'une extension nécessite une mise à jour. L'endroit le plus adéquat pour rechercher les versions officielles des extensions \LaTeX est <http://www.tex.ac.uk/CTANfind.html>

Les extensions sont en général fournies sous la forme de deux fichiers, l'un de suffixe `.dtx`, l'autre de suffixe `.ins`. Souvent un fichier `readme.txt` est joint et donne une brève description de l'extension, le mieux est de commencer par la lecture de ce fichier.

Ensuite, il faut, d'une part, installer l'extension et, d'autre part, extraire la documentation ; voici la manière de procéder :

1. Exécuter \LaTeX sur le fichier `.ins`. Ceci produira les fichiers `.sty`, `.def`, etc., dont \LaTeX a besoin.
2. Déplacer ces fichiers dans un répertoire adéquat, en général c'est dans `.../texmf/tex/latex` ou dans `.../localtexmf/tex/latex`.
3. Mettre à jour la base de données des noms de fichiers, la commande dépend de votre distribution \LaTeX , c'est en général `mktxlsr` ou `texhash`.

⁶. Voir aussi les extensions spécialisées `exam`, `examdesign` et `acrotex` disponibles sur CTAN.

⁷. La redéfinition d'environnements sera expliquée page 89.

Il faut ensuite extraire la documentation du fichier `.dtx` :

1. Exécuter `LATEX` sur le fichier `.dtx`. Ceci produira un fichier `.dvi` contenant la documentation (plusieurs exécutions de `LATEX` peuvent être nécessaires pour stabiliser les références croisées).
2. Vérifier si `LATEX` a produit un fichier `.idx`; si ce n'est pas le cas, passer à l'étape 5.
3. Pour produire l'index, exécuter la commande suivante :

```
makeindex -s gind.ist nom
```

 où `nom` désigne le nom du fichier principal sans suffixe.
4. Exécuter `LATEX` sur le fichier `.dtx` une nouvelle fois.
5. Enfin, produire un fichier PostScript ou PDF à imprimer pour une lecture plus confortable.

Parfois vous constaterez qu'un fichier `.glo` a été produit. Exécutez la commande suivante entre les étapes 4 et 5 :

```
makeindex -s gglo.ist -o nom.gls nom.glo
```

N'oubliez pas de relancer `LATEX` sur le fichier `.dtx` avant de passer à l'étape 5.

4.8. Protection des commandes « fragiles »

Les arguments de commandes telles que `\section` ou `\caption` etc., peuvent apparaître plusieurs fois dans le document (par exemple aussi dans la table des matières, les hauts de pages...), on dit qu'il s'agit d'arguments « mouvants » (*moving arguments*). Certaines commandes, comme `\footnote`, `\phantom` etc., ne produisent pas le résultat escompté quand elles sont exécutées comme argument de commandes de type `\section`, on dit qu'elles sont « fragiles », ce qui signifie qu'elles ont besoin de la protection d'un ... `\protect`.

La commande `\protect` n'a d'effet que sur la commande qui la suit immédiatement, mais *pas ses arguments* éventuels. La plupart du temps un `\protect` de trop ne produira aucun effet pervers.

Voici un exemple d'utilisation de `\protect` :

```
\section{Je suis prudent
  \protect\footnote{Je protège ma note de bas de page.}}
```

Chapitre 5

Documents hypertextes en PDF

Dans ce qui précède nous nous sommes intéressés à la production de documents papier. La diffusion de documents, notamment scientifiques, se faisant de plus en plus par le réseau, nous allons voir dans ce chapitre comment produire des documents « hypertextes » à partir de sources \LaTeX .

5.1. Documents hypertextes

Le qualificatif « hypertexte » associé à un document signifie qu'il offre des possibilités de navigation autres que la lecture linéaire des pages : ainsi un clic sur une référence à une partie (section, page etc.) permet de sauter directement à cette partie, ou encore un clic dans la table des matières ou dans l'index conduit directement à la page référencée.

Les pages WEB que l'on trouve actuellement sur les serveurs sont généralement codées en HTML (*HyperText Markup Language*). Ce format présente deux inconvénients lorsqu'il s'agit d'afficher des documents scientifiques :

1. Les possibilités en matière d'affichage de formules mathématiques sont si pauvres que les formules doivent le plus souvent être présentées sous forme d'images.
2. La mise en page ne peut pas être imposée par l'auteur, elle dépend essentiellement du « client » utilisé par le lecteur.

Bien qu'il soit possible de produire des documents en HTML à partir de sources \LaTeX ¹, les auteurs soucieux de qualité typographique préféreront utiliser le format PDF (*Portable Document Format*) qui préserve leur mise en page et permet également la navigation hypertexte. Les navigateurs modernes disposent d'extensions (*plugins*) capables d'afficher les documents PDF.

Un autre avantage des fichiers PDF par rapport aux fichiers DVI ou PostScript produits habituellement par \LaTeX est qu'ils sont affichables et imprimables sur *toutes les plateformes* (Unix, Windows, Macintosh) grâce au programme Acrobat Reader (*acroread*) fourni gratuitement par Adobe et normalement installé en standard.

1. Voir l'utilisation de `tex4ht` dans les *Cahiers GUTenberg* n° 37-38 (décembre 2000).

5.2. Documents en PDF pour le WEB

La production de fichiers PDF à partir de sources \LaTeX est devenue très simple grâce au moteur \pdfTeX développé par Hàn Thê Thành. \pdfTeX produit par défaut² des fichiers `.pdf` ; $\pdf\LaTeX$ est un format \LaTeX qui utilise le moteur \pdfTeX pour produire des fichiers `.pdf` à partir de sources \LaTeX .

Dans les installations modernes, notamment teTeX , fpTeX et CMacTeX , \pdfTeX et $\pdf\LaTeX$ sont prêts à l'emploi. Pour produire un fichier PDF au lieu d'un fichier DVI, il suffit de remplacer la commande `latex fichier.tex` par `pdflatex fichier.tex`³.

En \LaTeX comme en $\pdf\LaTeX$, il est vivement conseillé de spécifier le format de page (papier ou écran) utilisé. En effet, le format par défaut (*letter* : $8,5 \times 11$ pouces, soit 216×280 mm) n'est pas utilisé en Europe, il faut donc le redéfinir – soit en utilisant une option globale (`a4paper`, `a5paper`, etc.) pour les formats standard, par exemple :

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

– soit en donnant, dans le préambule, les valeurs souhaitées aux variables `\paperheight` et `\paperwidth` :

```
\setlength{\paperheight}{200mm}
```

```
\setlength{\paperwidth}{150mm}
```

Ceci est suffisant en \LaTeX ; en $\pdf\LaTeX$ aussi, à condition que l'extension `hyperref` soit utilisée (voir page 78). Sinon, il y a lieu d'ajouter dans le préambule, après avoir défini `\paperwidth` et `\paperheight`, la ligne :

```
\pdfpagewidth=\paperwidth \pdfpageheight=\paperheight
```

Dans la suite nous allons préciser les différences entre un source \LaTeX standard et un source destiné à $\pdf\LaTeX$. Les différences concernent trois points : les fontes à employer, le format des images à inclure, et la configuration des liens hypertextes.

5.2.1. Fontes

$\pdf\LaTeX$ permet d'utiliser toutes sortes de fontes (*bitmaps*, TrueType, PostScript type 1...) mais les fontes *bitmaps* sont à proscrire car Acrobat Reader les affiche très mal ; le mieux est de se limiter aux fontes PostScript type 1.

Dans les chapitres précédents la question du choix des fontes a été volontairement laissée de côté : \TeX disposant en standard d'un jeu de fontes cohérent

2. \pdfTeX peut aussi produire des fichiers `.dvi` comme le fait \TeX , mais ce n'est pas sa vocation.

3. Dans certaines installations, le lancement de ces commandes, `latex` ou `pdflatex`, est fait dans un menu.

assez bien adapté aux publications scientifiques, on utilise en général celui-ci. En fait, ce jeu de fontes existe en deux variantes : *Computer Modern* (CM), fontes à 128 caractères utilisées par défaut, et *Extended Cork* (EC), fontes à 256 caractères, que l'on active en ajoutant la ligne `\usepackage[T1]{fontenc}` dans le préambule.

Pour obtenir des césures correctes en français pour les mots accentués, il est nécessaire d'utiliser des fontes à 256 caractères (voir page 25). Les distributions modernes (TeXLive7 par exemple) contiennent une version type 1 des fontes EC, ce qui permet d'avoir à la fois un affichage lisible des fichiers PDF et des césures correctes en français. Toutefois, ces fontes, appelées *cm-super* n'ont pas la qualité graphique de leurs homologues CM type 1, en particulier les fichiers PDF produits sont beaucoup plus volumineux, ce qui peut être gênant lors des transmissions par le réseau.

Une autre solution consiste à utiliser les fontes virtuelles AE : il suffit pour cela de remplacer dans le préambule `\usepackage[T1]{fontenc}` par `\usepackage{ae}`. Les fontes AE ont également 256 caractères (donc pas de problème avec les césures françaises) mais les glyphes sont construits à partir des 128 caractères des fontes CM type 1 (de qualité professionnelle, mises en domaine public par BlueSky et Y&Y) : le caractère *é*, par exemple, est obtenu en superposant deux caractères des fontes CM, un *e* et un accent aigu. Les fichiers PDF produits sont assez compacts, leur seul inconvénient est que la recherche de chaînes de caractères accentués sous Acrobat Reader (fonction **Find**) ne fonctionne pas... c'est normal puisque le caractère *é* n'est pas présent en tant que tel dans le fichier PDF !

Enfin, rien n'empêche de faire appel à d'autres fontes PostScript type 1. L'allure du texte sera modifiée, en particulier, si la mise en page a été optimisée pour les fontes CM ou EC, tout sera à refaire... Parmi les fontes non commerciales, on pourra essayer par exemple l'extension `pxfonts` à base de fontes *Palatino* ou l'extension `txfonts` à base de fontes *Times*. Pour cela, il suffit d'ajouter dans le préambule, en plus de `\usepackage[T1]{fontenc}`, soit `\usepackage{pxfonts}`, soit `\usepackage{txfonts}`. Avec cette solution, les césures seront correctes en français, sous Acrobat Reader l'affichage sera lisible et la fonction **Find** permettra la recherche des chaînes de caractères accentués... Les spécialistes des fontes déplorent toutefois le manque de cohérence globale de ces systèmes⁴, à vous de juger !

4. Ces systèmes font appel à *Helvetica* pour les polices « Sans Serif » et à *Courier* pour les polices à chasse fixe (*typewriter*), il est vrai que ces trois familles n'ont pas été conçues pour fonctionner en harmonie...

Remarque importante : il arrive que pdfL^AT_EX affiche dans le fichier .log, un « avertissement » du type

Warning: pdftex (file eurmo10): Font eurmo10 at ... not found

Il *faut* trouver l'origine du problème et y remédier, faute de quoi une page (au moins) du fichier PDF créé *ne s'affichera pas du tout* : un seul caractère manquant suffit à empêcher l'affichage de la page où il se trouve !

5.2.2. Graphiques

La bonne façon d'inclure des images, avec L^AT_EX comme avec pdfL^AT_EX, est d'utiliser l'extension `graphicx` (voir page 63). Seule l'option change, si on utilise pdfL^AT_EX, l'option *doit être* `pdftex`, on code dans le préambule

```
\usepackage[pdftex]{color,graphicx}5
```

alors qu'en L^AT_EX l'option dépend du pilote d'impression utilisé (c'est souvent `dvips` : `\usepackage[dvips]{graphicx}`).

Utilisée avec L^AT_EX, l'extension `graphicx` attend des images au format EPS. Avec pdfL^AT_EX, elle accepte des images aux formats PDF, PNG, JPG, TIFF ou MetaPost... mais *pas* EPS !

Il est facile de convertir les images EPS au format PDF en utilisant le convertisseur `epstopdf`. Les formats PNG ou JPG sont bien adaptés aux photos, captures d'écrans, etc., car les fichiers PNG ou JPG sont bien plus compacts que les EPS ou PDF correspondants. Pour réaliser certaines figures géométriques, un langage de commandes peut être plus adapté qu'une construction à la souris avec un logiciel de dessin tel que XFig ou CorelDraw, dans ce cas l'emploi de MetaPost s'impose (voir le *Cahier GUTenberg* n° 41 également disponible en ligne [17]).

5.2.3. Liens hypertextes

La mise en place des liens hypertextes est faite par l'extension `hyperref`, que l'on charge par la commande `\usepackage{hyperref}`. `hyperref` doit être *la dernière* extension chargée dans le préambule. Cette extension convertit automatiquement en liens actifs tout ce qui est référencé en L^AT_EX : table des matières, index, notes marginales ou de bas de page, référence à des équations ou à des pages, etc.

De très nombreuses options sont disponibles pour paramétrer le fonctionnement d'`hyperref`, on les spécifie au choix

⁵. Il est naturel d'utiliser la couleur (extension `color`) pour des textes à afficher sur le WEB !

