Le premier rapport de M^{lle} Michu, étudiante en sciences humaines, réalisé tout en LaTeX

avec des illustrations, des tableaux, une bibliographie, une table des matières, une table des figures
et tout ça les doigts dans le nez

Bertrand Masson

27 janvier 2011

Table des matières

Introduction 1

1 Au commencement était LaTeX 3
1.1 LaTeX qu'est-ce ? . . . 3
1.1.1 LaTeX et la programation . . . 3
1.1.2 LaTeX et la typographie . . . 3
1.1.3 LaTeX *versus* traitements de texte . . . 4
1.2 LaTeX comment ça marche . . . 5
1.3 M^lle Michu n'est pas M^r Jourdain . . . 5
1.4 M^lle Michu n'est pas une neuneu . . . 6
1.5 M^lle Michu n'est pas une geek . . . 6
1.6 M^lle Michu a choisi d'utiliser TeXworks . . . 6
1.7 Où M^lle Michu découvre TeXworks . . . 8

2 Les premiers pas 11
2.1 Où M^lle Michu écrit sa première ligne de code . . . 11
2.2 Où M^lle Michu découvre les classes KOMA-Script . . . 11
2.2.1 La classe scrartcl . . . 12
2.2.2 La classe scrreprt . . . 12
2.2.3 La classe scrbook . . . 12
2.3 Où M^lle Michu découvre ce qu'est une commande LaTeX . . . 12
2.3.1 Les commandes . . . 15
2.3.2 Les environnements . . . 17
2.3.3 Les déclarations . . . 18
2.3.4 Environnement ou déclaration *That's the Question* ? . . . 19
2.4 Où M^lle Michu écrit enfin sa première ligne de code . . . 20
2.5 Où M^lle Michu est ébahie par la complétion automatique . . . 20
2.6 Où M^lle Michu choisi le format de page, la taille du texte, le recto-verso . . . 20
2.7 M^lle Michu perplexe doit indiquer l'encodage . . . 21
2.8 Où M^lle Michu poursuit son apprentissage de l'auto-complétion . . . 21
2.9 Où M^lle Michu n'a pas tout compris pour la complétion automatique 22
2.10 Où M^lle Michu fait appel à la police . . . 22
2.11 M^lle Michu veut parler français . . . 23
2.12 M^lle Michu achève le préambule . . . 23

3 M^lle Michu veut une méthode de travail 25
3.1 M^lle Michu divise pour mieux régner . . . 25
3.2 M^lle Michu appelle un fichier . . . 25
3.3 M^lle Michu introduit . . . 26
3.4 M^lle Michu sectionne . . . 26
3.4.1 Taille des divisions . . . 27

3.4.2 Arguments optionnels des divisions . . . 28
3.4.3 Formes sans numérotation . . . 29
3.4.4 Profondeur de la numérotation . . . 29
3.5 Mlle Michu confrontée aux compteurs . . . 30
1.1 Première section . . . 30
1.1.1 Première sous-section . . . 30
1.1.7 Deuxième sous-section . . . 30
1.1.10 Troisième sous-section . . . 30
1.1.7 Dernière sous-section . . . 31
3.6 Où Mlle Michu continue d'introduire . . . 31
3.6.1 Première difficulté pour Mlle Michu : les notes en bas de page 31
3.6.2 La bizarrerie - - . . . 33
3.6.3 La bizarrerie \, . . . 33
3.6.4 Mlle Michu et les nombres . . . 33
3.6.5 Mlle Michu capitalise petit . . . 34
3.6.6 Mlle Michu emphatique . . . 34
3.6.7 Mlle Michu change de paragraphe . . . 34
3.7 Mlle Michu dans l'espace . . . 35

4 Mlle Michu compile pour la première fois 39
4.1 Où Mlle Michu découvre avec horreur que ça ne compile pas . . . 39
4.2 Où Mlle Michu s'inquiète : comment retrouver l'erreur ? . . . 39
4.3 Où Mlle Michu s'exclame : Oh trop facile, je corrige de suite . . . 40
4.4 Où Mlle Michu se rend compte que retrouver les erreurs n'est pas toujours facile . . . 43
4.5 Où Mlle Michu s'étonne : c'est quoi tous ces fichiers ? . . . 45
4.6 Où Mlle Michu trouve fastidieux de changer de fichier pour compiler 45

5 Mlle Michu veut créer ses propres commandes 47
5.1 La première commande de Mlle Michu . . . 47
5.2 Mlle Michu et *xspace* . . . 48
5.3 Mlle Michu veut voir des exemples de commandes . . . 48
5.4 Mlle Michu modifie à nouveau ses fichiers . . . 49

6 Mlle Michu écrit en français 51
6.1 La ponctuation . . . 52
6.2 Mme, Melle . . . 52
6.3 1er, no , 1o . . . 52
6.3.1 1er, 2e,. . . ,ne . . . 52
6.3.2 1o , 2o ,. 53
6.3.3 no , nos ,No et Nos . . . 53
6.4 Le degré ° . . . 53
6.5 l'e dans l'a . . . 54
6.6 l'euro € . . . 54
6.7 Les accents . . . 54

7 Mlle Michu se lance dans l'insertion d'illustrations 55
7.1 Mlle Michu insère un petit pingouin . . . 55
7.2 Mlle Michu pour s'y retrouver organise son travail . . . 55

7.3 *LaTeX et les longueurs* . . . 58
7.3.1 Cadratin . . . 59
7.3.2 l'œil . . . 59
7.3.3 Système propre à LaTeX . . . 59
7.3.4 Unités absolues ou unités relatives, *that the question* ? . . . 60
7.4 Mlle Michu en veut un petit peu plus . . . 60
7.5 Mlle Michu flotte . . . 61
7.6 Où Mlle Michu apprend qu'il existe deux types de flottants . . . 61
7.7 Où Mlle Michu se rend compte qu'image et flottant c'est pas pareil 62
7.8 Où Mlle Michu doit respecter quelques règles . . . 62
7.9 Mlle Michu espère faire plier LaTeX . . . 64
7.10 Mlle Michu sachez que LaTeX à toujours raison . . . 64
7.11 Mlle Michu place sa première illustration . . . 66
7.12 Mlle Michu, la légende . . . 66
7.13 Mlle Michu veut citer son illustration : les références croisées . . . 67
7.14 Mlle Michu termine le premier chapitre de son rapport . . . 68

8 Mlle Michu fait des listes 77
8.1 Mlle Michu a besoin d'une liste . . . 77
8.2 Où Mlle Michu énumère . . . 78
8.3 Où Mlle Michu décrit . . . 78

9 Mlle Michu pense qu'il est urgent de s'occuper de la bibliographie 81
9.1 Mlle Michu panique, elle ne sait pas gérer des bases de données . . . 81
9.1.1 Mlle Michu saisi un article . . . 84
9.1.2 Mlle Michu saisi un livre . . . 84
9.2 Mlle Michu cite un ouvrage . . . 84
9.3 Mlle Michu fabrique sa bibliographie . . . 85
9.4 Mlle Michu compile sa bibliographie . . . 86

10 Au tableau Mlle Michu 91
10.1 Mlle Michu découvre tabular . . . 91
10.2 Mlle Michu va au tableau . . . 93
10.3 Ça déborde Mlle Michu . . . 95
10.4 Mlle Michu veut ses aises . . . 97
10.4.1 L'espace entre deux colonnes . . . 97
10.4.2 L'espace entre les lignes . . . 98
10.5 Mlle Michu veut des traits partiels . . . 98
10.6 Mlle Michu fusionne . . . 99
10.6.1 La fusion de colonnes . . . 99
10.6.2 La fusion de lignes . . . 99

11 Mlle Michu, le début de la fin 103
11.1 Mlle Michu à l'annexe . . . 103
11.1.1 Mlle Michu veut voir le mot annexe apparaitre . . . 103
11.2 À table Mlle Michu . . . 105
11.3 Mlle Michu des pieds à la tête . . . 105
11.4 Mlle Michu veut un titre . . . 106
11.5 Vous reprendrez bien une petite image Mlle Michu . . . 107
11.5.1 Les espaces verticaux . . . 107

11.5.2 Une image dans le titre 108

12 Il y en a un petit peu plus M^lle Michu, je vous le laisse ? 111
12.1 M^lle Michu mise en boîte 111
12.2 M^lle Michu fait du coloriage 113
12.2.1 Du texte en couleur 113
12.2.2 La feuille en couleur 113
12.2.3 Des boites de couleur 115
12.2.4 Intensité des couleurs 115
12.2.5 Couleur et minipage 116
12.3 M^lle Michu dans l'hypertexte 117
12.3.1 Liens en couleur 117
12.3.2 Lien vers un site web 118
12.3.3 hyperref et KOMA-Script 118
12.4 M^lle Michu brouillonne 119
12.5 M^lle Michu crée des commandes à arguments 119
12.5.1 Un seul argument 119
12.5.2 Plusieurs arguments 120
12.5.3 Avec option 121
12.5.4 Un truc un peu plus complexe 121
12.6 M^lle Michu et les césures 125
12.7 M^lle Michu à l'index 125
12.8 M^lle Michu confronté à l'angoisse de la page blanche 127
12.9 M^lle Michu numérologue 128
12.10 M^lle Michu et les *Overfull* et *Underfull* 128
12.10.1 *Overfull* 128
12.10.2 *underfull* 130

Conclusion 131

Introduction

Le but de ce tutoriel est d'apprendre à réaliser de A à Z un rapport (de stage, de maitrise, de ce que tu veux…) avec LaTeX. On va essayer de le faire avec le moins de réglages possibles, le plus « brut de décoffrage », pour ne pas te noyer dans les arcanes de la programmation LaTeXiènne, ça ce sera pour plus tard, quand tu maitriseras un document simple.

Le document final sera un fichier PDF [1] au format a4, en recto-verso et une police de taille 10 pt, prêt pour l'impression et lisible sur tout type d'ordinateur (Mac, Linux, Windows…) avec une très grande qualité typographique.

Pour la qualité du fond je ne peux pas faire grand chose pour toi, mais après tout c'est ton boulot ! Toutefois comme tu vas t'en rendre compte, LaTeX te déchargeant d'une grande partie du travail de mise en page, tu seras plus libre de te concentrer sur le fond. Avec LaTeX tu gagnes sur les deux tableaux.

Pour illustrer notre propos, je vais te montrer comment réaliser le rapport de la fouille d'un site préhistorique. Ce choix s'explique, d'une part parce que je suis archéologue préhistorien et inventer un site factice ne m'est pas très difficile et aussi parce qu'un tel rapport contient de nombreux éléments variés, comme des photos, des dessins au trait, des plans pleines pages, des graphiques, des tableaux et donc nous permettra d'aborder de nombreux aspects de la programmation LaTeXiène.

Tu vas étudier les fonctions LaTeX exactement dans l'ordre ou vont apparaître les difficultés. On aura un récit chronologique et non pas thématique. Ce ne sera pas forcément l'ordre dans lequel tu vas rencontrer les problèmes, lors de la rédaction de ton propre rapport, mais il ne devrait pas y avoir de grosses différences et de toute façon, je pense qu'à la fin de la lecture de ce tutoriel tu devrais être en mesure de réaliser ton document.

Quelques remarques avant de commencer. Tu rencontras certaines sections marquées en italique. Tu peux lors d'une première lecture te dispenser de les lire. Elles apportent des précisions ou de nouvelles notions dont tu n'auras pas forcément besoin dans un premier temps.

1. L'avantage du format PDF est qu'il préserve les polices, les images, les objets graphiques et la mise en forme de tout document source, quelles que soient l'application et la plate-forme utilisées pour le lire (Mac, Linux, Windows…), voir la définition sur Wikipédia.

1 Au commencement était LaTeX

> Il faut du temps pour apprendre à penser structure et contenu.
>
> Hans Hagen, l'auteur de ConTeXt

1.1 LaTeX qu'est-ce ?

LaTeX n'est pas un traitement de texte, c'est un ensemble de commandes informatiques, qui va construire ton texte, à partir des ordres que tu vas lui donner, en respectant les règles typographiques de la langue choisie et en prenant en compte l'ensemble de ton document.

1.1.1 LaTeX et la programation

Rassure toi, ces ordres n'ont pas besoins forcement d'être nombreux, mais pour obtenir un bon résultat, il faut au minimum indiquer la langue dans laquelle tu travailles, le format du papier, sa division (partie, chapitre section…) et le type de document que tu souhaites obtenir. On ne fabrique pas de la même manière un livre, un rapport de stage ou un article.

Pour simplifier LaTeX est un ensemble de macros [1] qui facilite l'utilisation du langage informatique TeX, qui est à l'œuvre dans la fabrication de ton document. TeX et LaTeX étant des logiciels libres [2], tu peux modifier ces macros pour les adapter à tes besoins, tu peux créer de nouvelles macros. De nombreux utilisateur de LaTeX ont crées des macros et les ont mises à la disposition de tous. Elles sont souvent regroupées dans des *packages* que tu charges en cas de besoins. C'est pour cette raison que l'on parle de « distribution LaTeX ». Une distribution LaTeX contient outre les programmes de base TeX et LaTeX, des milliers de *packages*, qui permettent, entre autre, d'écrire des formules de chimie, de faire des tableaux complexes, d'écrire en couleur, de mettre des images dans les marges, d'écrire en chinois et même en quenya la langue elfique inventé par J. R. R. Tolkien (). Les possibilités sont énormes et avant de ré-inventer la roue, cherche dans la liste des *packages* si quelqu'un n'a pas déjà proposé une solution à ton problème. Il existe plusieurs distributions LaTeX : MiKTeX sous Windows, TeXLive sous Linux et MacTeX sous Mac OS X.

1.1.2 LaTeX et la typographie

Les traitements de texte, obligent leurs utilisateurs à faire eux mêmes la mise en page de leurs documents. Malheureusement, rare, parmi eux, sont ceux qui maîtrisent les règles typographiques. La conséquence est que l'on s'est habitué

1. une macro est simplement un regroupement d'instruction afin d'obtenir une action. Par exemple `\includegraphics[width=12cm]{maPhoto}` va inclure l'image maPhoto dans ton texte en lui donnant une largeur de 12cm.
2. voir la définition de Wikipédia

à une typographie du pauvre. Le problème c'est que la typographie, ça ne sert pas uniquement à faire joli, son rôle principal est de faciliter la lecture. La typographie est une science ancienne, dont les premières règles ont été élaborées par les moines copistes et qui s'est développée avec l'invention de l'imprimerie par Gutenberg.

Pour ne pas détourner le lecteur du fond, il ne faut pas que des éléments typographiques ni que la mise en page n'entraîne de fatigue visuelle.

Par exemple une règle couramment admise est que la longueur d'une ligne de texte (en typographie on dit la justification) ne doit pas dépasser 60 à 70 caractères (espace compris). Au-delà l'œil fatigue et il est plus difficile de localiser la prochaine ligne. Pour un texte écrit sur une seule colonne, une justification de 65 caractères est considérée comme idéale. C'est pour éviter la fatigue visuelle que les journaux, qui cherchent pour des raisons d'économie à mettre le plus de texte par page, sont écrit en colonnes pour conserver une justification acceptable.

Le gris typographique est l'apparence grise que prend le corps de texte, quand tu le regardes de loin ou si tu plisses les yeux pour voir le texte flou. Plus ce gris est homogène plus la lecture du texte sera facile et agréable. Au contraire quand l'homogénéité de ce gris est rompue par par un mauvais interlignage, des « rivières » ou des « lézardes », (les lignes blanches composées d'espaces, les unes [3] sous les autres, qui traversent les textes mal fichus), l'attention portée au texte diminue car ces défauts détournent du texte et rendent la lecture difficile. Obtenir un bon gris n'est pas facile, il dépend de la police utilisée de sa taille de la justification, de l'interligne…

On pourrait encore parler de la taille des marges avec leurs modes de calcul (« le canon de l'atelier », « au neuvième ») et de bien d'autres choses. Toutes ces règles concourent à la conception d'un document agréable à lire permettant au lecteur de rester concentrer sur sa lecture. Si j'ai tenu à évoquer [4] ces questions d'ordre typographique, c'est pour essayer de te faire comprendre que l'on ne s'improvise pas typographe, et que d'avoir un logiciel qui s'en occupe est une chance.

1.1.3 LaTeX *versus* traitements de texte

Quand tu travailles avec LaTeX il faut que tu oublies tous tes réflexes d'utilisateur de traitement de texte, la philosophie est différente, le fonctionnement aussi. Un traitement de texte fait la mise en page au fur et à mesure sans connaitre la suite du texte. Quand je dis il fait la mise en page, il se contente juste de passer à la ligne quand il arrive sur la marge, le reste c'est toi qui le fait. L'exemple le plus frappant pour illustrer l'aberration de ce système est l'insertion d'images. Je ne sais pas si tu as déjà l'expérience de longs documents avec des illustrations sous un traitement de texte classique, on a toujours l'impression que c'est facile. Tu places l'image, tu ajustes la taille et la position avec des poignées. Ensuite tu passes une ligne, tu mets la légende, puis tu continues à taper le texte. Ça c'est dans le meilleur des cas. Si tu es dans le bas de la page et qu'il n'y a pas la place tu es éjecté page suivante et celle que tu viens de quitter, présente une grande plage de blanc. Si tu considères que de grands espaces

3. En typographie, l'espace qui sépare les mots et du genre féminin

4. Assez maladroitement, je le reconnais, n'étant pas typographe et ne maîtrisant pas suffisamment cette science. C'est d'ailleurs pour cela que j'utilise LaTeX.

vides au sein du texte ont une valeur esthétique, alors tout va bien. Enfin si tu figes ton texte car la moindre transformation décale l'ensemble de ton bel arrangement, t'obligeant à refaire toute la mise en page. Et te voila à reprendre les poignées pour triturer les images pour que la légende et l'image restent sur la même page, ou pour éviter les grandes plages de blanc crées.

LaTeX composant ton document après l'écriture de ton texte, il en a une vision complète et s'il n'a pas la place de positionner ton image en bas de page il continuera à placer du texte, en reportant ton image page suivante.

1.2 LaTeX comment ça marche

Il faut plusieurs étapes pour fabriquer un document avec LaTeX. Premièrement tu dois écrire ton texte. Tu ne t'occupes pas de la mise en page tu « frappes au kilomètre », tu restes concentré sur le fond. Tu as juste à ajouter de temps en temps une commande pour indiquer que tu veux changer de paragraphe, que tu veux ce mot en gras, ajouter une note en bas de page ou mettre la table des matières à cet endroit. Ce fichier est un fichier texte brut ou fichier texte simple ou fichier ASCII [5] que tu sauvegardes avec l'extension `.tex`. Ce fichier s'appelle *le source* [6].

Maintenant il faut transformer ce source en document final, c'est la *compilation*. Il existe plusieurs compilateurs, comme tu veux réaliser un document au format PDF, tu vas utiliser `pdflatex`. Tu obtiens un document conforme au standart PDF lisible par n'importe quel « lecteur PDF » outre Acrobat Reader d' Adobe qui est multiplateforme, il en existe de nombreux, une recherche dans un moteur de recherche internet te permettra de trouver ton bonheur.

1.3 M^lle Michu n'est pas M^r Jourdain

On ne peut pas faire du LaTeX sans le savoir, ni sans avoir une distribution LaTeX en état de marche. Ce tutoriel sera basé sur la TexLive 2010. Son gros avantage est d'être multiplateforme, Linux, Mac, et Windows. Si comme moi tu travailles sur plusieurs environnements, ton document passera de l'un à l'autre de façon totalement transparente, sans aucun problème.

Tu seras très certainement amené à lire les différentes fiches à Bébert pour approfondir certains sujets.

Je te conseille également de parcourir les documents suivants :

`http://www.tuteurs.ens.fr/logiciels/latex/manuel.html`

`http://www.ukonline.be/programmation/latex/`

`http://ftp.lip6.fr/pub/TeX/CTAN/info/lshort/french/lshort-fr.pdf`

Sur le site de Framabook le volume 5 : Tout... sur LaTeX...

`http://www.framabook.org/`

Pour les utilisateur de Mac : Introduction à LaTeX sous Mac OS X de Fabien Conus & Franck Pastor

5. Voir la définition du texte brut sur Wikipédia.

6. En informatique source est au masculin car c'est un raccourcis pour « le code source » qui est un ensemble d'instructions écrites dans un langage de programmation informatique.

1.4 Mlle Michu n'est pas une neuneu

L'installation d'une distribution LaTeX n'est pas l'objet de cette fiche. L'installation de la TeXLive2010 est assez triviale et je te renvoie au site suivant qui te l'explique pas à pas
version pdf :
http://www.tug.org/texlive/doc/texlive-fr/texlive-fr.pdf
version html :
http://www.tug.org/texlive/doc/texlive-fr/texlive-fr.html

Comme Mlle Michu n'est pas une neuneu, elle va s'en sortir « *finger in the nose* ». J'en profite pour faire quelques remarques. Il faut être le plus autonome possible, c'est une question de liberté élémentaire. Dépendre d'un tiers, ou être dominé par la machine, c'est être esclave. Donc essaye toujours de te débrouiller toute seule. Installe toi même tes logiciels. Dans certain cas tu vas en baver, car tu te trouveras confronté à des dépendances non résolue, à du matériel non reconnu…, mais chaque étape franchie, c'est un degré de liberté de gagné. Mais si tu lis ce tutoriel c'est que tu veux contrôler la fabrication de ton document et pas t'en remettre au bon vouloir de la machine.

1.5 Mlle Michu n'est pas une geek

Un fichier LaTeX est un simple fichier texte que te peux donc écrire avec n'importe quel éditeur de texte. Attention j'ai dit éditeur et pas traitement de texte. Il faudrait être sacrément vicieux pour écrire un fichier LaTeX avec *Microsoft Word*, bien que cela soit possible. Donc sous Linux des logiciels comme *gedit* et *KEdit*, ou le bloc-notes sous windows, font parfaitement l'affaire. Tu peux même faire du LaTeX en mode console, tu sais la petite boite noire dans laquelle les geeks pilotent leur ordi avec des lignes de commandes. M'enfin si tu sais manipuler *vim* ou *emacs* je doute que tu trouves un intérêt à ce tutoriel. Donc comme tu n'es pas une geek, le plus simple pour faire du LaTeX, est quand même d'utiliser un éditeur spécialisé, qui permet de créer, éditer, compiler tes fichiers LaTeX. Il en existe de nombreux, et certain comme *kile* pour Linux sont très (trop ?) sophistiqué, avec des icônes partout, et ressemble au traitement de texte classique. Voici le nom de plusieurs d'entre eux, une recherche sur internet, te donneras des renseignements complémentaires :

Linux : Kile, WineFish,Texmaker, TeXworks

Mac OS X : MacTeX, TeXShop, iTeXMac , Texmaker, TeXworks

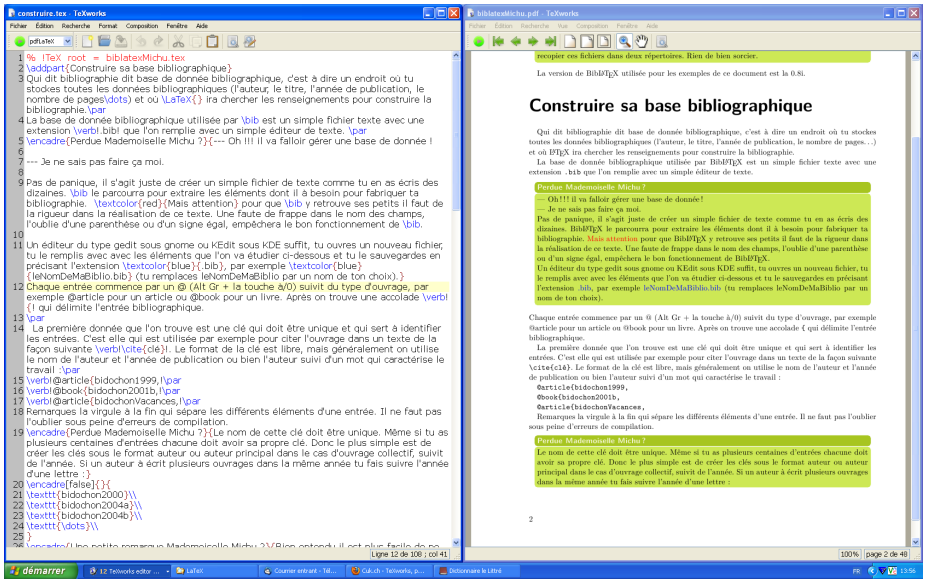
Windows : WinEdt, TeXnicCenter, Texmaker, TeXworks.

Liste non exhaustive.

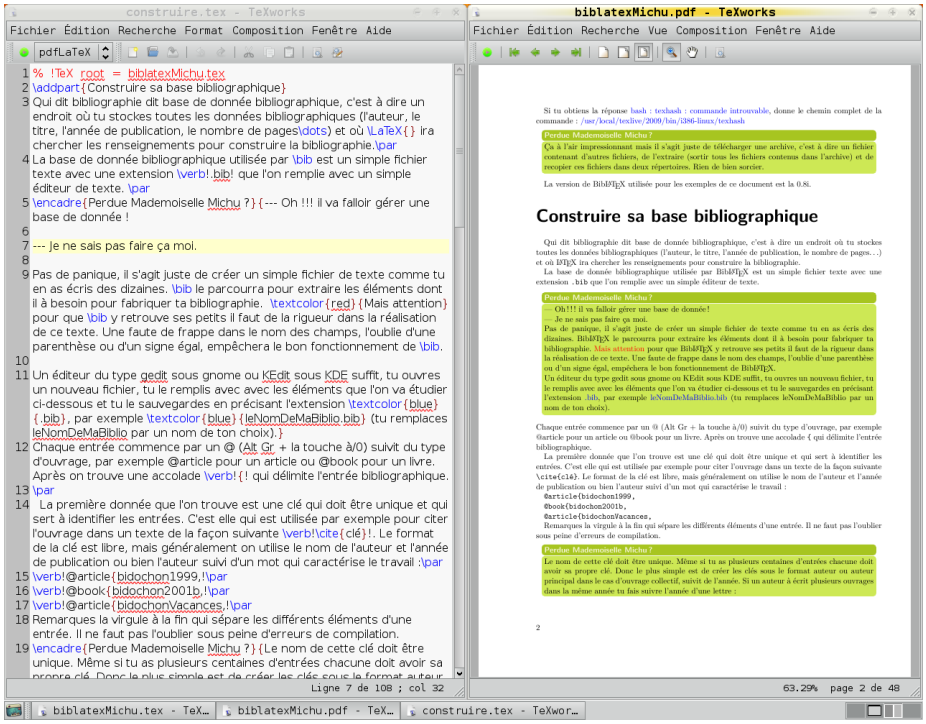
1.6 Mlle Michu a choisi d'utiliser TeXworks

Parmi tous ces éditeurs il faut faire un choix et comme c'est moi qui ai écrit ce tutoriel, c'est moi qui choisi, *and the winner is…* TeXworks. Je peux quand même te donner quelques arguments pour rendre ce choix moins arbitraire. TeXworks est :

– multiplateforme, tu conserves ainsi la même façon de travailler quelque soit l'environnement (même interface et mêmes fonctionnalités) ;



(a) sous windows XP



(b) sous Linux Debian

FIGURE 1.1 : TEXworks en action. Les deux fenêtre sont côte à côte et occupe tout l'écran ; à droite le pdf et à gauche le source. Le ratio différent des deux vues sont lié à des tailles d'écran différentes, 22 pouces à plat pour Windows et 19 pouces cathodique pour Linux.

– il reconnait l'UTF8, on en reparlera ;
– c'est l'éditeur de la TeXLive, pour les plateformes Windows et Mac OS X il est inclus, pour Linux il faut installer le paquet à part ;
– je n'aime pas les usines à gaz avec des icônes partout, encore moins quand il s'agit de faire du LaTeX. Pourquoi vouloir à tout pris que les logiciels qui traitent du texte se ressemblent. LaTeX n'est ni *Microsoft Word* ni *OpenOffice Writer* sa philosophie est différente et donc la façon de l'utiliser diffère également, surtout si tu veux être performant. Donc quitte à utiliser LaTeX allons y carrément avec TeXworks, son dépouillement et son autocomplétion.

Attention toutefois TeXworks est en court de développement et n'est pas considéré comme totalement stable, même s'il est parfaitement utilisable. La version utilisée ici est la 0.2.3 (r.466).

1.7 Où M^lle Michu découvre TeXworks

Figure 1.2 : TeXworks ouvert pour la première fois, légende dans le texte.

L'une des choses qui rebute le plus les débutants en LaTeX est le fait qu'il ne soit pas WYSIWYG (What You See Is What You Get, ce que vous voyez est ce que vous obtenez). TeXworks pallie un peu ce problème en coupant l'écran en deux fenêtres. La première contient *le source*, c'est à dire le fichiers `.tex` dans lequel tu saisis ton texte et les commandes LaTeX et la seconde un lecteur pdf qui permet de voir le résultat après compilation. La compilation c'est l'opération qui consiste à transformer le vilain crapaud qu'est le source, en un splendide document que tes collègues vont envier.

De plus ce lecteur PDF intégré permet de profiter des possibilités de synchronisation source-PDF. Si tu cliques avec le bouton gauche de la souris, en maintenant la touche Ctrl appuyée, dans l'une des fenêtres, l'autre se positionne au même endroit. Très pratique pour retrouver le code correspondant à une partie de ton document. Si ton source est constitué de plusieurs fichiers, le fichier codant la partie de ton rapport sera automatiquement ouvert si ce n'est déjà fait. Dans l'autre sens un Ctrl + bouton gauche de la souris dans le source, affiche la page correspondante du fichier `.pdf`, après compilation.

Les figures 1.1a et 1.1b, page 7 montrent TeXworks en action.

Tu trouveras, à cette adresse `http://www.leliseron.org/texworks/`, un manuel d'utilisation de TeXworks, qu'il est préférable d'avoir lu, même si l'on verras dans ce tutoriel comment utiliser l'éditeur.

Tu vas commencer par ouvrir TeXworks. Comme tu n'as pas encore de document il n'y a qu'une seule fenêtre qui occupe la moitié de l'écran. Cette fenêtre est très dépouillée. Tout en haut on a une série de menu déroulant (n° 1, figure 1.2), en dessous quelques icônes (n° 2), ensuite le cadre dans lequel on va écrire

notre fichier LATEX (n° 3). Un curseur clignotant t'invite à saisir ta première ligne (n° 4). Le chiffre (n° 5) est le numéro de ligne, si ta fenêtre en est dépourvu pas de panique on va arranger ça.

Tout d'abord tu vas configurer TEXworks pour qu'il corresponde à tes besoins. Tu vérifies que TEXworks pointe bien vers ta distribution LATEX. Tu vas donc dans *Édition* puis *Préférence*, dans l'onglet *Composition*, pour ouvrir la fenêtre illustrée figure 1.3. Dans le cas d'une TeXlive2010 sous Linux, il y a de forte chance que le chemin soit équivalent à : `/usr/local/textlive/2010/bin/i386-linux`.

Pour Windows ça doit ressembler à : `c:\texlive\2010\bin\win32`.

Tu dois trouver le chemin indiqué dans la fenêtre n° 1 fig 1.3. Si ce n'est pas le cas tu modifie en cliquant sur le signe plus pour ajouter un chemin et sur le signe moins pour supprimer un chemin erroné.

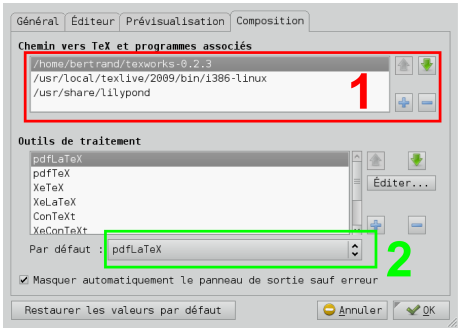


Figure 1.3 : Le menu « Composition ».