– soit entre crochets à l’appel d’hyperref :

```
\usepackage[pdftex]{hyperref}
```

– soit immédiatement après, par la commande `\hypersetup{options}`.

La seule option indispensable est `pdftex`, les autres permettent d’infléchir les réglages par défaut d’hyperref en ce qui concerne la présentation du document, les informations globales ou les liens, en voici quelques-unes (les valeurs par défaut sont en italique) :

`bookmarks (=true, false)` commande l’affichage ou non d’une colonne de signets (créés à partir de la table des matières) ;

`pdftoolbar (=true, false)` commande l’affichage ou non de la barre de menu d’Acrobat ;

`pdfmenubar (=true, false)` commande l’affichage ou non de la barre d’outils d’Acrobat ;

`pdffitwindow (=true, false)` ajuste ou non la taille de la fenêtre ouverte par Acrobat à celle du document ;

`pdftitle (=texte)` titre qui apparaît dans la rubrique Document Info du menu File d’Acrobat ;

`pdfauthor (=texte)` nom de l’auteur qui apparaît dans la même rubrique ;

`pdfnewwindow (=true, false)` ouvre ou non une nouvelle fenêtre lorsqu’on clique sur un lien qui conduit à un autre fichier (n’a aucun effet sur la navigation dans un même fichier) ;

`colorlinks (=true, false)` affiche les zones actives (liens) en couleur (par défaut ces zones sont encadrées) ; les couleurs dépendent du type de lien, elles peuvent être modifiées par les options :

`linkcolor (=color, red)` couleur des liens internes (sections, pages, etc.),

`citecolor (=color, green)` couleur des citations bibliographiques,

`filecolor (=color, magenta)` couleur des liens vers des fichiers *locaux*,

`urlcolor (=color, cyan)` couleur liens vers des URL externes.

Exemples :

1. Pour accepter tous les réglages par défaut d’hyperref on codera

```
\usepackage[pdftex]{hyperref}
```

2. Si on veut ajouter une colonne de signets et colorier les liens (les valeurs `=true` peuvent être omises) :

```
\usepackage[pdftex,bookmarks,colorlinks]{hyperref}
```

3. Pour un texte à imprimer en noir et blanc on choisit en général de ne pas encadrer les liens, mais on leur affecte la couleur noire car les niveaux de gris gênent la lecture :

```
\usepackage{hyperref}
```

```
\hypersetup{colorlinks,%
             citecolor=black,%
```

```
filecolor=black,%
linkcolor=black,%
urlcolor=black,%
pdftex}
```

4. Pour seulement renseigner la rubrique Document Info :

```
\usepackage[pdauthor={Pierre Desproges}%
pdftitle={Des femmes qui tombent},%
pdftex]{hyperref}
```

En plus des liens créés automatiquement par `hyperref` à partir des références croisées, il peut être utile de créer des liens spécifiques, ceci se fait avec la commande

`\href{url}{text}`

Exemples :

1. Le codage

L'association `\href{http://www.gutenberg.eu.org}{GUTenberg}` produira dans le fichier PDF « L'association GUTenberg », un clic sur le mot GUTenberg conduira au site de l'association.

2. Lorsque la destination du lien n'est pas une URL mais un fichier disponible localement, on peut encore utiliser `\href` :

La documentation complète est `\href{manual.pdf}{ici}`
On lira dans le fichier PDF « La documentation complète est ici », un clic sur le mot ici ouvrira le fichier `manual.pdf` du répertoire courant (le chemin est relatif au document courant).

3. L'auteur d'un article peut permettre à ses lecteurs de le joindre facilement par courrier électronique en utilisant `\href` dans la commande `\author` :

```
\author{\href{mailto:Paul.Tron@free.fr}{Paul Tron}}
```

La commande `\maketitle` (voir page 17) affichera le nom de l'auteur, il suffira au lecteur de cliquer sur celui-ci pour ouvrir une fenêtre contenant un formulaire de courrier électronique prêt à être adressé à l'auteur de l'article.

5.2.4. Problèmes possibles avec les liens

Des messages du genre suivant :

```
! pdfTeX warning (ext4): destination with the same identifier
(name{page.1}) has been already used, duplicate ignored
```

surviennent lorsqu'un compteur a été réinitialisé, par exemple dans la classe `book` par `\mainmatter` : le numéro de page est remis à 1 au début du premier

chapitre, mais la préface a aussi une page 1... Le problème est qu'une référence ultérieure à la page 1 du premier chapitre pointera sur la page 1 de la préface (c'est ce que dit le message « `duplicate ignored` »).

La parade consiste à ajouter `plainpages=false, pdfpagelabels` dans les options de l'extension `hyperref`, soit entre crochets, soit dans `\hypersetup{}`. Mais ceci ne fonctionne que pour le compteur de pages. Une solution plus radicale consiste à utiliser l'option `hypertextnames=false`, mais dans ce cas les références de l'index cessent d'être actives.

5.2.5. Problèmes avec les signets (*bookmarks*)

Les signets créés par `hyperref` à partir des commandes \LaTeX de sectionnement ne sont pas toujours conformes à ce qu'on attend, ceci est dû à des incompatibilités entre le format PDF et le mode de fonctionnement de \LaTeX . Lorsque ceci arrive, l'utilisateur en est averti par le message

```
Package hyperref Warning:
Token not allowed in a PDFDocEncoded string:
```

On est donc amené dans ce cas à prévoir deux codages différents, un pour \LaTeX et un pour PDF, c'est l'objet de la commande

```
\texorpdfstring{texte  $\TeX$ }{texte PDF}
```

Exemples :

1. Les expressions mathématiques ne passent pas dans les signets, on code par exemple :

```
\section{\texorpdfstring{\$E=mc^2\$}{E\ =\ mc\textttwosuperior}}
```

car `\section{\$E=mc^2\$}` produirait un signet de la forme « E=mc2 ».
2. Les changements de couleur ne passent pas non plus,

```
\section{\textcolor{red}{Rouge !}}
```

produirait un signet de la forme « redRouge! »: la commande `\textcolor` serait ignorée, son argument (red) serait imprimé et l'espace avant le '!' avalé. On code donc

```
\section{\texorpdfstring{\textcolor{red}{Rouge !}}{Rouge\ !}}
```
3. Lorsqu'on utilise l'extension `aeguill` les guillemets français sont remplacés dans les signets par leurs codes 174 et 175! En revanche, avec de vraies fontes à 256 caractères, il n'y a pas de problème⁶. Si on travaille avec `aeguill` on codera :

```
\section{\texorpdfstring{\og Guillemets \fg}%
{\string « Guillemets \string »}}
```

6. Avec $\MIT\TeX$ et `babel/frenchb`, pas de problème non plus.

4. Les ligatures œ et Œ disparaissent des signets, voici une parade un peu grossière (ligatures supprimées dans les signets) :
- ```
\section{\texorpdfstring{\OE UF, c\oe ur}{OEUF, coeur}}
```

### 5.2.6. Fichiers sources compatibles L<sup>A</sup>T<sub>E</sub>X et pdfL<sup>A</sup>T<sub>E</sub>X

L'idéal est d'avoir des fichiers sources qui se compilent aussi bien en L<sup>A</sup>T<sub>E</sub>X qu'en pdfL<sup>A</sup>T<sub>E</sub>X. En ce qui concerne les inclusions d'images, la solution est immédiate : il suffit d'*omettre systématiquement* le suffixe des noms de fichiers inclus par `\includegraphics` : lors d'une compilation en L<sup>A</sup>T<sub>E</sub>X, les fichiers de suffixe `.eps` seront inclus, tandis que pdfL<sup>A</sup>T<sub>E</sub>X inclura lui, les fichiers de suffixe `.png`, `.pdf`, `.jpg`, `.mps` ou `.tif` (par ordre décroissant de priorité).

Pour le reste (fontes spécifiques pour pdfL<sup>A</sup>T<sub>E</sub>X, appel ou non à `hyperref`, etc.) le plus simple est de faire appel à l'extension `ifpdf` qui définit une instruction conditionnelle `\ifpdf` qui sera *vraie* lorsqu'on cherchera à produire un fichier PDF (compilation en pdfL<sup>A</sup>T<sub>E</sub>X) et *fausse* sinon. On peut alors coder dans le préambule :

```
\usepackage{ifpdf}
\ifpdf
 \usepackage{aeguill}
 \usepackage[pdftex]{graphicx,color}
 \usepackage[pdftex]{hyperref}
\else
 \usepackage[T1]{fontenc}
 \usepackage[dvips]{graphicx}
\fi
```

Lors d'une compilation en L<sup>A</sup>T<sub>E</sub>X, seules les deux instructions suivant le `\else` seront exécutées, tandis qu'en pdfL<sup>A</sup>T<sub>E</sub>X, seules les trois instructions précédant le `\else` le seront.

Noter que, dans les installations récentes (TeXLive 6 et suivantes par exemple), le choix du pilote (`pdftex` ou `dvips`) à l'appel des extensions `graphicx` et `color` est fait automatiquement par les fichiers de configuration `graphics.cfg` et `color.cfg`.

### 5.3. Présentations à l'aide de pdfscreen

Les présentations de travaux scientifiques peuvent être faites soit au tableau noir, sous forme de transparents, soit de plus en plus fréquemment à l'aide d'un ordinateur relié à un projecteur vidéo.

pdfL<sup>A</sup>T<sub>E</sub>X, associé à l'extension pdfscreen permet de réaliser des présentations en PDF, aussi vivantes que celles produites par *PowerPoint*, mais *bien plus portables* car leur format (PDF) est indépendant du système d'exploitation utilisé.

L'extension pdfscreen charge graphicx, color et hyperref avec des options bien adaptées aux présentations sur écran, La figure 5.1 contient un exemple de fichier minimal à compiler avec pdfL<sup>A</sup>T<sub>E</sub>X et le résultat produit.

Pour réaliser ce type de document, on travaille en général en classe `article`. On commence par charger les extensions liées au contenu du document (ici seulement de quoi produire une présentation en français). On charge ensuite pdfscreen avec ses options

`screen` : présentation sur écran (l'alternative est `print` pour sortie sur imprimante);  
`panelright` panneau de navigation à droite, ou `panelleft` si on le veut à gauche (par défaut, il n'y a pas de panneau de navigation);  
`french` les noms des boutons du panneau de navigation seront en français<sup>7</sup>, (en anglais par défaut);  
`chocolate` jeu de couleurs du panneau de navigation, les autres choix possibles sont `gray`, `orange`, `palegreen`, `bluelace` et `blue` qui est le jeu de couleurs par défaut.

On précise ensuite le format d'affichage sur écran grâce aux commandes `\panelwidth` (un seul argument : la largeur du panneau de navigation), `\screensize` (deux arguments : hauteur et largeur de l'écran, panneau de navigation compris), `\marginwidth` (quatre arguments : marges gauche, droite, haute et basse du document; dans l'exemple, le document n'est pas centré horizontalement car il faut tenir compte des numéros de sections et sous-sections qui sont placés dans la marge gauche, il y a donc lieu d'augmenter celle-ci.

Il est possible d'ajouter un fond d'écran sous forme d'une image PDF, PNG ou JPG par la commande `\overlay{image}`, ou bien un fond uni dont on spécifie la couleur par la commande `\background{couleur}` (ce qui a été fait ici).

La commande `\emblem{logo}` permet d'ajouter un logo dans le panneau de navigation (*logo* doit être un fichier d'image reconnu : PDF, PNG, JPG...).

---

7. Cette option est *sans effet* sur le document lui-même.

---

```

\documentclass[pdftex,12pt]{article}
%%
\usepackage[latin1]{inputenc}
\usepackage[english,frenchb]{babel}
\usepackage{aeguill}
%%
\usepackage[screen,panelright,french,chocolate]{pdfscreen}
% Format de la page écran
\panelwidth=25mm
\screensize{150mm}{200mm} %%\screensize{height}{width}
\marginsize{42mm}{8mm}{10mm}{10mm}%%left, right, top, bottom
% Couleur ou image de fond d'écran
\overlayempty
\definecolor{mybg}{rgb}{1,0.9,0.7}
\backgroundcolor{mybg}
% Logo éventuel
\emblem{USTL}
%%
\usepackage{pause}
%%
\begin{document}
\begin{enumerate}
\item Bonne nouvelle\dots \pause
\item Mauvaise nouvelle
\end{enumerate}
\end{document}

```

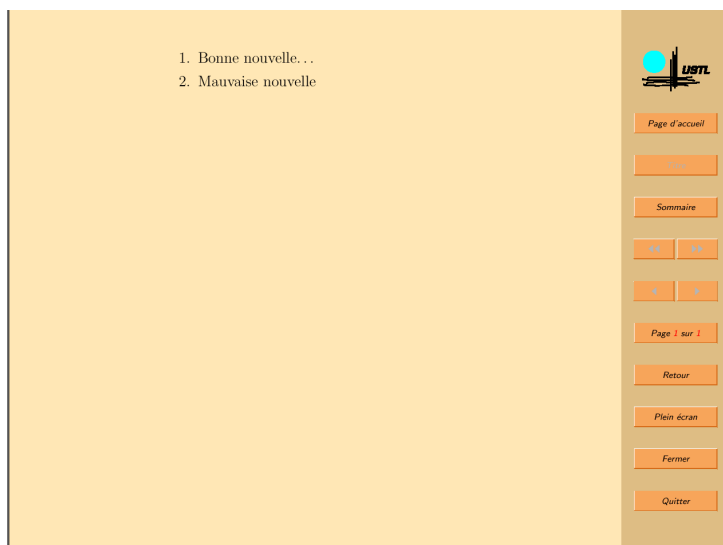


FIGURE 5.1 – Écran produit par le document ci-dessus

Si l'auteur ne souhaite pas révéler le contenu complet d'un transparent mais préfère l'afficher progressivement au rythme de son exposé, il devra

1. Faire appel à l'extension `pause` dans le préambule.
2. Ajouter une commande `\pause` à chaque point d'arrêt souhaité (ici entre les deux *item* de la liste).
3. Faire retraiter par `ppower4`<sup>8</sup> (*P<sup>4</sup> : Pdf Presentation Post-Processor*) le fichier `xy.pdf` produit par `pdfLATEX` :  
`ppower4 xy.pdf xyz.pdf`

Lors de la présentation de `xyz.pdf`, chaque appui sur la touche `<return>` dévoilera le morceau suivant du transparent<sup>9</sup>.

Pour contrôler le contenu de chaque transparent, on dispose, comme avec les classes `slides` et `seminar`, de l'environnement `\begin{slide} ... \end{slide}`. Le texte contenu dans un environnement `slide` est présenté centré verticalement dans la page, à condition bien sûr qu'il tienne dans le transparent.

Le panneau de navigation comporte par défaut les boutons suivants :

`Page d'accueil` conduit à l'URL indiquée en argument de la commande `\urlid{}`;

`Titre` conduit à la première page du document ;

`Sommaire` conduit à la table des matières ;

`<<`, `>>`, `<`, `>` conduisent respectivement à la première page, à la dernière, à la précédente ou à la suivante (par rapport à la page courante) ;

`Page n sur N` indique où on se trouve dans le document et permet de sélectionner le numéro de la prochaine page à afficher<sup>10</sup> ;

`Retour` revient à la page précédemment visitée (utile pour revenir au point de départ après avoir suivi un lien par exemple) ;

`Plein écran` bascule l'affichage en plein écran ou dans une fenêtre de dimensions plus réduites ;

`Fermer` ferme le document ;

`Quitter` ferme la fenêtre et quitte Acrobat Reader.

La compilation de l'exemple ci-dessus produit le message d'erreur :

8. `ppower4` est un programme écrit en `java`, il requiert la présence d'un interpréteur `java`.

9. En fait, `ppower4` crée, pour chaque commande `\pause`, une page intermédiaire contenant uniquement le texte qui la précède. Ainsi dans l'exemple présenté, le fichier `xy.pdf` a une seule page, tandis que `xyz.pdf` en a trois. Ceci n'a aucune influence sur les liens : ils fonctionnent toujours, y compris ceux qui font référence à un numéro de page.

10. La sélection d'une page par son numéro ne fonctionne pas dans les fichiers créés par `ppower4` : les valeurs de *n* et *N* sont celles du fichier PDF initial, mais `ppower4` a ajouté entre-temps des pages intermédiaires...

```
! pdfTeX warning (dest): name{contents} has been referenced
 but does not exist, replaced by a fixed one
```

ce qui est normal puisque le bouton Sommaire du panneau de navigation, qui devrait mener à la table des matières, ne pointe sur rien : il n'y a pas de commande `\tableofcontents` dans le document !

Si on préfère que la table des matières soit affichée dans le panneau de navigation (ce qui suppose que les entrées ne soient ni trop nombreuses ni trop longues<sup>11</sup>), il suffit d'ajouter `paneltoc` à la liste des options de `pdfscreen`.

Il est possible de personnaliser et même de redessiner complètement le panneau de navigation. Des effets spéciaux sont également prévus pour accompagner les changements de page... Consulter pour cela le manuel de `pdfscreen` qui existe en deux versions : à imprimer (`manual-print.pdf`), ou à visualiser à l'écran (`manual-screen.pdf`), ces deux fichiers accompagnent normalement la distribution `pdfscreen`.

---

11. Penser à utiliser l'argument optionnel `[titre court]` des commandes de sectionnement, voir page 29.

---

## Chapitre 6

# Personnalisation de L<sup>A</sup>T<sub>E</sub>X

---

Les documents produits avec les commandes que vous avez apprises jusqu'ici sont déjà de très bonne qualité. Même s'ils manquent de fantaisie, ils obéissent à toutes les règles de l'art de la typographie, ce qui les rend agréables à lire.

Mais il y a des situations où L<sup>A</sup>T<sub>E</sub>X ne propose pas de commande ni d'environnement adapté à vos besoins, ou bien où le résultat produit par une commande existante ne répond pas à votre attente.

Dans ce chapitre, vous allez apprendre à définir de nouvelles commandes L<sup>A</sup>T<sub>E</sub>X et à modifier la présentation par défaut.

### 6.1. Vos propres commandes, environnements et extensions

Vous avez sûrement constaté que toutes les commandes décrites dans cet ouvrage sont présentées dans un cadre et sont référencées dans l'index qui se trouve à la fin. Au lieu d'utiliser à chaque fois l'ensemble des commandes L<sup>A</sup>T<sub>E</sub>X nécessaires, nous avons créé une extension dans laquelle nous avons défini de nouvelles commandes et de nouveaux environnements adaptés à cet usage. Ainsi nous pouvons simplement écrire :

```
\begin{lsccommand}
\ci{dum}
\end{lsccommand}
```



\dum

Dans cet exemple, nous utilisons à la fois un nouvel environnement appelé `lsccommand` qui est responsable du tracé du cadre et une nouvelle commande appelée `\ci` qui compose le nom de la commande et qui produit l'entrée correspondante dans l'index. Vous pouvez le vérifier en cherchant la commande `\dum` dans l'index à la fin de ce document ; vous y trouverez une entrée pointant vers chaque page où la commande `\dum` est mentionnée.

Si nous décidons un jour que nous ne souhaitons plus voir de cadre autour du nom des commandes, il nous suffira de modifier la définition de l'environnement

`\scommand` pour définir un nouveau style. C'est bien plus simple (et efficace) que de parcourir tout le document pour remplacer une à une toutes les commandes qui tracent les cadres.

### 6.1.1. Nouvelles commandes

Pour définir de nouvelles commandes, utilisez la commande :

```
\newcommand{nom}[num]{définition}
```

Cette commande prend deux arguments obligatoires : le *nom* de la commande à créer et sa *définition*. L'argument *num* entre crochets est optionnel ; il indique le nombre de paramètres qu'utilisera la nouvelle commande (au plus 9) ; s'il est omis, la commande n'accepte pas d'argument.

Les deux exemples ci-dessous vous aiderons à saisir le principe. Le premier exemple définit une nouvelle commande appelée `\ucil` qui est une abréviation de « une courte introduction à L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> ». Une telle commande pourrait être utile si vous aviez à citer de nombreuses fois le titre de ce livre.

```
\newcommand{\ucil}
 {Une courte (?)
 introduction à \LaTeXe}
% dans le document :
Voici \og \ucil \fg \dots
```

```
Voici « Une courte (?) introduction à
LATEX 2ε »...
```

L'exemple suivant montre comment utiliser l'argument *num*. La séquence `#1` est remplacée par le paramètre réel. Pour utiliser plus d'un paramètre, continuez avec `#2`, etc.

```
\newcommand{\uxil}[1]
 {Une \emph{#1}
 introduction à \LaTeXe}
% dans le document :
\begin{itemize}
\item \uxil{courte}
\item \uxil{rapide}
\end{itemize}
```

```
- Une courte introduction à LATEX 2ε
- Une rapide introduction à LATEX 2ε
```

`\newcommand` ne peut redéfinir une commande existante, L<sup>A</sup>T<sub>E</sub>X vous en averti pas un message d'erreur si vous essayez de le faire. Si vous voulez explicitement remplacer une commande existante, utilisez `\renewcommand` qui utilise la même syntaxe que `\newcommand`.

Dans certains cas, vous aurez peut-être besoin de `\providecommand`. Elle fonctionne comme `\newcommand`, mais au lieu d'envoyer un message d'erreur lorsqu'on tente d'écraser une commande déjà définie, elle ignore simplement la nouvelle définition.

Rappel : L<sup>A</sup>T<sub>E</sub>X supprime en général les espaces qui suivent une commande, (voir page 5).

### 6.1.2. Nouveaux environnements

Il est également possible de définir de nouveaux environnements. La commande `\newenvironment` se présente de la manière suivante :

```
\newenvironment{nom}[num]{avant}{après}
```

Comme `\newcommand`, `\newenvironment` autorise l'emploi de paramètres dont le nombre est donné par l'argument optionnel *num*. Le contenu de l'argument *avant* est exécuté avant que le contenu de l'environnement ne soit traité. Le contenu de l'argument *après* est traité lorsque l'on rencontre la commande `\end{nom}`. Les paramètres éventuels ne peuvent être utilisés que dans *avant*.

L'exemple ci-dessous illustre l'utilisation de `\newenvironment`.

```
\newenvironment{king}
{\rule{1ex}{1ex}%
 \hspace{\stretch{1}}}
{\hspace{\stretch{1}}%
 \rule{1ex}{1ex}}
```

■ Mes chers sujets, ... ■

```
\begin{king}
Mes chers sujets, \dots
\end{king}
```

Les commandes utilisées dans l'exemple ci-dessus seront présentées plus loin : pour la commande `\rule`, voir page 103, pour `\stretch`, voir page 95 enfin, pour plus d'informations sur `\hspace`, voir page 95.

`\newenvironment` n'autorise pas la redéfinition d'un environnement existant. Pour redéfinir un environnement existant, utilisez `\renewenvironment` qui utilise la même syntaxe que `\newenvironment`.

### 6.1.3. Votre propre extension

Si vous définissez plein de nouveaux environnements et de nouvelles commandes, le préambule de votre document va s'allonger dangereusement. Il peut

alors devenir intéressant de créer une extension contenant toutes ces nouvelles définitions. Avec la commande `\usepackage` vous pourrez faire appel à votre extension dans votre document.

Écrire une extension consiste principalement à copier le préambule de votre document dans un fichier à part, dont le nom se termine par `.sty`. Il y a une commande spécifique à utiliser sur la première ligne de votre extension :

```
\ProvidesPackage{nom de l'extension}
```

`\ProvidesPackage` indique à  $\text{\LaTeX} 2_{\epsilon}$  le nom de l'extension afin notamment de lui permettre de produire des messages d'erreur significatifs. La figure 6.1 montre un exemple d'extension simple qui reprend les commandes définies dans les exemples précédents.

---

```
% Exemple d'extension par Tobias Oetiker
\ProvidesPackage{demopack}
\newcommand{\ucil}{Une courte (?) introduction à \LaTeXe}
\newcommand{\uxil}[1]{Une \emph{#1}
 introduction à \LaTeXe}
\newenvironment{king}{\begin{quote}}{\end{quote}}
```

---

FIGURE 6.1 – Exemple d'extension

## 6.2. Polices et tailles des caractères

### 6.2.1. Commandes de changement de police

$\text{\LaTeX}$  choisit la police de caractères et sa taille en fonction de la structure logique du document (sections, notes de bas de page...). Dans certains cas, on voudrait pouvoir changer la taille de la police à la main. Pour cela, utilisez les commandes listées dans les tableaux 6.1 et 6.2. La taille exacte de chaque police est un choix qui dépend de la classe de document et de ses options. La table 6.3 donne les tailles absolues en points pour les commandes présentes dans les classes de document standard.

```
{\small Les romains
petits et \textbf{gras}
régèrent sur}
{\Large la belle
et grande \textit{Italie}.}
```

Les romains petits et **gras** régèrent sur  
la belle et grande *Italie*.

Une caractéristique importante de L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> est que les différents attributs d'une police peuvent être modifiés indépendamment les uns des autres. Cela signifie que vous pouvez exécuter des commandes de changement de taille ou même de changement de police tout en conservant l'attribut gras ou italique.

En mode *mathématique*, vous pouvez utiliser les commandes de changement de police pour quitter provisoirement le mode mathématique et saisir du texte normal. Pour changer les attributs de la police en mode mathématique, il existe un jeu de commandes spéciales. Reportez-vous au tableau 3.19 page 61.