Sans entrer dans le détail, il faut que tu saches qu'il existe plusieurs compilateurs, en fonction du type de document que tu veux obtenir, par exemple le compilateur *latex* qui produit un fichier au format `.dvi` (le format d'origine des documents TEX). Le but étant de réaliser un fichier pdf tu vas utiliser *pdflatex* comme compilateur. Tout en bas de la fenêtre, tu choisis pdflatex dans le menu déroulant (n° 2 fig 1.3).

Maintenant tu ouvres l'onglet « Éditeur » et tu vas régler la coloration syntaxique. Tu choisis LATEX dans le menu déroulant n° 1 de l'image 1.4. Ceci permet d'avoir un source (le document `.tex` s'appelle un source) coloriés ou les commandes LATEX apparaissent en couleur, pour bien les repérer dans le document. Ainsi les commentaires (précédé de %) seront en rouge, les environnements (commence par `\begin{}` et se termine par `\end{}`) seront en vert et les commandes en bleu. Ton texte est en noir.

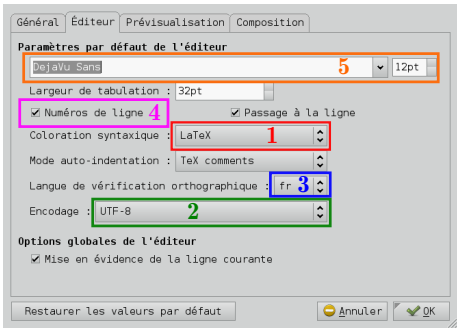


Figure 1.4 : Le menu « Éditeur ».

Et l'on passe à l'encodage. Ce choix est important car il permet à nous autres européens d'avoir des caractères accentués directement au clavier. Je te conseille de prendre `utf8` qui est reconnu par tous les environnements et qui évite d'avoir les signes accentués remplacés par des signes cabalistiques quand tu changes de machine (n° 2 fig. 1.4). C'est de toute façon la valeur par défaut de TEXworks.

Pour avoir la correction orthographique, sous Windows il faut au préalable avoir récupéré des dictionnaires (ceux d'OpenOffice ou de Thunderbird que tu récupères sur les sites web). Tu places les fichiers `.aff` et `.dic` dans le répertoire `<home>\Texworks\dictionaries`. Pour Linux il faut au préalable avoir installer pour le français, *aspell* et *aspell-fr* ou bien *hunspell* et *hunspell-fr*. Maintenant tu peux indiquer le dictionnaire par défaut, n° 3 fig. 1.4.

aspell et *hunspell* ne sont pas des éléments de la TEXlive, mais de la distribution Linux qui s'installe avec par exemple synaptic sous Ubuntu et Debian, urpmi, yast,... pour d'autre distribution.

Une dernière chose à faire avant de refermer cet onglet, est de cocher « numé-

rotation de lignes » qui permet de retrouver plus facilement les erreurs signalées par le compilateur, n° 4 fig. 1.4 .

Voila tu as maintenant un TeXworks près a fonctionner. Bien entendu ces changements ne sont à faire que la première fois ou tu utilises l'éditeur.

Tu peux changer si tu le désires d'autres paramètres comme la taille des icônes, masquer la fenêtre des erreurs en fin de compilation, changer la police de l'éditeur.

Ce n'est pas la police du document final, mais uniquement celle qui apparaît à l'écran. Tu la changes (ainsi que sa taille) uniquement pour un meilleur confort de travail.

2 Les premiers pas

2.1 Où M^lle Michu écrit sa première ligne de code

La première chose à faire c'est d'indiquer à LaTeX le type de document que l'on souhaite réaliser. En effet sous LaTeX il existe plusieurs types de documents prédéfinis, qui ne donneront pas le même résultat. Il existe des styles pour écrire des articles, des rapports, un livre, des lettres, faire des présentations à la « PowerPoint », une thèse,...

Ces styles (on dit classe en LaTeX, *class* en anglais) définissent la mise en page, la taille des marges, le nombre et le type de divisions (pas de chapitre pour un article par exemple),...

Il y a trois principales classes en standard :

article pour écrire un article ;

book pour écrire un livre ;

report pour écrire un rapport.

Tu veux faire un rapport (*report* en anglais) donc tu vas mettre comme première ligne ceci :

```
\documentclass{report}
```

Seulement voila on est des européens et la classe *report* est faite pour les américains et leur format de papier bizarre. Et oui les américains utilisent le format de papier « letter » (21,6 x 27,9 cm), alors que sur le continent européen on utilise le format « a4 » (21 x 29,7 cm). Les mises en page des classes classiques de LaTeX sont adaptées au format « letter ».

Donc on va plutôt utiliser la classe *scrreprt*, crée pour tous les européens par Markus Kohm, un allemand, qui provient de KOMA-Script (ne t'inquiète pas ce package est dans la TexLive).

2.2 Où M^lle Michu découvre les classes KOMA-Script

Les classes KOMA-Script qui correspondent aux types de documents sont :

rédiger un	classes standards	classe KOMA-Script
article	article	scrartcl
rapport	report	scrreprt
livre	book	scrbook

Les noms des classes KOMA-Script sont un peu barbares, imprononçables et ne sont pas facile à retenir. Le *scr* évoque script de KOMA-Script, la suite est une abréviation plus ou moins heureuse des classes classiques.

Toutes les commandes utilisées dans cette annexe concernent la version 3.00 de KOMA-Script. Avec cette version beaucoup de commandes KOMA-Script ont changées de nom. C'est la version qui équipe les TexLive 2009 et 2010.

2.2.1 La classe scrartcl

La classe article sert pour faire des petits documents de quelques pages, destiné à être publier avec d'autres articles dans des revues. Il n'y a pas de chapitre. Le sectionnement est le suivant : parties (`\part{}`), sections (`\section{}`), sous-sections (`\subsection{}`), sous-sous-sections (`\subsubsection{}`), paragraphes (`\paragraph{}`) et sous-paragraphes (`\subparagraph{}`).

Tu n'es pas obligé d'utiliser tous les niveaux de sectionnement. Je t'indique toutes les possibilités à ta disposition. Par exemple `\part{}` est peu utilisé dans un article. Tu peux concevoir un article avec un seul niveau de sectionnement, `\section{}`

Le titre est sur la même page que le début du texte, ainsi que les parties (`\part{}`) qui ne sont pas sur des pages séparées. Cette classe accepte un résumé et des annexes.

```
\documentclass{scrartcl}
```

La figure 2.1 montre l'aspect des deux premières pages d'un article fictif, à plusieurs parties.

2.2.2 La classe scrreprt

La classe rapport est faite pour des document plus long, se satisfaisant à eux-même. Aux divisions de la classe article s'ajoute le chapitre(`\chapter{}`). Comme pour la classe article on peut insérer un résumé. Le titre, le résumé, les entêtes des parties se trouvent sur des pages impaires séparées.

```
\documentclass{scrreprt}
```

La figure 2.2 montre les première pages d'un rapport fictif, divisé en parties, chapitres, sections... et avec un résumé.

2.2.3 La classe scrbook

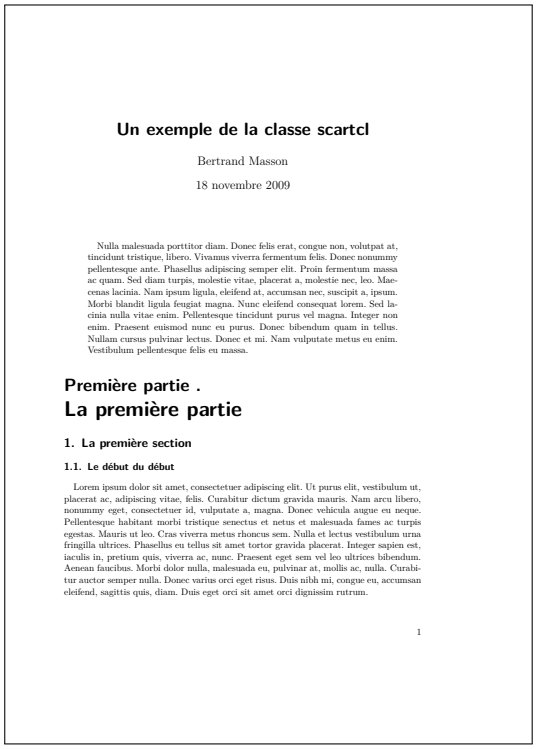
La classe livre est faite pour écrire ... un livre.

Il n'y pas de grandes différences par rapport à la classe rapport, si ce n'est l'ajout des commandes `\frontmatter`, `\mainmatter` et `\backmatter` qui gèrent les préfaces et postfaces.

Toutefois le comportement de certaine commande n'est pas le même qu'avec la classe rapport, comme par exemple la gestion de la page de titre.

2.3 Où M^lle Michu découvre ce qu'est une commande LaTeX

Avant de saisir ta première commande, il est temps de t'expliquer ce qu'est une commande LaTeX. Ça va être un peu long et un peu rébarbatif, mais com-



Un exemple de la classe scartcl

Bertrand Masson

18 novembre 2009

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Première partie .
La première partie

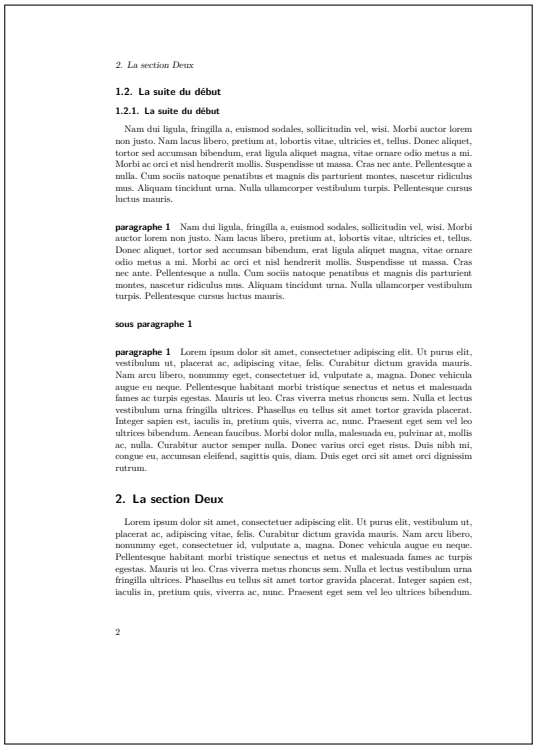
1. La première section

1.1. Le début du début

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

1

(**a**) page 1



2. La section Deux

1.2. La suite du début

1.2.1. La suite du début

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

paragraphe 1 Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

sous paragraphe 1

paragraphe 1 Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

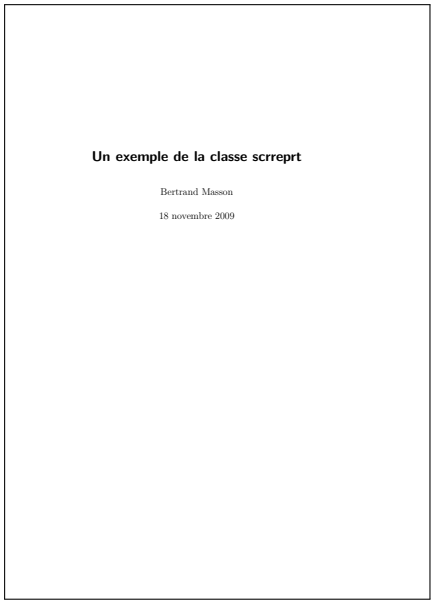
2. La section Deux

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum.

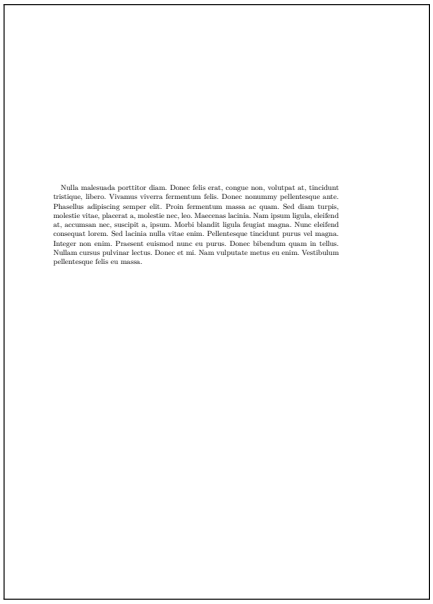
2

(**b**) page 2

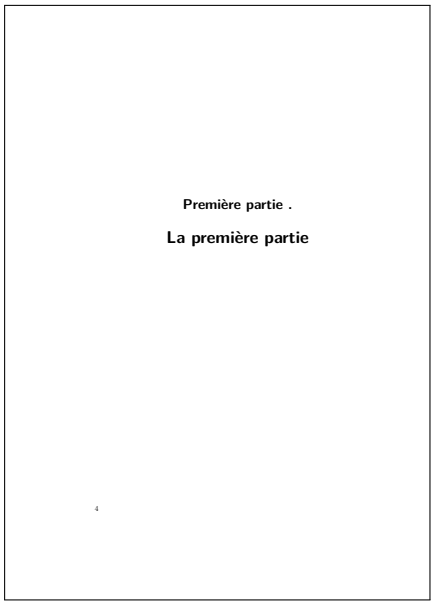
Figure 2.1 : Les deux premières pages d'un article fictif, réalisé avec la classe *scrartcl*.



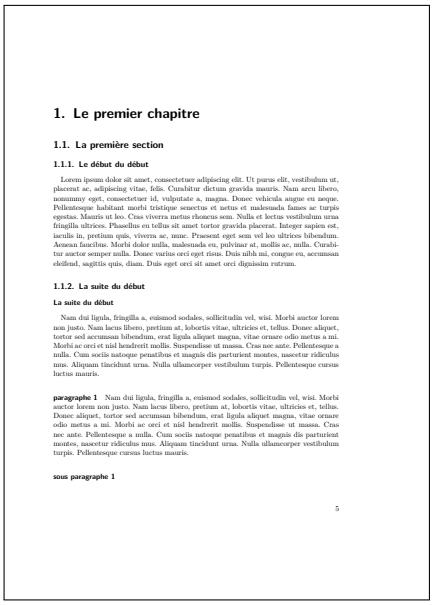
(**a**) page 1



(**b**) page 3



(**c**) page 4



(**d**) page 5

Figure 2.2 : Les premières pages d'un rapport fictif, réalisé avec la classe *scrreprt*. Attention, en réalité, entre ces pages ci-dessus il y a une page blanche.

prendre le fonctionnent des commandes LaTeX t'éviteras de nombreuses erreurs. Accroches-toi c'est parti.

LaTeX fonctionne à la manière du HTML qui sert à fabriquer les pages web. Il utilise des balises qui vont formater le texte qu'elles contiennent, comme par exemple mettre en gras, centrer,... et des commandes, comme `\par` qui permet de changer de paragraphe ou `\LaTeX` qui écrit LaTeX. Pour les distinguer du reste du texte, ces commandes et balises commencent toujours par `\`.

2.3.1 Les commandes

Il y a deux types de commandes, avec ou sans arguments.

Les commandes sans argument

Voici une liste de commande sans argument :
`\og` qui écrit le guillemet ouvrant «
`\fg` qui écrit le guillemet fermant »
`\'` l'accent aigu qu'en LaTeX tu peux placer sur n'importe quel lettre `\'B` donne Á, `\'z`, ź
`\no` pour écrire n°
`\oe` pour écrire œ
`\dots` pour écrire ... (ce qui n'est pas pareil que ... trois points écrits au clavier)

Donc quand LaTeX rencontre un `\`, il reconnait le début d'une commande. Pour la fin des commandes sans arguments c'est un peu plus compliqué car il y a plusieurs types de caractères qui indiquent à LaTeX la fin de la commande :

- l'espace ;
- tout caractère qui n'est pas une lettre ;
- `\␣` (␣ symbolise l'espace, c'est pas un signe cabalistique à taper, la fin de commande ici est donc une contre-barre plus un espace) ;
- une double accolade `{}`vide.

La première conséquence c'est qu'aucune commande ou balise LaTeX ne contient de chiffres, de caractères de ponctuation (, ; :) ou de tiret (souviens-t-en quand tu créeras tes commandes, et oui, dans ce tutoriel tu vas apprendre à créer des commandes).

Quand tu utilises l'espace pour terminer une commande, celle-ci est « mangée » par LaTeX, elle n'apparait pas dans le texte final. Un exemple si j'écris dans le source

Code

```
\LaTeX la suite
```

l'espace après le X de LaTeX indique la fin de commande mais est « mangée » et on obtient dans le document final (le .pdf)

Résultat

LaTeXla suite

M^lle Michu
— T'es couillon tu n'a qu'a mettre 2 espaces une qui est mangée et l'autre qui reste ! ! !

Et bien non c'est pas possible pour LaTeX 1, 2, 3 ou 100 000 espaces c'est pareil, il n'en voit qu'une seule. On en reparlera plus tard.

Pour obtenir le bon résultat il faut écrire l'une de ces possibilités

Code

```
\LaTeX\ la suite
\LaTeX{} la suite
```

qui donne

Résultat

LaTeX la suite
LaTeX la suite

Par contre tu peux écrire

Code

```
\LaTeX, la suite
\LaTeX. La suite
```

Résultat

LaTeX, la suite
LaTeX. La suite

Que l'espace soit mangée n'a pas que des inconvénients. Essayons d'écrire « Le cœur a ses raisons que la raison ne connaît pas ». La commande LaTeX pour le œ est `\oe`. Si j'écris `c\oeur` je vais obtenir une erreur car la commande `\oeur`, n'existe pas. Il faut donc écrire `c\oe` ur, l'espace étant mangé j'obtiens cœur. Tu aurais pu écrire aussi

Code

```
Le c\oe{}ur a ses raisons que la raison ne connaît pas
```

Résultat

Le cœur a ses raisons que la raison ne connaît pas

Par contre

Code

```
Le c\oe\ ur a ses raisons que la raison ne connaît pas
```

ne marche pas

Résultat

Le cœ ur a ses raisons que la raison ne connaît pas

et `c\oe\ur` renvoie une erreur car `\ur` est une commande qui n'existe pas.

Tu as pu remarquer qu'il n'est pas nécessaire d'insérer une espace au début de la commande.

Les commandes à argument

Les commandes à argument sont faites pour traiter des objets, des mots ou textes courts, qui sont placés entre accolades. Voici des exemples de commandes à argument :
`\textbf{un mot}` pour mettre en gras : **un mot** ;

`\includegraphics{pingouin.jpg}` pour inclure une image

Certaines commandes peuvent prendre plusieurs arguments :

`\rotatebox{45}{texte}` pour faire subir une rotation de 45° à un texte texte

Certaines commandes peuvent prendre des options. Toutes les options LaTeX s'écrivent entre crochets [ ] et sont séparées par des virgules, s'il y en à plusieurs. Par exemple si tu veux doubler la taille du pingouin :

`\includegraphics[scale=2]{pingouin.jpg}` qui donne

Tu peux dans certain cas emboîter les commandes :
`\textbf{\textit{un mot}}` pour un mot en gras et italique : ***un mot***

Par contre certaines actions sont impossibles, comme par exemple mettre un nouveau paragraphe dans un texte en gras :
`\textbf{un mot \par un autre mot}`, provoque une erreur de compilation. Pour traiter des textes plus grand avec des éléments de formatage il faut utiliser les environnements.

2.3.2 Les environnements

Les environnements commence toujours par `\begin{nomEnvironnement}` et se termine par `\end{nomEnvironnement}`. Tout ce qui se trouve entre ces deux bornes seront affecté par l'action de l'environnement choisi. Le nom de l'environnement obéi aux même règles que les noms de commandes, pas de ponctuations, de nombres, de caractères accentués. Par exemple pour centrer une partie d'un document :

Code

```
\begin{center}
Tout ce qui
\par
est \textbf{ici} est centré.
\end{center}
Mais pas après.
```

Résultat

Tout ce qui
est **ici** est centré.

Mais pas après.

Tu peux emboîter les environnements, mais pas les imbriquer, le premier exemple est bon, le deuxième retourne une erreur de compilation.

Code

```
\begin{environnement1}
texte ou rien
\begin{environnement2}
texte
\end{environnement2}
texte ou rien
\end{environnement1}
```

Code

```
\begin{environnement1}
texte ou rien
\begin{environnement2}
texte
\end{environnement1}
texte ou rien
\end{environnement2}
```

C'est une des erreurs les plus fréquente quand on débute. Elles ne sont pas toujours faciles à démêler, surtout quand les environnements sont éloignés.

2.3.3 Les déclarations

Il existe un dernier type de commande : les déclarations. Ce sont des commandes dont l'action débute à l'endroit où tu places cette commande et qui se poursuit jusqu'à la fin de ton document, sauf si elle rencontre :

- la fin d'un environnement dans lequel elle est intégré ;
- une autre commande qui annule ou modifie son action.

Voici pour être plus clair des exemples, pages suivantes, avec la commande `\tiny`, qui permet d'écrire en tout petit .

Code

```
Voici un texte écrit dans la taille courante de mon document. Si je
place la commande\tiny, tout ce qui suit est donc écrit en tout pe-
tit et ça va continuer jusqu'à la fin. Enfin pas vraiment car là je
vais placer la commande \large qui écrit en grand et annule l'action
de la commande tiny. Par contre \color{blue} n'a pas d'action sur
la commande large car son rôle est de changer la couleur du texte.
Tu as compris ? \tiny Maintenant tout est en bleu sauf si tu changes
\color{red} la couleur.
```

Résultat

Voici un texte écrit dans la taille courante de mon document. Si je place la commande, tout ce qui suit est donc écrit en tout petit et ça va continuer jusqu'à la fin. Enfin pas vraiment car là je vais placer la commande qui écrit en grand et annule l'action de la commande tiny. Par contre n'a pas d'action sur la commande large car son rôle est de changer la couleur du texte. Tu as compris ? Maintenant tout est en bleu sauf si tu changes la couleur.

Un exemple avec la commande dans un environnement.

```
Voici un texte écrit dans la taille courante de mon document. Maintenant je vais centrer la suite du texte. \begin{center} Voila mon texte est centré.\par
La commande \tiny me le met en tout petit et son action s'arrête là \end{center} car j'ai quitté l'environnement center dans lequel j'ai mis la commande tiny.
```

Code

Voici un texte écrit dans la taille courante de mon document. Maintenant je vais centrer la suite du texte.

Voila mon texte est centré.
La commande me le met en tout petit et son action s'arrête là

car j'ai quitté l'environnement center dans lequel j'ai mis la commande tiny.

Résultat

Remarque au passage qu'un environnement provoque automatiquement un saut de ligne avec un espace au début et à la fin.

Il existe un autre moyen de contraindre une déclaration, c'est de l'enfermer entre des accolades { }.

```
Un texte {\tiny avec la commande tiny} contrainte par des accolades.
```

Code

Un texte avec la commande tiny contrainte par des accolades.

Résultat

On a fait le tour des commandes LaTeX. J'espère que tu n'es pas trop noyée. Mais ne t'inquiètes pas, on se familiarise rapidement avec ce système qui finalement est assez logique.

2.3.4 Environnement ou déclaration *That's the Question* ?

Certaines fonctions possèdent plusieurs types de commande. Pour mettre en gras on a vu la forme à argument `\textbf{texte}`, et bien il existe une forme déclarative `\bfseries` et un environnement `\begin{bfseries}` ... `\end{bfseries}`. Pour la gestion des polices les environnements ne sont rarement utilisés. Si tu dois formater un mot ou un petit groupe de mot la forme à arguments `\textbf{texte}` est plus facile, par contre quand tu crées une commande [1] la forme déclarative (`\bfseries`) est plus indiquée. De même quand le formatage du texte intervient dans un autre environnement. Par exemple si tu veux mettre ton texte en rouge et en gras, tu peux coder cela de cette façon :

```
Un texte normal et{\color{red}\bfseries un texte en rouge et en gras}.
```

Code

Un texte normal et **un texte en rouge et en gras**.

Résultat

1. et oui avec LaTeX tu peux créer tes commandes, on étudiera cette fonction dans un prochain chapitre

2.4 Où Mlle Michu écrit enfin sa première ligne de code

Revenons à nos moutons et écrivons notre première ligne, tu te souviens, celle qui nous permet de déclarer le type de document que l'on veux réaliser (un rapport avec la classe *scrreprt*), donc tu tapes :

```
\do
```

STOP !!

Pas la peine d'aller plus loin. Tapes maintenant sur la touche Tab de ton clavier et miracle

`\documentclass{|}` s'écrit tout seul et le curseur est placé entre les accolades. Tu n'as plus qu'à taper *scrreprt*.

```
\documentclass{scrreprt}
```

2.5 Où Mlle Michu est ébahie par la complétion automatique

Fabuleux, c'est ce que l'on appelle la complétion automatique (ou auto-complétion). Apprend à l'utiliser et fait le systématiquement, tu gagneras beaucoup de temps. Je sais que la période est à la souris et au *clicodrome*, mais crois moi faire tout à partir du clavier est beaucoup plus rapide et efficace que l'utilisation de la souris.

Un exemple : la commande pour mettre en gras sous LaTeX est `\textbf{}`. Avec la complétion automatique il me suffit de taper `bf` et Tab et d'écrire le mot ou la phrase à mettre en gras. Compare avec les actions d'écrire le mot, de prendre la souris, de sélectionner ce mot ou cette phrase, de cliquer sur l'icône qui met en gras et de revenir au clavier. Et bien pendant que tu as fait tout cela, moi j'ai déjà écrit la deuxième phrase.

Tu me rétorqueras « oui mais il faut apprendre un maximum de commande avant d'être efficace ». Dans la réalité le nombre de commandes à savoir n'est pas si important, c'est souvent les mêmes qui reviennent. Et de toute façon tous les logiciels demandent un apprentissage. Il a bien fallu que tu apprennes que l'icône avec un G (ou un B) dessus servait à mettre en gras. Tu n'as pas le choix, TeXworks est dépourvu d'icône de mise en forme. Donc arrête la mauvaise fois et au boulot.

2.6 Où Mlle Michu choisi le format de page, la taille du texte, le recto-verso

```
\documentclass{screprt}
```

Avec cette commande tu auras un document avec un format de papier a4, un texte de taille 12 pt et en recto simple, qui sont les valeurs par défaut de KOMA-Script. Comme tu veux un document en 10 pt et en recto-verso, tu vas ajouter les options suivantes

```
\documentclass[10pt,twoside]{screprt}
```

LaTeX n'accepte que 3 tailles de texte 10, 11 et 12 pt. KOMA-Script accepte n'importe quelle taille, mais attention certaine commandes LaTeX risquent de ne pas

aimer et tu peux te retrouver avec des problèmes imprévus si tu utilises une fontes de 14 pt par exemple. Mais quelle idée de prendre une fonte de 14 pt pour écrire un rapport, à moins de n'avoir rien à dire et faire du remplissage.

Attention

Choisir une taille de fonte ne veut pas dire que l'intégralité du document sera de cette taille. Tu pourras écrire des portions de texte en plus petit ou en plus grand. Mais les commandes pour le faire (`\small` ou `\Huge`, par exemple) sont proportionnelle à la taille choisie ici.

L'option `twoside` permet d'avoir un document en recto-verso, ce qui a une influence sur les marges, la numérotation des pages, les entêtes,...

2.7 Mlle Michu perplexe doit indiquer l'encodage

La police de caractères utilisée par défaut par LaTeX (appelée CM ou Computer Modern) ne possède pas les caractères tel que â, é, ï, ç, ù... Il faut donc charger une police adéquate par :

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

La première ligne `\usepackage[T1]{fontenc}` permet à LaTeX d'utiliser des fontes modernes T1 possédant tous les caractères utilisées par les langues d'Europe de l'ouest.

La deuxième ligne `\usepackage{lmodern}` charge la fonte *lmodern* qui est un équivalent, avec tous les caractère spéciaux, de Computer Modern.

On a un autre problème à régler avant de pouvoir taper des textes en français avec LaTeX, lui faire reconnaître les caractères accentués du clavier et éviter d'avoir à taper `\'e` pour é. Pour cela il suffit d'ajouter :

```
\usepackage[utf8]{inputenc}
```

Donc pour avoir des accents qui vont bien avec une belle police, il faut fournir à LaTeX ces trois commandes :

Code

```
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{lmodern}
```

2.8 Où Mlle Michu poursuit son apprentissage de l'auto-complétion

Ici aussi l'auto-complétion fait des miracles. Il te suffit de taper `\us` puis Tab et tu obtiens

```
\usepackage{}
```

Ce qui n'est pas vraiment ce que tu veux puisqu'il manque les crochets. Donc au lieu de taper une seule fois sur Tab tu tapes une deuxième fois. Et tu obtiens

```
\usepackage[|]{•}
```

Taper plusieurs fois sur Tab fera défiler toutes les options possibles commençant par `\us`. Tu choisis celle qui te convient. Le curseur est placé entre les crochets tu inscris donc :

`\usepackage[utf8]{•}`

Tu peux remarquer qu'il y a un • entre les parenthèses. Tu peux te déplacer sur ce • avec la combinaison de touches Ctrl Tab . Tu le remplaces par

`\usepackage[utf8]{inputenc}`

Tu verras par la suite que certaines commandes possèdent plusieurs • afin de remplir plusieurs champs. Donc pour passer de l'un à l'autre tu utilises la la combinaison de touches Ctrl tab et pour revenir en arrière la combinaison de touches Ctrl Shift tab . Et oui il te faut 3 doigts pour y parvenir, mais tu vas très vite acquérir une certaine agilité au niveau des doigts, très profitable et je te rappelle qu'il y a 2 touches Shift sur un clavier, une pour chaque main.

Tu procèdes de même pour la deuxième commande.

2.9 Où M^lle Michu n'a pas tout compris pour la complétion automatique

M^lle Michu : Il y a des fois ou tu tapes le début de la commande avec la barre (`\`) et d'autre comme pour mettre en gras « bf » qui n'est pas le début de la commande (`\textbf{}`). C'est quoi ce bazar ?

Effectivement une petite explication s'impose. Si je tape `\te`, je suis obligé de taper 6 fois sur Tab pour compléter la commande pour écrire en gras et 11 fois pour l'italique, car des commandes commençant par `\te`, il y en a un certain nombre. De la même façon vu le nombre d'environnement différent (les trucs qui commencent par un `\begin`), réaliser la complétion en tapant `\be`, ne risque pas de faciliter la frappe. Donc pour améliorer le système TEXworks a défini une liste de raccourci que tu trouveras à la fin de ce manuel `http://www.leliseron.org/texworks/`. C'est pour cette raison que tu peux taper bf+tab pour mettre en gras, bce+tab pour centrer (le b pour begin et le ce pour center), se+tab pour section, sse pour sous-section,... Dans la suite de ce document j'indiquerais en magenta le raccourci pour la complétion (bf + Tab , bce + Tab , se + Tab ...).

Dans le manuel cité ci-dessus tu trouveras la façon de créer tes propres raccourcis. Si « bf » ne te conviens pas, tu peux faire en sorte que « gr » te donne le même résultat.

2.10 Où M^lle Michu fait appel à la police

Le nombre de police de caractère est limité sous LaTeX (enfin c'est pas tout à fait exact mais choisir une police exotique nous entraînerait trop loin et n'entre pas dans le propos de ce tutoriel). Le but est quand même de faire un rapport scientifique, pas un magazine « people ». Le rôle de la typographie est de faciliter la lecture afin que le lecteur reste concentré sur le fond. Les polices de LaTeX répondent à cette nécessité. Si tu trouves la police *lmodern* trop LaTeXiène, voici quelques polices que tu peux essayer (tu remplaces dans `\usepackage{lmodern}` *lmodern* par le nom donné) :

– *fourier* pour la police Utopia ;

– *mathptmx* pour la police Times ;
– *mathpazo* pour la police Palatino ;
– *helvet* pour la police Helvetica ;
– *kpfonts* la police utilisée dans ce document.

Pour plus de polices je te renvoie au site :
http://www.tug.dk/FontCatalogue/.