Lors des changements de taille, les accolades jouent un rôle essentiel. Elles sont utilisées pour former des *groupes* qui limitent la portée de la plupart des commandes de L<sup>A</sup>T<sub>E</sub>X.

TABLE 6.1 – Polices

---

|                           |                   |                               |                  |
|---------------------------|-------------------|-------------------------------|------------------|
| <code>\textrm{...}</code> | roman             | <code>\textsf{...}</code>     | sans serif       |
| <code>\texttt{...}</code> | typewriter        |                               |                  |
| <code>\textmd{...}</code> | medium            | <code>\textbf{...}</code>     | <b>bold face</b> |
| <code>\textup{...}</code> | upright           | <code>\textit{...}</code>     | <i>italic</i>    |
| <code>\textsl{...}</code> | <i>slanted</i>    | <code>\textsc{...}</code>     | SMALL CAPS       |
| <code>\emph{...}</code>   | <i>emphasised</i> | <code>\textnormal{...}</code> | document font    |

---

TABLE 6.2 – Tailles des polices

---

|                            |             |                     |            |
|----------------------------|-------------|---------------------|------------|
| <code>\tiny</code>         | minuscule   | <code>\Large</code> | plus grand |
| <code>\scriptsize</code>   | très petit  | <code>\LARGE</code> | très grand |
| <code>\footnotesize</code> | assez petit | <code>\huge</code>  | énorme     |
| <code>\small</code>        | petit       | <code>\Huge</code>  | géant      |
| <code>\normalsize</code>   | normal      |                     |            |
| <code>\large</code>        | grand       |                     |            |

---

TABLE 6.3 – Tailles en points dans les classes standard

| taille                     | 10pt (défaut) | option 11pt | option 12pt |
|----------------------------|---------------|-------------|-------------|
| <code>\tiny</code>         | 5pt           | 6pt         | 6pt         |
| <code>\scriptsize</code>   | 7pt           | 8pt         | 8pt         |
| <code>\footnotesize</code> | 8pt           | 9pt         | 10pt        |
| <code>\small</code>        | 9pt           | 10pt        | 11pt        |
| <code>\normalsize</code>   | 10pt          | 11pt        | 12pt        |
| <code>\large</code>        | 12pt          | 12pt        | 14pt        |
| <code>\Large</code>        | 14pt          | 14pt        | 17pt        |
| <code>\LARGE</code>        | 17pt          | 17pt        | 20pt        |
| <code>\huge</code>         | 20pt          | 20pt        | 25pt        |
| <code>\Huge</code>         | 25pt          | 25pt        | 25pt        |

Il aime les `{\LARGE` grands et  
`{\small` les petits}  
 caractères}.

Il aime les grands et les petits  
 caractères.

Les commandes de changement de taille modifient également l'interligne, mais seulement si le paragraphe se termine dans la portée de la commande de changement de taille, c'est-à-dire si l'accolade fermante `}` est placée après la fin de paragraphe. Comparez l'interligne dans les deux exemples suivants qui ne diffèrent que par la position de la commande `\par` :

`{\Large` Ne lisez pas ceci ! Ce  
 n'est pas vrai ! Croyez-moi !`\par}`

Ne lisez pas ceci ! Ce n'est  
 pas vrai ! Croyez-moi !

`{\Large` Ne lisez pas ceci ! Ce  
 n'est pas vrai ! Croyez-moi !`}``\par`

Ne lisez pas ceci ! Ce n'est  
 pas vrai ! Croyez-moi !

Si vous voulez utiliser une commande de modification de la taille pour tout un paragraphe ou même plus, vous pouvez utiliser la syntaxe des environnements à la place de celle des commandes.

```
\begin{Large}
Ceci n'est pas vrai
Mais, qu'est-ce qui l'est
de nos jours\dots
\end{Large}
```

Ceci n'est pas vrai Mais,  
qu'est-ce qui l'est de nos  
jours...

### 6.2.2. Attention danger

Il est maladroit d'utiliser, de manière répétitive, de telles commandes de changement explicite de police, en effet ces commandes vont à l'encontre de la philosophie de L<sup>A</sup>T<sub>E</sub>X qui est de séparer les aspects logiques et visuels d'un document.

Pour préserver le balisage logique, il est préférable de définir une nouvelle commande qui effectuera le changement de style souhaité. Supposons que vous vouliez mettre en valeur les mots liés au concept de danger, vous pourriez procéder ainsi :

```
% dans le préambule ou dans
% une extension :
\newcommand{\danger}[1]
 {\textbf{#1}}
% dans le document :
Défense d'\danger{entrer}.
Cette pièce contient une
\danger{machine} d'origine
inconnue.
```

Défense d'**entrer**. Cette pièce contient  
une **machine** d'origine inconnue.

Cette approche sera précieuse si vous décidez plus tard d'utiliser une autre représentation typographique du concept danger que le changement de gras. Elle évitera d'avoir à rechercher et remplacer une à une toutes les occurrences de `\textbf` correspondant à la notion de danger, il suffira de modifier la commande `\danger`.

### 6.2.3. Un conseil

Pour conclure cette promenade au pays des commandes de changement de police, voici un (mauvais) conseil :

---

**N'oubliez pas !** Plus vous utilisez de polices dans un document, plus il DEVIENT *lisible et bien présenté*.

## 6.3. Espacement

### 6.3.1. Entre les lignes

Pour obtenir un interligne plus grand pour un document, vous pouvez utiliser la commande

```
\linespread{facteur}
```

dans le préambule de votre document. Utilisez `\linespread{1.3}` pour un interligne « un et demi » et `\linespread{1.6}` pour un « double » interligne. L'interligne par défaut est 1.

### 6.3.2. Mise en page d'un paragraphe

Il y a deux paramètres qui jouent sur l'apparence d'un paragraphe. En insérant une définition telle que :

```
\setlength{\parindent}{0pt}
\setlength{\parskip}{1ex plus 0.5ex minus 0.2ex}
```

dans le préambule, vous supprimez le retrait des débuts de paragraphe (1<sup>re</sup> définition) et vous augmentez l'espace entre deux paragraphes (2<sup>e</sup> définition) de `1ex` (voir la signification de `ex` dans le tableau 6.4 page 96). Les `plus 0.5ex` et `minus 0.2ex` autorisent  $\TeX$  à dilater (de maximum `0.5ex`) ou comprimer (de maximum `0.2ex`) l'espace vertical entre les paragraphes si ceci améliore la mise en page. Attention, la deuxième définition a également une influence sur la table des matières : ses lignes deviennent également plus espacées. Pour éviter cela, vous pouvez déplacer ces commandes du préambule vers le corps du document, après la commande `\tableofcontents` (ou bien ne pas les utiliser du tout, car de nombreux typographes préfèrent un retrait en début de paragraphe à une augmentation de l'espacement, pour séparer les paragraphes).

Pour forcer un retrait au début d'un paragraphe, utilisez la commande

```
\indent
```

au début du paragraphe<sup>1</sup>. Bien sûr cela ne marche que si `\parindent` n'est pas nul.

Pour créer un paragraphe sans retrait, utilisez :

```
\noindent
```

en tête du paragraphe.

Il est possible de commencer un paragraphe par une lettrine en utilisant l'extension `lettrine` [16] :

```
\lettrine[options]{lettrine}{texte}
```

Celle de la page iii se code : `\lettrine{C}{e document}`

### 6.3.3. Espacement horizontal

$\text{\LaTeX}$  détermine l'espacement entre les mots et les phrases automatiquement. Pour ajouter de l'espacement horizontal, utilisez :

```
\hspace{longueur}
```

Si une telle espace doit être conservée, même lorsqu'elle tombe en début ou en fin de ligne, utilisez `\hspace*`. Dans le cas le plus simple, *longueur* est simplement un nombre suivi d'une unité. Les unités les plus importantes sont listées dans le tableau 6.4.

Ceci `\hspace{1.5cm}` est une espace de 1.5 cm.

```
Ceci est une espace de 1.5 cm.
```

La commande :

```
\stretch{n}
```

---

1. Pour obtenir un retrait systématique au début du premier paragraphe de chaque section, utilisez l'extension `indentfirst` de l'ensemble `tools`. Le recours à `indentfirst` est toutefois inutile en français si on utilise `babel/frenchb`.

TABLE 6.4 – Unités T<sub>E</sub>X

---

|    |                                               |              |
|----|-----------------------------------------------|--------------|
| mm | millimètre                                    | □            |
| cm | centimètre = 10 mm                            | ┌───┐        |
| in | pouce <sup>a</sup> = 25,4 mm                  | ┌──────────┐ |
| pt | point ≈ 1/72 pouce ≈ 0,35 mm                  | ▯            |
| em | largeur d'un cadratin dans la police courante | ┌┐           |
| ex | hauteur d'un « x » dans la police courante    | ┌┐           |

---

*a. inch en anglais. NdT.*

---

produit une espace élastique. Elle se dilate jusqu'à occuper tout l'espace libre disponible sur la ligne. Si deux commandes `\hspace{\stretch{n}}` sont exécutées sur la même ligne, les largeurs de deux espaces sont proportionnelles au facteur d'élasticité  $n$ .

```
x\hspace{\stretch{1}}%
x\hspace{\stretch{3}}x
```



Il est souvent préférable d'exprimer les dimensions des espaces en unités dépendant de la police utilisée plutôt qu'en unités fixes (cm, mm, pt). Les unités **em** (cadratin, soit approximativement la largeur d'un M dans la police courante) et **ex** (demi-cadratin) sont faites pour ça :

```
{\Large}big\hspace{1em}y\\
{\tiny}tin\hspace{1em}y
```



### 6.3.4. Espacement vertical

L'espacement vertical entre les paragraphes, sections, sous-sections, etc. est déterminé automatiquement par  $\text{\LaTeX}$ . En cas de besoin, de l'espace supplémentaire *entre deux paragraphes* peut être inséré avec la commande :

```
\vspace{longueur}
```

Cette commande doit normalement être utilisée entre deux lignes vides. Si l'espacement doit être conservé en haut ou en bas d'une page, utilisez la version étoilée de la commande `\vspace*`.

La commande `\stretch` en association avec `\pagebreak` permet d'imprimer du texte sur la dernière ligne d'une page ou de centrer verticalement du texte sur une page : `\vspace{\stretch{n}}` agit comme un ressort vertical de puissance proportionnelle à  $n$ .

Du texte \dots

```
\vspace{\stretch{1}}
```

Ceci sera imprimé sur la dernière ligne.\pagebreak

De l'espace supplémentaire entre deux lignes du *même* paragraphe ou à l'intérieur d'une table peut être obtenu par la commande :

```
\[longueur]
```

Les *longueurs* peuvent s'exprimer en unités absolues (`\[1cm]`, `\[3mm]`) ou relatives à la police de caractère courante : `\[2ex]`, (deux fois la hauteur de la lettre  $x$ ), `\space{0.5\baselineskip}` (la moitié de l'interligne courant).

Les commandes `\smallskip`, `\medskip` et `\bigskip` ajoutent des espaces verticaux élastiques dont les valeurs moyennes sont respectivement 1, 2 et 4 mm environ.

## 6.4. Disposition d'une page

Les dimensions des marges et des autres paramètres régissant la présentation d'un document sont calculées automatiquement en fonction, notamment, des options passées à la commande `\documentclass` (`12pt`, `a4paper`, `b5paper`, etc.). Il arrive que les valeurs prédéfinies ne conviennent pas, la figure 6.2 montre tous les paramètres qui peuvent être modifiés. Cette figure a été réalisée avec l'extension `layout` de l'ensemble `tools`.

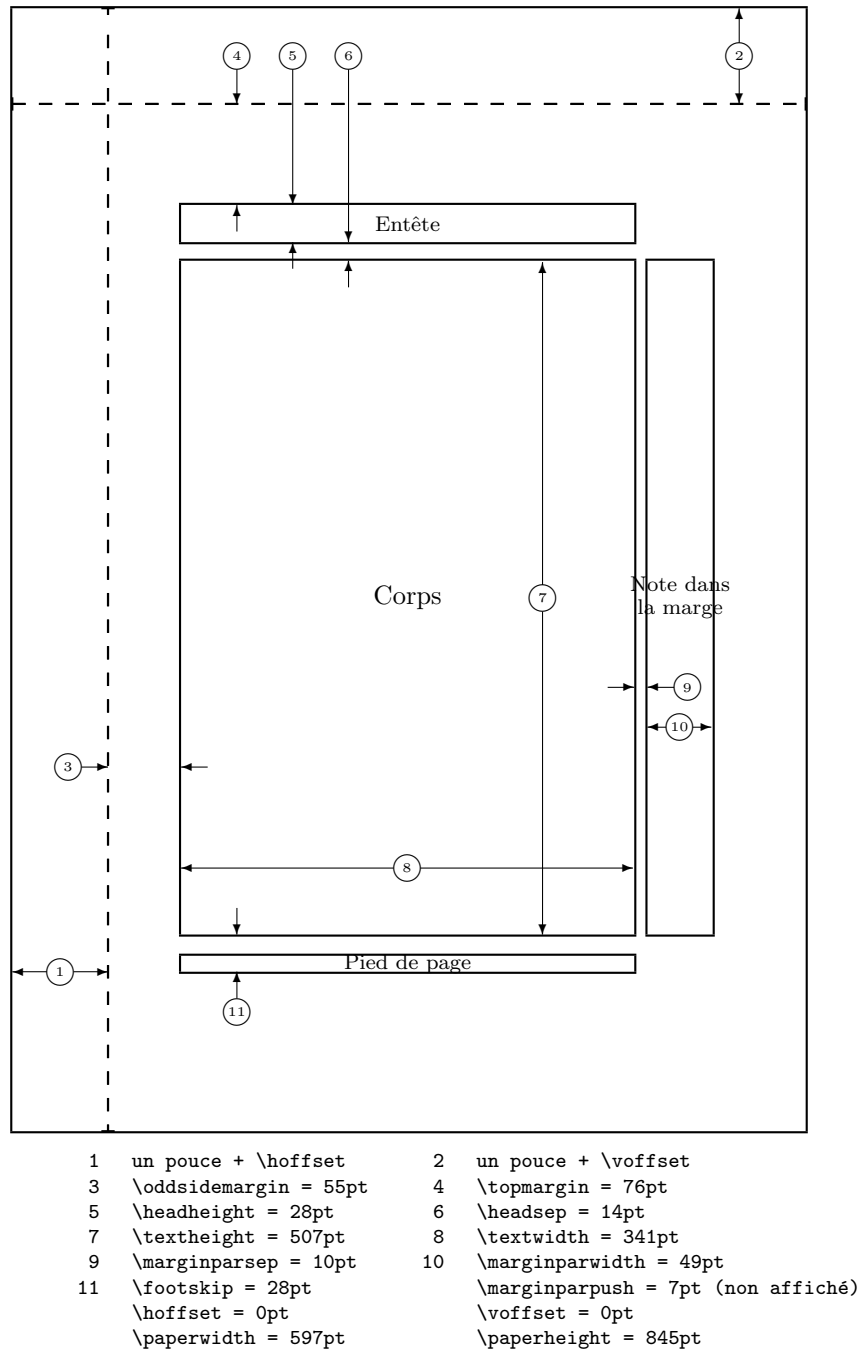


FIGURE 6.2 – Paramètres de la disposition d'une page

**Attendez** avant de vous lancer dans « élargissons un peu ce texte », prenez le temps de réfléchir ! Comme souvent avec  $\text{\LaTeX}$ , il y a de bonnes raisons pour disposer les pages de cette façon.

Sans doute, comparée avec une page standard produite avec MS Word, une page de  $\text{\LaTeX}$  à l'air anormalement étroite. Mais regardez votre livre préféré<sup>2</sup> et comptez le nombre de caractères sur une ligne normale. Vous verrez qu'il n'y a guère plus de soixante-six caractères par ligne. L'expérience montre qu'un texte devient moins lisible si le nombre de caractères par ligne dépasse trop cette valeur, les lignes trop longues fatiguant l'œil. Ceci explique aussi pourquoi les journaux utilisent plusieurs colonnes.

Ainsi, si vous élargissez le corps du texte, ayez conscience que vous le rendez aussi moins lisible. Ceci dit, si vous tenez vraiment à modifier les paramètres qui contrôlent la disposition d'une page, voici comment procéder.  $\text{\LaTeX}$  dispose de deux commandes pour modifier les dimensions. Elles sont généralement utilisées dans le préambule.

La première commande affecte une valeur fixe à un paramètre de dimension :

```
\setlength{paramètre}{longueur}
```

La seconde ajoute une longueur à un paramètre de dimension :

```
\addtolength{paramètre}{longueur}
```

La seconde commande est en pratique plus utile que `\setlength`, parce qu'elle permet de travailler en relatif par rapport à la taille par défaut. Pour ajouter un centimètre à la largeur du texte, tout en gardant le texte centré, nous utiliserions les commandes suivantes dans le préambule :

```
\addtolength{\hoffset}{-0.5cm}
\addtolength{\textwidth}{1cm}
```

Dans ce contexte, il peut être intéressant de faire appel à l'extension `calc`, qui permet d'utiliser des expressions algébriques traditionnelles pour ajouter, soustraire, multiplier ou diviser des dimensions. Avec l'extension `calc`, la commande suivante est correcte :

```
\setlength{\oddsidemargin}{(\pagewidth-\textwidth)/2-1in}
```

---

2. Un vrai livre, imprimé par un grand éditeur...

## 6.5. Jouons un peu avec les dimensions

Il faut éviter autant que possible de donner des dimensions absolues aux figures, tableaux, etc. Il est préférable de les définir relativement aux dimensions des éléments de base de la page : on choisira `\textwidth` comme largeur d'une figure pour lui faire occuper toute la largeur de la page ; si celle-ci change par la suite la figure s'adaptera automatiquement.

Les trois commandes suivantes permettent de déterminer la largeur, la hauteur et la profondeur<sup>3</sup> d'une chaîne de caractères.

```
\settoheight{variable}{texte}
\settodepth{variable}{texte}
\settowidth{variable}{texte}
```

L'exemple ci-dessous montre une utilisation possible de ces commandes :

```
\flushleft
\newenvironment{vardesc}[1]{%
 \settowidth{\parindent}{#1\ }
 \makebox[Opt][r]{#1\ }}{}
\begin{displaymath}
a^2+b^2=c^2
\end{displaymath}

\begin{vardesc}{0ù}a,
b sont les cotés adjacents à
l'angle droit d'un triangle
rectangle, \par
c est l'hypoténuse du
triangle, \par
d n'est pas utilisé ici.
Étonnant non ?
\end{vardesc}
```

$$a^2 + b^2 = c^2$$

Où  $a$ ,  $b$  sont les cotés adjacents à l'angle droit d'un triangle rectangle,  
 $c$  est l'hypoténuse du triangle,  
 $d$  n'est pas utilisé ici. Étonnant non ?

## 6.6. Boîtes

L<sup>A</sup>T<sub>E</sub>X construit ses pages en empilant des boîtes. Chaque caractère est une petite boîte qui est collée à d'autres boîtes-caractères pour former un mot.

---

<sup>3</sup>. La hauteur concerne la partie située au-dessus de la ligne de base, la profondeur celle située en dessous.

Ceux-ci sont alors assemblés à d'autres mots, avec une colle spéciale qui est élastique pour permettre de comprimer ou d'étirer des suites de mots afin de remplir exactement une ligne sur la page.

Reconnaissons qu'il s'agit d'une description simpliste de ce qui se passe réellement, mais le fait est là : T<sub>E</sub>X travaille avec des boîtes et de la colle. Les caractères ne sont pas les seuls à pouvoir former des boîtes. Virtuellement tout peut être mis dans des boîtes, y compris d'autres boîtes. Chaque boîte est ensuite traitée par L<sup>A</sup>T<sub>E</sub>X comme s'il s'agissait d'un simple caractère.

Dans les chapitres précédents vous avez déjà rencontré quelques boîtes, même si nous ne l'avons pas signalé. L'environnement `\tabular` et la commande `\includegraphics`, par exemple, produisent tous les deux des boîtes. Il est ainsi facile d'aligner deux illustrations ou deux tables côte à côte. Il suffit de s'assurer que la somme de leurs largeurs ne dépasse pas la largeur du texte.

Il est aussi possible d'emballer un paragraphe dans une boîte :

```
\parbox[pos]{largeur}{texte}
```

on peut également utiliser un environnement :

```
\begin{minipage}[pos]{largeur} texte \end{minipage}
```

le paramètre `pos` peut être l'une des lettres `c`, `t` ou `b` pour contrôler l'alignement vertical de la boîte par rapport à la base du texte précédent. `largeur` est une dimension indiquant la largeur de la boîte.

`\parbox` et `minipage` opèrent sur un ou plusieurs paragraphes (on parle de boîtes verticales). Il existe d'autres types de boîtes qui opèrent sur des morceaux de lignes (on parle de boîtes horizontales). L'une d'elles nous est déjà connue. Il s'agit de `\mbox` qui combine simplement une série de boîtes pour en former une nouvelle ; elle peut être utilisée pour empêcher L<sup>A</sup>T<sub>E</sub>X de couper une ligne entre deux mots. La commande `\makebox` généralise `\mbox` (`\mbox` est définie en fait comme une `\makebox` sans argument optionnel).

```
\makebox[largeur][pos]{texte}
```

Le paramètre `largeur` définit la largeur de la boîte vue de l'extérieur<sup>4</sup>. En plus des longueurs usuelles vous pouvez également utiliser `\width`, `\height`,

---

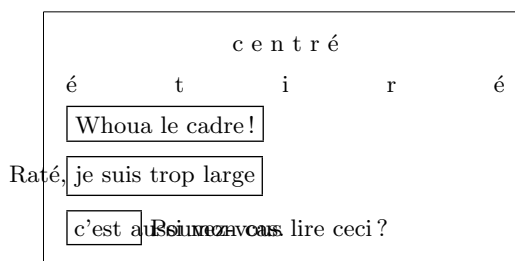
4. Cela signifie qu'elle peut être plus petite que la largeur du contenu de la boîte. Dans un cas extrême on peut même positionner la largeur à `0pt`; ainsi le texte dans la boîte sera placé sans influencer les boîtes adjacentes.

`\depth` et `\totalheight`<sup>5</sup> dans l'expression du paramètre `largeur`. Leurs valeurs sont obtenues à partir des dimensions réelles du `texte`. Le paramètre `pos` est l'une des lettres `c` (`center`) pour centrer le texte, `l` (`left flush`) pour l'aligner à gauche, `r` (`right flush`) pour l'aligner à droite, ou `s` (`spread`) pour le répartir horizontalement dans la boîte.

La commande `\framebox` fonctionne de la même façon que `\makebox`, mais elle ajoute un cadre autour du texte.

L'exemple suivant vous montre des exemples d'utilisation des commandes `\makebox` et `\framebox` :

```
\makebox[\textwidth]{%
 c e n t r é}\par
\makebox[\textwidth][s]{%
 é t i r é}\par
\framebox[1.1\width]{Whoua
 le cadre !} \par
\framebox[0.8\width][r]{Raté,
 je suis trop large} \par
\framebox[1cm][l]{c'est aussi
 mon cas.}
Pouvez-vous lire ceci ?
```

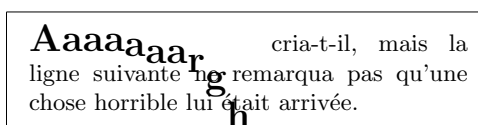


Les boîtes peuvent être déplacées verticalement grâce à la commande

```
\raisebox{élévation}[profondeur][hauteur]{texte}
```

L'argument obligatoire *élévation* fixe l'amplitude du déplacement vertical, les deux arguments optionnels permettent de modifier les dimensions verticales de la boîte. Vous pouvez utiliser `\width`, `\height`, `\depth` et `\totalheight` dans les trois premiers paramètres afin d'agir en fonction de la taille du texte contenu dans la boîte.

```
\raisebox{0pt}[0pt][0pt]{\Large%
\textbf{Aaaa}\raisebox{-0.3ex}{a}%
\raisebox{-0.7ex}{aa}%
\raisebox{-1.2ex}{r}%
\raisebox{-2.2ex}{g}%
\raisebox{-4.5ex}{h}}
cria-t-il, mais la ligne suivante
ne remarqua pas qu'une chose
horrible lui était arrivée.
```



5. `\totalheight` est la somme de `\height` (ce qui est au-dessus de la ligne de base) et de `\depth` (ce qui est en dessous de la ligne de base).

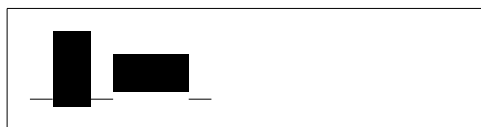
## 6.7. Filets

Quelques pages plus haut vous avez peut-être remarqué la commande :

```
\rule[élévation]{largeur}{hauteur}
```

En utilisation normale, elle produit une simple boîte noire.

```
\rule{3mm}{.1pt}%
\rule[-1mm]{5mm}{1cm}%
\rule{3mm}{.1pt}%
\rule[1mm]{1cm}{5mm}%
\rule{3mm}{.1pt}
```



C'est utile pour produire des filets horizontaux ou verticaux. Le filet horizontal sur la page de titre par exemple a été tracé à l'aide d'une commande `\rule`.

Un cas particulier est celui du filet de largeur nulle et de hauteur spécifiée. En typographie professionnelle ceci est appelé un montant. Ils sont utilisés pour s'assurer qu'un élément aura une hauteur donnée. Vous pouvez par exemple l'utiliser dans un environnement `tabular` pour garantir la hauteur d'une ligne.

```
\begin{tabular}{|c|}
\hline
\rule{1pt}{4ex}Étai\dots\\
\hline
\rule{0pt}{4ex} montant \\
\hline
\end{tabular}
```



Un filet de largeur nulle a également été utilisé page 40 pour construire le carré de 5 centimètres de côté.