Tu trouveras également des exemples de polices de caractères dans le document de Marc Baudoin « Apprends LaTeX ! », chapitre 8.5 « Changer la police de caractères », page 94, disponible à cette adresse : http://www.babafou.eu.org/Apprends_LaTeX/Apprends_LaTeX.pdf

2.11 M^lle Michu veut parler français

Chaque langue possède des règles typographiques qui lui sont propre. On ne coupe pas les mots de la même manière en anglais et en français. Les items d'une listes commence par • en anglais et – en français. En anglais et en allemand on ne met pas d'espace avant les signes doubles (; :!?), en français on met une espace insécable avant et une espace normale après pour le : et pour les ;!? ont met avant une espace fine insécable et une espace normale après [2]

La langue naturel de LaTeX est l'anglais. Le rôle du package *babel* est de permettre à LaTeX de parler d'autres langues, à l'aide d'extension comme *french* pour le français [3].

```
\usepackage[french]{babel}
```

Grâce à ce package les césures des mots en fin de ligne seront correctes, les espaces entre les signes de ponctuation seront respectés, l'entête des chapitres sera écrit « Chapitre » et non « Chapter », les légendes des figures commenceront par « Figure » ...

Attention

Il est recommandé que ce package *babel* soit le dernier package chargé, car il modifie le comportement de certains autres packages.

2.12 M^lle Michu achève le préambule

Pour l'instant le préambule, c'est comme cela que l'on appelle la première partie du source qui comprend les paramètres de mise en page et le chargement des packages dont tu as besoin, comprend 5 lignes :

Code

```
\documentclass[10pt,twoside]{scrreprt}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{lmodern}
\usepackage[frenchb]{babel}
```

On va s'arrêter là pour l'instant. On complétera le préambule au fur et à mesure de nos besoins. Maintenant tu vas ajouter deux balises `\begin{document}`

2. L'espace insécable est une espace qui ne peut être coupée par un saut de ligne. Elle s'écrit ~ en LaTeX (`\,` pour l'espace fine insécable).
3. *french* est maintenue par Daniel Flipo, la dernière version est la 2.5a (21 août 2010)

et \end{document}. C'est dans cet environnement que tu vas saisir le texte de ton rapport.

```
\documentclass[10pt,twoside]{scrreprt}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{lmodern}
\usepackage[frenchb]{babel}
\begin{document}

\end{document}
```

3 M^lle Michu veut une méthode de travail

Je vais t'indiquer une méthode de travail. C'est celle que j'utilise pour la rédaction de tous mes rapports, mais bien entendu tu peux faire autrement. Dans ce tutoriel, nous allons étudier les éléments dans l'ordre, la page de titre puis l'introduction le premier chapitre... les annexes. Mais dans la pratique je ne procède pas de cette manière. En général je me préoccupe du titre à la fin et il est rare que j'écrive tous les chapitres dans l'ordre. Ça c'est le gros avantage de LaTeX sur les traitements de texte classiques, comme tu ne t'occupes pas de la mise en page, tu peux faire les choses dans l'ordre que tu veux LaTeX se débrouillera tout seul pour recoller les morceaux correctement. Tu peux toujours rajouter des éléments même en début de texte sans avoir à te préoccuper du décalage que cela va occasionner sur les images, les tableaux,... déjà insérés. Tu verras LaTeX gère cela très bien.

3.1 M^lle Michu divise pour mieux régner

Quand j'écris un rapport je ne le fais pas dans un seul fichier, mais je crée autant de fichier que de chapitres ou de section. C'est plus facile pour s'y retrouver, surtout si le document est important. Tu peux diviser ton document comme tu l'entends, tu n'es pas obligé de respecter la division par section ou chapitre. Tu peux mettre plusieurs sections dans un fichier ou avoir plusieurs fichiers pour une section. Le but est de pouvoir accéder rapidement et facilement à n'importe quel partie de ton document. Tu organises ton travail comme tu veux. Pour ce tutoriel on va diviser le document de la façon suivante :

monRapport.tex c'est le fichier maitre, qui contient le préambule et l'appel aux autres fichiers ;

introduction.tex comme son nom l'indique, contient le texte d'introduction du document ;

chapitre1.tex là aussi le nom est explicite. En général je ne l'intitule pas de cette manière mais avec un mot du titre, car c'est plus parlant et il se peut que ce chapitre ne reste pas toujours le premier ;

... la suite des fichiers par chapitre ou section ou partie de section ;

conclusion.tex ce serait pas mal de mettre la conclusion dans ce fichier ;

annexe1.tex une annexe, pour ce tutoriel il n'y en aura qu'une seule mais tu peux en créer plusieurs.

3.2 M^lle Michu appelle un fichier

La commande pour appeler un fichier est `\input{monFichier}` Tu n'indiques pas l'extension `.tex`. Le fichier appelé doit être dans le même répertoire que le

fichier appelant. Il parait assez logique d'avoir préalablement créé un répertoire qui contiendra tous les documents nécessaires à la réalisation de ton rapport. Tu vas donc créer ton fichier maître *monRapport.tex*, qui contient les lignes suivantes

Code

```
\documentclass[10pt,twoside]{scrreprt}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{lmodern}
\usepackage[frenchb]{babel}

\begin{document}
\input{introduction}
\end{document}
```

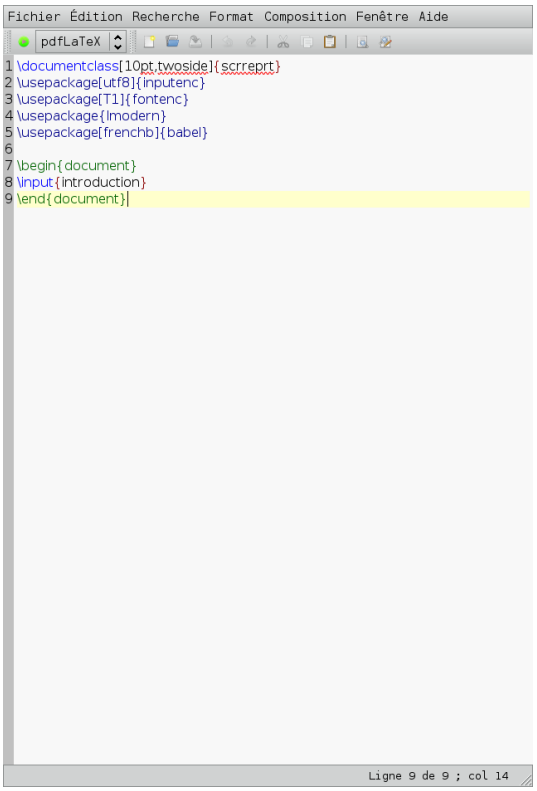


FIGURE 3.1 : Vue du fichier maître dans la fenêtre TEXworks. Remarque les effets de la coloration syntaxique.

Tu dois donc avoir dans ta fenêtre TEXworks quelque chose qui ressemble à la figure 3.1. Il est temps de sauvegarder ton premier fichier, menu *Fichier* sous menu *Sauvegarder*. Pour notre exemple tu vas l'appeler `monRapport.tex`. Ce sera notre fichier maître.

3.3 M^lle Michu introduit

Comme tu fais appel au fichier *introduction.tex*, tu vas le créer. Un appel à un fichier qui n'existe pas va produire une erreur lors de la compilation. Ctrl N ouvre une nouvelle fenêtre TEXworks. Tu as remarqué on ne suit déjà plus l'ordre que j'avais indiqué, mais c'est pas grave on s'occupera du titre plus tard.

On va rentrer dans le vif du sujet, et tu vas te rendre compte que le travail avec LATEX est différent de celui fourni avec un traitement de texte classique. Avec LATEX tu dois juste indiquer par une commande que l'introduction est un chapitre et c'est tout. LATEX va ce charger tout seul de la numérotation, de la taille et de la graisse de la police. Ce n'est pas ton problème, toi tu t'occupes du texte de ton introduction.

3.4 M^lle Michu sectionne

Le sectionnement est différent pour les classes livre et rapport d'une part (présence de chapitre) et article d'autre part (pas de chapitre). LATEX attribue un numéro de niveau à ces divisions, qui est utilisé dans certaines options comme la profondeur de la numérotation (voir section 3.4.4 p.29). Les divisions sont automatiquement numérotées.

article (scrartcl)			livre (scrbook) et rapport (scrreprt)		
nom	niveau	numérotation	nom	niveau	numérotation
\part	0	Première partie	\part	-1	Première partie
\section	1	1.	\chapter	0	1.
\subsection	2	1.1.	\section	1	1.1.
\subsubsection	3	1.1.1.	\subsection	2	1.1.1.
\paragraph	4	pas de numérotation	\subsubsection	3	pas de numérotation
\subparagraph	5	pas de numérotation	\paragraph	4	pas de numérotation
			\subparagraph	5	pas de numérotation

Table 3.1 : Différentes divisions en fonction du type de document

```
\chapter{Un chapitre}
\section{Une section}
\subsection{Une sous section}
\subsubsection{Une sous sous section}
\paragraph{Un paragraphe}
\subparagraph{Un sous paragraphe}
```

Code

1 Un chapitre

1.1 Une section

1.1.1 Une sous section

Une sous sous section

Un paragraphe

Un sous paragraphe

Résultat

3.4.1 Taille des divisions

KOMA-Script possède 3 formats de taille de divisions différents que l'on modifie par l'option *headings* : *headings=big* (option par défaut), *headings=normal*, *headings=small*. Si tu veux une petite taille des titres voici comment faire :

```
\documentclass[10pt,twoside,headings=small]{scrreprt}
```

Code

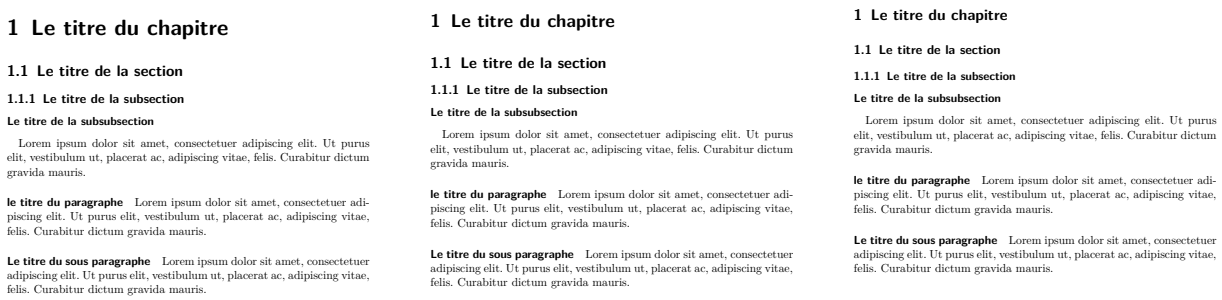


Figure 3.2 : Les trois formats de taille de divisions de KOMA-Script, de gauche à droite *headings=big*, *headings=normal*, *headings=small*.

3.4.2 Arguments optionnels des divisions

Toutes les divisions se présentent de la même façon :

```
\nomDivision[titre court]{titre de la division}
```

[titre court] permet d'indiquer un autre titre qui sera utilisé dans la table des matières ou les entêtes. Cela est très utile par exemple pour indiquer une forme raccourcie d'un titre long , qui déborde dans la table des matières ou dans l'entête.

Tu peux aussi forcer une division à s'écrire sur plusieurs lignes à l'aide de \\ (voir la section 3.7 page 35 pour les commandes de saut de ligne :

Code

```
\section{Un titre sur\\ deux lignes}
```

Mais cette méthode peut introduire des effets indésirables dans la table des matières ou les entêtes. Pour éviter ces problèmes il vaut mieux réécrire le titre sans \\ dans l'argument optionnel :

Code

```
\section[Un titre sur une ligne]{Un titre sur\\ deux lignes}
```

Le titre optionnel sert également dans le cas où l'on désire mettre une note de bas de page dans un titre. L'apparition problématique de la note de bas de page dans le sommaire peut être évité par l'emploie du titre optionnel :

Code

```
\section[Un titre sans note de bas de page]{Un titre avec une note
de bas de page\footnote{La note}}
```

Remarque

Les commandes comme \section sont appelées des commandes mobiles car elle apparaissent plusieurs fois dans le document. La même section apparaitra dans le texte, dans la table des matières et dans l'entête. Les commandes comme \footnote, dite fragile, n'aime pas trop se retrouver dans des commandes mobiles. Donc pour éviter des problèmes de compilation il vaut mieux les protéger par la commande \protect

```
\section[Un titre sans note de bas de page]{Un titre avec une note
de bas de page\protect\footnote{La note}}
```
Code

3.4.3 Formes sans numérotation

Les commandes *addpart*, *addchap* et *addsec* produisent des divisions sans numérotation qui apparaissent dans la table des matières et dans l'entête. On se sert de cette forme notamment pour l'introduction ou la conclusion qui ne sont généralement pas numérotées.

```
\addpart[version courte]{titre}
\addchap[version courte]{titre}
\addsec[version courte]{titre}
```
Code

Il existe également une version étoilé qui ne produit d'entrée ni dans la table des matières ni dans l'entête.

```
\addpart*{titre}
\addchap*{titre}
\addsec*{titre}
```
Code

Ce sont des commandes KOMA-Script, qui n'existent pas dans les classes classiques (book, article, report).
Attention

3.4.4 Profondeur de la numérotation

Un petit rappel pour te signaler que la lecture des parties dont le titre est en italique n'est pas indispensable dans un premier temps.
Remarque

Par défaut pour LaTeX la numérotation des divisions se fait jusqu'à `\subsection` comprise, au-delà les divisions ne sont plus numérotées. Cela correspond au niveau 2 (voir le tableau 3.1 p. 27).

Cette profondeur de numérotation est contrôlée par le compteur secnumdepth dont la valeur par défaut est 2. Tu peux modifier sa valeur par la commande :

```
\setcounter{secnumdepth}{nombre}
```
Code

Par exemple `\setcounter{secnumdepth}{4}` étendra la numérotation jusqu'au `\paragraph`.

Si tu veux que toutes les divisions de ton document soit numérotées, utilise une grande valeur comme 10 `\setcounter{secnumdepth}{10}`.

Au contraire avec une profondeur de -2 aucune division ne sera numérotée. (part la division la plus grande à la valeur -1). Dans ce cas tu n'es plus obligé d'utiliser les commandes KOMA-Script comme `\addsec{}`.

Si tu ne connais pas le niveau de profondeur de ton document et que tu veuilles ajouter un niveau de numérotation supplémentaire, tu peux utiliser la commande `\addtocounter{secnumdepth}{1}`, tu mets `\addtocounter{secnumdepth}{2}` pour 2 niveaux supplémentaires et ainsi de suite.

Toutes ces commandes se placent avant le `\begin{document}`.

3.5 Mlle Michu confrontée aux compteurs

Il existe un certain nombre de compteurs prédéfinis dans LaTeX, comme par exemple les compteurs de pages, de chapitres, de sections, de numéros de figures,... Tu seras peut-être amenée à en modifier quelques uns pour adapter le comportement par défaut de LaTeX à tes souhaits (voir la modification du numéro de page section 12.9, page 128). Ce n'est pas difficile, il existe deux commandes, `\setcounter{compteur}{nouvelleValeur}` qui attribue au compteur la nouvelle valeur et `\addtocounter{compteur}{nouvelleValeur}` qui incrémente de la nouvelle valeur le compteur (se nombre peut être négatif pour diminuer le compteur). Voici des exemples avec le compteur de sous-section. Le compteur des division de texte à le même nom que la division donc notre compteur s'appelle *subsection*. Voici une première sous-section de la première section du chapitre 1, obtenue par le code suivant :

Code

```
\section{Première section} \subsection{Première sous-section}
```

Résultat

1.1 Première section

1.1.1 Première sous-section

Pour la deuxième sous-section on va mettre le compteur à 6. Attention pour bien comprendre ce qui se passe , il faut savoir que la commande `\subsection` *incrémente de 1* le compteur *puis affiche* le résultat. On attribue la valeur du compteur de subsection à 6, la commande `\subsection` l'incrémente 6+1=7 et affiche le résultat :

Code

```
\setcounter{subsection}{6}
\subsection{Deuxième sous-sous-section}
```

Résultat

1.1.7 Deuxième sous-section

On va incrémenter de 2 pour la troisième, donc notre compteur vaut 7+2=9 et la commande `\subsection` l'incrémente 9+1=10 :

Code

```
\addtocounter{subsubsection}{2}
\subsection{Troisième sous-section}
```

Résultat

1.1.10 Troisième sous-section

Enfin on va diminuer le compteur pour la dernière le compteur vaut 10-4=6 et la commande `\subsection` l'incrémente 6+1=7 :

Code

```
\addtocounter{subsection}{-4}
\subsection{Dernière sous-section}
```

Résultat

1.1.7 Dernière sous-section

3.6 Où M^lle Michu continue d'introduire

En général l'introduction est un chapitre non numéroté. On vient de le voir, KOMA-Script possède une commande pour faire cela `\addchap{Le titre}`. Donc notre fichier *introduction.tex* contient

Code

```
\addchap{Introduction}
La découverte d'un nouveau gisement moustérien dans cette région
du Balafon Occidental est toujours un événement. Surtout quand
cette découverte à lieu dans la vallée du Pételasoiffe, appelée
par de nombreux auteur la \og{} Vallée de l'Omo\fg{} de l'Europe du
Nord\footnote{Rendons à César ce qui\dots c'est au professeur Raoul
\bsc{Dumortier} que l'on doit la première mention de cette
expression.}. Le site d'Harfouëte -- La Mare au Diable, découvert
en 2009, outre son insertion dans un contexte micro-régional riche,
faisant déjà l'objet de très nombreuses recherches\footnote{Une
synthèse sous la direction de Robert \bsc{de Latour} devrait
paraître à la fin de l'année 2010.}, présente un grand intérêt
intrinsèque. En effet la présence de \emph{plusieurs foyers} et de
\emph{reste osseux} rendent ce site unique dans cette région.
\par
Le gisement d'Harfouëte -- La Mare au Diable est situé à proximité
de deux sites mondialement connu. En amont, à 500\,m à vol d'oiseau
se trouve le site de \og{} La roche Penchée\fg{}, fouillé depuis les
années 1887. À 1,5\,km en aval, se situe le gisement moustérien de
\og{}La Castellerie Rouge\fg{}, étudié depuis près de vingt ans par
l'équipe américaine du professeur A. L. \bsc{Robinson}.
```

Voici le fichier *introduction.tex*. On a déjà vu dans les précédentes pages la signification de certaines commandes utilisées dans celle-ci.
`\og` qui écrit le guillemet ouvrant « , *o* pour ouvrez et *g* pour guillemet ;
`\fg` qui écrit le guillemet fermant », *f* pour fermez et *g* pour guillemet ;
`\dots` pour écrire … qui sont les véritables trois points typographiques, ce qui n'est pas pareil que ... trois points écrits au clavier. Quatre nouvelles commandes sont apparues : `\footnote{}`, `\bsc{}`, `\par`, `\emph{}` et deux bizarreries -- et `\,`.

3.6.1 Première difficulté pour M^lle Michu : les notes en bas de page

Avec LaTeX ce n'est pas une difficulté. Il te suffit juste d'employer la commande `\footnote{Ici le texte de ta note de bas de page}` que tu places dans

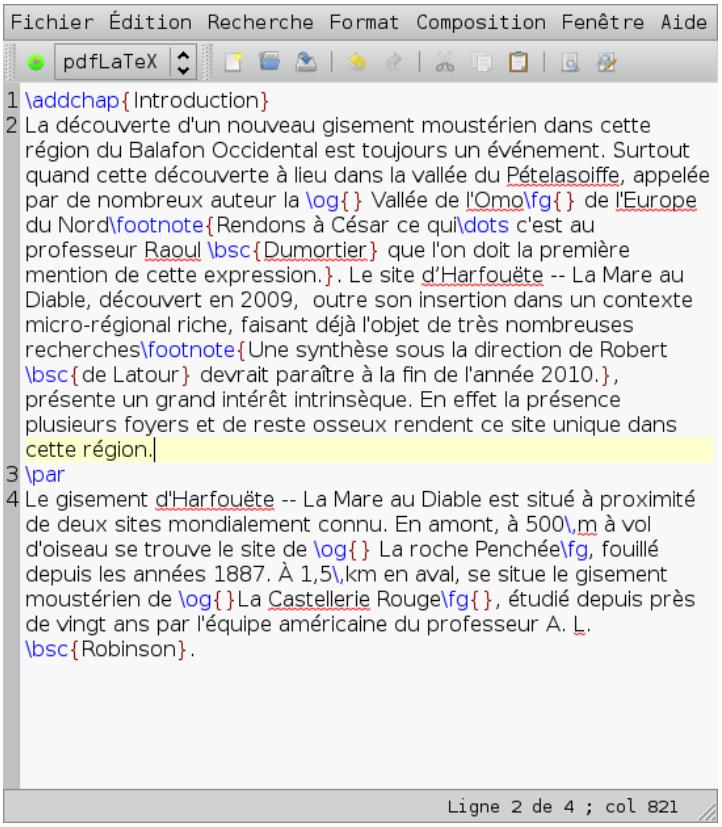


Figure 3.3 : L'introduction dans TEXworks. Tu remarques l'intérêt de la coloration syntaxique. Les commandes LATEX se détache nettement du texte. Les mots soulignés de rouge indique les mots mal orthographiés. Ici des noms propres inconnus du dictionnaire. Remarque également que pour TEXworks une ligne se termine par un « retour de chariot ». pour ce dernier point voir la section 3.7 p. 35

le texte à l'endroit ou te veux voir apparaître l'appelle de la note. LaTeX s'occupera de l'attribution du numéro et de son placement.

3.6.2 La bizarrerie --

En typographie française il existe 3 types de tirets (pour plus de détail, voir l'article sur wikipedia : http://fr.wikipedia.org/wiki/Tiret). Le trait court qui sert dans les césures, comme trait d'union ou pour séparer les intervalles (2000 - 2010 par exemple) et qui s'écrit « - » (le signe moins de ton clavier numérique). Le tiret moyen (demi-cadratin) qui sert pour les énumérations, c'est également le signe moins des soustractions (5 – 3 = 2) et qui s'écrit « -- » (deux signes moins de ton clavier numérique accolés). Enfin le tiret long (cadratin) utilisé pour introduire les dialogues, ou pour remplacer les parenthèses qui s'écrit « --- » — trois signes moins de ton clavier numérique accolés —.

3.6.3 La bizarrerie \,

Je ne sais pas si tu auras besoin de manipuler beaucoup de chiffres et d'unités dans ton document, mais moi dans mes rapports archéologiques, j'en consomme énormément. Pour cela comme pour le reste il existe des règles typographiques. En français les décimales sont délimitées par une virgule (c'est un point en anglais). Les milliers sont séparés par une espace fine insécable (une virgule en anglais, décidément ces anglais ne font rien comme tout le monde !). Les chiffres sont séparés des unités également par une espace fine insécable. L'espace fine est deux fois plus petite que l'espace de la barre d'espace de ton clavier. Insécable signifie qu'elle ne peut être coupée en fin de ligne. Le nombre et les unités ne peuvent pas être sur deux lignes différentes. La commande `\,` permet de créer une espace fine insécable.

Dans la pratique je ne procède pas de cette façon pour écrire mes unités. J'utilise le package *numprint* qui gère l'écriture des nombres. Les deux mesures de mon introduction peuvent s'écrire avec *numprint* : `\np[m]{500}` et `\np[km]{1,5}`, de cette façon je suis sûr que les normes typographiques françaises seront respectées.

3.6.4 *Mlle Michu et les nombres*

Si tu dois écrire beaucoup de nombres, il est préférable d'utiliser le package numprint (`\usepackage{numprint}`) qui associé à french produira la bonne typographie.

code		résultat
123456789	=	123456789
`\numprint{123456789}`	=	123 456 789
`\numprint{3,14159265358979323846}`	=	3,141 592 653 589 793 238 46
`\numprint{12.34}`	=	12,34

En plus tu peux préciser les unités de cette façon : `\numprint[kg]{12,34}` = 12,34 kg.

Si tu veux utiliser le raccourci `\np{}` au lieu de `\numprint{}`, il te faut charger le package de cette façon : `\usepackage[np]{numprint}`

3.6.5 M^lle Michu capitalise petit

Une des règles en typographie française veut que le nom de famille soit écrit en PETITES CAPITALES. C'est le rôle de la commande `\bsc{Le Nom}`.

Attention

La commande `\bsc{}` est une commande de l'option *french* du package *babel*. La commande LaTeX est `\textsc{}`. La différence entre les deux est que `\bsc` empêche la coupure du mot en fin de ligne, conformément aux règles typographiques françaises gérant les noms propres.

3.6.6 M^lle Michu emphatique

La commande `\emph{}` permet de mettre une portion de texte en emphase. En typographie française mettre en relief un mot ou un groupe de mot se fait en utilisant l'italique dans un texte droit (police romaine) et en romain dans un texte en italique. C'est ce que fait `\emph{}` :

Code

```
Comme le faisait remarquer Bébert dans son \emph{excellente étude}
sur la bière \og \textit{Une bière \emph{sans houblon}, ne vaut pas
la peine d'être bue}\fg
```

Résultat

Comme le faisait remarquer Bébert dans son *excellente étude* sur la bière « *Une bière* sans houblon, *ne vaut pas la peine d'être bue* »

Si tu veux insister lourdement tu peux mettre la portion de texte en gras avec la commande `\textbf{}`.

Puisque que l'on est dans les changement de fonte voici un tableaux récapitulatif des commandes de changement de fonte :

Commande	Déclartion	Résultat	auto-complétion
`\textrm{...}`	`{\rmfamily ...}`	texte en roman	rm + Tab
`\textsf{...}`	`{\sffamily ...}`	texte sans serif	sf + Tab
`\texttt{...}`	`{\ttfamily ...}`	machine à écrire	tt + Tab
`\textup{...}`	`{\upshape ...}`	écriture droite (défaut)	up + Tab
`\textit{...}`	`{\itshape ...}`	*texte en italique*	it + Tab
`\textsl{...}`	`{\slshape ...}`	*écriture penchée*	sl + Tab
`\textsc{...}`	`{\scshape ...}`	PETITES CAPITALES	sc + Tab
`\textmd{...}`	`{\mdseries ...}`	médium (défaut)	md + Tab
`\textbf{...}`	`{\bfseries ...}`	**texte en gras**	bf + Tab

3.6.7 M^lle Michu change de paragraphe

`\par` permet de changer de paragraphe. Avec `\par`, il est temps d'aborder la gestion des espaces et des paragraphes sous LaTeX.

3.7 Mlle Michu dans l'espace

LaTeX compose le texte en suivant des règles typographiques précises dépendantes de la langue utilisée. Concernant les espaces voici quelques unes de ces règles :

Les mots sont séparés par une espace et une seule. Donc tu peux toujours ajouter autant d'espace, de tabulation entre les mots, LaTeX les ignore et n'en mettra qu'une seule. Pour illustrer le propos un extrait de « Les misérables » de Victor Hugo.

Code

```
L'armée plia brusquement de tous les côtés à la fois, de Hougomont,
de la Haie-Sainte,          de Papelotte,                  de
Plancenoit. Le cri trahison ! fut suivi du cri          sauve-qui-
peut ! Une armée qui se débande, c'est un dégel. Tout fléchit, se
fêle,                          craque, flotte, roule, tombe, se
heurte, se hâte,                          se précipite.
```

donne après compilation

Résultat

L'armée plia brusquement de tous les côtés à la fois, de Hougomont, de la Haie-Sainte, de Papelotte, de Plancenoit. Le cri trahison ! fut suivi du cri sauve-qui-peut ! Une armée qui se débande, c'est un dégel. Tout fléchit, se fêle, craque, flotte, roule, tombe, se heurte, se hâte, se précipite.

En typographie française, les paragraphes sont uniquement marqué par un passage à la ligne avec une indentation (un retrait) de la première ligne. Il n'y a *pas* de ligne blanche entre deux paragraphes. Avec LaTeX il y a deux manières de changer de paragraphe, ajouter une ligne blanche dans le source ou ajouter la commande \par.

Code

```
L'armée plia brusquement de tous les côtés à la fois, de Hougomont,
de la Haie-Sainte, de Papelotte, de Plancenoit.

Le cri trahison ! fut suivi du cri sauve-qui-peut ! Une armée qui se
débande, c'est un dégel.
\par
Tout fléchit, se fêle, craque, flotte, roule, tombe, se heurte, se
hâte, se précipite. Désagrégation inouïe.\par Ney emprunte un che-
val, saute dessus, et, sans chapeau, sans cravate, sans épée...
```

Résultat

L'armée plia brusquement de tous les côtés à la fois, de Hougomont, de la Haie-Sainte, de Papelotte, de Plancenoit.

Le cri trahison ! fut suivi du cri sauve-qui-peut ! Une armée qui se débande, c'est un dégel.

Tout fléchit, se fêle, craque, flotte, roule, tombe, se heurte, se hâte, se précipite. Désagrégation inouïe.

Ney emprunte un cheval, saute dessus, et, sans chapeau, sans cravate, sans épée,...

De la même façon que pour les espaces, plusieurs lignes seront ignorées par LaTeX.

Code

```
L'armée plia brusquement de tous les côtés à la fois, de Hougomont,
de la Haie-Sainte, de Papelotte, de Plancenoit.


Le cri trahison ! fut suivi du cri sauve-qui-peut ! Une armée qui se
débande, c'est un dégel.

\par

Tout fléchit, se fêle, craque, flotte, roule, tombe, se heurte, se
hâte, se précipite. Désagrégation inouïe.
```

donne après compilation

Résultat

L'armée plia brusquement de tous les côtés à la fois, de Hougomont, de la Haie-Sainte, de Papelotte, de Plancenoit.

Le cri trahison ! fut suivi du cri sauve-qui-peut ! Une armée qui se débande, c'est un dégel.

Tout fléchit, se fêle, craque, flotte, roule, tombe, se heurte, se hâte, se précipite. Désagrégation inouïe.

Des doubles \\ permettent de passer à la ligne sans changer de paragraphe. Tu n'as donc pas de retrait de la première ligne.

Code

```
L'armée plia brusquement de tous les côtés à la fois, de Hougomont,
de la Haie-Sainte, de Papelotte, de Plancenoit.\\
Le cri trahison ! fut suivi du cri sauve-qui-peut ! Une armée qui
se débande, c'est un dégel. Tout fléchit, se fêle, craque, flotte,
roule, tombe, se heurte, se hâte, se précipite. Désagrégation in-
ouïe.
```

Résultat

L'armée plia brusquement de tous les côtés à la fois, de Hougomont, de la Haie-Sainte, de Papelotte, de Plancenoit.
Le cri trahison ! fut suivi du cri sauve-qui-peut ! Une armée qui se débande, c'est un dégel. Tout fléchit, se fêle, craque, flotte, roule, tombe, se heurte, se hâte, se précipite. Désagrégation inouïe.

Attention

Un retour de chariot (la touche Entrée de ton clavier) ne provoque pas de passage à la ligne dans ton document final

```
L'armée plia brusquement de tous les côtés à la fois, de Hougomont,
de la Haie-Sainte, de Papelotte, de Plancenoit.
Le cri trahison ! fut suivi du cri sauve-qui-peut ! Une armée qui se
débande, c'est un dégel.
Tout fléchit, se fêle, craque, flotte, roule, tombe, se heurte, se
hâte, se précipite.
Désagrégation inouïe.
Ney emprunte un cheval, saute dessus, et, sans chapeau, sans cra-
vate, sans épée...
```

Code

L'armée plia brusquement de tous les côtés à la fois, de Hougomont, de la Haie-Sainte, de Papelotte, de Plancenoit. Le cri trahison ! fut suivi du cri sauve-qui-peut ! Une armée qui se débande, c'est un dégel. Tout fléchit, se fêle, craque, flotte, roule, tombe, se heurte, se hâte, se précipite. Désagrégation inouïe. Ney emprunte un cheval, saute dessus, et, sans chapeau, sans cravate, sans épée,...

Résultat

Par contre dans TEXworks un retour de chariot provoque la numérotation d'une nouvelle ligne. Cette caractéristique peut te permettre d'aérer ton source sans modifier le résultat final. Compare les figures 3.3, p. 32 et 3.4, p. 38. Ces deux sources produisent exactement le même résultat.