FIN



---

## Bibliographie

---

- [1] Leslie Lamport. *L<sup>A</sup>T<sub>E</sub>X: A Document Preparation System*. Addison-Wesley, Reading, Massachusetts, second edition, 1994, ISBN 0-201-52983-1.
- [2] Donald E. Knuth. *The T<sub>E</sub>Xbook*, Volume A of *Computers and Typesetting*, Addison-Wesley, Reading, Massachusetts, second edition, 1984, ISBN 0-201-13448-9.
- [3] Michel Goossens, Frank Mittelbach et Alexander Samarin. *The L<sup>A</sup>T<sub>E</sub>X Companion*. Addison-Wesley, Reading, Massachusetts, 1994, ISBN 0-201-54199-8. Cet ouvrage a été traduit en français (même titre) chez Campus Press, ISBN 2-7440-0897-4.
- [4] Bernard Desgraupes. *L<sup>A</sup>T<sub>E</sub>X Apprentissage, guide et référence*. Vuibert, 2000, ISBN 2-7117-8658-7.
- [5] Christian Rolland. *L<sup>A</sup>T<sub>E</sub>X par la pratique*. O'Reilly, 1999, ISBN 2-84177-073-7.
- [6] L<sup>A</sup>T<sub>E</sub>X3 Project Team. *L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> for authors*. Distribué avec L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> dans `usrguide.tex`.
- [7] L<sup>A</sup>T<sub>E</sub>X3 Project Team. *L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> for Class and Package writers*. Distribué avec L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> dans `clsguide.tex`.
- [8] L<sup>A</sup>T<sub>E</sub>X3 Project Team. *L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> Font selection*. Distribué avec L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> dans `fntguide.tex`.
- [9] David P. Carlisle. *Packages in the 'graphics' bundle*. Distribué avec les extensions « graphics » dans `grfguide.tex`.
- [10] Rainer Schöpf, Bernd Raichle, Chris Rowley. *A New Implementation of L<sup>A</sup>T<sub>E</sub>X's verbatim Environments*. Distribué avec l'ensemble « tools » dans `verbatim.dtx`.
- [11] American Mathematical Society. *AMS-L<sup>A</sup>T<sub>E</sub>X Version 2.0 User's guide*. Distribué avec les extensions AMS-L<sup>A</sup>T<sub>E</sub>X dans `amsl.doc.tex`.
- [12] Kristoffer H. Rose. *Xy-pic User's Guide*. Téléchargeable sur CTAN avec la distribution Xy-pic.
- [13] Bernard Gaille. *Notice d'utilisation du style french multilingue*. Disponible avec le style `french` sur <http://frenchpro.free.fr>.
- [14] Yves Perrousseau. *Manuel de typographie française élémentaire*. Ateliers Perrousseau éditeur, 1995, ISBN 2-911220-00-5.
- [15] Keith Reckdahl. *Using EPS Graphics in L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> Documents* qui explique tout ce que vous avez toujours voulu savoir et même plus sur les fichiers

---

PostScript et leur utilisation avec L<sup>A</sup>T<sub>E</sub>X. Disponible en ligne sur CTAN :  
[/info/epslatex.ps](#)<sup>1</sup>

- [16] Daniel Flipo. Le fichier `lettrine.dtx` est disponible en ligne sur CTAN :  
[CTAN:/macros/latex/contrib/supported/lettrine/](#)<sup>1</sup>.  
Le fichier `OOREADME` qui l'accompagne explique comment installer l'extension et produire la documentation et les exemples d'utilisation.
- [17] John D. Hobby, traduction française de P. Fournier et J.-C. Charpentier.  
« MetaPost », *Cahier GUTenberg* n° 41. Disponible en ligne sur <http://www.gutenberg.eu.org/article88.html>
- [18] FAQ<sup>2</sup> de Marie-Paule Kluth, maintenue par Benjamin Bayart. Disponible en ligne sur <http://www.grappa.univ-lille3.fr/FAQ-LaTeX/>

---

1. Voir page vi la liste des sites CTAN.

2. FAQ signifie *Frequently Asked Questions* ou Foire Aux Questions.

---

# Index

---

**Symboles**

|                                    |                   |
|------------------------------------|-------------------|
| <code>\!</code> .....              | 48                |
| <code>"</code> .....               | 20                |
| <code>\$</code> .....              | 41                |
| <code>\(</code> .....              | 41                |
| <code>\)</code> .....              | 41                |
| <code>\,</code> .....              | 42, 48            |
| <code>-</code> .....               | 20                |
| <code>—</code> .....               | 20                |
| <code>\-</code> .....              | 19                |
| <code>—</code> .....               | 20                |
| <code>—</code> .....               | 20                |
| <code>., espace après</code> ..... | 27                |
| .....                              | 21                |
| <code>.aux</code> .....            | 12                |
| <code>.cls</code> .....            | 11                |
| <code>.dtx</code> .....            | 11                |
| <code>.dvi</code> .....            | 11                |
| <code>.fd</code> .....             | 11                |
| <code>.idx</code> .....            | 12                |
| <code>.ilg</code> .....            | 12                |
| <code>.ind</code> .....            | 12                |
| <code>.ins</code> .....            | 11                |
| <code>.lof</code> .....            | 12                |
| <code>.log</code> .....            | 12                |
| <code>.lot</code> .....            | 12                |
| <code>.sty</code> .....            | 11                |
| <code>.tex</code> .....            | 11                |
| <code>.toc</code> .....            | 12                |
| <code>\@</code> .....              | 27                |
| <code>\[</code> .....              | 41                |
| <code>&amp;</code> .....           | 35                |
| <code>\</code> .....               | 17, 33–35, 97     |
| <code>\*</code> .....              | 17                |
| <code>\]</code> .....              | 41                |
| <code>~</code> .....               | 27                |
| <b>A</b>                           |                   |
| A4 (papier) .....                  | 9, 76             |
| A5 (papier) .....                  | 9                 |
| å .....                            | 23                |
| accent .....                       | 22                |
| aigu .....                         | 23                |
| circonflexe .....                  | 23                |
| grave .....                        | 23                |
| accolade .....                     | 91                |
| accolades .....                    | 5, 47             |
| <code>\addtolength</code> .....    | 99                |
| æ .....                            | 23                |
| aeguill .....                      | 81                |
| <code>align</code> .....           | 50                |
| alignement décimal .....           | 36                |
| <code>ambsy</code> .....           | 54                |
| <code>amfonts</code> .....         | 43, 61            |
| <code>amsmath</code> .....         | 41, 42, 46–52, 54 |
| <code>amssymb</code> .....         | 55, 61            |
| <code>\and</code> .....            | 29                |
| <code>antislash</code> .....       | 5                 |
| <code>\appendix</code> .....       | 28, 30            |
| <code>\ar</code> .....             | 65                |
| <code>\arccos</code> .....         | 46                |
| <code>\arcsin</code> .....         | 46                |
| <code>\arctan</code> .....         | 46                |
| <code>\arg</code> .....            | 46                |
| <code>array</code> .....           | 49                |
| article (classe) .....             | 8                 |
| <code>\author</code> .....         | 29                |
| avantages de $\LaTeX$ .....        | 3                 |
| <b>B</b>                           |                   |
| B5 (papier) .....                  | 9                 |
| <code>babel</code> .....           | 24–26             |
| <code>babel/frenchb</code> .....   | 20, 28, 32        |
| <code>\backmatter</code> .....     | 30                |
| backslash .....                    | 5                 |
| <code>\begin</code> .....          | 31                |

---

|                                     |               |                                     |        |
|-------------------------------------|---------------|-------------------------------------|--------|
| <code>\bibitem</code> .....         | 68            | <code>\]</code> .....               | 41     |
| <code>\Big</code> .....             | 48            | <code>\addtolength</code> .....     | 99     |
| <code>\big</code> .....             | 48            | <code>\and</code> .....             | 29     |
| <code>\Bigg</code> .....            | 48            | <code>\appendix</code> .....        | 28, 30 |
| <code>\bigg</code> .....            | 48            | <code>\ar</code> .....              | 65     |
| <code>\binom</code> .....           | 46            | <code>\arccos</code> .....          | 46     |
| blackboard bold .....               | 43            | <code>\arcsin</code> .....          | 46     |
| blancs .....                        | 4             | <code>\arctan</code> .....          | 46     |
| <code>bm</code> .....               | 54            | <code>\arg</code> .....             | 46     |
| <code>\bmod</code> .....            | 46            | <code>\author</code> .....          | 29     |
| boîte .....                         | 100           | <code>\backmatter</code> .....      | 30     |
| bold face .....                     | 91            | <code>\begin</code> .....           | 31     |
| <code>\boldsymbol</code> .....      | 54            | <code>\bibitem</code> .....         | 68     |
| book (classe) .....                 | 8             | <code>\Big</code> .....             | 48     |
|                                     |               | <code>\big</code> .....             | 48     |
| <b>C</b>                            |               | <code>\Bigg</code> .....            | 48     |
| cédille .....                       | 23            | <code>\bigg</code> .....            | 48     |
| calc .....                          | 99            | <code>\binom</code> .....           | 46     |
| <code>\caption</code> .....         | 39, 73        | <code>\bmod</code> .....            | 46     |
| caractères réservés .....           | 5             | <code>\boldsymbol</code> .....      | 54     |
| caractères spéciaux .....           | 22            | <code>\caption</code> .....         | 39, 73 |
| <code>\cdot</code> .....            | 45            | <code>\cdot</code> .....            | 45     |
| <code>\cdots</code> .....           | 48            | <code>\cdots</code> .....           | 48     |
| center .....                        | 32            | <code>\chapter</code> .....         | 28     |
| césure .....                        | 18            | <code>\chaptermark</code> .....     | 70     |
| <code>\chapter</code> .....         | 28            | <code>\ci</code> .....              | 87     |
| <code>\chaptermark</code> .....     | 70            | <code>\cite</code> .....            | 68     |
| <code>\ci</code> .....              | 87            | <code>\cleardoublepage</code> ..... | 40     |
| <code>\cite</code> .....            | 68            | <code>\clearpage</code> .....       | 40     |
| <code>\cleardoublepage</code> ..... | 40            | <code>\cos</code> .....             | 46     |
| <code>\clearpage</code> .....       | 40            | <code>\cosh</code> .....            | 46     |
| codage .....                        | 10            | <code>\cot</code> .....             | 46     |
| color .....                         | 78, 82, 83    | <code>\coth</code> .....            | 46     |
| commandes                           |               | <code>\csc</code> .....             | 46     |
| <code>\!</code> .....               | 48            | <code>\date</code> .....            | 29     |
| <code>\(</code> .....               | 41            | <code>\ddots</code> .....           | 48     |
| <code>\)</code> .....               | 41            | <code>\deg</code> .....             | 46     |
| <code>\,</code> .....               | 42, 48        | <code>\degres</code> .....          | 26     |
| <code>\-</code> .....               | 19            | <code>\depth</code> .....           | 102    |
| <code>\@</code> .....               | 27            | <code>\det</code> .....             | 46     |
| <code>\[</code> .....               | 41            | <code>\dim</code> .....             | 46     |
| <code>\%</code> .....               | 17, 33-35, 97 | <code>\displaystyle</code> .....    | 52     |
| <code>\*</code> .....               | 17            | <code>\documentclass</code> .....   | 8      |

---

|                                     |          |                                    |             |
|-------------------------------------|----------|------------------------------------|-------------|
| <code>\dots</code> .....            | 21       | <code>\lg</code> .....             | 46          |
| <code>\dq</code> .....              | 27       | <code>\lim</code> .....            | 46          |
| <code>\dum</code> .....             | 87       | <code>\liminf</code> .....         | 46          |
| <code>\emph</code> .....            | 31       | <code>\limsup</code> .....         | 46          |
| <code>\end</code> .....             | 31       | <code>\linebreak</code> .....      | 17          |
| <code>\eqref</code> .....           | 42       | <code>\linespread</code> .....     | 94          |
| <code>\exp</code> .....             | 46       | <code>\listoffigures</code> .....  | 39          |
| <code>\footnote</code> .....        | 30, 73   | <code>\listoftables</code> .....   | 39          |
| <code>\frac</code> .....            | 44       | <code>\ln</code> .....             | 46          |
| <code>\framebox</code> .....        | 39, 102  | <code>\log</code> .....            | 46          |
| <code>\frenchspacing</code> .....   | 28       | <code>\mainmatter</code> .....     | 30          |
| <code>\frontmatter</code> .....     | 29       | <code>\makebox</code> .....        | 101, 102    |
| <code>\fussy</code> .....           | 18       | <code>\makeindex</code> .....      | 69          |
| <code>\gcd</code> .....             | 46       | <code>\maketitle</code> .....      | 29          |
| <code>\height</code> .....          | 101, 102 | <code>\mathbb</code> .....         | 43          |
| <code>\hline</code> .....           | 35       | <code>\mathrm</code> .....         | 52          |
| <code>\hom</code> .....             | 46       | <code>\mathversion</code> .....    | 54          |
| <code>\href</code> .....            | 80       | <code>\max</code> .....            | 46          |
| <code>\hspace</code> .....          | 89, 95   | <code>\mbox</code> .....           | 19, 22, 101 |
| <code>\hyphenation</code> .....     | 18       | <code>\min</code> .....            | 46          |
| <code>\idotsint</code> .....        | 49       | <code>\multicolumn</code> .....    | 37          |
| <code>\ieme</code> .....            | 26       | <code>\newcommand</code> .....     | 88          |
| <code>\ier</code> .....             | 26       | <code>\newenvironment</code> ..... | 89          |
| <code>\iiiint</code> .....          | 49       | <code>\newline</code> .....        | 17          |
| <code>\iiint</code> .....           | 49       | <code>\newpage</code> .....        | 17          |
| <code>\iint</code> .....            | 49       | <code>\newtheorem</code> .....     | 53          |
| <code>\include</code> .....         | 12       | <code>\No</code> .....             | 26          |
| <code>\includegraphics</code> ..... | 64       | <code>\noindent</code> .....       | 95          |
| <code>\includeonly</code> .....     | 12       | <code>\nolinebreak</code> .....    | 17          |
| <code>\indent</code> .....          | 95       | <code>\nombre</code> .....         | 26          |
| <code>\index</code> .....           | 69       | <code>\nopagebreak</code> .....    | 17          |
| <code>\inf</code> .....             | 46       | <code>\overbrace</code> .....      | 45          |
| <code>\input</code> .....           | 13       | <code>\overleftarrow</code> .....  | 45          |
| <code>\int</code> .....             | 46       | <code>\overline</code> .....       | 45          |
| <code>\item</code> .....            | 32       | <code>\overrightarrow</code> ..... | 45          |
| <code>\ker</code> .....             | 46       | <code>\pagebreak</code> .....      | 17          |
| <code>\label</code> .....           | 30, 42   | <code>\pageref</code> .....        | 30          |
| <code>\LaTeX</code> .....           | 20       | <code>\pagestyle</code> .....      | 10          |
| <code>\LaTeXe</code> .....          | 20       | <code>\paragraph</code> .....      | 28          |
| <code>\ldots</code> .....           | 48       | <code>\parbox</code> .....         | 101         |
| <code>\left</code> .....            | 47       | <code>\parindent</code> .....      | 94          |
| <code>\leftmark</code> .....        | 70       | <code>\parskip</code> .....        | 94          |
| <code>\lettrine</code> .....        | 95       | <code>\part</code> .....           | 28          |

- 
- |                                       |             |                                            |           |
|---------------------------------------|-------------|--------------------------------------------|-----------|
| <code>\pause</code> .....             | 85          | <code>\tanh</code> .....                   | 46        |
| <code>\phantom</code> .....           | 51, 73      | <code>\TeX</code> .....                    | 20        |
| <code>\pmod</code> .....              | 46          | <code>\texorpdfstring</code> .....         | 81        |
| <code>\Pr</code> .....                | 46          | <code>\text</code> .....                   | 52        |
| <code>\printindex</code> .....        | 70          | <code>\textbackslash</code> .....          | 5         |
| <code>\prod</code> .....              | 46          | <code>\textrm</code> .....                 | 52        |
| <code>\protect</code> .....           | 73          | <code>\textstyle</code> .....              | 52        |
| <code>\providecommand</code> .....    | 89          | <code>\thispagestyle</code> .....          | 11        |
| <code>\ProvidesPackage</code> .....   | 90          | <code>\times</code> .....                  | 45        |
| <code>\qqquad</code> .....            | 42, 48      | <code>\title</code> .....                  | 29        |
| <code>\quad</code> .....              | 42, 48      | <code>\today</code> .....                  | 20        |
| <code>\raisebox</code> .....          | 102         | <code>\totalheight</code> .....            | 102       |
| <code>\ref</code> .....               | 30, 42      | <code>\ud</code> .....                     | 49        |
| <code>\renewcommand</code> .....      | 88          | <code>\underbrace</code> .....             | 45        |
| <code>\renewenvironment</code> .....  | 89          | <code>\underline</code> .....              | 31, 45    |
| <code>\right</code> .....             | 47          | <code>\up</code> .....                     | 26        |
| <code>\right.</code> .....            | 47, 49      | <code>\usepackage</code> 9, 21, 23, 24, 90 |           |
| <code>\rightmark</code> .....         | 70          | <code>\vdots</code> .....                  | 48        |
| <code>\rule</code> .....              | 39, 89, 103 | <code>\vec</code> .....                    | 45        |
| <code>\scriptscriptstyle</code> ..... | 52          | <code>\verb</code> .....                   | 34, 35    |
| <code>\scriptstyle</code> .....       | 52          | <code>\verbatiminput</code> .....          | 71        |
| <code>\sec</code> .....               | 46          | <code>\vspace</code> .....                 | 97        |
| <code>\section</code> .....           | 28, 73      | <code>\widehat</code> .....                | 45        |
| <code>\sectionmark</code> .....       | 70          | <code>\widetilde</code> .....              | 45        |
| <code>\setlength</code> .....         | 94, 99      | <code>\width</code> .....                  | 101, 102  |
| <code>\settodepth</code> .....        | 100         | <code>\xymatrix</code> .....               | 65        |
| <code>\settoheight</code> .....       | 100         | comment .....                              | 6, 71, 72 |
| <code>\settowidth</code> .....        | 100         | commentaires .....                         | 6         |
| <code>\sin</code> .....               | 46          | <code>\cos</code> .....                    | 46        |
| <code>\sinh</code> .....              | 46          | <code>\cosh</code> .....                   | 46        |
| <code>\sloppy</code> .....            | 18          | <code>\cot</code> .....                    | 46        |
| <code>\sqrt</code> .....              | 44          | <code>\coth</code> .....                   | 46        |
| <code>\stackrel</code> .....          | 46          | crochets .....                             | 5, 47     |
| <code>\stretch</code> .....           | 89, 95, 97  | <code>\csc</code> .....                    | 46        |
| <code>\subparagraph</code> .....      | 28          |                                            |           |
| <code>\subsection</code> .....        | 28          | <b>D</b>                                   |           |
| <code>\subsectionmark</code> .....    | 70          | délimiteurs .....                          | 47        |
| <code>\substack</code> .....          | 47          | <code>\date</code> .....                   | 29        |
| <code>\subsubsection</code> .....     | 28          | dcolumn .....                              | 36        |
| <code>\sum</code> .....               | 46          | <code>\ddots</code> .....                  | 48        |
| <code>\sup</code> .....               | 46          | <code>\deg</code> .....                    | 46        |
| <code>\tableofcontents</code> ...     | 29, 94      | <code>\degres</code> .....                 | 26        |
| <code>\tan</code> .....               | 46          | <code>\depth</code> .....                  | 102       |

- 
- description ..... 32
  - `\det` ..... 46
  - deux colonnes ..... 9
  - `\dim` ..... 46
  - dimensions ..... 95
  - `\displaymath` ..... 41
  - `\displaystyle` ..... 52
  - disposition d'une page ..... 97
  - `\documentclass` ..... 8
  - `\dots` ..... 21
  - double interligne ..... 94
  - `\dq` ..... 27
  - `\dum` ..... 87
  
  - E**
  - `\emph` ..... 31, 91
  - emplacement ..... 38
  - `empty` ..... 10
  - en-tête ..... 10
  - `\end` ..... 31
  - `enumerate` ..... 32
  - environnement ..... 31
  - environnements
    - `align` ..... 50
    - `array` ..... 49
    - `center` ..... 32
    - `comment` ..... 6, 71, 72
    - `description` ..... 32
    - `displaymath` ..... 41
    - `enumerate` ..... 32
    - `equation` ..... 42
    - `figure` ..... 37, 39, 63
    - `flushleft` ..... 32
    - `flushright` ..... 32
    - `itemize` ..... 32
    - `lscommand` ..... 87, 88
    - `math` ..... 41
    - `minipage` ..... 101
    - `multline` ..... 50
    - `quotation` ..... 33
    - `quote` ..... 33
    - `split` ..... 50
    - `subarray` ..... 47
    - `table` ..... 37, 39, 63
    - `tabular` ..... 35
    - `thebibliography` ..... 68
    - `verbatim` ..... 34, 71
    - `verse` ..... 33, 34
  - `\eqref` ..... 42
  - `equation` ..... 42
  - équations
    - longues ..... 50
    - système d' ..... 50
  - espace ..... 4
    - après une commande ..... 5
    - en début de ligne ..... 4
  - espace insécable ..... 27
  - espacement
    - horizontal ..... 95
    - mathématique ..... 48
    - vertical ..... 97
  - `eucal` ..... 61
  - `eufrak` ..... 61
  - `euro` ..... 21
  - `eurofont` ..... 21
  - `europs` ..... 21
  - `eurosans` ..... 21
  - `eurosym` ..... 21
  - executive (papier) ..... 9
  - `\exp` ..... 46
  - exposant ..... 44
  - `exscale` ..... 10, 48
  - extension ..... 6, 9
    - `.def` ..... 72
    - `.dtx` ..... 11, 72, 73
    - `.dvi` ..... 13, 64, 73
    - `.eps` ..... 64
    - `.fd` ..... 11
    - `.glo` ..... 73
    - `.idx` ..... 12, 69, 73
    - `.ind` ..... 70
    - `.ins` ..... 11, 72
    - `.ps` ..... 64
    - `.sty` ..... 11, 72
    - `.tex` ..... 11
  - extensions

aeguill ..... 81  
 amsbsy ..... 54  
 amsfons ..... 43, 61  
 amsmath ... 41, 42, 46–52, 54  
 amssymb ..... 55, 61  
 babel ..... 24–26  
 babel/frenchb ..... 20, 28, 32  
 bm ..... 54  
 calc ..... 99  
 color ..... 78, 82, 83  
 dcolumn ..... 36  
 eucal ..... 61  
 eufrak ..... 61  
 eurofont ..... 21  
 europs ..... 21  
 eurosans ..... 21  
 eurosym ..... 21  
 exscale ..... 10, 48  
 fancyhdr ..... 70, 71  
 fontenc ..... 10, 25  
 french ..... 24  
 graphicx ... 63, 64, 78, 82, 83  
 hyperref ..... 76, 78–83  
 ifpdf ..... 82  
 ifthen ..... 10  
 indentfirst ..... 95  
 inputenc ..... 10, 23, 25  
 latexsym ..... 10  
 layout ..... 97  
 lettrine ..... 95  
 longtabular ..... 37  
 makeidx ..... 10, 69  
 marvosym ..... 21  
 mathrsfs ..... 61  
 mltex ..... 25  
 pause ..... 85  
 pdfscreen ..... 83, 86  
 showidx ..... 70  
 supertabular ..... 37  
 syntonly ..... 10  
 verbatim ..... 6, 71, 72  
 xy ..... 65

**F**

fancyhdr ..... 70, 71  
 fichier source ..... 4  
 figure ..... 37  
 figure ..... 37, 39, 63  
 filet ..... 103  
 fleqn ..... 9  
 flushleft ..... 32  
 flushright ..... 32  
 font  
   \emph ..... 91  
   \footnotesize ..... 91  
   \Huge ..... 91  
   \huge ..... 91  
   \LARGE ..... 91  
   \Large ..... 91  
   \large ..... 91  
   \normalsize ..... 91  
   \scriptsize ..... 91  
   \small ..... 91  
   \textbf ..... 91  
   \textit ..... 91  
   \textmd ..... 91  
   \textnormal ..... 91  
   \textrm ..... 91  
   \textsc ..... 91  
   \textsf ..... 91  
   \textsl ..... 91  
   \texttt ..... 91  
   \textup ..... 91  
   \tiny ..... 91  
 fontenc ..... 10, 25  
 \footnote ..... 30, 73  
 \footnotesize ..... 91  
 formules ..... 41  
 \frac ..... 44  
 fraction ..... 44  
 \framebox ..... 39, 102  
 french ..... 24  
 \frenchspacing ..... 28  
 \frontmatter ..... 29  
 \fussy ..... 18