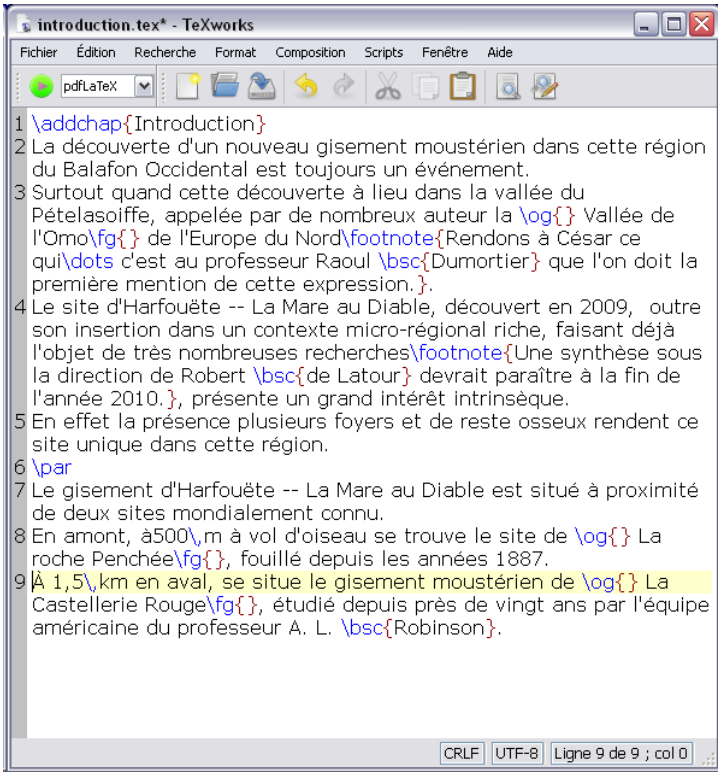


Figure 3.4 : Les retours de chariot provoquent la numérotation d'une nouvelle ligne dans TEXworks sans pour autan fabriquer un saut de ligne dans le document final. Compare avec la figure 3.3 p. 32.

4 Mlle Michu compile pour la première fois

Bien que cela ne soit pas très utile à ce stade, je vois bien que tu veux voir ce que cela donne *en vrai*. Donc tu vas faire ta première compilation.

Tout d'abord il faut passer sur la fenêtre de notre source maître. En effet compiler le fichier *introduction.tex* va produire des erreurs car il ne possède pas de préambule obligatoire pour la compilation. Tu cliques sur le menu *Fenêtre* nº 1 figure 4.1.

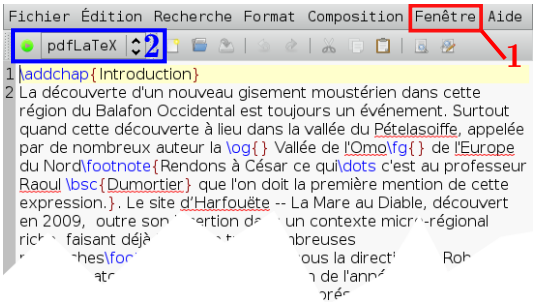


Figure 4.1 : Localisation du menu « Fenêtre » dans TEXworks

En bas, tu as la liste de tous les fichiers `.tex` ouverts, tu choisis ton source maître, *monRapport.tex*, figure 4.2. Théoriquement si tu as bien suivi les instructions pour la configuration de TEXworks tu dois avoir dans ce menu *pdfLaTeX* de sélectionné, nº 2 figure 4.1. Si ce n'est pas le cas tu le choisis dans le menu déroulant. Pour compiler tu cliques sur la flèche verte, ou tu tappes Ctrl T et c'est parti. Au bas de la fenêtre TEXworks, une nouvelle fenêtre (dite de log) va apparaître dans laquelle va défiler tout une série de phrases. Ce sont toutes les instructions que LATEX exécute. C'est également dans cette fenêtre que seront écrit les erreurs rencontrées par le compilateur, mais nous en reparlerons plus tard. Si la compilation c'est déroulée sans encombre une nouvelle fenêtre doit apparaître à droite contenant la première page de ton document en pdf. Il porte le nom du fichier maitre : `monRapport.pdf`

4.1 Où Mlle Michu découvre avec horreur que ça ne compile pas

Mlle Michu qui a pourtant suivit à la lettre mes instructions, lance la compilation et *enfer et damnation* tout s'arrête, la compile ne se termine pas et elle n'obtient pas son fichier `.pdf`

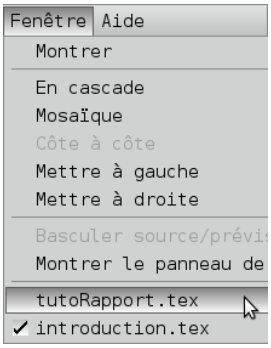


Figure 4.2 : Le menu « Fenêtre » dans TEXworks

Pas de panique. Tu as du faire une erreur dans une ou plusieurs commandes. Utiliser LATEX demande une certaine rigueur. Une commande mal écrite, une parenthèse oubliée et le compilateur plante.

4.2 Où Mlle Michu s'inquiète : comment retrouver l'erreur ?

Mlle Michu
— Comment je fais moi pour retrouver une erreur dans les dizaines de lignes

écrites ?

Quand le compilateur de LaTeX rencontre une erreur il s'arrête. Dans TeXworks tu remarques que la flèche verte qui lance la compilation s'est transformée en une croix rouge. (no 1 fig. 4.3, p. 41).

En bas de la fenêtre de TeXworks est apparue une nouvelle fenêtre. Cette fenêtre est est la fenêtre de log où le compilateur note toutes ses actions. C'est ici que tu trouveras les renseignements concernant l'erreur de compilation. La flèche no 2 (fig. 4.3, p. 41) indique la ligne où est notée le fichier dans lequel se trouve l'erreur ici :

(./introduction.tex (c :/texlive/2009/texmf-dist/tex/latex/lm/t1lmss.fd)

Donc notre erreur est dans le fichier *introduction.tex*. La deuxième partie de la phrase donne le chemin d'instalation de la TeXlive.

La flèche no 3 (fig. 4.3, p. 41) donne le type d'erreur ! Undefined control sequence. (commande inconnue). La flèche no 4 montre le numéro de la ligne de ton fichier *introduction.tex*, où se situe l'erreur l.2, donc ligne 2. Suit ensuite (flèche no 5) un extrait de cette ligne avec un passage à la ligne juste après l'endroit où le compilateur situe l'erreur

... e de l'Omo\fg{} de l'Europe du Nord\fotnote
{Rendons à César ce qui...

Tu te rends compte qu'à la ligne 2 du fichier *introduction.tex* tu as écrit `\Fotnote` au lieu de `\Footnote`.

Le ? (flèche no 6) indique que LaTeX attend une action de ta part. Sous la flèche no 7 tu as une console, dans laquelle tu vas saisir ta réponse (un curseur doit y clignoter).

Les prochaines versions de TeXworks présenterons un nouvel onglet *LaTeX errors* dans la zone des logs qui facilite la lecture des erreurs, voir fig 4.4, page 42. Avant de commencer à paniquer, je te préviens que tu risque de trouver dans cette fenêtre tout un tas de lignes d'écrivant des erreurs du type *Overfull* ou *Underfull*. Pour l'instant nous allons les ignorer. J'aborde ce point à la fin de cet ouvrage à la section 12.10, page 128, car tu ne possèdes pas encore toutes les notions pour comprendre la correction de ce type d'erreurs.

4.3 Où Mlle Michu s'exclame : Oh trop facile, je corrige de suite

La première chose à faire c'est de taper sur la touche entrée de ton clavier pour entrer dans la console, puis plusieurs possibilités s'offrent à toi :

1. tu retapes sur la touche entrée de ton clavier pour poursuivre la compilation, l'erreur est ignorée (donc la note de bas de page ne sera pas écrite) et la compilation se poursuit jusqu'à la fin ou à la prochaine erreur. Parfois l'erreur est trop grave et la compilation ne peut se poursuivre tu passes donc à la solution 2 ;
2. tu saisis x plus la touche entrée pour arrêter la compilation (tu peux également taper Ctrl t). La fenêtre de log reste affichée et tu peux donc encore y voir le message d'erreur, tu corriges dans le source et tu re-compiles ;
3. tu saisis un h plus la touche entrée pour obtenir plus d'aide, c'est pas toujours plus explicite mais peut parfois aider ;

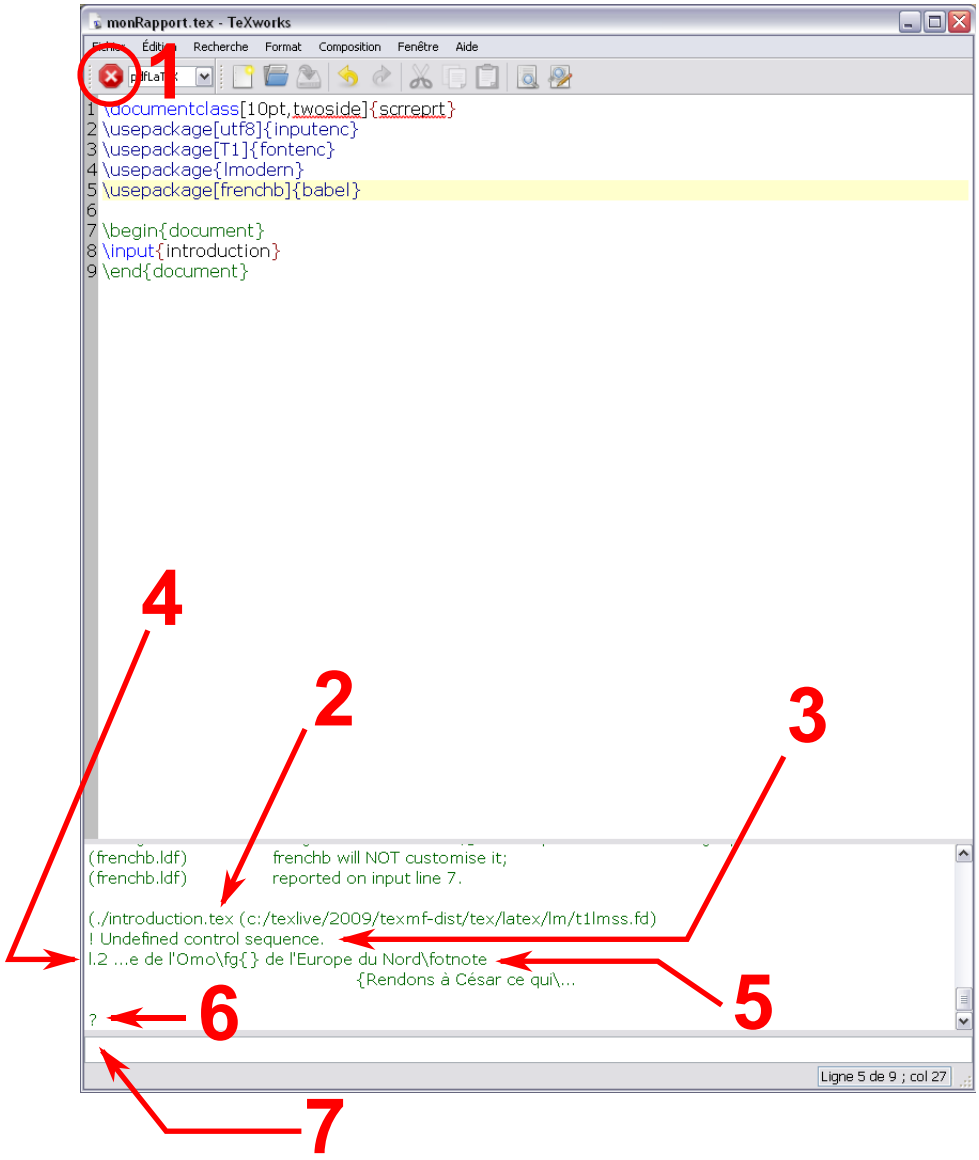


Figure 4.3 : Fenêtre de TEXworks après une erreur de compilation

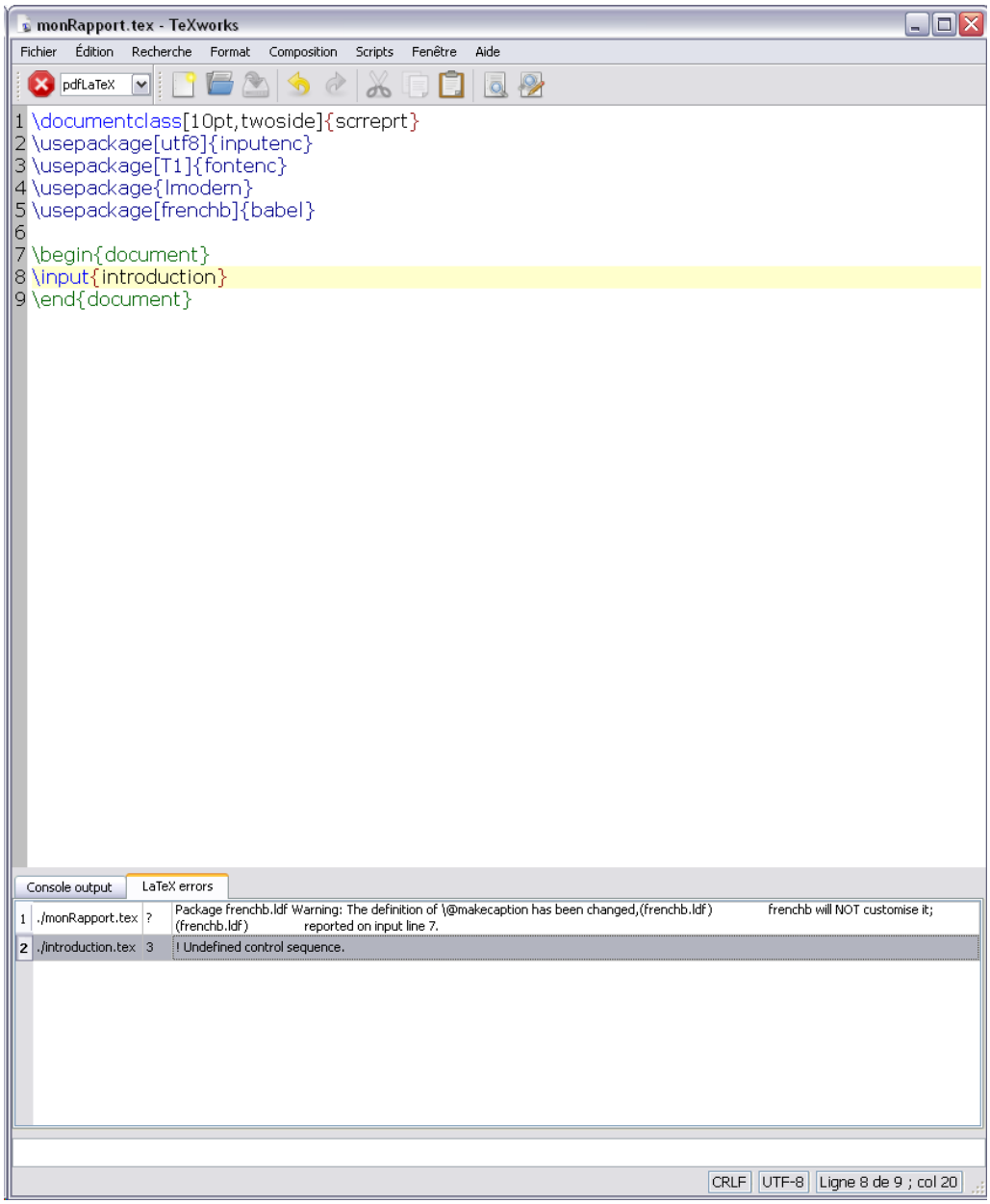


Figure 4.4 : Fenêtre d'une version en développement de TEXworks après une erreur de compilation. Un deuxième onglet (LATEX errors) a été ajouté, il contient un tableau récapitulant les erreurs. La deuxième ligne (en grisé ici) correspond à notre erreur. La première colonne indique le fichier concerné, la deuxième le numéro de ligne et la troisième le type d'erreur. Un double clic sur cette ligne te renvoie directement au passage incriminé.

4. tu peux saisir une commande de remplacement. Pour cela tu saisis un i et entrée puis la commande de remplacement et entrée .

Attention

Avec cette dernière action, il n'y a pas correction dans le fichier source, il faudra faire la correction dans le source après, ça permet juste de vérifier la correction en poursuivant la compilation.

Dans tous les cas le log sera toujours visible et tu pourras le faire défiler pour lire l'ensemble des erreurs.

Dans la pratique lorsque le compilateur rencontre une erreur, je la corrige immédiatement dans le source et je poursuis la compilation (touche *entrée* dans le log) à la recherche d'autre erreurs éventuelles que je corrige au fur et à mesures. Cette première compilation terminée, je relance une compilation qui prend en compte les correction effectuées.

4.4 Où Mlle Michu se rend compte que retrouver les erreurs n'est pas toujours facile

Tu viens de voir le cas le plus simple, la commande mal écrite. J'en profite pour te rappeler que LaTeX est sensible à la casse et donc par exemple `\LaTeX` et `\LateX` sont deux commandes distinctes, c'est le genre d'erreur pas toujours facile à trouver.

Tu vas voir un cas plus compliqué à corriger, l'oublie d'une parenthèse fermante. Supposons que tu veilles mettre les mots « vallée du Pételasoiffe » en italique. Pour cela tu utilises la commande suivante

```
\textit{vallée du Pételasoiffe}
```

Malheureusement tu as oublié la dernière parenthèse. La compilation renvoie une erreur voir figure 4.6 p. 44. On retrouve le nom du fichier où c'est produit l'erreur (no 1), le type d'erreur Runaway argument ? (argument manquant) (no 2) un extrait de la phrase concernée (no 3)

Par contre le numéro de ligne n'indique pas la ligne concernée mais la ligne qui appelle le fichier fautif dans le document maître. En effet LaTeX ne s'est rendu compte de l'erreur qu'à la fin de *introduction.tex* car il ne sait pas quelle partie du texte tu veux mettre en italique.

Si tu connais bien ton texte tu peux retrouver et corriger facilement l'erreur, sinon tu peux tenter une recherche Ctrl F sur le terme « {vallée du Pételasoiffe ».

Avec l'habitude, d'une part tu feras plus attention à la bonne écriture des diverses commandes, d'autre part tu retrouveras plus facilement les erreurs, les indications données par le log te devenant familière. L'utilisation de la complétion automatique, en écrivant correctement les commandes, diminue grandement les sources d'erreur.

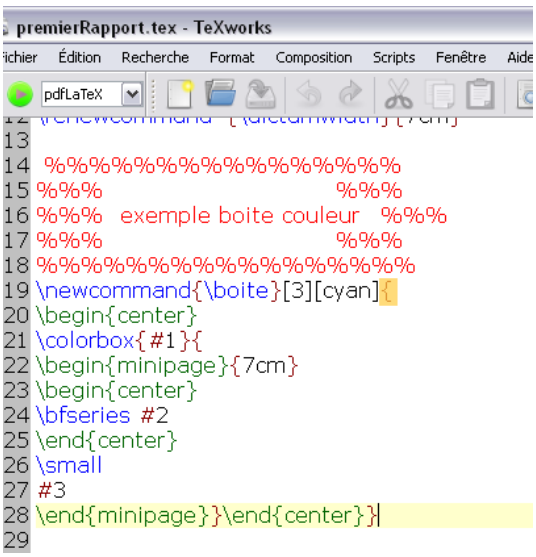


Figure 4.5 : Dans TeXworks quand ton curseur passe sur une parenthèse (ici la dernière), un rectangle orange surligne la parenthèse correspondante.

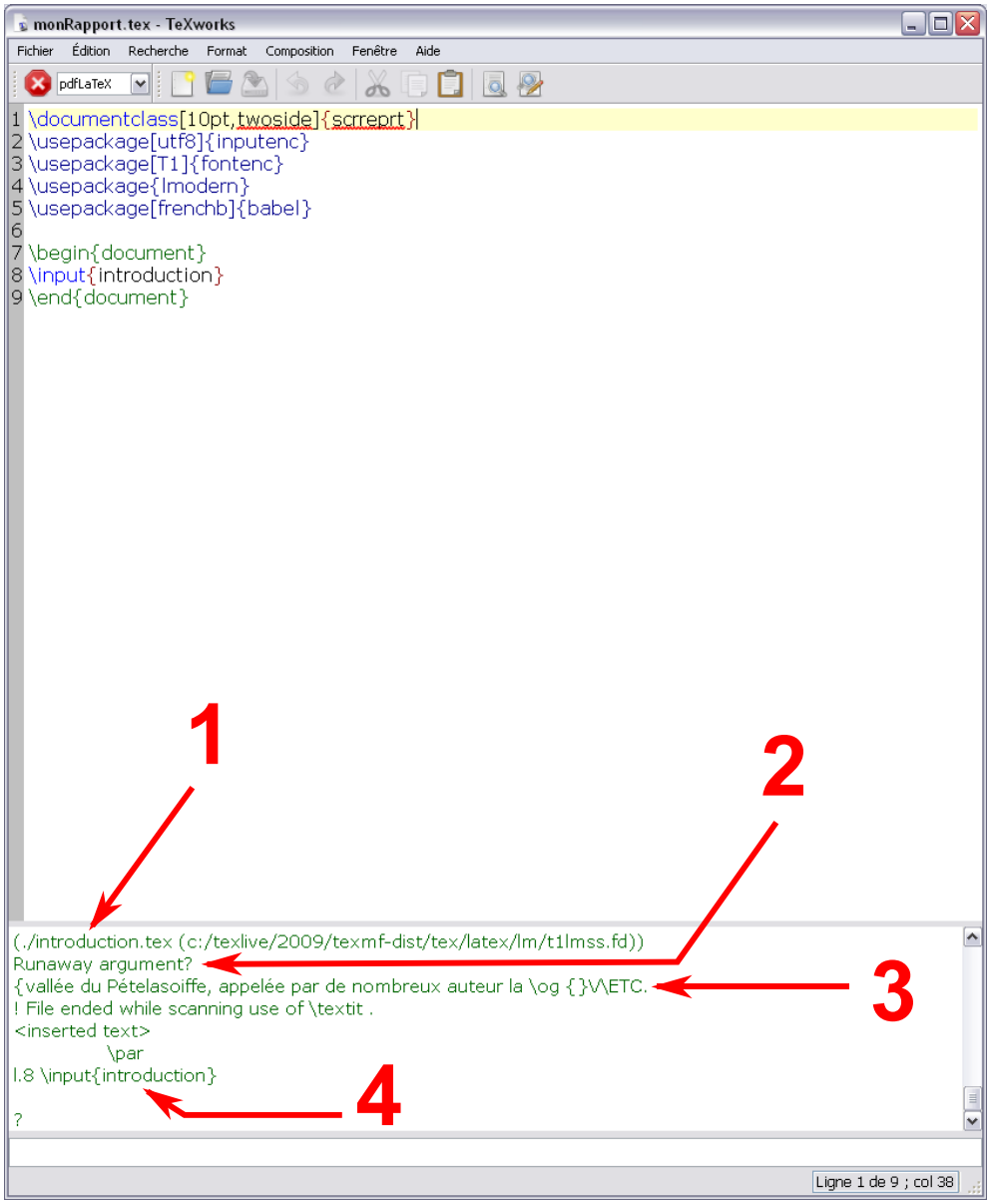


Figure 4.6 : Fenêtre de TEXworks après une erreur de compilation

Pour les parenthèses TEXworks t'est d'une grande utilité car quand ton curseur passe sur une parenthèse, il surligne la parenthèse correspondante (fig 4.5, page 43). Ceci est très pratique pour contrôler les imbrication de parenthèses.

LATEX , n'est pas plus barbare que les autres logiciels, il demande comme tout objet informatique un minimum d'apprentissage, mais tu le vaux bien.

Si tout va bien tu dois obtenir le fichier `.pdf` fig. 4.7, p.46. Ici je n'ai pas mis « vallée du Pételasoiffe » en italique.

4.5 Où Mlle Michu s'étonne : c'est quoi tous ces fichiers ?

Si tu regardes dans le répertoire contenant tes fichiers, tu constateras que LATEX a créé toutes une séries de fichiers, dits auxiliaires, tous nommés `monRapport`, mais avec des extensions différentes. Le fichier `.log` contient les informations de la compilation (ce que tu as vu défiler dans la fenêtre de log). À ce propos dans les *Préférences* de TEXworks, onglet *Composition*, tout en bas il y a une case à cocher pour masquer (ou non) automatiquement la fenêtre de log, si LATEX ne rencontre pas d'erreurs.

Le fichier `.aux` est utilisé, entre autre, par LATEX pour les références croisées, on en reparlera. Le fichier `.toc` sert pour fabriquer la table des matières.

Si tu trouves que tous ces fichiers encombrent ton répertoire tu peux tous les effacer, ils sont automatiquement générés à chaque compilation et n'ont d'utilités que lors de celle-ci. Si tu sauvegardes ton travail sur d'autre support (*ce qui est obligatoire*) il est inutile de les prendre en compte.

N'efface pas tes fichiers `.tex`, ni les fichier `.bib` utilisés pour la bibliographie, que l'on fabriquera plus tard, tu perdrais tout ton travail. Tu n'effaces pas non plus le fichier `.pdf` qui est quand même ce que tu veux obtenir, mais c'est moins grave car tu pourras toujours le refabriquer à partir des sources.

Pour éviter toutes erreurs sous TEXworks dans le menu *Fichier* tu trouveras une commande *Supprimer les Fichiers Auxiliaires...*

4.6 Où Mlle Michu trouve fastidieux de changer de fichier pour compiler

Et Mlle Michu a raison. Je te bassine depuis le début, avec le fait qu'utiliser la souris c'est nul, et là je te fait triturer le mulot à tout va. Il existe bien entendu un raccourci clavier pour la flèche verte, c'est Ctrl T . Et encore plus fort tu peux indiquer dans ton fichier *introduction.tex* le fichier maitre. Tu ajoutes au tout début de ton fichier (sur la première ligne) l'instruction suivante

```
%!TeX root = monRapport.tex
```

Je te rappelle qu'en LATEX, un % débute des commentaires qui ne sont pas interprétés par le compilateur, mais qui dans ce cas sont compris de TEXworks, qui va compiler sans problème les sources. Donc tout en restant dans ton introduction, un simple Ctrl T fait apparaître le fichier pdf. Voilà qui est beaucoup mieux.

Introduction

La découverte d'un nouveau gisement moustérien dans cette région du Balafon Occidental est toujours un événement. Surtout quand cette découverte à lieu dans la vallée du Pételasoiffe, appelée par de nombreux auteur la « Vallée de l'Omo » de l'Europe du Nord [1]. Le site d'Harfouëte – La Mare au Diable, découvert en 2009, outre son insertion dans un contexte micro-régional riche, faisant déjà l'objet de très nombreuses recherches [2], présente un grand intérêt intrinsèque. En effet la présence plusieurs foyers et de reste osseux rendent ce site unique dans cette région.

Le gisement d'Harfouëte – La Mare au Diable est situé à proximité de deux sites mondialement connu. En amont, à 500 m à vol d'oiseau se trouve le site de « La roche Penchée », fouillé depuis les années 1887. À 1,5 km en aval, se situe le gisement moustérien de « La Castellerie Rouge », étudié depuis près de vingt ans par l'équipe américaine du professeur A. L. Robinson.

1. Rendons à César ce qui. . .c'est au professeur Raoul Dumortier que l'on doit la première mention de cette expression.
2. Une synthèse sous la direction de Robert de Latour devrait paraître à la fin de l'année 2010.

Figure 4.7 : La première page de ton rapport après compilation.

5 M^lle Michu veut créer ses propres commandes

Eh oui, elle est comme ça M^lle Michu, elle sait à peine aligner deux commandes LaTeX qu'elle veut déjà créer les siennes. Il faut avouer que M^lle Michu n'a pas tord. Créer des commandes avec LaTeX est très facile et bougrement efficace, alors pourquoi s'en priver ?

Tu as remarqué que déjà dans notre introduction on a employé plusieurs fois le nom du site archéologique, objet de notre rapport. Ce nom est particulièrement long, Harfouëte – La Mare au Diable, trouver une façon efficace de l'écrire ferait gagner du temps et éviterais les erreurs. Es-tu sûr au bout de la centième page d'avoir toujours mis un a minuscule au a du « au » et le tiret séparateur c'est toujours un – et pas un - ou un — ?

Imaginons que tu viennes de terminer ton rapport, et que tu le fasses relire par un de tes collègues, spécialiste de la préhistoire de la vallée du Pételasoiffe. Ce dernier te fait remarquer que le site que tu as fouillé à déjà été mentionné par plusieurs auteurs mais sous le nom de « La Mare des Diables » et que dans un souci de cohérence il serait préférable que tu renommes ton site « La Mare des Diables ». Si tu as créé une commande, pas de problème, tu fais le changement dans celle-ci tu recompiles et l'intégralité de ton rapport est modifié. Dans le cas contraire, c'est la galère il te faut retrouver tous tes « La Mare au Diable » dans tous tes fichiers ! ! !

5.1 La première commande de M^lle Michu

Pour construire une commande tu utilises la commande `\newcommand*` de la façon suivante : `\newcommand*{\nomDeTaCommande}{CodeLaTeX}`

Je te rappelle qu'une commande LaTeX commence toujours par un `\` et ne doit contenir que des lettres, en aucun cas des chiffres ou des signes de ponctuation. On va appeler notre commande qui veut remplacer le nom du site « `\site` » :

```
\newcommand*{\site}{Harfouëte -- La Mare au Diable}
```

Voilà c'est pas plus compliqué que cela. Tu places cette définition dans le préambule de ton source LaTeX entre l'entête dans lequel tu charges les packages et le `\begin{document}`. Tu l'utilises comme toutes les commandes LaTeX en plaçant dans ton source `\site`. Après compilation tous les `\site` seront remplacés par Harfouëte – La Mare au Diable.

Remarque

Pourquoi étoiler les \newcommand ?
Donald E. Knuth le créateur de TEX a créé deux types de commandes (on dit aussi macro) les longues (forme non étoilée) qui peuvent inclure un changement de paragraphe et les courtes (forme étoilée) qui n'en contiennent pas.
La différence se situe lors de la recherche d'erreur notamment dans le cas d'oubli de l'accolade fermente.
Dans le cas d'une macro longue LATEX va lire le fichier jusqu'au bout avant de se rendre compte de l'erreur (c'est le cas que nous avons évoqué dans un chapitre précédent, 4.4 p. 43). Par contre avec une macro courte LATEX se rendra compte de l'erreur à la fin du paragraphe, la détection de l'erreur sera simplifié et LATEX pourra poursuivre la composition du reste du texte normalement.

5.2 Mlle Michu et *xspace*

On a déjà évoqué les problèmes d'espaces à la fin des commandes (voir 2.3.1 p. 15) Pour éviter d'avoir à taper \site{} on va utiliser la commande \xspace du package du même nom, que l'on charge par la commande \usepackage{xspace}. Notre commande devient :

Code

```
\newcommand*{\site}{Harfouëte -- La Mare au Diable\xspace}
```

Maintenant tu n'es plus obligé de te préoccuper des espaces dans ta commande et \site suffit.

L'autre avantage du package *xspace* est que couplé avec l'option *french* de *babel* tu n'es plus obligé d'écrire \og{} ou \fg{}, tu peux écrire \og et \fg, grâce à *xspace*, l'espace après la commande n'est plus « mangée ».

5.3 Mlle Michu veut voir des exemples de commandes

Voici une petites série d'exemples, ils sont extraits de mes documents professionnels donc ne te préoccupe pas du jargon archéologique. Tout d'abord j'emploie énormément d'abréviation qui me facilite la saisie comme :

Code

```
\newcommand*{\sra}{Service Régional de l'Archéologie du Nord -- Pas-de-Calais\xspace}
\newcommand*{\spf}{Société préhistorique française\xspace}
\newcommand*{\bulspf}{Bulletin de la \spf\xspace}
\newcommand*{\gallia}{\textit{Gallia Préhistoire}\xspace}
```

Comme tu peux le voir dans les exemples ci-dessus, tu peux mettre d'autres commandes LATEX à l'intérieur de ta commande et même des commandes que tu viens de créer.

J'utilise également beaucoup de macros pour garantir l'homogénéité de mon document, par exemple il y a plusieurs façons d'écrire « avant Jésus Christ » et « après Jésus Christ », des commandes résolvent le problème.

```
\newcommand*{\apjc}{apr. J.-C.\xspace}
\newcommand*{\avjc}{av. J.-C.\xspace}
\newcommand*{\carbon}{\up{14}C\xspace}
```

Code

Dans la dernière macro, très pratique pour les dates radio-carbone, la commande `\up{}` permet d'écrire en exposant, on obtient ^{14}C.

LaTeX possède beaucoup de commandes internes, il peut arriver que le nom que tu choisisses soit attribué, dans ce cas LaTeX te prévient lors de la compilation par une remarque explicite. Tu peux créer d'autre types de commandes, plus complexe, comme les commandes à arguments, mais nous verrons cela plus tard quand nous en auront besoin.

5.4 M^lle Michu modifie à nouveau ses fichiers

Si tu appliques les conseils que l'on a vu dans ces derniers chapitres ton fichier source `monRapport.tex` doit ressembler au code suivant. J'y ai ajouté des commentaires (précédés par % et en rouge avec la coloration syntaxique) afin de me souvenir du rôle de chaque package. C'est très pratique au début de l'utilisation de LaTeX car on oublie parfois l'utilité d'un package, surtout que cette liste va s'agrandir. Le fichier `monRapport.tex` :

```
\documentclass[10pt,twoside]{scrreprt}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{lmodern}
\usepackage[np]{numprint} % pour écrire les nombres, option np pour
                          \np{} et pas \numprint{}
\usepackage{xspace} % pour la gestion de \og et \fg sans {}
\usepackage[frenchb]{babel}

%%%%%%%%%%%%%%%%%%%%%%%%
%%% Commandes perso %%%
%%%%%%%%%%%%%%%%%%%%%%%%
\newcommand*{\site}{Harfouëte -- La Mare au Diable\xspace}


\begin{document}
\input{introduction}
\end{document}
```

Code

Remarque l'utilisation des commentaires (%) pour structurer ton source (le titre « commandes perso »). Le fichier `introduction.tex` est devenu :

```
% !TeX root = monRapport.tex
\addchap{Introduction}
La découverte d'un nouveau gisement moustérien dans cette région
du Balafon Occidental est toujours un événement. Surtout quand
cette découverte à lieu dans la vallée du Pételasoiffe, appelée
par de nombreux auteur la \og Vallée de l'Omo\fg de l'Europe du
Nord\footnote{Rendons à César ce qui\dots c'est au professeur Raoul
\bsc{Dumortier} que l'on doit la première mention de cette expres-
sion.}. Le site d'\site, découvert en 2009, outre son insertion dans
un contexte micro-régional riche, faisant déjà l'objet de très nom-
breuses recherches\footnote{Une synthèse sous la direction de Robert
\bsc{de Latour} devrait paraître à la fin de l'année 2010.}, pré-
sente un grand intérêt intrinsèque. En effet la présence plusieurs
foyers et de reste osseux rendent ce site unique dans cette région.
\par
Le gisement d'\site est situé à proximité de deux sites mondiale-
ment connu. En amont, à \np[m]{500} à vol d'oiseau se trouve le
site de \og La roche Penchée\fg, fouillé depuis les années 1887.
À \np[km]{1,5} en aval, se situe le gisement moustérien de \og La
Castellerie Rouge\fg, étudié depuis près de vingt ans par l'équipe
américaine du professeur A. L. \bsc{Robinson}.
```

6 M^lle Michu écrit en français

Continuons notre rapport, notre introduction terminée, nous allons commencer notre rapport par un historique des recherches sur le site d'Harfouête – La Mare au Diable. Pour cela nous allons créer un nouveau fichier que nous allons appeler `historique.tex`. Pour créer un nouveau fichier tu tapes Ctrl N ou tu cliques sur l'icône

Voici ses premières lignes :

```
% !TeX root = monRapport.tex
\chapter{Historique des recherches}
\section{Les prospections de Salomon \bsc{Caine}}
Salomon \bsc{Caine} (1865-1932) comme tous les cultivateurs de la
\vallee fut très tôt sensibilisé à la préhistoire, et c'est natu-
rellement qu'il se mit à ramasser des silex taillés qu'il découvrait
lorsqu'il labourait ses terres avec ses b\oe ufs. À partir des an-
nées 1910, sur les conseils de M\up{elle} Marie-Odile \bsc{Dupain du
Tertre}, qui fouillait le gisement de \og La Roche Penchée\fg situé
sur des terres appartenant à Salomon \bsc{Caine}, celui-ci commença
à classer ses découvertes et surtout à noter avec précision sur un
plan cadastral l'endroit des ses trouvailles.
```

Code

La pluspart des commandes présentes ici te sont déjà familière. La seule commande que tu ne connait pas, et pour cause, est `\vallee`, car c'est une commande que j'ai créée. Elle remplace « vallée du Pételasoiffe » car j'ai l'impression que je vais l'employer souvent.

```
\newcommand*{\vallee}{vallée du Pételasoiffe\xspace}
```

Code

L'emploie de commande tel que `\oe` et `\up{elle}` montre qu'il est temps d'en finir avec la francisation de notre texte.

Je vais résumer dans le tableau suivant toutes les commandes permettant de franciser ton texte que l'on a déjà rencontré dans les chapitres précédant :

Commande	Rôle	Exemple	voir
`\og`	guillemet ouvrant	«	section 3.6 page31
`\fg`	guillemet fermant	»	section 3.6 page31
`\bsc{}`	pour écrire les noms propres	Robert Bidochon	section 3.6.5 page34
`\oe`	pour écrire l'e dans l'o	œ	section 2.3.1 page15
`\np{}`	pour écrire les nombres et les unités	18 256,59 km	section 3.6.4 page33

Écrire en français, c'est le rôle du package *french* (voir la section 2.11 p. 23). Pour te faciliter la tache il te propose plusieurs commandes que je vais te décrire.

6.1 La ponctuation

Comme je l'ai indiqué (section 2.11 page 23), en français qui dit signe double dit double espace un devant et un derrière.

Et là miracle, *french* ajoutera automatiquement une espace insécable devant tous les signes doubles même si tu l'a oublié. Tu n'a même pas à te préoccuper de savoir si c'est une espace fine ou normale.

Code

```
Un texte: écrit; n'importe! comment?
```

Résultat

donne

Un texte : écrit ; n'importe! comment ?

Attention

french insère une espace avant, mais pas après :
`Un texte:écrit.`
donne
Un texte :écrit.

6.2 M^me, M^elle

C'est la commande `\up{}` qui permet d'écrire correctement M^me, M^elle, M^mes, M^elles :

Code

```
M\up{me}, M\up{elle}, M\up{mes}, M\up{elles}
```

Remarque

Une petite remarque typographique en passant :
En français l'abréviation de Monsieur c'est M. (un M majuscule avec un point) et pas Mr, M^r.
Mr est réservé pour les sujets de sa gracieuse majesté britannique quelque soit la langue de l'ouvrage, c'est l'abréviation de Mister comme Mr Bond.
De même l'abréviation de Messieurs c'est MM. (deux M majuscules avec un point) et pas Ms, Mrs ou M^rs. Encore une foi Ms ou Mrs sont à réserver pour des sujets britanniques, Miss ou Mistress.

6.3 1^er, n°, 1°

6.3.1 1^er, 2^e,...,n^e

Premier s'abrège avec `1\ier` ce qui donne 1^er et deuxième avec `2\ieme` qui imprime 2^e. Pourquoi ne pas utiliser `1\up{ier}` ou `2\up{ieme}` ? Parce que en typographie française le ier de premier s'écrit ^er et pas ^ier, le ième de deuxième, troisième, dixième... s'écrit ^e et pas ^ième, ^me... En utilisant les raccourcis de *french* tu es sûr de respecter cette règle. Donc voici les commandes à utiliser :

Code

```
1\ier, 1\iers, 1\iere, 1\ieres, 2\ieme, 2\iemes, 10\ieme,
10\iemes...
```

qui impriment :

Résultat

1er, 1ers,1re,1res, 2e, 2es, 10e, 10es ...

6.3.2 1° , 2° ,...

\primo, \secundo, \tertio, \quarto, donnent respectivement 1° , 2° ,3° , 4° . Pour les nombres suivants tu utilises \FrenchEnumerate{} : \FrenchEnumerate{8} pour 8° et \FrenchEnumerate{256} pour 256° .

6.3.3 n° , nos ,N° et Nos

n° , nos ,N° et Nos s'écrivent \no, \nos,\No et \Nos. *french* ajoute automatiquement une espace insécable entre n° et le chiffre, donc tu codes : \no1 pour n° 1 et \nos12, 25 et 38 pour nos 12, 25 et 38. Tu peux avoir l'impression que \no 1 pour n° 1 et \nos 12, 25 et 38, c'est pareil. Eh bien non car dans ce cas l'espace entre le n° et le chiffre n'est pas insécable et tu peux te retrouver dans certain cas avec le n° en fin de ligne et le chiffre au début de la suivante, et ça c'est pas beau.

Remarque

L'utilisation du package *xspace*, évite d'avoir à taper 1\ier{}, \secundo{}... pour ne pas voir l'espace suivante se faire manger.

6.4 Le degré °

Le degré s'écrit avec \degres. Attention ce n'est pas un petit o (la lettre) comme dans n° , mais un petit rond °. Si tu utilises l'encodage *utf8*, comme ça doit être le cas si tu suis ce tutorial, tu peux utiliser le ° du clavier.

Tu pourras voir dans la littérature le symbole degrés écrit \degre mais attention ce n'est pas la bonne taille de °.

Code

```
20\degres
20\degre
```

Résultat

20°
20°

En typographie française les mesures de température s'écrive 20 °C (avec une espace) et pas 20°C (sans espace). Dans ce cas *french* n'ajoute pas d'espace donc tu dois coder comme ceci

Code

```
20\,\degres C
ou bien
20\,°C
en utilisant le degré du clavier
```

6.5 l'e dans l'a

Directement au clavier (avec l'encodage *utf8*) avec œ par Alt Gr O le æ par Alt Gr A le Æ par Shift Alt Gr A le Œ par Shift Alt Gr O . Sinon les commandes suivantes donnent le même résultat :
\oe ⟹ œ
\OE ⟹ Œ
\ae ⟹ æ
\AE ⟹ Æ

6.6 l'euro €

Tu as peut être besoin décrire des sommes en euro. Pour cela il te faudra charger le package *eurosym* qui te fourni deux commandes \euro ⟹ € et \EUR{10} ⟹ €10

Attention

Avec l'euro le package *xspace* ne te seras d'aucun secours, si ton symbole € précède un mot comme ici, tu dois l'écrire \euro{}.

Comme tu peux le constater l'euro précède le chiffre. Il n'y a que pour la langue allemande que l'euro est à droite. Si tu veux malgré tout un euro à droite tu charges le package de cette façon :

Code

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

6.7 Les accents

On a déjà vu le problèmes des accents qui peuvent être tapés directement au clavier. Mais tu peux avoir besoin ponctuellement pour écrire un mot ou un nom étranger d'une lettre accentuée qui n'est pas présente sur ton clavier, voici les solutions. La commande \' (la touche qui contient également le 4 de ton clavier) ajoute un accent aigu et ce sur n'importe quel lettre \'f ⟹ f́, \'R ⟹ Ŕ. L'accent grave s'obtient avec \` (la touche Alt Gr è ) \`g ⟹ g̀, \`A ⟹ À. Pour les autres accents c'est le même principe, voici un tableau qui les résume :

\^x ⟹ x̂	\"x ⟹ ẍ	\H x ⟹ x̋	\~x ⟹ x̃
\c x ⟹ x̧	\k x ⟹ x̨	\=x ⟹ x̄	\b x ⟹ x̲
\.x ⟹ ẋ	\d x ⟹ ẋ	\r x ⟹ x̊	\u x ⟹ x̆
\v x ⟹ x̌	⟹	⟹	⟹

7 M^lle Michu se lance dans l'insertion d'illustrations

Comme M^lle Michu dans son texte parle des cadastres de Salomon Caine et qu'elle en possède des reproductions, elle se dit que se serait une bonne chose d'en faire figurer un exemplaire.

Nous sommes parti pour une explication qui risque d'être longue, car s'il est facile d'insérer une image dans un texte, le faire de façon efficace est plus compliqué et va faire appel à des notions comme les flottants, qui n'existent pas dans les traitements de texte. Les forums de discutions consacré à LaTeX sont remplis de questions de LaTeXiens novices n'arrivant pas à placer correctement leurs images. Donc nous allons prendre notre temps et j'espère que tu n'iras pas grossir la cohorte des débutants qui se sont emmêlé les pinceaux dans les flottants.

7.1 M^lle Michu insère un petit pingouin

Nous allons reprendre notre petit pingouin ( ) que tu as rencontré dans les premières page de ce document.

L'insertion d'images c'est le boulot du package *graphicx*, il te faut donc ajouter dans l'entête de ton document maître la ligne suivante :

Code

```
\usepackage{graphicx}
```

Le compilateur qui fabrique les `.pdf` (*pdflatex*) n'accepte que trois formats d'images `.png`, `.jpg` et `.pdf`. Donc la première chose à faire est de vérifier le format de tes images et le cas échéant à les modifier. Notre petit pingouin s'appelle Tux et son image `tux.png`. La commande pour le placer dans le texte est donc :

Code

```
\includegraphics{tux.png}
```

Attention

Comme aucun répertoire n'est indiqué LaTeX va chercher l'image uniquement dans le répertoire courant, c'est à dire celui qui contient le fichier `historique.tex`

7.2 M^lle Michu pour s'y retrouver organise son travail

Pour faciliter la gestion de tes documents je te conseille de placer tes images dans un répertoire « images » dans le répertoire où se trouve tes fichiers LaTeX. Dans ce cas, la commande pour placer notre petit pingouin est donc :

Code

```
\includegraphics{./images/tux.png}
```

Tu peux définir le chemin du répertoire ou LaTeX doit aller chercher les images par : `\graphicspath{{./images/}}`. L'expression `./images/`, signifie dans le répertoire « images » du répertoire courant. Tu places cette commande dans le préambule de ton source (avant `\begin{document}`).

Si tu as trié tes illustrations dans différents répertoires en fonction de leur nature (schémas, photos, dessins), il te faudra utiliser la commande :

Code

```
\graphicspath{{./shema/}{./photos/}{./dessins/}...{./dossierN/}}
```

Attention

Même si tu ne déclares qu'un seul répertoire il doit être entouré de {}, et n'oublie pas la barre de fraction (/) finale.

Tu peux t'abstenir de préciser l'extension si tu as mis dans le préambule de ton document la commande :

Code

```
\DeclareGraphicsExtensions{.png,.jpg,.pdf}
```

Dans l'exemple ci-dessus, LaTeX va d'abord chercher l'existence d'un fichier en `.png`, s'il ne trouve pas, un fichier en `.jpg` et enfin en `.pdf`. Tu peux bien évidemment changer l'ordre.

Dans les exemples qui suivent on va considérer que j'ai déclaré le chemin et les extensions. Voici le nouveau fichier `monRapport.tex`

Code

```
\documentclass[10pt,twoside]{scrreprt}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{lmodern}
\usepackage[np]{numprint} % pour écrire les nombres, option np pour
                          \np{} et pas \numprint{}
\usepackage{xspace} % pour la gestion de \og et \fg sans {}
\usepackage[frenchb]{babel}

%%%%%%%%%%%%%%%%%%%%%%%
%%% Commandes perso %%%
%%%%%%%%%%%%%%%%%%%%%%%
\newcommand*{\site}{Harfouëte -- La Mare au Diable\xspace}

%%%%%%%%%%%%%%%%%%%%%%%
%%%      images       %%%
%%%%%%%%%%%%%%%%%%%%%%%
\graphicspath{./images/}
\DeclareGraphicsExtensions{.png,.jpg,.pdf}

\begin{document}
\input{introduction}
\input{historique}
\end{document}
```

On va placer notre première image dans un document LaTeX. Voici le code (pour la complétion automatique le raccourci est inc + Tab) :

Code

```
Voici comment insérer une image, \incudegraphics{tux}. C'est un
petit pingouin.
```

Résultat

Voici comment insérer une image,

. C'est un petit pingouin

Enfin pas si petit que cela. Sans précision `\includegraphics` affiche le dessin dans sa taille réelle soit pour notre pingouin 111 x 111 pixels. Pour y remédier `\includegraphics` accepte des options du type option=valeur, séparées par des virgules.

Code

```
\includegraphics[option1=valeur1,option2=valeur2]{image}
```

Tu peux commencer par modifier l'échelle de ton image par l'option `scale`. Des valeurs supérieures à 1 augmentent la taille de l'image (`scale=2` double la taille), des valeurs comprises entre 0 et 1 diminuent la taille (`scale=0.5` divise par 2 les dimensions). Pour la complétion automatique le raccourci est inc + Tab pour obtenir `\incudegraphics{|}` et tu appuies une deuxième fois sur Tab pour avoir `\includegraphics[|]{•}`

Code

```
Voici comment insérer une image, \includegraphics[scale=0.2]{tux}.
C'est un petit pingouin.
```

Résultat

Voici comment insérer une image, . C'est un petit pingouin

Voila qui est beaucoup mieux. Tu peux également donner directement les dimensions souhaitées pour l'image. L'option `width` règle la largeur de l'image et `height` sa hauteur. Si tu ne donnes qu'une seule dimension les proportions de l'image seront conservées. Voici différents exemples :

Code

```
Voici comment insérer une image, \includegraphics[width=1cm]{tux}.
C'est un petit pingouin.
```

Résultat

Voici comment insérer une image, . C'est un petit pingouin

Code

```
Voici comment insérer une image,
\includegraphics[width=4cm,height=1cm]{tux}. C'est un petit
pingouin.
```

Résultat

Voici comment insérer une image, . C'est un petit pingouin

Remarque

LaTeX gère une grande quantité de longueurs, aussi bien absolues, comme me millimètre, le point,... que relatives comme la longueur du texte (`\textwidth`), le cadratin (em), qui représente la largeur de la lettre M dans la police courante.

7.3 LaTeX et les longueurs

On manipule pas mal de longueurs avec LaTeX. Contrairement au traitement de texte, dans lesquels tu utilises la souris et des poignées de réglage, sous LaTeX tu es obligé, comme on vient de le voir pour l'insertion d'image, de préciser un certain nombre de dimensions. Ces mesures de distance se répartissent en deux groupes :

Les unités absolues : ces unités ont toujours la même valeur quelque soit la forme du document et sont indépendantes de LaTeX. Par exemple le point, le centimètre, le millimètre ;

Les unités relative : ces unités dépendent des caractéristiques de ton documents. Par exemple le cadratin, qui représente la largeur de la lettre M dans la police courante, dépend bien évidement la police utilisée.
`\textwidth` qui représente la longueur du texte dépend de la mise en page de ton document.

Je ne vais pas te décrire la totalité des unités absolues prise ne charge par LaTeX, car elles sont nombreuses et je doute, que par exemple, tu ais besoin d'utiliser le cicéro qui vaut 12 didots, lui même valant 1,07 pt. Mais si tu as besoin d'unités exotiques ou typographiques, une recherche sur internet, devrait répondre à tes questions.

L'unité absolue de base de LaTeX est une unité typographique le point noté pt et qui vaut 0,35 mm. Moi je préfère utiliser le centimètre (cm) et le millimètre (mm), beaucoup plus parlant pour nous enfants de la révolution de 1789 (2,54 cm = 72,27 pt).

Les unités relatives sont tout aussi nombreuses en voici quelques unes que tu pourrais trouver utiles.

7.3.1 Cadratin

Le *cadratin* (em), qui représente la largeur de la lettre M dans la police courante, mais qui, dans les faits, équivaut à la largeur des corps. Par exemple dans un texte tapé en corps 11, un cadratin mesurera 11 points.

Le tiret long ou tiret cadratin — que l'on écrit en LaTeX « --- » et que tu utilises si tu veux retranscrire un dialogue, vaut 1em.

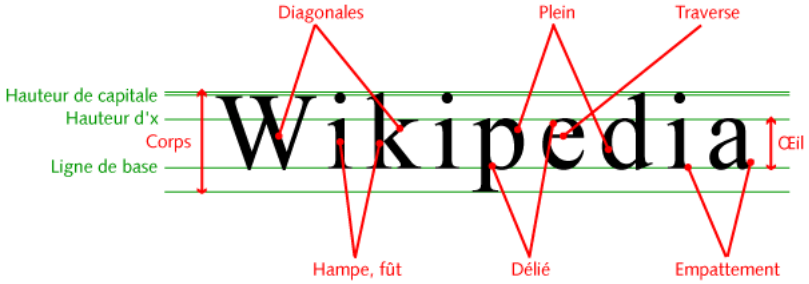
Le tiret court (demi-cadratin) – que l'on écrit en LaTeX « -- » et qui est utilisé dans les listes, vaut 0.5 em.

Le quart de cadratin (0.25 em) - que l'on écrit en LaTeX « - » qui est utilisé pour les césures ou les mots ou noms composés.

L'espace fine qui ce met devant les signes doubles comme : ; ! ou ? s'écrit « `\,` » en LaTeX et vaut un quart de quadratin.

7.3.2 l'œil

Cette unité, notée « ex » correspond à la hauteur du x bas-de-casse. Voici un petit schéma emprunté à Wikipédia pour mieux comprendre.



Je l'utilise essentiellement pour définir des hauteurs, comme par exemple l'espace entre deux lignes `\vspace{1.5ex}`, c'est à dire, égal à 150% de la hauteur du x bas-de-casse, ou l'épaisseur d'un trait `\rule{5cm}{0.5ex}`.

7.3.3 Système propre à LaTeX

Ces unités sont essentiellement utilisées dans la mise en page des documents. Préférant utiliser les formats prédéfinit par KomaScript, ou à la rigueur le package geometry , je les utilise peu. Toutefois certaines d'entre-elles sont incontournables.

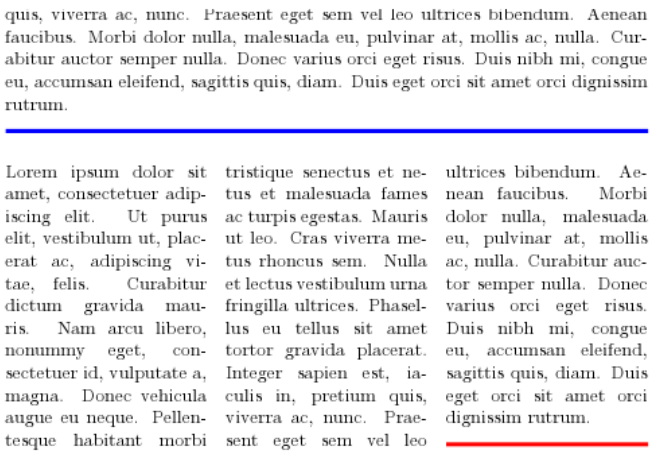
`\textwidth`

Représente la longueur d'une ligne texte dans la page. Ci-dessous je vais tracer une ligne de la largeur d'une ligne de texte par la commande `\rule{\textwidth}{0.5ex}`et une suivante moitié plus petite `\rule{\textwidth}{0.5ex}` (`{0.5ex}`indique l'épaisseur du trait).

`\linewidth`

Représente la longueur de la ligne de texte dans l'environnement courant. La plupart du temps `\linewidth` = `\textwidth`, mais quand on est dans un environnement minipage (voir section 12.1, page 111) ou dans un environnement

multi-colonnes les 2 valeurs sont différentes. Dans le dessin ci-dessous la ligne bleue représente `\textwidth` et la rouge `\linewidth`



`\textheight`

Représente la hauteur du texte dans la page. J'utilise cette unité pour calculer la taille des images, pleine page, plus haute que large, pour qu'elle ne déborde pas de la page de la façon suivante :

```
\includegraphics[height=0.8\textheight]{monImage}
```

7.3.4 Unités absolues ou unités relatives, *that the question* ?

Je ne sais pas si il y a une réponse définitive, tout ce que je peux te répondre, c'est ce que je fais.

En gros pour tout ce qui concerne le texte, comme la taille d'un saut de ligne ou d'une indentation, la taille d'un tableau et de ses colonnes, j'utilise les distances relatives, car si, par exemple, je change le type de police ou la dimension des marges, les distances ainsi définies vont évoluer en fonction de ces nouvelles données et les proportions recherchées vont être conservées.

Par contre pour tout ce qui concerne la page, comme placement d'objet, la taille des marges, ..., j'utilise les unités absolues. Le format de page A4 étant définit en centimètre (21x29,7cm) je me repère mieux dans la page en utilisant les centimètres. De même pour la taille des photos, un format de 9x13 cm est plus facile à se représenter que 0.8`\textheight`.

7.4 Mlle Michu en veut un petit peu plus

Maintenant tu sais placer une image dans un document LaTeX. Mais ce n'est pas exactement ce que tu veux. Tu aimerais avoir une image centrée, avec une légende en dessous, avec si possible un numéro de figure comme l'illustration suivante :

Figure 7.1 : Jeux de mains, jeux de vilains.

Pour parvenir à ce résultat il faut employer les *flottants*.

7.5 M^lle Michu flotte

Sous LaTeX en utilisant `\includegraphics` que l'on a vu dans la section précédente, quand tu places une image dans ton source, tu ne sais pas à quel endroit de la page (ni sur quelle page) tu te trouves, puisque la compilation n'a pas eu lieu. Tu ne sais même pas s'il reste assez de place sur la page pour la mettre à cet endroit. Prenons un exemple, tu veux insérer une image après une portion de texte. Seulement, pas de chance, après compilation la fin de ce morceau de texte se trouve à 6 cm du bas de la page. Avec la meilleure volonté du monde LaTeX ne pourra y faire tenir ton image de 9x13 cm. LaTeX va donc la positionner en haut de la page suivante et laisser au bas de la page précédente un blanc de 6 cm. Enfin pas vraiment car LaTeX étant plus subtil que les traitements de texte classiques, va répartir ce blanc entre les différents espaces inter-paragraphe, s'ils existent, de la page. Pas terrible !

Pour régler ces problèmes du placement des figures, LaTeX offre une autre voix.

Au lieu de dire à LaTeX je veux une image ici, tu vas lui dire, j'aimerais, s'il te plaît, que tu places mon image ici. Comme tu as été très poli avec LaTeX, il va faire un effort. Il ne peut te garantir qu'il pourra la mettre exactement ici mais il va essayer de la placer au plus près. Ton image va donc flotter dans le texte en fonction de la place disponible et de critères esthétiques, connus de LaTeX. Elle pourra être placée avant ou après son point d'insertion. Voila pourquoi en langage LaTeX on les appelle des flottants.

7.6 Où M^lle Michu apprend qu'il existe deux types de flottants

Il existe deux types de flottants, ceux pour insérer des images, appelé `\figure` et ceux pour insérer des tableaux, `\table`. Les flottants sont des environnements, ils commencent donc par `\begin{}` et finissent par `\end{}`. Ces environnements servent à créer des objets flottants dans lesquels tu peux mettre

ce que tu veux, du texte, des images, des tableaux, ou n'importe quel objet LaTeX. La différence entre les deux environnements ce situe au niveau de la légende et de la table associée. Dans le cas de *figure* tu auras une numérotation en « Figure 1, Figure 2,..., Figure n » et une *Tables des figures* et dans l'environnement *table*, une numérotation en « Table 1, Table 2,..., Table n » et une *Liste des tableaux*. La légende est produite par la commande `\caption` et le référencement par `\label`. Voici deux exemples que l'on va détailler par la suite :

Code

```
\begin{figure}[!htbp]
\includegraphics[width=12cm]{monImage}
\caption{La légende de mon image.}
\label{labelImage}
\end{figure}
```

Code

```
\begin{table}[!htbp]
\begin{tabular}
... données du tableau ...
\end{tabular}
\caption{La légende du tableau.}
\label{labelTableau}
\end{table}
```

7.7 Où Mlle Michu se rend compte qu'image et flottant c'est pas pareil

Il ne faut pas confondre flottants et image (ou tableau). Le flottant correspond à tout ce qui est entre les bornes `\begin{image}` et `\end{image}`(ou `\begin{table}` et `\end{table}`), donc l'image (ou le tableau) *et* la légende. Ce qui est à l'intérieur du flottant est indivisible et forme un bloc. Souviens-t-en quand tu écriras des légendes longues.

La figure 7.2 p. 63 montre bien la différence.

7.8 Où Mlle Michu doit respecter quelques règles

Si tu ne veut pas galérer avec les flottants il est nécessaire de bien en comprendre le fonctionnement et de respecter quelques règles.

L'emploi de flottants a plusieurs conséquences. Premièrement tu ne peux pas employer des formules du type « voir sur l'image ci-dessous ». En effet ton image peut très bien être en haut de la page où sur la page suivante. Tu dois employer des trucs du style, « voir l'image n° *x* page *y* ». On verra comment faire cela, c'est ce que l'on appelle des *références croisées*.

Pour conserver une certaine esthétique typographique LaTeX à des règles de positionnement des illustrations par rapport au texte. La première est que le nombre total de flottants permis sur une page est de 3 par défaut, avec un maximum de 2 en haut de la page et 1 seul en bas de page.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Lorem ipsum dolor sit amet,

FIGURE 0.1: Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit,

FIGURE 0.2: Nulla malesuada porttitor diam.

vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Lorem ipsum dolor sit amet, consectetuer adipiscing elit.

1

FIGURE 7.2 : Différence entre flottant et image. J'ai mis les légendes en rouge pour bien visualiser l'espace occupé par les flottants, marqué par le cadre bleu.

Une autre règle observée par LaTeX est qu'une page mélangeant des images (ou des tableaux) et du texte, contient au minimum 20% de texte. Donc si 1 ou 2 flottants occupent 80% de la surface le ou les suivants seront déplacés sur la page suivante.

LaTeX préfère placer les flottants en haut de page, même avant leur appel.

7.9 Mlle Michu espère faire plier LaTeX

Comme je te l'ai déjà expliqué l'endroit ou tu positionnes le flottant dans le source n'est qu'une indication de placement, LaTeX de toute façon n'en fera qu'à sa tête. Tu peux quand même préciser ton souhait par des options que tu places entre [ ] :

! : les règles de placement vues précédemment, comme le nombre de flottants par page, sont ignorées ;

h : là ou il apparait dans le source (h pour here) ;

t : en haut de la page (t pour top) ;

b : en bas de la page (b pour bottom) ;

p : sur une page à part, sans texte (p pour page). Cette page est appelée page flottante.

Tu peux mettre plusieurs options (sans les séparer par une virgule), leur ordre n'a aucune importance, LaTeX utilise toujours le même ordre (!, h, t, b, p) jusqu'à rencontrer un placement indiqué et réalisable. `\begin{figure}[tbh]`, `\begin{figure}[bth]`, `\begin{figure}[hbt]`, c'est pareil et c'est traité par LaTeX en *h*, *t*, *b*.

Notes que les options doivent être mise après `\begin{figure}`, `\begin[htb]{figure}` produit une erreur de compilation, tu dois écrire `\begin{figure}[htb]`.

7.10 Mlle Michu sachez que LaTeX à toujours raison

Il y a une chose fondamentale à connaitre au sujet des options de placement. LaTeX n'étudiera que les options mentionnées. Imaginons que tu aies une illustration qui occupe 40% d'une page et que tu fournisses comme option **[hb]** LaTeX étudiera *h* puis *b*, les autres (*t* et *p*) seront ignorées.

Je peux t'annoncer tout de suite que *b* sera ignorée car ton illustration dépasse les 30% autorisée pour une image de bas de page. Quand à *h* si après compilation il se trouve en haut de page elle peut passer (si d'autres images n'occupent pas la place), mais s'il est en bas de page, elle sera également refusée. LaTeX ne pourra la placer, et l'illustration ira dans la file d'attente des figures non placées (*unprocessed float*).