- 
- G**
- `\gcd` ..... 46
  - `graphicx` ..... 63, 64, 78, 82, 83
  - `graphiques` ..... 63
  - `grec`
    - `alphabet` ..... 44
  - `groupe` ..... 91
  - `guillemets` ..... 20
- H**
- `headings` ..... 10
  - `\height` ..... 101, 102
  - `\hline` ..... 35
  - `\hom` ..... 46
  - `horizontal`
    - `accolade` ..... 45
    - `filet` ..... 103
    - `trait` ..... 45
  - `\href` ..... 80
  - `\hspace` ..... 89, 95
  - `\Huge` ..... 91
  - `\huge` ..... 91
  - `hyperref` ..... 76, 78–83
  - `\hyphenation` ..... 18
- I**
- `i et j sans points` ..... 23
  - `\idotsint` ..... 49
  - `\ieme` ..... 26
  - `\ier` ..... 26
  - `ifpdf` ..... 82
  - `ifthen` ..... 10
  - `\iiiint` ..... 49
  - `\iiint` ..... 49
  - `\iint` ..... 49
  - `\include` ..... 12
  - `\includegraphics` ..... 64
  - `\includeonly` ..... 12
  - `\indent` ..... 95
  - `indentfirst` ..... 95
  - `index` ..... 68
  - `\index` ..... 69
  - `indice` ..... 44
- `\inf` ..... 46
  - `\input` ..... 13
  - `inputenc` ..... 10, 23, 25
  - `\int` ..... 46
  - `intégrale` ..... 46
  - `interligne` ..... 94
  - `international` ..... 24
  - `italic` ..... 91
  - `\item` ..... 32
  - `itemize` ..... 32
- J**
- `justification` ..... 17
- K**
- `\ker` ..... 46
  - Knuth, Donald E. .... 1
- L**
- `\label` ..... 30, 42
  - Lamport, Leslie ..... 1
  - `\LARGE` ..... 91
  - `\Large` ..... 91
  - `\large` ..... 91
  - `\LaTeX` ..... 20
  - `LATEX 2.09` ..... 2
  - `LATEX 2ε` ..... 2
  - `LATEX3` ..... 1, 4
  - `\LaTeXe` ..... 20
  - `latexsym` ..... 10
  - `layout` ..... 97
  - `\ldots` ..... 48
  - `\left` ..... 47
  - `\leftmark` ..... 70
  - `legal (papier)` ..... 9
  - `leqno` ..... 9
  - `letter (papier)` ..... 9
  - `lettrine` ..... 95
  - `\lettrine` ..... 95
  - `\lg` ..... 46
  - `ligature` ..... 22
  - `\lim` ..... 46
  - `\liminf` ..... 46
  - `\limsup` ..... 46

- 
- |                                    |             |                                    |     |
|------------------------------------|-------------|------------------------------------|-----|
| <code>\linebreak</code> .....      | 17          | <code>\newline</code> .....        | 17  |
| <code>\linespread</code> .....     | 94          | <code>\newpage</code> .....        | 17  |
| <code>\listoffigures</code> .....  | 39          | <code>\newtheorem</code> .....     | 53  |
| <code>\listoftables</code> .....   | 39          | <code>\No</code> .....             | 26  |
| livre .....                        | 8           | <code>\noindent</code> .....       | 95  |
| <code>\ln</code> .....             | 46          | <code>\nolinebreak</code> .....    | 17  |
| <code>\log</code> .....            | 46          | <code>\nombre</code> .....         | 26  |
| longtabular .....                  | 37          | <code>\nopagebreak</code> .....    | 17  |
| lsccommand .....                   | 87, 88      | <code>\normalsize</code> .....     | 91  |
|                                    |             | notitlepage .....                  | 9   |
| <b>M</b>                           |             |                                    |     |
| <code>\mainmatter</code> .....     | 30          | <b>O</b>                           |     |
| <code>\makebox</code> .....        | 101, 102    | objets flottants .....             | 37  |
| makeidx .....                      | 10, 69      | œ.....                             | 23  |
| <code>\makeindex</code> .....      | 69          | oneside .....                      | 9   |
| makeindex .....                    | 69          | openany .....                      | 9   |
| <code>\maketitle</code> .....      | 29          | openright .....                    | 9   |
| marges .....                       | 97          | option .....                       | 8   |
| marvosym .....                     | 21          | <code>\overbrace</code> .....      | 45  |
| math .....                         | 41          | overfull hbox .....                | 18  |
| mathématiques .....                | 41          | <code>\overleftarrow</code> .....  | 45  |
| accents .....                      | 45          | <code>\overline</code> .....       | 45  |
| espaces .....                      | 48          | <code>\overrightarrow</code> ..... | 45  |
| fonctions .....                    | 46          | <b>P</b>                           |     |
| <code>\mathbb</code> .....         | 43          | <code>\pagebreak</code> .....      | 17  |
| <code>\mathrm</code> .....         | 52          | <code>\pageref</code> .....        | 30  |
| mathrsfs .....                     | 61          | <code>\pagestyle</code> .....      | 10  |
| <code>\mathversion</code> .....    | 54          | papier                             |     |
| <code>\max</code> .....            | 46          | A4 .....                           | 9   |
| <code>\mbox</code> .....           | 19, 22, 101 | A5 .....                           | 9   |
| <code>\min</code> .....            | 46          | letter .....                       | 9   |
| minipage .....                     | 101         | taille du .....                    | 9   |
| Mittelbach, Frank .....            | 1           | <code>\paragraph</code> .....      | 28  |
| mltex .....                        | 25          | paragraphe .....                   | 15  |
| modulo .....                       | 46          | paramètres .....                   | 5   |
| moins (signe) .....                | 20          | <code>\parbox</code> .....         | 101 |
| montant .....                      | 103         | parenthèses .....                  | 47  |
| <code>\multicolumn</code> .....    | 37          | <code>\parindent</code> .....      | 94  |
| multiline .....                    | 50          | <code>\parskip</code> .....        | 94  |
|                                    |             | <code>\part</code> .....           | 28  |
| <b>N</b>                           |             |                                    |     |
| <code>\newcommand</code> .....     | 88          | pause .....                        | 85  |
| <code>\newenvironment</code> ..... | 89          | <code>\pause</code> .....          | 85  |

- 
- pdfL<sup>A</sup>T<sub>E</sub>X ..... 76, 78, 83, 85  
pdfscreen ..... 83, 86  
\phantom ..... 51, 73  
pieds de page ..... 10  
plain ..... 10  
\pmod ..... 46  
point ..... 20  
points de suspension ..... 21, 48  
    diagonaux ..... 48  
    verticaux ..... 48  
police ..... 90  
PostScript encapsulé ..... 63  
\Pr ..... 46  
préambule ..... 6  
prime ..... 45  
\printindex ..... 70  
\prod ..... 46  
\protect ..... 73  
\providecommand ..... 89  
\ProvidesPackage ..... 90
- Q**  
\qqquad ..... 42, 48  
\quad ..... 42, 48  
quotation ..... 33  
quote ..... 33
- R**  
références croisées ..... 30  
racine carrée ..... 44  
\raisebox ..... 102  
rapport ..... 8  
recto simple ..... 9  
recto-verso ..... 9  
\ref ..... 30, 42  
\renewcommand ..... 88  
\renewenvironment ..... 89  
report (classe) ..... 8  
\right ..... 47  
\right. .... 47, 49  
\rightmark ..... 70  
roman ..... 91  
\rule ..... 39, 89, 103
- S**  
sans serif ..... 91  
scandinaves (caractères) ..... 23  
\scriptscriptstyle ..... 52  
\scriptsize ..... 91  
\scriptstyle ..... 52  
\sec ..... 46  
\section ..... 28, 73  
\sectionmark ..... 70  
\setlength ..... 94, 99  
\settodepth ..... 100  
\settoheight ..... 100  
\settowidth ..... 100  
showidx ..... 70  
\sin ..... 46  
\sinh ..... 46  
slanted ..... 91  
slides (classe) ..... 8  
\sloppy ..... 18  
\small ..... 91  
small caps ..... 91  
somme ..... 46  
split ..... 50  
\sqrt ..... 44  
\stackrel ..... 46  
\stretch ..... 89, 95, 97  
style de page  
    empty ..... 10  
    headings ..... 10  
    plain ..... 10  
    subarray ..... 47  
\subparagraph ..... 28  
\subsection ..... 28  
\subsectionmark ..... 70  
\substack ..... 47  
\subsubsection ..... 28  
\sum ..... 46  
\sup ..... 46  
supertabular ..... 37  
symboles  
    gras ..... 54  
    mathématiques ..... 55  
syntonly ..... 10

- 
- systèmes d'équations ..... 50
- T**
- table ..... 37, 39, 63  
 tableau ..... 37  
 \tableofcontents ..... 29, 94  
 tabular ..... 35  
 taille
  - de la police par défaut ..... 9
  - des polices ..... 90
  - des polices mathématiques 51
  - du papier ..... 9, 76
  - prédéfinies ..... 91
 \tan ..... 46  
 \tanh ..... 46  
 \TeX ..... 20  
 \texorpdfstring ..... 81  
 \text ..... 52  
 \textbackslash ..... 5  
 \textbf ..... 91  
 \textit ..... 91  
 \textmd ..... 91  
 \textnormal ..... 91  
 \textrm ..... 52, 91  
 \textsc ..... 91  
 \textsf ..... 91  
 \textsl ..... 91  
 \textstyle ..... 52  
 \texttt ..... 91  
 \textup ..... 91  
 thebibliography ..... 68  
 \thispagestyle ..... 11  
 tilde ..... 27, 45  
 \times ..... 45  
 \tiny ..... 91  
 tiret ..... 20  
 \title ..... 29  
 titlepage ..... 9  
 titre du document ..... 9, 29  
 \today ..... 20  
 \totalheight ..... 102  
 transparents ..... 8  
 twocolumn ..... 9
- U**
- \ud ..... 49  
 umlaut ..... 23  
 \underbrace ..... 45  
 underfull hbox ..... 18  
 \underline ..... 31, 45  
 unités ..... 95, 96  
 \up ..... 26  
 upright ..... 91  
 \usepackage ..... 9, 21, 23, 24, 90
- V**
- \vdots ..... 48  
 \vec ..... 45  
 vecteurs ..... 45  
 \verb ..... 34, 35  
 verbatim ..... 6, 71, 72  
 verbatim ..... 34, 71  
 \verbatiminput ..... 71  
 verse ..... 33, 34  
 vertical
  - espacement ..... 97
  - points de suspension ..... 48
 virgule ..... 20  
 \vspace ..... 97
- W**
- \widehat ..... 45  
 \widetilde ..... 45  
 \width ..... 101, 102  
 WYSIWYG ..... 2, 3
- X**
- xy ..... 65  
 \xymatrix ..... 65

## Cahiers GUTenberg – Hors série n° 2, février 2003

ISSN : 1140-9304. *Dépôt légal* : 1<sup>er</sup> trimestre 2003.

*Tirage* : 1 200 exemplaires.

*Impression* : Imprimerie Louis-Jean, B.P. 87, 05003 Gap.

*Directeur de la publication et rédacteur en chef* : Jacques ANDRÉ.

*Rédacteur de ce numéro* : Daniel FLIPO.

*Comité de rédaction* : Jacques ANDRÉ, Thierry BOUCHE, Michel BOVANI,  
Daniel FLIPO, Pierre FOURNIER, Yannis HARALAMBOUS,  
Gilles PÉREZ-LAMBERT, Éric PICHERAL, Christophe PYTHOUD,  
Emmanuel SAINT-JAMES.

*Rédaction* :

Cahiers GUTenberg, c/o IRISA, Campus universitaire de Beaulieu,  
35042 RENNES Cedex, France.

gut@irisa.fr ; fax : +33 2 99 84 71 71

- 
- Veuillez m'abonner aux *Cahiers GUTenberg* pour 4 numéros  
à partir du numéro \_\_\_\_\_ à titre de :  
 membre de l'association (31 €)  
 non-membre (46 €).

- Veuillez me faire parvenir les *Cahiers GUTenberg* suivants :

| Nombre | cahier numéro                                | prix unitaire* | prix total |
|--------|----------------------------------------------|----------------|------------|
|        | 30 (ATypI & Blanchard, octobre 1998)         | 8 €            |            |
|        | 31 (numéro courant)                          | 15 €           |            |
|        | 32 (GUT'99 Lyon, mai 1999)                   | 15 €           |            |
|        | 33-34 (XML, décembre 1999)                   | 30 €           |            |
|        | 35-36 (GUT'2000 Toulouse, mai 2000)          | 22 €           |            |
|        | 37-38 (numéro courant, décembre 2000)        | 22 €           |            |
|        | 39-40 (GUT'2001 Metz, mai 2001)              | 22 €           |            |
|        | 41 (Metapost, novembre 2001)                 | 22 €           |            |
|        | * Les prix sont en euros TVA et port compris | Total :        |            |

Adresse d'expédition :

Nom :

Société :

Adresse :

Code+Ville :

Pays :

Téléphone/Fax :

messagerie :

Date :

Signature :

---

À retourner **accompagné de votre règlement** à :  
Association GUTenberg, 2 rue des Boutons d'or, F-05000 Gap  
secretariat@gutenberg.eu.org ; fax +33 4 92 53 35 60