Pour améliorer la situation tu peux utiliser **[!hb]**. Le point d'exclamation (**!**) empêchant LaTeX d'utiliser les règles d'esthétique, ta figure pourra être placée en bas de la page.

LaTeX en plus des règles esthétiques vues précédemment, suit également des règles pour le placement des flottants :

– une figure ne peut être placée que dans les endroits spécifiés dans la liste des options ;

– les flottants ne peuvent pas être placés sur les pages antérieurs au placement. Ils peuvent quand même se trouver avant le point d'insertion mais uniquement sur la même page. Par contre ils peuvent se retrouver sur n'importe quelle page postérieure ;
– les flottants sont placés dans l'ordre. Aucune figure même petite, ne peut être placée avant une autre. Une figure non placée va empêcher le placement de toutes les suivantes. Par contre une table (si elle respecte l'ordre des tables) peut passer avant une figure.

Pour toutes ces raisons il faut à tout prix bannir les options uniques (`[h]`, `[b]`, `[t]` et `[p]`). Si tu ne précises pas d'option LaTeX, utilise `[tbp]`. Plus tu donnes d'options, mieux LaTeX traite le placement des flottants. Les options suivantes marchent bien `[htbp]`, `[tbp]`, `[htp]` et `[tp]`.

Le nombre de flottants stockés dans liste des flottants non placés ne peut dépasser 18. Au-delà tu obtiens une erreurs de compilation « *Too many unprocessed floats* ».

Les flottants de la liste sont placés automatiquement à la fin d'un chapitre ou à la fin du document pour les documents comme les articles qui n'ont pas de chapitre.

La commande `\clearpage` place tous les flottants en attente et commence une nouvelle page.

L'option *p* permet de placer un flottant sur une page sans texte. Cette page est une page flottante. Elle obéit également à des règles. Par défaut 50% de cette page doit être occupé par un flottant. On ne peut donc pas mettre une seule petite illustration sur cette page. Si l'image est trop petite LaTeX complétera cette page flottante avec l'image suivante.

J'espère que le fonctionnement des flottants n'a plus de secret pour toi. L'une des principales erreurs des débutants LaTeXiens, est de mettre des conditions de placement trop strictes. Tu fais du LaTeX donc oublies tes réflexes WYSIWYG. Laisse à LaTeX le plus de liberté possible pour le placement de tes illustrations, tu verras il ne se débrouille pas si mal. Quand tu lis un rapport, le fait que l'image ne soit pas immédiatement à côté du texte n'est pas très gênant.

Quelques conseils :

– LaTeX est fait pour écrire du texte, il équilibre très bien du texte et quelques images, mais tu vas avoir des problèmes si tu essais d'intégrer un peu de texte dans un flot d'images. Dans ce cas je te conseille de regrouper tes images sur des pages indépendantes.
– De la même manière si tes tableaux sont nombreux ou très long, regarde si tu ne peux pas les placer en annexe, ton texte risquant d'être illisible, car éparpillé entre les flottants.
– S'occuper du placement des images au fur et à mesure de l'écriture du texte est une perte de temps. Le placement précis des images se fait en dernier quand le document est près à être imprimé. Dans un premier temps tu les places au fil de ton texte avec des options de placement les plus ouvertes possibles, `[htbp]` est un bon choix. Quand arrive le moment de l'impression tu peux restreindre les options de placement, déplacer la ligne de commande de flottant quelques paragraphes au-dessus ou en-dessous, jouer avec la taille de l'image, regrouper des images proches sur une page réservée,...
– Si ton image doit être à un endroit précis de ton texte, n'utilise pas de flot-

tant. Tu peux ajouter une légende à un élément hors flottant avec la commande `\captionof` du package `caption`.

Remarque

Avec la TEXlive, l'accès à la documentation est très facile. Dans une console, tu tapes `texdoc leNomDuPackage` pour afficher la doc au format pdf dans le lecteur pdf par défaut de ta machine. Par exemple pour accéder à la documentation du package `caption` tu saisi : `textdoc caption`.
Pour les windowsien peu habitué à cet instrument, tu trouves la console, dans le menu « démarrer », « Tous les programmes », « Accessoires » sous le nom de « Invite de commandes ».

7.11 Mlle Michu place sa première illustration

Revenons à notre rapport et à l'insertion d'une représentation d'un des cadastres de Salomon Caine. Notre image est grande, 20 x 29 cm, donc le meilleur placement est une pleine page et même dans se cas il nous faut la réduire. Réduire la largeur de notre image à la largeur du texte devrait donner un bon résultat. La largeur du texte d'un document LaTeX est donné par `\textwidth` (voir la section 7.3.3 p. 59). Le placement de notre image se fait par le code :

Code

```
\begin{figure}[p]
\includegraphics[width=\textwidth]{cadastre}
\end{figure}
```

La complétion s'obtient avec bfi + Tab + Tab .
Stop, pas la peine de me le faire remarquer, je sais, j'ai dit qu'il fallait bannir les options uniques comme `[p]`, mais ici l'image est tellement grande qu'elle ne peut tenir que sur une pleine page, donc je suis sûr de mon coup.

7.12 Mlle Michu, la légende

Les légendes s'obtiennent à l'aide de `\caption{Texte de la légende}`. C'est tout, LaTeX s'occupe de la numérotation de ta figure. Puisque la classe notre document est *scrreprt* (la classe rapport de KOMA-Script) nos illustrations sont numérotées par chapitre. Pour le chapitre 1 on aura Figure 1.1, Figure 1.2, ... Figure 1.n pour le chapitre 2 Figure 2.1, Figure 2.2, ... Figure 2.n et ainsi de suite. Le code devient :

```
\begin{figure}[p]
\includegraphics[width=\textwidth]{cadastre}
\caption{Extrait d'un cadastre annoté par Salomon Caine.}
\end{figure}
```

La complétion s'obtient avec \ca + Tab .

De la même manière que pour les titres des sections, il existe une possibilité d'indiquer une légende courte, afin de ne pas mettre le bazard dans la liste des figures :

```
\caption[légende courte]{légende longue}
```

7.13 M^lle Michu veut citer son illustration : les références croisées

M^lle Michu devient exigeante. Effectivement comme notre image est sur une page différente, il devient nécessaire d'en faire l'appel dans le texte, quelque chose du genre « voir fig. x, page y ». La première chose à faire est de marquer l'endroit du texte auquel on voudra faire référence en utilisant pour cela la commande \label{motClé}. Le mot clé sert pour établir la référence. Attention il doit être unique. Si tu as utilisé plusieurs fois le même mot clé, tu auras un avertissement lors de la compilation :

LaTeX Warning : There were multiply-defined labels.

Tu peux référencer tout environnement *portant un numéro* (les théorèmes, les chapitres, les sections, les équations, les figures...). Il te suffit de placer le label juste après l'environnement que tu veux référencer.. Par exemple pour référencer une section :

```
\section{Le titre de ma section}\label{motClé}
```

Pour référencer un flottant, il est impératif de mettre le label après la commande \caption puisque c'est elle qui génère le numéro de la figure. Donc si je reprend notre exemple avec le cadastre de Salomon Caine :

```
\begin{figure}[p]
\includegraphics[width=\textwidth]{cadastre}
\caption{Extrait d'un cadastre annoté par Salomon Caine.}
\label{cadastreCaine}
\end{figure}
```

Pour faire les références dans le texte, tu disposes alors des deux commandes \ref{motClé} et \pageref{motClé}. La première inscrira le numéro de chapitre, section, le numéro de la figure, du tableau,... La commande \pageref, comme son nom l'indique, donnera le numéro de la page. Si nous reprenons notre exemple, notre texte devient :

Code

```
% !TeX root = monRapport.tex
\chapter{Historique des recherches}
\section{Les prospections de Salomon \bsc{Caine}}
Salomon \bsc{Caine} comme tous les cultivateurs de la \vallee fut
très tôt sensibilisé à la préhistoire, et c'est naturellement
qu'il se mit à ramasser des silex taillés qu'il découvrait lors-
qu'il labourait ses terres avec ses b\oe ufs. À partir des années
1910, sur les conseils du professeur Pierre-Marie \bsc{Dupain du
Tertre}, qui fouillait le gisement de \og La Roche Penchée\fg si-
tué sur des terres appartenant à Salomon \bsc{Caine}, celui-ci
commença à classer ses découvertes et surtout à noter avec préci-
sion sur un plan cadastral l'endroit des ses trouvailles (voir fig.
\ref{cadastreCaine} p. \pageref{cadastreCaine}).
```

Tu aurais pu également écrire « (voir figure \ref{cadastreCaine} page \pageref{cadastreCaine}) », \ref n'indique qu'un numéro et pas le type de l'environnement.

L'utilisation des références croisées nécessite deux compilations successives. Un fichier avec l'extension .aux est créé lors de la première compilation contenant les numéros associés à la référence, qui sont utilisés lors de la seconde compilation. Tu peux donc faire référence à des \label définis plus loin dans le texte et même dans d'autres fichiers.

Si dans ton texte tu fais appel à des référence non définie (soit par oublie du \label soit par une faute dans l'orthographe de ton mot clé) \ref et \pageref seront remplacé par des « ?? ».

7.14 M^lle Michu termine le premier chapitre de son rapport

Il est temps de terminer ce premier chapitre, il ne nous permet plus d'apprendre de nouvelles notions LaTeXiènes. Voici le code complet :

Code

```
% !TeX root = monRapport.tex
\chapter{Historique des recherches}
\section{Les prospections de Salomon \bsc{Caine}}
Salomon \bsc{Caine} comme tous les cultivateurs de la \vallee fut
très tôt sensibilisé à la préhistoire, et c'est naturellement
qu'il se mit à ramasser des silex taillés qu'il découvrait lors-
qu'il labourait ses terres avec ses b\oe ufs. À partir des années
1910, sur les conseils du professeur Pierre-Marie \bsc{Dupain du
Tertre}, qui fouillait le gisement de \og La Roche Penchée\fg si-
tué sur des terres appartenant à Salomon \bsc{Caine}, celui-ci
commença à classer ses découvertes et surtout à noter avec préci-
sion sur un plan cadastral l'endroit des ses trouvailles (voir fig.
\ref{cadastreCaine} p. \pageref{cadastreCaine}).
```

```
\begin{figure}[p]
\includegraphics[width=\textwidth]{cadastre}
\caption{Extrait d'un cadastre annoté par Salomon Caine.}
\label{cadastreCaine}
\end{figure}
C'est ainsi qu'au lieudit \og La mare au Diable\fg avait noté la dé-
couverte d'une industrie moustérienne non usée, composée de milliers
d'artefacts de silex, de centaines de nucléus Levallois et d'une
cinquantaine d'éclat préférentiels (fig \ref{carteCaine}).
\begin{figure}[ht]
\centering
\includegraphics{gisement}
\caption{Situation du gisement de \og La mare au Diable\fg d'après
la carte publié par Salomon \bsc{Caine} en 1923.}
\label{carteCaine}
\end{figure}
\section{Résultats des sondages préliminaires (2003)}
Dans le cadre d'une prospection thématique sur \og le Moustérien
de la \vallee en 2002, nous avons recherché l'emplacement de ce gi-
sement et des gisements voisins signalés par Salomon \bsc{Caine},
par des sondages mécaniques ponctuels. Le sondage numérotée \og
Sond13\fg nous a permis de retrouver l'emplacement précis du gi-
sement du \site. Il se situe sur le versant exposé à l'Est d'une
petite vallée sèche, le Ravin de Tilleuls, à \np[m]{650} de son
confluent avec la \vallee (en rive méridionale). Un sondage très li-
mité a été ouvert le long du talus (bande de \np[m]{5} de long sur
\np[m]{0.5} de large, soit environ \np[m\up{2}]{2,5}), pour vérifier
la densité des artefacts et la position stratigraphique du niveau
(fig. \ref{sondage13} p. \pageref{sondage13}).

\par
\begin{figure}[ht]
\centering
\includegraphics[width=9cm]{vueSondage}
\caption{Vue du sondage réalisée en 2003.}
\label{sondage13}
\end{figure}

Le niveau archéologique, exempt par ailleurs de tout cailloutis, se
situait -semblait-il- au sommet d'un limon humifère brun foncé (sol
gris forestier du Début Glaciaire weichsélien) épais de quelques dé-
cimètres et reposant directement sur le cailloutis de base composé
de blocs et de graviers de silex. Le pendage du niveau archéologique
est assez sensible vers l'Est, un peu plus prononcé que la pente ac-
tuelle (10\degres). L'épaisseur de l'horizon humifère décroît dans
cette direction ; latéralement, l'industrie semblait donc reposer di-
rectement sur le cailloutis. \par
```

Code suite

Code suite

```
L'industrie en silex recueillie en 2003 (84 artefacts coordonnés sur
plan et 60 artefacts supérieurs à \np[mm]{20} hors-stratigraphie)
comprend des éclats corticaux, des éclats de mise en forme de nu-
cléus, deux nucléus Levallois de méthode linéale, de tailles ex-
trêmes : l'un très petit (46 x 57 x \np[mm]{18}), l'autre beaucoup
plus grand (138 x 131 x \np[mm]{43}), 2 gros nucléus ébauchés (pe-
sant respectivement \np[g]{1820} et \np[g]{2170}). Bien que les
remontages n'aient pas été tentés systématiquement, 3 éclats ont
pu être remontés sur leur bloc d'origine. Aucun outil n'a été re-
cueilli. Le faible échantillon dont nous disposions permettait sim-
plement de noter la domination, sur le plan technologique, de la
méthode Levallois, avec des variantes probables dont la significa-
tion restait à découvrir.
Maintenant je vais mettre du texte factice car le but de ce tutoriel
c'est quand même \LaTeX\ et pas l'étude d'un faux site préhisto-
rique. \lipsum[1-2]
```

Quelques remarques sur ce code dont tu connais pratiquement toutes les commandes. Tout d'abord la façon d'écrire m^2 à l'aide de la commande `\up{texte}` qui met le texte en exposant. Deuxièmement la puissance de l'option *french* de *babel* qui écrit les correctement les décimales même si elle sont notées à l'anglo-saxonne : `\np[m]{0.5}` donne 0,5 m. La commande `\lipsum`[1-2], dont tu n'auras pas besoin écrit un texte en pseudo-latin pour faire du remplissage.

Les cinq pages suivante donne le résultat après compilation.

Introduction

La découverte d'un nouveau gisement moustérien dans cette région du Balafon Occidental est toujours un événement. Surtout quand cette découverte à lieu dans la vallée du Pételasoiffe, appelée par de nombreux auteur la « Vallée de l'Omo » de l'Europe du Nord [1]. Le site d'Harfouëte – La Mare au Diable, découvert en 2009, outre son insertion dans un contexte micro-régional riche, faisant déjà l'objet de très nombreuses recherches [2], présente un grand intérêt intrinsèque. En effet la présence plusieurs foyers et de reste osseux rendent ce site unique dans cette région.

Le gisement d'Harfouête – La Mare au Diable au Diable est situé à proximité de deux sites mondialement connu. En amont, à 500 m à vol d'oiseau se trouve le site de « La roche Penchée », fouillé depuis les années 1887. À 1,5 km en aval, se situe le gisement moustérien de « La Castellerie Rouge », étudié depuis près de vingt ans par l'équipe américaine du professeur A. L. Robinson.

1. Rendons à César ce qui... c'est au professeur Raoul Dumortier que l'on doit la première mention de cette expression.

2. Une synthèse sous la direction de Robert de Latour devrait paraître à la fin de l'année 2010.

1 Historique des recherches

1.1 Les prospections de Salomon Caine

Salomon Caine comme tous les cultivateurs de la vallée du Pételasoiffe fut très tôt sensibilisé à la préhistoire, et c'est naturellement qu'il se mit à ramasser des silex taillés qu'il découvrait lorsqu'il labourait ses terres avec ses bœufs. À partir des années 1910, sur les conseils du professeur Pierre-Marie Dupain du Tertre, qui fouillait le gisement de « La Roche Penchée » situé sur des terres appartenant à Salomon Caine, celui-ci commença à classer ses découvertes et surtout à noter avec précision sur un plan cadastral l'endroit des ses trouvailles (voir fig. 1.1 p. 3). C'est ainsi qu'au lieudit « La mare au Diable » avait noté la découverte d'une industrie moustérienne non usée, composée de milliers d'artefacts de silex, de centaines de nucléus Levallois et d'une cinquantaine d'éclat préférentiels (fig 1.2).

1.2 Résultats des sondages préliminaires (2003)

Dans le cadre d'une prospection thématique sur « le Moustérien de la vallée du Pételasoiffe » en 2002, nous avons recherché l'emplacement de ce gisement et des gisements voisins signalés par Salomon Caine, par des sondages mécaniques ponctuels. Le sondage numérotée « Sond13 » nous a permis de retrouver l'emplacement précis du gisement du Harfouête – La Mare au Diable. Il se situe sur le versant exposé à l'Est d'une petite vallée sèche, le Ravin de Tilleuls, à 650 m de son confluent avec la vallée du Pételasoiffe (en rive méridionale). Un sondage très limité a été ouvert le long du talus (bande de 5 m de long sur 0,5 m de large, soit environ 2,5 m^2), pour vérifier la densité des artefacts et la position stratigraphique du niveau (fig. 1.3 p. 5).

Le niveau archéologique, exempt par ailleurs de tout cailloutis, se situait -semblait-il- au sommet d'un limon humifère brun foncé (sol gris forestier du Début Glaciaire weichsélien) épais de quelques décimètres et reposant directement sur le cailloutis de base composé de blocs et de graviers de silex. Le pendage du niveau archéologique est assez sensible vers l'Est, un peu plus prononcé que la pente actuelle (10°). L'épaisseur de l'horizon humifère décroît dans cette direction ; latéralement, l'industrie semblait donc reposer directement sur le cailloutis.

L'industrie en silex recueillie en 2003 (84 artefacts coordonnés sur plan et 60 artefacts supérieurs à 20 mm hors-stratigraphie) comprend des éclats corticaux, des éclats de mise en forme de nucléus, deux nucléus Levallois de méthode linéale, de tailles extrêmes : l'un très petit (46 x 57 x 18 mm), l'autre beaucoup plus grand (138 x 131 x 43 mm), 2 gros nucléus ébauchés (pesant respectivement 1 820 g et 2 170 g). Bien que les remontages n'aient pas été tentés systématiquement, 3 éclats ont pu être remontés

FIGURE 1.1: Extrait d'un cadastre annoté par Salomon CAINE.

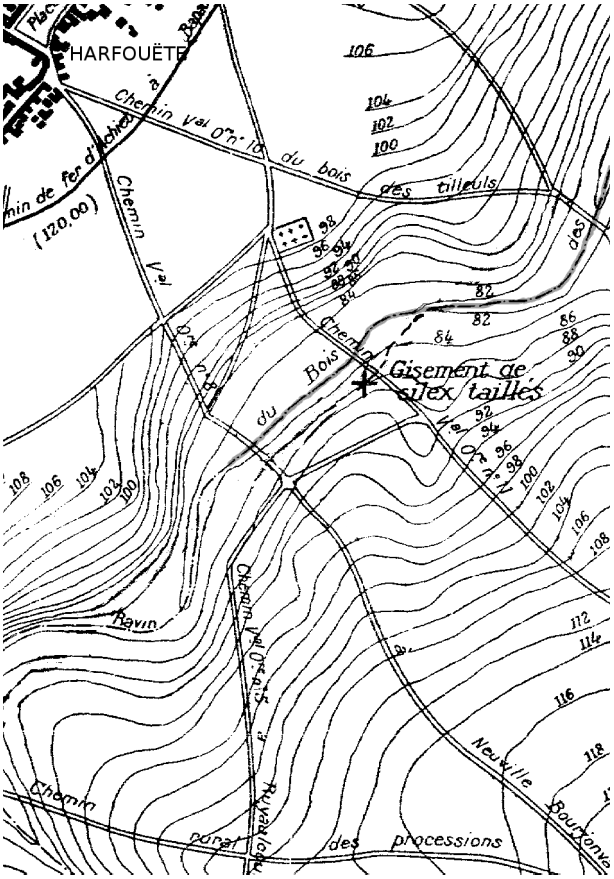


FIGURE 1.2: Situation du gisement de « La mare au Diable » d'après la carte publié par Salomon CAINE en 1923.

sur leur bloc d'origine. Aucun outil n'a été recueilli. Le faible échantillon dont nous disposions permettait simplement de noter la domination, sur le plan technologique, de la méthode Levallois, avec des variantes probables dont la signification restait à découvrir.

Maintenant je vais mettre du texte factice car le but de ce tutoriel c'est quand même LaTeX et pas l'étude d'un faux site préhistorique. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis,

FIGURE 1.3: Vue du sondage réalisé en 2003.

viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

8 M^lle Michu fait des listes

Nous poursuivons notre rapport avec un deuxième chapitre consacré à une partie que j'appellerais « technico-administrative ». C'est dans cette partie que j'indique de quels moyens financiers et humains nous avons disposé pour réaliser la fouille, quels étaient les buts scientifiques et quelles méthodes, mises en œuvre pour y parvenir. Je ne vais pas entrer dans le détail, mais l'une des caractéristique de cette partie est qu'elle consomme beaucoup de listes.

8.1 M^lle Michu a besoin d'une liste

M^lle Michu aimerait bien mettre la liste de toutes les personnes qui ont participé au chantier de fouille. Rien de plus simple, il faut utiliser l'environnement *itemize* :

```
% !TeX root = monRapport.tex
\chapter{La campagne de fouilles 2004}
\section{Méthodologie}
La campagne de fouilles s'est déroulée du 1\up{er} juillet au 29
août 2004. Nous remercions chaleureusement tous ceux qui ont
participé, aux fouilles et aux tâches afférentes :
\begin{itemize}
\item Robert Bidochon ;
\item Raymonde Bidochon ;
\item Raoul Dieulefit (Université de Trifouilli-les-Oies) ;
\item Simone Cageot dite Pépète.
\end{itemize}
```

Code

2. La campagne de fouilles 2004

2.1. Méthodologie

La campagne de fouilles s'est déroulée du 1er juillet au 29 août 2004. Nous remercions chaleureusement tous ceux qui ont participé, aux fouilles et aux tâches afférentes :

- Robert Bidochon ;
- Raymonde Bidochon ;
- Raoul Dieulefit (Université de Trifouilli-les-Oies) ;
- Simone Cageot dite Pépète.

Résultat

8.2 Où Mlle Michu énumère

Pour la suite du texte nous allons faire économie du détail des moyens financiers pour passer directement aux buts de la campagne de fouille 2004. Pour cette partie Mlle Michu a également besoin d'une liste mais voudrait que celle-ci soit numéroté. Tout aussi facile que la précédente liste, pour faire une énumération il faut utiliser l'environnement *enumerate*

Code

```
Cette première campagne de fouille vise trois objectifs :
\begin{enumerate}
\item préciser la stratigraphie ;
\item délimiter l'emprise du gisement par une série de sondages
manuels ;
\item réaliser la fouille d'une surface de \np[m²]{20} autour des
artefacts trouvé lors du sondage de 2003 ;
\end{enumerate}
```

Résultat

Cette première campagne de fouille vise trois objectifs :

1. préciser la stratigraphie ;
2. délimiter l'emprise du gisement par une série de sondages manuels ;
3. réaliser la fouille d'une surface de 20 m² autour des artefacts trouvé lors du sondage de 2003.

8.3 Où Mlle Michu décrit

Avec le chapitre suivant consacré à la description de la stratigraphie, nous allons aborder le dernier type de liste, les descriptions, avec l'environnement *description*. Dans ce type de liste les items s'écrive de cette façon : `\item`[l'entête]la description

Code

```
\chapter{La chronostratigraphie}
Le dégagement de plusieurs coupes le long du talus du chemin creux
et le creusement de plusieurs sondages dans le champ surplombant le
site (voir fig. \ref{planCoupe} p. \pageref{planCoupe}), ont livré
une série quaternaire relativement complète et complexe à
interpréter. Il a fallu d'abord recenser toutes les unités sédimen-
tologigues présentes et les localiser dans les différentes sections,
rechercher ensuite les rapports pouvant exister entre les séries
observées sur le talus et dans le champ, enfin reconsidérer les
```

```
dynamiques de mise en place ainsi que les approches paléoclimatiques
et chronologiques.
\section{Unités sédimentologiques}
\begin{description}
\item[A1 -]Substratum grahitique ;
\item[A2 -]Blocs plus ou moins arénisés in situ avec aréne dans les
anfractuosités (base du sondage 6) ;
\item[B1 -] Head ancien, avec petite pédogénèse lessivée (Bt peu
développé, taches blanches 10 YR 6/4), réduit à une échine dans le
sondage 6. Peut-être représenté dans le sondage 1 par la blocaille
située sous la dune de sable grossier ? ;
\item[C -] Base des limons sableux lités avec Bt en bandes incorpo-
rant encore des lambeaux du head B1. Zone localement cryoturbée.
\end{description}
```

Code suite

3. La chronostratigraphie

Le dégagement de plusieurs coupes le long du talus du chemin creux et le creusement de plusieurs sondages dans le champ surplombant le site (voir fig. 3.1 p. 25), ont livré une série quaternaire relativement complète et complexe à interpréter. Il a fallu d'abord recenser toutes les unités sédimentologigues présentes et les localiser dans les différentes sections, rechercher ensuite les rapports pouvant exister entre les séries observées sur le talus et dans le champ, enfin reconsidérer les dynamiques de mise en place ainsi que les approches paléoclimatiques et chronologiques.

Résultat

3.1. Unités sédimentologiques

A1 - Substratum grahitique ;

A2 - Blocs plus ou moins arénisés in situ avec aréne dans les anfractuosités (base du sondage 6) ;

B1 - Head ancien, avec petite pédogénèse lessivée (Bt peu développé, taches blanches 10 YR 6/4), réduit à une échine dans le sondage 6. Peut-être représenté dans le sondage 1 par la blocaille située sous la dune de sable grossier ? ;

C - Base des limons sableux lités avec Bt en bandes incorporant encore des lambeaux du head B1. Zone localement cryoturbée.

Si tu veux faire ressortir un peu plus l'entête tu peux le mettre en gras de cette façon : `\item[\bfseries A1 -]`.

Comme tu peux le voir fabriquer des listes avec LaTeX est trivial.

9 M^lle Michu pense qu'il est urgent de s'occuper de la bibliographie

Et M^lle Michu n'a pas tord. Réaliser un document en sciences humaines sans bibliographie tient de la gageure. Maintenant que tu commences à mieux comprendre le fonctionnement de LaTeX, il est temps de se pencher sur la réalisation de la bibliographie, car il est quand même assez logique d'insérer les renvois bibliographiques au fur et à mesure de l'avancement du travail. Comme Bébert est un mec sympa, il a rédigé un petit document sur le sujet que tu trouveras ici : « BibLaTeX expliqué à Mademoiselle Michu, étudiante en Sciences Humaines » . Par la suite je vais expliquer les grandes lignes de la réalisation d'une bibliographie, mais si tu veux réellement t'en sortir, je ne pense pas que tu puisses faire l'impasse de la lecture des fiches de Bébert.

Pour fabriquer notre bibliographie nous allons utiliser BibTeX qui est un logiciel et un format de fichier conçu par Oren Patashnik et Leslie Lamport en 1985 pour LaTeX. Il sert à gérer et traiter des bases bibliographiques. Toutefois il n'est pas très facile d'usage, surtout pour les travaux en sciences humaines et je trouve préférable d'utiliser le package BibLaTeX de Philipp Lehman, qui facilite grandement le travail avec BibTeX.

La bibliographie d'un document est une liste structurée d'ouvrages cités dans le corps du texte et placée à la fin d'un article ou un rapport. Pour la réaliser il faut un endroit où sont stockées les données bibliographiques (auteur, titre, année de publication,...) un système pour faire des références dans le texte, et un modèle pour la mise en forme de la bibliographie.

Pour ne pas alourdir ce tutoriel qui est déjà bien fourni, nous allons nous limiter, pour notre bibliographie, à la saisie d'articles et de livres, mais il existe bien d'autre type de documents. De même nous limiterons les champs de saisie au strict minimum.

9.1 M^lle Michu panique, elle ne sait pas gérer des bases de données

Pas de panique, il s'agit juste de créer un simple fichier de texte comme tu en as écris des dizaines. BibLaTeX le parcourra pour extraire les éléments dont il a besoin pour fabriquer ta bibliographie.

Attention

Pas un document « word » (extension `.doc`) ou « open office » (`.opt`). En informatique, un fichier texte ou fichier texte brut ou fichier texte simple ou fichier ASCII, est un fichier dont le contenu représente uniquement une suite de caractères imprimables, d'espaces et de retours à la ligne (wikipédia). Ils ont en général une extension en `.txt`
Pour pouvoir l'utiliser comme fichier bibliographique tu lui donneras `.bib` comme extension.

Attention

Pour que BibLaTeX y retrouve ses petits il faut de la rigueur dans la réalisation de ce texte. Une faute de frappe dans le nom des champs, l'oublie d'une parenthèse ou d'un signe égal, empêchera le bon fonctionnement de BibLaTeX.

Un éditeur du type gedit sous GNOME, KEdit sous KDE ou le bloc notes de windows suffit, tu ouvres un nouveau fichier, tu le remplis avec les éléments que l'on va étudier ci-dessous et tu le sauvegardes en précisant l'extension `.bib`, par exemple `leNomDeMaBiblio.bib` (tu remplaces leNomDeMaBiblio par un nom de ton choix).

Comme tu travailles depuis le début avec TeXworks autant continuer avec lui, il fera très bien l'affaire. Tu ouvres un nouveau fichier Ctrl N et l'on va commencer à remplir notre base bibliographique.

Chaque entrée commence par un @ (Ctrl à) suivit du type d'ouvrage, par exemple @article pour un article ou @book pour un livre. Après on trouve une accolade { qui délimite l'entrée bibliographique.

La première donnée que l'on trouve est une clé qui doit être unique et qui sert à identifier les entrées. C'est elle qui est utilisée par exemple pour citer l'ouvrage dans un texte de la façon suivante `\cite{clé}`. Le format de la clé est libre, mais généralement on utilise le nom de l'auteur et l'année de publication ou bien l'auteur suivi d'un mot qui caractérise le travail :

```
@article{bidochon1999,
@book{bidochon2001b,
@article{bidochonVacances,
```

Remarque

Remarque la virgule à la fin qui sépare les différents éléments d'une entrée. Il ne faut pas l'oublier sous peine d'erreurs de compilation.

Remarque

Le nom de cette clé doit être unique. Même si tu as plusieurs centaines d'entrées chacune doit avoir sa propre clé. Donc le plus simple est de créer les clés sous le format auteur ou auteur principal dans le cas d'ouvrage collectif, suivit de l'année. Si un auteur à écrit plusieurs ouvrages dans la même année tu fais suivre l'année d'une lettre :

```
bidochon2000
bidochon2004a
bidochon2004b
...
```

Ensuite viennent les différents champs constituants l'entrée bibliographique comme le titre, l'auteur, l'éditeur, l'année... Ils sont du type `champ = {valeur},` et sont séparés par une virgule.

les valeurs des champs sont encadrées d'accolades (on peut également utiliser les doubles quotes (`champ = "valeur",`). Il ne faut pas oublier les {} et la virgule sous peine d'erreurs de compilation.

Tu peux omettre les accolades pour les valeurs numériques, comme l'année. `year={1990}` ou `year=1990` sont équivalents.

Une accolade finale } termine l'enregistrement. Contrairement aux commandes LaTeX il n'y a pas de \ précédant les noms de champs. Voici des exemples d'enregistrements. Tu n'es pas obligé de passer à la ligne entre chaque champ, mais dans ce cas le fichier est moins lisible. La virgule à la fin du *dernier* champ n'est pas obligatoire. Les espaces entre les différents éléments d'un champ ne sont pas nécessaires, ils améliorent seulement la lisibilité du code. Les deux lignes ci-dessous sont équivalentes :

```
author = {Bébert Lafiche}
author={Bébert Lafiche}
```

Voici quelques exemples :

```
@article{beber2009,
author = {Bébert Lafiche},
title = {Introduction à BibLaTex},
journaltitle = {Les fiches à Bébert},
number = {5},
year = {2009}
}

@book{bakounineAnarchie,
author = {Bakounine Michel},
title = {Le socialisme libertaire},
publisher = {Denoël},
year = {1973},
location = {Paris}}

@online{bebert,
title = {Les fiches à Bébert},
year = {2009},
url = {http://bertrandmasson.free.fr/},
}

 @collection{Ioakimidis1974,title={Histoires d'extraterrestres},
 editor={Demètre Ioakimidis},year=1974, publisher={Le Livre de Poche},\\number=3763,
 pages=414}
```

Toutes ces entrées sont valides, mais la dernière est nettement moins lisible et rend les corrections difficiles.

9.1.1 Mlle Michu saisi un article

Revenons un peu en arrière. Dans le début de notre rapport nous avons évoqué plusieurs publications. Il serait bien de les citer. Robert De Latour a écrit un article intitulé « Historique des fouilles menées depuis 1887 sur le site Moustérien de La Roche Penchée », publié en 1999 dans un numéro des Cahiers d'Archéologie Balafonienne .

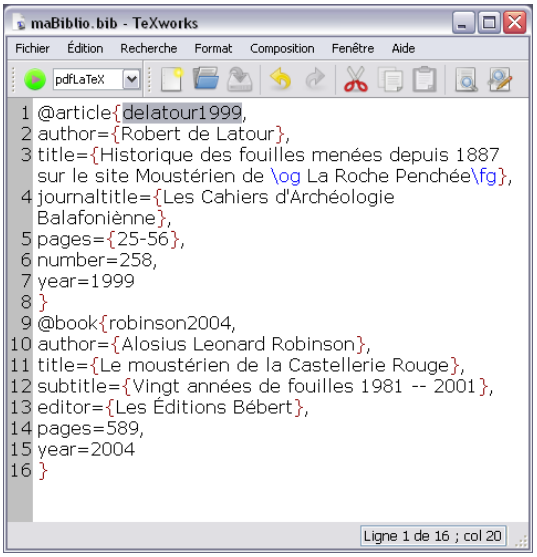


Figure 9.1 : La base bibliographique saisie dans TEXworks.

L'entrée dans la base bibliographique ressemble à :

```
@article{delatour1999,
author={Robert de Latour},
title={Historique des fouilles menées depuis 1887 sur le site
Moustérien de \og La Roche Penchée\fg},
journaltitle={Les Cahiers d'Archéologie Balafonienne},
pages={25-56},
number=258,
year=1999
}
```

9.1.2 Mlle Michu saisi un livre

Le professeur A. L. Robinson a écrit un bouquin sur ses fouilles à « La Castellerie Rouge », voici l'entrée :

```
@book{robinson2004,
author={Alosius Leonard Robinson},
title={Le moustérien de la Castellerie Rouge},
subtitle={Vingt années de fouilles 1981 -- 2001},
editor={Les Éditions Bébert},
pages=589,
year=2004
}
```

Comme tu peux le voir la création d'une base bibliographique n'a rien de sorcier, le nom des champ est explicite.

Il nous faut maintenant sauver notre base bibliographique. Sous TEXworks tu choisis dans le menu *Fichier Sauvegarder comme...* (Ctrl Shift S). Dans la liste *Fichiers de type* tu choisis *Bases de données BibTeX*, tu lui donnes le nom que tu veux, pour l'exemple on va prendre `maBiblio`.

9.2 Mlle Michu cite un ouvrage

Pour faire référence à un ouvrage dans ton texte rien de plus simple il te suffit d'utiliser la commande `\cite{clé}`. BibLATEX possède une autre commande `\parecite{clé}` qui encadre notre citation de parenthèse, c'est cette dernière que nous allons utiliser dans notre introduction :

```
% !TeX root = monRapport.tex
\addchap{Introduction}
La découverte d'un nouveau gisement moustérien dans cette région
du Balafon Occidental est toujours un événement. Surtout quand
cette découverte à lieu dans la vallée du Pételasoiffe, appelée
par de nombreux auteur la \og Vallée de l'Omo\fg de l'Europe du
Nord\footnote{Rendons à César ce qui\dots c'est au professeur
Raoul \bsc{Dumortier} que l'on doit la première mention de cette
expression.}. Le site d'Harfouëte -- La Mare au Diable, découvert
en 2009, outre son insertion dans un contexte micro-régional riche,
faisant déjà l'objet de très nombreuses recherches\footnote{Une
synthèse sous la direction de Robert \bsc{de Latour} devrait
paraître à la fin de l'année 2010.}, présente un grand intérêt
intrinsèque. En effet la présence plusieurs foyers et de reste
osseux rendent ce site unique dans cette région.
\par
Le gisement d'Harfouëte -- La Mare au Diable est situé à proximité
de deux sites mondialement connu. En amont, à \np[m]{500} à vol
d'oiseau se trouve le site de \og La roche Penchée\fg, fouillé
depuis les années 1887 \parecite{delatour1999}. À \np[km]{1,5}
en aval, se situe le gisement moustérien de \og La Castellerie
Rouge\fg, étudié depuis près de vingt ans par l'équipe américaine
du professeur A. L. \bsc{Robinson} \parecite{robinson2004}.
```

Code

9.3 M^lle Michu fabrique sa bibliographie

Maintenant que tu possèdes une base bibliographie stockée dans un fichier `maBiblio.bib`, on va voir comment en extraire les données pour réaliser une bibliographie. On va ajouter des commandes dans notre fichier maitre. Tout d'abord on va bien évidement charger le package BibLaTeX
`\usepackage{biblatex}` Ce package accepte des options. Nous allons nous intéresser à celle qui concerne la mise en forme de la bibliographie. Il y en a plusieurs mais nous ne retiendrons que authoryear qui est la plus adaptée au ouvrage en sciences humaines. Les appels dans le texte et la bibliographie sont du type auteur + année.

```
\usepackage[style=authoryear]{biblatex}
```

Comme notre rapport est en français nous allons ajouter le package *csquotes* qui permet d'avoir les guillemets françaises dans la bibliographie. Il nous faut indiquer à LaTeX le nom de notre base bibliographique

```
\bibliography{maBiblio}
```

Enfin nous devons indiquer où insérer la bibliographie en général à la fin du rapport par la commande :

```
\printbibliography
```

Voici notre nouveau fichier maitre :

Code

```
\documentclass[10pt,twoside]{scrreprt}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{lmodern}
\usepackage[np]{numprint} % pour écrire les nombres, option np pour
                          % \np{} et pas \numprint{}
\usepackage{xspace} % pour la gestion de \og et \fg sans {}
\usepackage[style=authoryear]{biblatex} % gestion de la
                         % bibliographie par auteurs + années
\bibliography{maBiblio} % chargement de la base bibliographique
\usepackage{csquotes}% pour avoir les guillemets français
                         % dans la biblio
\usepackage[frenchb]{babel}

%%%%%%%%%%%%%%%%%%%%%%%
%%% Commandes perso %%%
%%%%%%%%%%%%%%%%%%%%%%%
\newcommand*{\site}{Harfouëte -- La Mare au Diable\xspace}
\newcommand*{\vallee}{vallée du Pételasoiffe}

%%%%%%%%%%%%%%%%%%%%%%%
%%%      images      %%%
%%%%%%%%%%%%%%%%%%%%%%%
\graphicspath{./images/}
\DeclareGraphicsExtensions{.png,.jpg,.pdf}


\begin{document}
\input{introduction}
\input{historique}
\input{methode}
\printbibliography
\end{document}
```

9.4 Mlle Michu compile sa bibliographie

À ce stade de la rédaction de notre rapport, compiler la bibliographie n'est pas très utile mais puisque c'est le sujet de ce chapitre je vais te montrer comment faire. Pour obtenir la bibliographie dans ton document final, il faut réaliser la compilation en plusieurs étapes.

1. tu exécutes pdfLaTeX, ce qui provoque l'écriture des citations dans le fichier .aux
2. tu exécutes BibTeX (Attention BibTeX et non pas BibLaTeX) ce qui permet la création à partir du fichier .aux, d'un fichier .bbl qui contient les références et la bibliographie formatée.
3. une dernière exécution de pdfLaTeX incorpore les éléments contenus dans le fichier .bll dans ton document .pdf final.

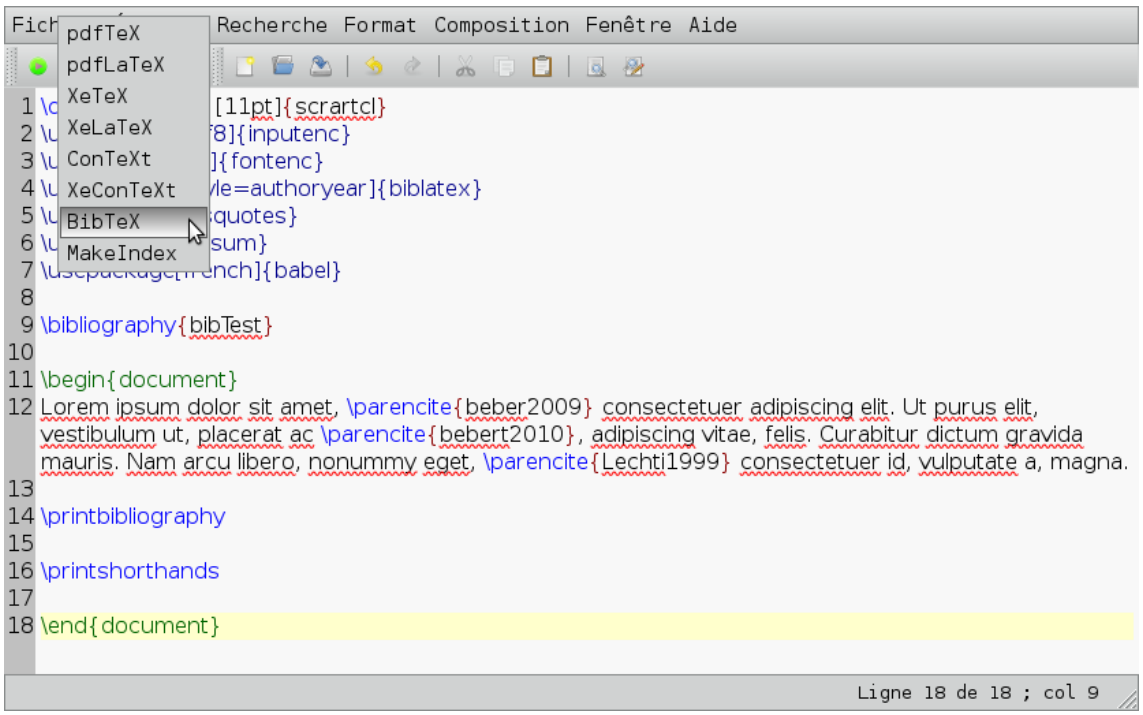


Figure 9.2 : Changement de compilateur avec TEXworks

L'image 9.2 te montre comment procéder au changement de compilateur avec TEXworks. Tu cliques dans le menu déroulant à côté de la flèche verte, et tu choisis pdflatex puis bibtex et à nouveau pdflatex. La flèche verte lance la compilation à chaque fois (raccourci clavier Ctrl T).

Si tu as des erreurs dans ton fichier .bib le compilateur bibtex te les indiquera de la même façon que pdflatex.

Les deux pages suivantes te montrent les pages *Introduction* et *Bibliographie* obtenues après compilation.

Introduction

La découverte d'un nouveau gisement moustérien dans cette région du Balafon Occidental est toujours un événement. Surtout quand cette découverte à lieu dans la vallée du Pételasoiffe, appelée par de nombreux auteur la « Vallée de l'Omo » de l'Europe du Nord [1]. Le site d'Harfouëte – La Mare au Diable, découvert en 2009, outre son insertion dans un contexte micro-régional riche, faisant déjà l'objet de très nombreuses recherches [2], présente un grand intérêt intrinsèque. En effet la présence plusieurs foyers et de reste osseux rendent ce site unique dans cette région.

Le gisement d'Harfouête – La Mare au Diable au Diable est situé à proximité de deux sites mondialement connu. En amont, à 500 m à vol d'oiseau se trouve le site de « La roche Penchée », fouillé depuis les années 1887 (Latour 1999). À 1,5 km en aval, se situe le gisement moustérien de « La Castellerie Rouge », étudié depuis près de vingt ans par l'équipe américaine du professeur A. L. Robinson (Robinson 2004).

1. Rendons à César ce qui... c'est au professeur Raoul Dumortier que l'on doit la première mention de cette expression.

2. Une synthèse sous la direction de Robert de Latour devrait paraître à la fin de l'année 2010.

Bibliographie

Latour, Robert de (1999). « Historique des fouilles menées depuis 1887 sur le site Moustérien de « La Roche Penchée » ». Dans : *Les Cahiers d'Archéologie Balafonièn*ne 258, p. 25–56.

Robinson, Alosius Leonard (2004). *Le moustérien de la Castellerie Rouge. Vingt années de fouilles 1981 – 2001*. Éd. par Les Éditions Bébert, p. 589.

10 Au tableau M^lle Michu

Poursuivant notre rapport nous arrivons enfin à la description du mobilier archéologique découvert. Lors de cette première campagne de fouille plusieurs secteurs ont été explorés. Afin de pouvoir comparer l'industrie lithique provenant de ces différents secteurs un petit tableau serait très utile. Nous allons passer un peu de temps sur les tableaux, car les sciences humaines sont de grandes consommatrices de tableaux.

10.1 M^lle Michu découvre tabular

La réalisation de tableaux se fait au moyen de l'environnement tabular. On n'a pas besoin de charger un package, c'est compris dans LaTeX. Comme tous les environnements LaTeX tu commences par un begin `\begin{tabular}` et tu finis par un end `\end{tabular}`.

La première chose à faire est de définir le nombre et le type de colonnes. L'environnement tabular possède trois types de colonnes :

centrée : noté c (**c**enter) ;

justifiée à gauche : notée l (**l**eft) ;

justifiée à droite : notée r (**r**ight).

Voici comment tu déclares un tableau avec trois colonnes respectivement centrée, justifiée à droite et justifiée à gauche :

```
\begin{tabular}{crl}

\end{tabular}
```

Code

Le remplissage du tableau se fait à l'aide de & qui permet de changer de colonne et de \\ qui change de ligne.

```
\begin{tabular}{crl}
ligne 1 colonne 1 & ligne 1 colonne 2 & ligne 1 colonne 3 \\
ligne 2 colonne 1 & ligne 2 colonne 2 & ligne 2 colonne 3 \\
centrée & à droite & à gauche \\
\end{tabular}
```

Code

ligne 1 colonne 1 | ligne 1 colonne 2 | ligne 1 colonne 3
ligne 2 colonne 1 | ligne 2 colonne 2 | ligne 2 colonne 3
centrée | à droite | à gauche

Résultat

J'avoue il n'est pas évident de voir dans ce résultat un tableau, car il maque un élément, les lignes. Pour tracer des lignes horizontale tu utilises la commande \hline, placée après le changement de ligne et pour les ligne verticale le symbole | (*pipe* en anglais) obtenus par AltGr 6, que tu places avec le descriptif de colonne.

Code

```
\begin{tabular}{|c|r|l|}
\hline
ligne 1 colonne 1 & ligne 1 colonne 2 & ligne 1 colonne 3 \\
\hline
ligne 2 colonne 1 & ligne 2 colonne 2 & ligne 2 colonne 3 \\
\hline
centrée & à droite & à gauche \\
\hline
\end{tabular}
```

Résultat

ligne 1 colonne 1	ligne 1 colonne 2	ligne 1 colonne 3
ligne 2 colonne 1	ligne 2 colonne 2	ligne 2 colonne 3
centrée	à droite	à gauche

Tu n'es pas obligé de laisser des espaces ou des sauts de ligne avant les &\\ et |. Il ne sont utiles que pour une meilleur lisibilité du code. Les espaces ajoutées, n'ont aucune influence sur l'espacement entre le texte et les filets. L'exemple suivant donnera exactement le même résultat :

Code

```
\begin{tabular}{| c| r |l|}
\hline ligne 1 colonne 1&ligne 1 colonne 2&ligne 1 colonne 3
\\\hline
ligne 2 colonne 1&ligne 2 colonne 2& ligne 2 colonne 3 \\\hline
centrée & à droite & à gauche \\\hline
\end{tabular}
```

Pour obtenir plusieurs lignes horizontales ou verticales, il te suffit d'ajouter des | ou des \hline.

Code

```
\begin{tabular}{||c|r|||l|}
\hline
ligne 1 colonne 1 & ligne 1 colonne 2 & ligne 1 colonne 3 \\
\hline\hline
ligne 2 colonne 1 & ligne 2 colonne 2 & ligne 2 colonne 3 \\
\hline\hline\hline
centrée & à droite & à gauche \\
\hline \end{tabular}
```

Résultat

ligne 1 colonne 1	ligne 1 colonne 2	ligne 1 colonne 3
ligne 2 colonne 1	ligne 2 colonne 2	ligne 2 colonne 3
centrée	à droite	à gauche

Attention

Il doit y avoir nombreDeColonne – 1 **&** (il n'y en a pas ni au début de la ligne, ni à la fin) sous peine d'erreurs de compilation. Par exemple pour un tableau de trois colonnes, chaque ligne doit avoir deux **&**
Une cellule vide est notée **&&** ou **& &**

Code

```
\begin{tabular}{|c|r|l|}
\hline
& ligne 1 colonne 2 & ligne 1 colonne 3 \\\hline
ligne 2 colonne 1 && ligne 2 colonne 3 \\\hline
centrée & à droite & à gauche \\\hline
\end{tabular}
```

Résultat

	ligne 1 colonne 2	ligne 1 colonne 3
ligne 2 colonne 1		ligne 2 colonne 3
centrée	à droite	à gauche

10.2 Mlle Michu va au tableau

Fière de ses connaissances nouvellement acquises Mlle Michu se lance dans la réalisation de son premier tableau.

Code

```
\chapter{Étude archéologique}
\section{Les matières premières}
Les préhistoriques ont apporté sur le site un silex crétacé sous
forme de rognons, qu'ils ont puisé dans les argiles à silex situé à
quelques centaines de mètres. Dans la nappe alluviale du Pételasoif,
ils ont prélevés des blocs de grès et de grès quartzite ainsi que
quelques rares rognons de silex tertiaire.
\begin{tabular}{|r|r|r|r|}
\hline
& Sondage 1 & Sondage 2 & Fouille du talus \\\hline
Silex& 99\% & 73\% & 60\% \\\hline
Grès & 0\% & 14\% & 23\% \\\hline
Grès quartzite & 0\% & à gauche \\\hline
Silex tertiaire& 1\% & 3\% & 7\% \\\hline
\end{tabular}
```

Remarque

% fait partie des caractères réservés par LaTeX. Je te rappelle qu'il sert à introduire des commentaires dans ton document qui ne seront pas interprété par le compilateur. Donc si tu veux écrire des pourcentages tu dois employer la commande \%

Résultat

4. Étude archéologique

4.1. Les matières premières

Les préhistoriques ont apporté sur le site un silex crétacé sous forme de rognons, qu'ils ont puisé dans les argiles à silex situées à quelques centaines de mètres. Dans la nappe alluviale du Pételasoif, ils ont prélevé des blocs de grès et de grès quartzite ainsi que quelques rares rognons de silex tertiaire.

	Sondage 1	Sondage 2	Fouille du talus
Silex	99%	73%	60%
Grès	0%	14%	23%
Grès quartzite	0%	10%	10%
Silex tertiaire	1%	3%	7%

M^lle Michu est assez satisfaite de son tableau, mais elle aimerait bien qu'il soit centré et lui adjoindre une légende. Et la elle se souvient qu'il existe un flottant pour les tableaux et que ce serait pas une mauvaise idée de l'employer.

Code

```
\chapter{Étude archéologique}
\section{Les matières premières}
kLes préhistoriques ont apporté sur le site un silex crétacé
sous forme de rognons, qu'ils ont puisé dans les argiles à silex
situées à quelques centaines de mètres. Dans la nappe alluviale du
Pételasoif, ils ont prélevé des blocs de grès et de grès quartzite
ainsi que quelques rares rognons de silex tertiaire.
\begin{table}[htbp]
\begin{center}
\begin{tabular}{|r|r|r|r|}
\hline
& Sondage 1 & Sondage 2 & Fouille du talus \\\hline
Silex& 99\% & 73\% & 60\% \\\hline
Grès & 0\% & 14\% & 23\% \\\hline
Grès quartzite & 0\% & à gauche \\\hline
Silex tertiaire& 1\% & 3\% & 7\% \\\hline
\end{tabular} \end{center} \caption{Répartition des matières
premières dans les différents secteurs fouillés} \label{matPrem}
\end{table}
```

4 Étude archéologique

4.1 Les matières premières

Les préhistoriques ont apporté sur le site un silex crétacé sous forme de rognons, qu'ils ont puisé dans les argiles à silex situé à quelques centaines de mètres. Dans la nappe alluviale du Pételasoif, ils ont prélevés des blocs de grès et de grès quartzite ainsi que quelques rares rognons de silex tertiaire.

Résultat

	Sondage 1	Sondage 2	Fouille du talus
Silex	99%	73%	60%
Grès	0%	14%	23%
Grès quartzite	0%	10%	10%
Silex tertiaire	1%	3%	7%

TABLE 4.1: Répartition des matières premières dans les différents secteurs fouillés

10.3 Ça déborde M^lle Michu

M^lle Michu emballée par le résultat se lance immédiatement dans la réalisation d'un autre tableau :

Résultat

	Sondage 1	Sondage
Nucléus non Levallois	3	
Nucléus Levallois de modalité linéale à préparation centripète	1	
Nucléus Levallois de modalité récurrente bipolaire à deux plans de frappe opposés	0	
Total		

Oups !!! ça déborde. En effet tabular dimensionne la taille de la colonne à la taille de la plus grande ligne et très vite on déborde de la page, d'autant plus que les colonnes de type `{l c r}` ne reconnaissent aucun saut de ligne (ni en passant une ligne blanche ni en utilisant `\par`.

Tu peux améliorer le résultat en diminuant la taille de la fonte :

Code

```
{\tiny \begin{tabular} ... la suite du tableau ...\end{tabular}}
```

Résultat

	Sondage 1	Sondage 2	Fouille du talus
Nucléus non Levallois	3	3	5
Nucléus Levallois de modalité linéale à préparation centripète	1	2	8
Nucléus Levallois de modalité récurrente bipolaire à deux plans de frappe opposés	0	1	5
Total			

Effectivement notre tableau ne déborde plus, mais il n'est pas très lisible. Il existe une autre solution. tabular possède un autre type de colonne p{valeurUnité}
Cette commande permet de fixer la largeur de la colonne. Elle reconnait bien entendue toutes les unités comprises par LaTeX (voir section 7.3, page 58). Le même tableau que précédemment avec une première colonne fixée à 3 cm.

Résultat

	Sondage 1	Sondage 2	Fouille du talus
Nucléus non Levallois	3	3	5
Nucléus Levallois de modalité linéale à préparation centripète	1	2	8
Nucléus Levallois de modalité récurrente bipolaire à deux plans de frappe opposés	0	1	5
Total	4	6	18

Cette colonne ne produit que des colonnes justifiées. Pour améliorer la disposition de ton texte à l'intérieur de chaque colonne, tu disposes des commandes suivante \par pour les saut de paragraphe, \centering pour centrer ton texte, \raggedright (fer à droite) et \raggedright (fer à gauche). Si tu veux changer l'alignement de ton texte pour toute la colonne tu dois l'indiquer dans chaque cellule.

Code

```
\begin{tabular}{|p{5cm}|r|r|r|}
\hline
& Sondage 1 & Sondage 2 & Fouille du talus \\\hline
\centering Nucléus non Levallois& 3 & 3 & 5 \\\hline
\raggedright Nucléus Levallois de modalité linéale à préparation
centripète & 1 & 2 & 8 \\\hline
Nucléus Levallois de\par modalité récurrente bipolaire\par à deux
plans de frappe opposés & 0 & 1& 5 \\\hline
\raggedleft Total& 4& 6& 18\\\hline
\end{tabular}
```

	Sondage 1	Sondage 2	Fouille du talus
Nucléus non Levallois	3	3	5
Nucléus Levallois de modalité linéale à préparation centripète	1	2	8
Nucléus Levallois de modalité récurrente bipolaire à deux plans de frappe opposés	0	1	5
Total	4	6	18

Résultat

10.4 M^lle Michu veut ses aises

Si tu trouves que les colonnes de ton tableaux sont trop serrées tu peux modifier l'espacement entre 2 lignes avec `\arraystretch` et l'espace entre deux colonnes avec `\tabcolsep`.

10.4.1 L'espace entre deux colonnes

`\tabcolsep` correspond à la valeur de l'espace avant ou après un | (donc la moitié de l'espace inter-colonne). Tu le modifies à l'aide des commandes `\setlength` ou `\addtolength` (voir l'anexe 7.3 p. 58). Sa valeur par défaut est de 6 pt.

case 1	case 2	case 3
case 4	case 5	case 6

Résultat

Le même tableau avec une valeur de `\tabcolsep` de 12 pt.

case 1	case 2	case 3
case 4	case 5	case 6

Résultat

Le code du tableau ci-dessus :

```
{\setlength{\tabcolsep}{12pt}
\begin{tabular}{|c|c|c|}
\hline
case 1 & case 2 & case 3\\\hline
case 4 & case 5 & case 6\\\hline
\end{tabular}}
```

Code

Remarque les deux accolades qui encadre le code, elles permettent de limiter l'action de `\tabcolsep` à ce seul tableau. Si tu ne les mets pas tous les tableaux qui suivent dans ton texte seront affecté par l'élargissement de l'espace inter-colonnes (ça peut être un choix délibéré). Si tu veux que que tous

les tableaux de ton document présentent des espaces inter-colonne plus large tu places la commande `\setlength{\tabcolsep}{12pt}` dans le préambule avant le `\begin{document}`.

10.4.2 L'espace entre les lignes

Il te faut modifier la valeur de `\arraystretch` pour augmenter l'espacement entre 2 lignes. La valeur de `\arraystretch` est un coefficient multiplicateur : 0.5 diminue l'espace de moitié, 2.0 le double. Sa valeur par défaut est de 1.0.

Attention

Attention cette commande se modifie à l'aide de `\renewcommand`.

Le même tableau que précédemment avec une valeur de `\arraystretch` de 2.

Résultat

case 1	case 2	case 3
case 4	case 5	case 6

Code

```
{\renewcommand{\arraystretch}{2}
\begin{tabular}{|c|c|c|}
\hline
case 1 & case 2 & case 3\\\hline
case 4 & case 5 & case 6\\\hline
\end{tabular}}
```

Même remarque que ci-dessus pour les accolades entourrant le code.

10.5 Mlle Michu veut des traits partiels

`\cline{colonneDébut-colonneFin}`, trace des lignes partielles :
`\cline{1-1}` trace une ligne uniquement dans la 1re colonne, `\cline{2-4}` du début de la colonne 2 à la fin de la colonne 4.

Code

```
\begin{tabular}{|1|1|1|1|}
\hline case \no1 & case \no2 & case \no3 & case \no4 \\
\cline{1-1}
case \no5 & case \no6 & case \no7 & case \no8 \\
\cline{2-4}
case \no9 & case \no10 & case \no11 & case \no12 \\
\cline{1-2}\cline{4-4}
case \no13 & case \no14 & case \no15 & case \no16 \\
\hline
\end{tabular}
```

case n° 1	case n° 2	case n° 3	case n° 4
case n° 5	case n° 6	case n° 7	case n° 8
case n° 9	case n° 10	case n° 11	case n° 12
case n° 13	case n° 14	case n° 15	case n° 16

Résultat

10.6 M^lle Michu fusionne

10.6.1 La fusion de colonnes

La fusion de colonnes se fait avec la commande `\multicolumn{nombCol}{alignement}{contenu}` où alignement vaut l,c,r. Tu n'es pas obligé de respecter le choix que tu as fait à l'initialisation des colonnes ; regarde le comportement des cellules n° 1 et n° 10.

```
\begin{tabular}{|l|l|l|l|}
\hline
\multicolumn{2}{|r|}{Cellules fusionnées} & case \no3
& case \no4 \\\hline
case \no5 &\multicolumn{2}{l|}{Cellules fusionnées} & case \no8\\
\hline
case \no9 &\multicolumn{3}{|c|}{Cellules fusionnées}\\\hline
case \no13 & case \no14 & case \no15 & case vsargno16\\
\hline
\end{tabular}
```

Code

Cellules fusionnées		case n° 3	case n° 4
case n° 5	Cellules fusionnées		case n° 8
case n° 9	Cellules fusionnées		
case n° 13	case n° 14	case n° 15	case n° 16

Résultat

Attention aux | dans multicolumn qui peuvent tracer 2 traits comme dans la case n° 10, il vaut mieux procéder comme pour la case n° 6 et enlever le | de gauche dans |c|.

Attention

10.6.2 La fusion de lignes

La fusion de lignes nécessite le package *multirow* (`\usepackage{multirow}`) et se fait avec la commande
`\multirow{nombreLignes}{largeur}{contenu}`. Tu peux remplacer `{largeur}` par `{*}`, l'* indique une largeur automatique, adapté au contenu de la cellule.

Code

```
\begin{tabular}{|l|l|l|l|}
\hline
Le texte \no1 & Le texte \no2 & ligne1&
Une longue ligne pour remplir\\\hline
\multirow{3}{*}{texte3} &\multirow{3}{3cm}{texte4}
&\multirow{2}{4cm}{texte5}\\
\cline{3-3}
&& ligne3 &\\\cline{3-4} && ligne4 & Fin \\\hline
\end{tabular}
```

Résultat

Le texte n° 1	Le texte n° 2	ligne1	Une longue ligne pour remplir
texte3	texte4	ligne2	texte5
		ligne3	
		ligne4	Fin

Attention

N'oublies pas de saisir des cellules vides pour celles qui sont fusionnées. Pour chaque ligne du tableau tu dois toujours avoir un nombre de & égal à nombreDeColonne – 1.

Attention la commande \multirow ne se comporte pas comme \multicolumn, elle ne supprime pas les traits en trop. Il te faut utiliser à bon escient les \cline

Code

```
\begin{tabular}{|l|c|}
\hline
{\bfseries colonne \no1}&{\bfseries colonne \no2}\\\hline
Le texte \no1 &\multirow{2}{4cm}{cellule sur 2 lignes}\\\hline
Le texte \no2 &\\\hline
\end{tabular}
```

Résultat

colonne n° 1	**colonne n° 2**
Le texte n° 1	~~cellule sur 2 lignes~~
Le texte n° 2	

Code

```
\begin{tabular}{|l|c|}
\hline
{\bfseries colonne \no1}&{\bfseries colonne \no2}\\\hline
Le texte \no1 &\multirow{2}{4cm}{cellule sur 2 lignes}\\\cline{1-1}
Le texte \no2 &\\\hline
\end{tabular}
```

colonne n° 1	colonne n° 2
Le texte n° 1	cellule sur 2 lignes
Le texte n° 2	

Résultat

Il y aurait encore beaucoup à dire sur les tableaux, mais avec ces premiers éléments tu devrais pouvoir réaliser un grand nombre de tableaux différents. Bébert, tu sais le mec sympa qui avait déjà réalisé une fiche sur la bibliographie a également pondu une série de fiches sur les tableaux : Les fiches de Bébert. Vraiment sympa ce Bébert !

11 M^lle Michu, le début de la fin

Maintenant tu as tous les éléments pour terminer toute seule le rapport. Je te laisse faire. Il nous reste quelques éléments à voir pour le finaliser : la page de titre les tables des matières et des figures, la numérotation des pages...

11.1 M^lle Michu à l'annexe

Les annexes sont des chapitres particuliers dont la numérotation est alphanumérique : A , A.1, A.1.1, B, B.1, B.2, ...

C'est la commande `\appendix` qui est à l'origine des annexes. Tu la places à l'endroit où tu veux voir apparaitre tes annexes. Voici un exemple de code :

```
\chapter{Conclusion}
Le texte
...
de ta conclusion
\appendix
\chapter{Première annexe}
\section{Première section}
...
\chapter{Deuxième annexe}
...
```

Code

Ce qui te donne pour la première annexe :

A. Première anexe

Résultat

A.1. Première section

Le texte ...

11.1.1 M^lle Michu veut voir le mot annexe apparaitre

Il te suffi d'ajouter `\KOMAoptions{chapterprefix}` juste avant `\appendix`.

Code

```
\chapter{Conclusion}
Le texte
...
de ta conclusion
\KOMAoptions{chapterprefix}
\appendix
\chapter{Première annexe}
\section{Première section}
...
\chapter{Deuxième annexe}
...
```

Résultat

Annexe A.

Première anexe

A.1. Première section

Le texte ...

Remarque

Si tu places la commande `\KOMAoptions{chapterprefix}` dans l'entête de ton document maître, tu retrouveras le comportement par défaut de la classe *report* qui introduit les chapitres par le mot « Chapitre » suivi du numéro

Code

```
...
\KOMAoptions{chapterprefix}
\begin{document}
\chapter{Titre du chapitre}
\section{Titre de la première section}
...
```

Chapitre 1

Titre du chapitre

Résultat

1.1 Titre de la première section

11.2 À table M^lle Michu

Pour paraphraser Boris Vian « la question de la table des matières, se résout en un quart d'heure, c'est de celles qu'on écarte ». (d'après « La java des bombes atomiques »). Tu places un `\tableofcontents` à l'endroit où tu veux qu'elle apparaisse. Tout aussi facilement `\listeoffigures` te créera une liste des figures et `\listoftables` une liste des tableaux.

Attention

Il faut deux compilations pour obtenir le bon résultat. Lors de la première LaTeX écrit les données utiles dans un fichier `.toc` pour les tables des matières et un fichier .lof pour les figures (dans notre cas `monRapport.toc` et `monRapport.lof`). La deuxième compilation pour inclure la table des matières ou la liste des figures dans notre document. Si tu modifies ton sectionnement alors que tu as déjà une table des matières, cette nouvelle entrée n'apparaitra qu'au bout de deux compilations.

11.3 M^lle Michu des pieds à la tête

Par défaut LaTeX ne place que le numéro de page, centré dans le pied de page pour les documents en simple page et sur le bord externe de la feuille pour les recto-verso. Tu peux changer ce comportement à l'aide de `\pagestyle{style}` où `style` peut prendre les valeurs :

empty : sans entête, ni pied de page, donc plus de numéro de page ;

plain : le style par défaut, uniquement le numéro de page en pied de page ;

headings : certaines informations sont affichées dans l'entête. Elles dépendent du type de document. Pour un rapport en recto-verso, tu as le titre du chapitre courant, sur les pages paires et le titre de la section courantes sur les pages impaires.

Tu places cette commande dans le préambule de ton document maître.

Remarque

Même avec l'emploi de `\pagestyle{headings}` la première page d'un chapitre ne possède pas d'entête.

Si pour une raison ou pour une autre tu désires qu'une page ait un comportement différent, du reste du document, tu ajoutes au niveau de cette page la commande : `\thispagestyle{style}`.

Cette gestion des entêtes est assez sommaire. Si tu veux aller plus loin tu devras examiner le package `fancyhdr`.

11.4 M^lle Michu veut un titre

LaTeX fabrique automatiquement le titre, pour cela il faut lui fournir le nom de l'auteur, le titre de l'ouvrage, éventuellement un sous-titre. Pour ce faire tu places dans le préambule de ton source maître les éléments suivant :

Code

```
\author{Le nom de l' auteur}
\title{Le titre de l'ouvrage}
\subtitle{Le sous titre}
```

Le titre se fabrique avec la commande `\maketitle` que tu places juste après le `\begin{document}`.

Si je reprend un extrait de notre document maître `monRapport.tex` :

Code

```
\documentclass[10pt,twoside]{scrreprt}

...
\usepackage[frenchb]{babel}

...
%%%%%%%%%%%%%%%%%%%%%%%
%%%      titre      %%%
%%%%%%%%%%%%%%%%%%%%%%%
\author{M^lle Michu}
\title{Le gisement moustérien d'\site}
\subtitle{Première campagne de fouille 6 septembre - 15 octobre
2010}
\begin{document}
\maketitle
\input{introduction}
\input{historique}
\input{methode}
\printbibliography
\end{document}
```

Ce qui nous donne :

Le gisement moustérien d'Harfouête – La Mare au Diable

Première campagne de fouille 6 septembre – 15 octobre 2010

Résultat

M^lle Michu

26 novembre 2010

Comme tu peux le voir la date du jour est automatiquement ajoutée. Si tu ne veux pas voir apparaitre la date tu places au même niveau que l'auteur et le titre la commande `date` vide : `\date{}`. Si tu veux une date particulière indépendante de celle du jour courant, tu l'inscris dans le champ date : `\date{14 juillet 1789}`.

La coupure du titre au niveau du nom du site n'est pas du meilleur effet. Tu peux passer une ligne à l'aide de `\\`.

```
\title{Le gisement moustérien\\ d'\site}.
```

11.5 Vous reprendrez bien une petite image M^lle Michu

Tu peux améliorer ta page de titre sans trop de difficulté, en y insérant une image. Pour cela on va avoir besoin d'une notion que l'on a pas encore étudié.

11.5.1 Les espaces verticaux

Tu n'y a peut être pas prêter attention mais jusqu'à maintenant on a utilisé aucun espace vertical pour la réalisation de notre rapport. C'est très bien ainsi, LaTeX ce débrouille très bien tout seul. Les commandes `\title` et `\subtitle` ne sont pas prévu à l'origine pour y mettre des images donc il va falloir les aider. LaTeX reconnait plusieurs commandes pour gérer des espaces verticaux. Tout d'abord `\bigskip`, `\medskip` et `\smallskip` pour placer un grand, moyen et petit espace. Et la commande `\vspace{dimension}` pour un espace dont tu fixes la hauteur.

Attention

Ces commandes n'insère pas un saut de ligne. Elles ne gèrent que la taille de l'espace entre les lignes ou les objets. Tu dois ajouter au préalable un saut de ligne par, une ligne blanche, les commandes `\par` ou `\\`.

Code

```
Une première ligne.\\
\smallskip
Une deuxième ligne.\\
\medskip
Troisième ligne.\\
\bigskip
Quatrième ligne.\\
\vspace{1.5cm}
Dernière ligne.
```

Résultat

Une première ligne.
Une deuxième ligne.
Troisième ligne.

Quatrième ligne.

Dernière ligne.

11.5.2 Une image dans le titre

Tu vas donc la placer dans `\subtitle{}` à l'aide de la commande `\includegraphics[]{}` et tu vas la séparer du titre par des commandes `\vspace{}`.

Code

```
\subtitle{Première campagne de fouille 6 septembre - 15 octobre 2010
\\\vspace{1cm}
\includegraphics[scale=0.3]{image}\\
\vspace{2cm}}
```

Le premier espace sépare l'image du sous-titre et le second du nom de l'auteur. Si tu n'as pas de sous-titre tu modifies un peu le code :

Code

```
\subtitle{\vspace{1cm}
\includegraphics[scale=0.3]{image}\\
\vspace{2cm}}
```

Le résultat de ce petit bidouillage est sur la page suivante.

Le gisement moustérien d'Harfouête – La Mare au Diable

Première campagne de fouille 6 septembre – 15 octobre 2010

M^{lle} Michu

26 novembre 2010

12 Il y en a un petit peu plus M^lle Michu, je vous le laisse ?

On va voir dans ce chapitre plusieurs trucs dont on a pas eu besoin pour rédiger notre rapport archéologique mais dont tu peux avoir l'utilité ou qui embelliront ton document facilement.

12.1 M^lle Michu mise en boîte

Tu peux avoir besoin ponctuellement de boîte pour mettre une portion de texte en évidence (une définition, un aparté, une longue citation,...).

LaTeX reconnaît de nombreux types de boîtes, il n'est pas question de les étudier tous dans le cadre de cet ouvrage, on va seulement examiner les minipages. Comme son nom l'indique, il s'agit de créer une petite page à l'intérieur de ta page. Tu pourra y intégrer tous les éléments de formatage que tu utilises dans le reste de ton document. C'est un environnement dont voici la syntaxe (La complétion s'obtient avec bmi + Tab + Tab) :

```
\begin{minipage}{largeur}
Le texte à mettre dans la minipage
...
\end{minipage}
```

Voici un exemple dans un texte :

```
Tout d'abord un texte qui ne veut rien dire qui sert uniquement à
faire du remplissage, pour montrer l'utilisation des minipages.
\begin{minipage}{7cm}
\small
Encore du texte pour faire du remplissage mais celui-ci est à
l'intérieur de ma minipage. J'ai utilisé une fonte plus petite
(commande small).

On peut utiliser des sauts paragraphes.
...
\end{minipage}

Et enfin, texte qui ne veut rien dire qui sert uniquement à faire
du remplissage, pour montrer l'utilisation des minipages.
```

Résultat

Tout d'abord un texte qui ne veut rien dire qui sert uniquement à faire du remplissage, pour montrer l'utilisation des minipages.

Encore du texte pour faire du remplissage mais celui-ci est à l'intérieur de ma minipage. J'ai utilisé une fonte plus petite (commande small).
On peut utiliser des sauts paragraphes.
...

Et enfin, texte qui ne veut rien dire qui sert uniquement à faire du remplissage, pour montrer l'utilisation des minipages.

Remarque

Comme tu peux le constater les paragraphes à l'intérieur de la minipage ne sont pas indentés

Tu peux centrer ta minipage :

Code

```
Tout d'abord un texte qui ne veut rien dire qui sert uniquement à
faire du remplissage, pour montrer l'utilisation des minipages.
\begin{center}
\begin{minipage}{7cm}
\small
Encore du texte pour faire du remplissage mais celui-ci est à
l'intérieur de ma minipage. J'ai utilisé une fonte plus petite
(commande small).

On peut utiliser des sauts paragraphes.
...
\end{minipage}
\end{center}

Et enfin, texte qui ne veut rien dire qui sert uniquement à faire
du remplissage, pour montrer l'utilisation des minipages.
```

Résultat

Tout d'abord un texte qui ne veut rien dire qui sert uniquement à faire du remplissage, pour montrer l'utilisation des minipages.

Encore du texte pour faire du remplissage mais celui-ci est à l'intérieur de ma minipage. J'ai utilisé une fonte plus petite (commande small).
On peut utiliser des sauts paragraphes.
...

Et enfin, texte qui ne veut rien dire qui sert uniquement à faire du remplissage, pour montrer l'utilisation des minipages.

12.2 Mlle Michu fait du coloriage

La couleur ce gère avec le package *xcolor* (`\usepackage{xcolor}`), qui contient 19 couleurs prédéfinies : black, white, red, green, blue, cyan, magenta, yellow, lime, brown, cyan, darkgray, gray, lightgray, magenta, olive, orange, pink, purple, teal, violet.

Tu peux charger d'autres systèmes de couleur, pour augmenter le nombre de couleur prédéfinie :

dvipsnames : 68 couleurs prédéfinies ;

svgnames : 150 couleurs prédéfinies ;

x11names : 317 couleurs prédéfinies ;

Tu fais appel à ces systèmes par une commande du type `\usepackage[dvipsnames]{xcolor}` ou plusieurs en même temps `\usepackage[svgnames,dvipsnames]{xcolor}`

La figure 12.1 page 114, te donne la liste pour *dvipsnames*, pour les 2 autres je te renvoie à la documentation d'*xcolor* pages 38–40.

Remarque

Un petit rappel : pour acceder à la documentation des packages de la TeXLive il te suffit de taper dans une console :
`texdoc nomduPackage` donc pour xcolor
`texdoc xcolor`

12.2.1 Du texte en couleur

Tu as deux possibilités pour mettre du texte en couleur. Pour les textes courts sans saut de paragraphe tu peux utiliser la commande à argument `\textcolor{la couleur}{Le texte à mettre en couleur}`. Pour des portions de texte plus long ou comprenant des sauts de paragraphes, tu emploies la déclaration `\color{la couleur}`, dont tu limites la portée par des `{}`.

Code

```
Voici un exemple de \textcolor{red}{texte rouge}. Un autre exemple
{\color{RawSienna} un fragment de texte avec une couleur dvipsnames
et

un saut de paragraphe}. Le reste est en noir.
```

Résultat

Voici un exemple de texte rouge. Un autre exemple un fragment de texte avec une couleur dvipsnames et
un saut de paragraphe. Le reste est en noir.

12.2.2 La feuille en couleur

Pour changer la couleur de la feuille (en blanc par défaut sous LaTeX), tu dois utiliser la commande : `\pagecolor{couleur}`. La portée de cette commande commence à la page courante où a été déclarée la commande jusqu'à la fin du document ou à une nouvelle commande `\pagecolor{couleur}`. Pour affecter tout le document place la commande dans le préambule (avant le `\begin{document}`).

Figure 12.1 : Liste des 68 couleurs prédéfinies de l'option dvipsnames. Attention aux majuscules qui sont obligatoires.

12.2.3 Des boites de couleur

Pour obtenir un texte sur un fond de couleur tu utilises la commande `\colorbox{Couleur du fond}{texte}` :

Code

```
Voici un \colorbox{red}{texte sur fond rouge}.
```

Résultat

Voici un texte sur fond rouge.

Et évidemment pour mettre un texte en couleur sur fond de couleur, tu combines les deux commandes :

Code

```
Voici un \colorbox{blue}{\textcolor{red}{texte rouge sur fond
bleu}}.
```

Résultat

Voici un texte rouge sur fond bleu.

Pour avoir un cadre coloré il faut changer de type de boite : `\fcolorbox{couleur de la bordure}{couleur du fond}{texte}`

Code

```
\fcolorbox{red}{cyan}{Un texte en noir sur fond bleu et bordé de
rouge}
```

Résultat

Un texte en noir sur fond bleu et bordé de rouge

Tu peux régler l'épaisseur de la bordure avec `\setlength{\fboxrule}{valeur unité}`

Remarque

N'oublie pas la portée des commandes. En fonction de l'endroit où tu places le `\setlenght`... et de l'emploi ou non d'accolades, tu affecteras une seule, plusieurs ou toutes tes boites.

12.2.4 Intensité des couleurs

Si tu trouves les couleurs un peu trop flashi pour ton usage, tu diminues leur intensité de la manière suivante :

Code

```
{\color{red!50} qui indique 50% de rouge},{\color{yellow!25} indique
25% de jaune,}{\color{yellow} indique 100% de jaune,...}
```

Résultat

qui indique 50% de rouge, indique 25% de jaune, indique 100% de jaune,...

12.2.5 Couleur et minipage

Voici un exemple un peu plus complexe pour créer des encarts qui ont de la gueule. J'y ai mis un maximum de chose pour te montrer ce qu'il est possible de faire. Pour colorer mon encart j'ai placé une minipage dans une fcolorbox.

Code

```
Tout d'abord un texte qui ne veut rien dire qui sert uniquement à
faire du remplissage, pour montrer l'utilisation des minipages.
\begin{center}
\fcolorbox{red}{cyan}{
\begin{minipage}{7cm}
\begin{center}
\bfseries Le titre de ma minipage
\end{center}
\small
Encore du texte pour faire du remplissage mais celui-ci est à
l'intérieur de ma minipage. J'ai utilisé une fonte plus petite
(commande small). Voici une note de bas de page\footnote{La note
de bas de page qui est numéroté avec des lettres à l'intérieur de la
minipage}

On peut utiliser des sauts paragraphes.

Une liste :
\begin{itemize}
\item premier item ;
\item deuxième item ;
\item le dernier.
\end{itemize}
Une autre note de bas de page\footnote{La deuxième note}. Et enfin
un petit pingouin \includegraphics[width=1cm]{tux}.
\end{minipage}
}% cette accolade ferme la fcolorbox
\end{center}
Et enfin, texte qui ne veut rien dire qui sert uniquement à faire du
remplissage, pour montrer l'utilisation des minipages.
```

Je n'ai pas mis le résultat dans un encart coloré pour que tu puisses voir correctement le résultat (je ne me suis pas foulé pour le choix des couleurs) :

Tout d'abord un texte qui ne veut rien dire qui sert uniquement à faire du remplissage, pour montrer l'utilisation des minipages.

Le titre de ma minipage

Encore du texte pour faire du remplissage mais celui-ci est à l'intérieur de ma minipage. J'ai utilisé une fonte plus petite (commande small). Voici une note de bas de page [a]
On peut utiliser des sauts paragraphes.
Une liste :
– premier item ;
– deuxième item ;
– le dernier.
Une autre note de bas de page [b]. Et enfin un petit pingouin .

a. La note de bas de page qui est numéroté avec des lettres à l'intérieur de la minipage
b. La deuxième note

Et enfin, texte qui ne veut rien dire qui sert uniquement à faire du remplissage, pour montrer l'utilisation des minipages.

12.3 Mlle Michu dans l'hypertexte

Si ton document final n'est pas fait uniquement pour l'impression, et si tu veux qu'il puisse également être lu sur écran, il devient intéressant d'y ajouter des hyperliens. De cette façon tu pourras cliquer sur tous les objets référencés comme la table des matières, les listes de figures, les références bibliographiques, les références dans le texte, les notes de bas de page, les index et te retrouver automatiquement sur la zone référencée.

Une seule commande, tu charges le package *hyperref* et il se charge de tout.

Ce package comporte une multitude d'options, je ne vais t'en indiquer quelques unes, les plus basiques.

12.3.1 Liens en couleur

colorlinks=true permet de colorer les liens

linkcolor=couleur la couleur des liens internes *red* par défaut

citecolor=couleur la couleur des citations bibliographiques *green* par défaut

urlcolor=couleur la couleur des liens url *magenta* par défaut

Donc pour avoir les liens internes roses, les citations bibliographiques jaunes et les url en violet tu codes de cette façon :

Code

```
\usepackage[colorlinks=true,linkcolor=pink,citecolor=yellow,urlcolor=violet]{hyperref}
```

Pour avoir les liens avec les couleurs par défaut cette commande suffit :

Code

```
\usepackage[colorlinks=true]{hyperref}
```

Si tu ne mets pas l'option *colorlinks=true* tous tes liens seront encadrés par une boite de couleur comme ceci : 12

12.3.2 Lien vers un site web

Si tu veux insérer des liens vers des sites web voici comment faire toujours avec le package *hyperref* :

- `\url{l'url du site}` pour un lien sur l'adresse
- `\href{l'url du site}{mot ou groupe de mot}`pour un lien sur un ou plusieurs mots

Code

```
\url{http://bertrandmasson.free.fr/}
\href{http://bertrandmasson.free.fr/}{Les fiches à Bébert}
```

Résultat

http://bertrandmasson.free.fr/
Les fiches à Bébert

12.3.3 hyperref et KOMA-Script

La façon dont tu ordonnes le package *hyperref* et les autres packages peut avoir une influence sur son bon fonctionnement. D'autre part *hyperref* et KOMA-Script ne sont pas encore entièrement compatible et tu peux te trouver avec des renvois décalés, notamment au niveau de la table des matières. Pour éviter les problèmes Markus Kohm a créé le package *scrhack*, qui modifie certaines instructions des packages d'autres auteurs afin d'assurer la compatibilité. Il te faut quand même charger le package hyperref et ses options. Voici un extrait de l'entête de ce manuel comme exemple :

Code

```
\documentclass[11pt]{scrbook}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
...
\usepackage[colorlinks=true]{hyperref}
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% pour rendre compatible %%%
%%%%    hyperref et koma     %%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

\usepackage{scrhack}
\KOMAoptions{hyperref=false}
...
\usepackage[frenchb]{babel}
...
```

12.4 M^lle Michu brouillonne

Si tu utilises beaucoup d'images dans ton document, la compilation va devenir très vite fastidieuse. Pour accélérer le mouvement, il existe l'option *draft* qui passe en mode brouillon. Dans ce mode les images ne sont pas chargées, elles sont remplacées par un cadre vide à leurs dimensions exactes. Tu peux donc vérifier ta mise en page facilement, tu enlèves l'option *draft* quand ton document est prêt.

Code
```
\documentclass[11pt,draft]{scrbook}
```

Attention
Dans le mode *draft* les liens créés par *hyperref* sont inactifs.

12.5 M^lle Michu crée des commandes à arguments

On a vu dans le chapitre 5 « M^lle Michu veut créer ses propres commandes », page 47, comment créer des commandes simples, mais tu peux tout aussi facilement créer des commandes à arguments.

12.5.1 Un seul argument

Code
```
\newcommand{\nomCommande}[nombre d'arguments]{code LaTeX}
```

L'argument est l'objet que tu veux transformer par ta commande. Par exemple *mot* qui est l'argument dans la commande suivante qui le met en gras :

Code
```
\textbf{mot}
```

Voici un exemple d'une commande à deux arguments le premier défini l'angle de rotation, le deuxième l'objet qui subit la rotation :

Code
```
\rotatebox{45}{texte}
```

LaTeX n'accepte que 9 arguments au maximum. Dans la partie *code*, ces arguments sont notés #1, #2,...#9.

Comme exemple, nous allons créer une commande `\imp` (*imp*ortant) pour mettre en relief des mots. Il s'agit d'une commande avec un seul argument :

Code
```
\newcommand{\imp}[1]{\emph{#1}}
```

Tu utilises ta nouvelle commande comme toutes les commandes à argument :

Code
```
\imp{mot important}
```

qui donne

Résultat

mot important

Je te sens pour le moins perplexe M^lle Michu. Tu te dis il est complètement givré le Bébert, il crée une commande qui existe déjà ! Certes je reconnais volontiers une certaine part de folie, mais pas tant que ça sur ce coup là. Avec LaTeX, il faut essayer de toujours séparer le fond de la forme, cela rend ton document plus malléable. Dans un premier temps j'ai décidé que mes mots importants seraient mis en emphase. Donc effectivement les deux commandes `\imp{}` et `\emph{}` sont identiques. Mais finalement dans un deuxième temps, je me suis ravisé et je préférais voir mes mots important en gras. Et hop je change ma commande `\imp{}` :

Code

```
\newcommand{\imp}[1]{\textbf{#1}}
```

Et toi tu fais comment avec tous tes `\emph{}` ?

Et puis non, je vais les mettre en rouge mes mots importants, c'est chouette le rouge :

Code

```
\newcommand{\imp}[1]{\color{red}#1}
```

Et toi qui venais de changer tous tes `\emph{}` en `\textbf{}` !

12.5.2 Plusieurs arguments

Pour étudier une commande à plusieurs arguments, on va reprendre notre exemple de `\imp{}` en couleur. Pourquoi se limiter à la couleur rouge ?

Code

```
\newcommand{\imp}[2]{\color{#1}#2}
```

Maintenant on peut changer de couleur à volonté :

Code

```
\imp{red}{mot important}
```

Résultat

mot important

Et

Code

```
\imp{blue}{mot important}
```

Résultat

mot important

Attention à l'ordre des arguments surtout s'ils sont nombreux :

Code

```
\imp{mot important}{blue}
```

provoque une erreur de compilation.

12.5.3 Avec option

Pour ce type de commande il existe une méthode plus subtile, surtout si tu vas utiliser plus de mots en rouge que dans une autre couleur. Tu te souviens qu'il existe des commandes à options (voir page 17). Et bien on va créer ce type de commande.

Code

```
\newcommand{\nomCommande}[nombre d'arguments][valeur de
l'option]{code LaTeX}
```

Donc pour notre exemple :

Code

```
\newcommand{\imp}[2][red]{\color{#1}#2}
```

Attention

Tu ne peux créer qu'une seule option et c'est toujours l'argument #1

Code

```
\imp{mot important}
```

Résultat

mot important

Et

Code

```
\imp[blue]{mot important}
```

Résultat

mot important

12.5.4 Un truc un peu plus complexe

Voici un exemple un peu plus complexe, reprenant notre minipage en couleur.

Code

```
\newcommand{\boiteJaune}[2]{% cette accolade ouvre la partie code de la commande
\begin{center}
\colorbox{yellow}{
\begin{minipage}{7cm}
\begin{center}
\bfseries #1
\end{center}
\small
#2
\end{minipage}
}% cette accolade ferme la colorbox
\end{center}
}% cette accolade ferme la partie code de la commande
```

Maintenant tu emploies :

```
\boiteJaune{Le titre}{
Encore du texte pour faire du remplissage mais celui-ci est à
l'intérieur de ma minipage. J'ai utilisé une fonte plus petite
(commande small). Voici une note de bas de page\footnote{La note
de bas de page qui est numéroté avec des lettres à l'intérieur de la
minipage}

On peut utiliser des sauts paragraphes.

Une liste :
\begin{itemize}
\item premier item ;
\item deuxième item ;
\item le dernier.
\end{itemize}
Une autre note de bas de page\footnote{La deuxième note}. Et enfin
un petit pingouin \includegraphics[width=1cm]{tux}.
}% cette accolade ferme la partie code de la commande
```

Le titre

Encore du texte pour faire du remplissage mais celui-ci est à l'intérieur de ma minipage. J'ai utilisé une fonte plus petite (commande small). Voici une note de bas de page [a]
On peut utiliser des sauts paragraphes.
Une liste :
– premier item ;
– deuxième item ;
– le dernier.
Une autre note de bas de page [b]. Et enfin un petit pingouin .

a. La note de bas de page qui est numéroté avec des lettres à l'intérieur de la minipage
b. La deuxième note

En recopiant le code et en changeant la couleur de la boite (et le nom de la commande) tu peux te créer plein de boites de couleurs différentes. Tu peux également utiliser les options pour changer la couleur :

```
\newcommand{\boite}[3][cyan]{% cette accolade ouvre la partie code de la commande
\begin{center}
\colorbox{#1}{
\begin{minipage}{7cm}
\begin{center}
\bfseries #2
\end{center}
\small
#3
\end{minipage}
}% cette accolade ferme la colorbox
\end{center}
}% cette accolade ferme la partie code de la commande
```

```
\boite[red!25]{Le titre}{
Encore du texte pour faire du remplissage mais celui-ci est à
l'intérieur de ma minipage. J'ai utilisé une fonte plus petite
(commande small).}
```

Le titre

Encore du texte pour faire du remplissage mais celui-ci est à l'intérieur de ma minipage. J'ai utilisé une fonte plus petite (commande small).

Tu peux choisir d'utiliser l'option pour la taille de la boite :

```
\newcommand{\boite}[4][7]{% cette accolade ouvre la partie code de la commande
\begin{center}
\colorbox{#2}{
\begin{minipage}{#1cm}
\begin{center}
\bfseries #3
\end{center}
\small
#4
\end{minipage}
}% cette accolade ferme la colorbox
\end{center}
}% cette accolade ferme la partie code de la commande
```

```
\boite[10]{red!25}{Le titre}{
Encore du texte pour faire du remplissage mais celui-ci est à
l'intérieur de ma minipage. J'ai utilisé une fonte plus petite
(commande small).}
```

Le titre

Encore du texte pour faire du remplissage mais celui-ci est à l'intérieur de ma minipage. J'ai utilisé une fonte plus petite (commande small).

Quand précédemment j'ai attiré ton attention sur l'ordre des arguments, cela concernait l'ordre lors de l'utilisation de la commande, pas dans sa création ou tu peux mettre les arguments dans l'ordre que tu veux (c'est d'ailleurs obligatoire ci-dessus car seul le #1 peut être optionnel), tu peux même employer un argument plusieurs fois si nécessaire.

Code

```
\newcommand{\boite}[4][7]{% cette accolade ouvre la partie code de la commande
\begin{center}
\colorbox{#3!10}{
\begin{minipage}{#1cm}
\begin{center}
\color{#3}\bfseries #2
\end{center}
\small
#4
\end{minipage}
}% cette accolade ferme la colorbox
\end{center}
}% cette accolade ferme la partie code de la commande
```

Comme j'ai inversé, pour la démonstration, le numéro d'argument du titre et de la couleur tu dois saisir de cette façon :

Code

```
\boite[10]{Le titre}{red}{
Encore du texte pour faire du remplissage mais celui-ci est à
l'intérieur de ma minipage. J'ai utilisé une fonte plus petite
(commande small).}
```

Le titre

Encore du texte pour faire du remplissage mais celui-ci est à l'intérieur de ma minipage. J'ai utilisé une fonte plus petite (commande small).

On va arrêter là pour l'étude des commandes. Si j'ai insisté lourdement sur leur utilisation c'est que leur rôle est très important. Tout d'abord elles vont te faciliter la saisie et ensuite en séparant le fond et la forme, elles vont te permettent de modifier très facilement ton document, afin de l'adapter à différents supports (impression ou lecture sur écran, document en noir et blanc ou en couleur, ...)

12.6 M^lle Michu et les césures

Chaque langue a ses règles, en français on ne coupe pas un mot sur la dernière syllabe muette, par exemple césure ne sera jamais coupé comme ceci : césu- re. Ces règles sont prises en charge par l'option *french* du package *babel*.

Ceci peut entraîner des défauts dans la mise en page. Il arrive parfois que LaTeX ne parvienne pas, en raison des règles de césure, à couper un mot en fin de ligne et que celui-ci déborde dans la marge. Dans ce cas tu peux forcer la césure en employant `\-` de cette façon :

Code

```
césu\-re
```

ce qui va permettre la césure de « césure » sur la dernière syllabe muette.

Si au contraire tu ne veux pas que *french* coupe un mot, il suffit de le placer dans une boite comme ceci : `\mbox{césure}`. Cette méthode est très utile pour éviter qu'un numéro de téléphone ne soit coupé par LaTeX :

Code

```
\mbox{01 02 03 04 05}
```

L'inconvénient c'est que la portée des deux commandes que l'on vient de voir est locale. Si tu veux autoriser la césure sur la syllabe muette pour tous les mots « césure » et empêcher toutes césures du mot « énorme », il te faut utiliser la commande `\hyphenation{cé-su-re,énorme}` que tu places dans le préambule de ton document.

Attention

La commande `\hyphenation{césu-re}`, interdit la coupure entre cé et sure, si tu veux conserver la première césure, tu dois écrire `\hyphenation{cé-su-re}`.

12.7 M^lle Michu à l'index

Ton travail nécessite peut être la création d'un index. La première chose à faire est de placer dans ton document des commandes d'indexation. Cette commande est du type `\index{mot à indexer}`. Par exemple pour indexer les mots *moustérien* et *racloir* dans le texte suivant :

Code

```
La découverte d'un nouveau gisement moustérien\index{moustérien}
dans cette région du Balafon Occidental est toujours un évenement.
...
L'outillage moustérien\index{moustérien} du secteur B est dominé par
les racloirs\index{racloir} convexes ...
```

Il existe plusieurs types d'entrée d'index. Les entrées hiérarchiques qui permettent d'avoir des entrées d'index de ce type :

Résultat

Racloir
  concave, 35, 45,68
  convexe, 36, 52
  droit 31

Tu les codes :

Code

```
L'outillage moustérien du secteur b est dominé par les racloirs
convexes\index{racloir!convexe}...
```

Les entrées symboliques permettent de dissocier l'entrée d'index et son ordre de tri. `\index{latex@\LaTeX}` permet de classer LaTeX dans les mots commençant par *la*...

Cette entrée est utile pour classer au bon endroit, les mots ayant une première lettre accentuée : `\index{enorme@énorme}` qui sera classé parmi la lettre e. Tu peux également te servir de cette forme pour une mise en forme particulière de certaines entrées :

Code

```
\index{bebert@\textbf{Bébert}}
```

ajoute, **Bébert** en gras dans l'index, classé à bebert.

Pour la mise en forme du numéro de page (par exemple en italique) tu utilises le pipe (|, Alt Gr 6) :

Code

```
\index{moustérien|textit}
```

Attention

Ici il n'y a pas de \ devant la commande de mise en forme

Maintenant que tu as placé tes entrées d'index, il te faut indiquer deux choses à LaTeX : l'endroit où tu veux voir apparaître ton index (en général à la fin) par la commande `\printindex` et qu'il doit générer un index par `\makeindex`.

Code

```
\makeindex
\begin{document}
Le document avec les entrées d'index...
\printindex
\end{document}
```

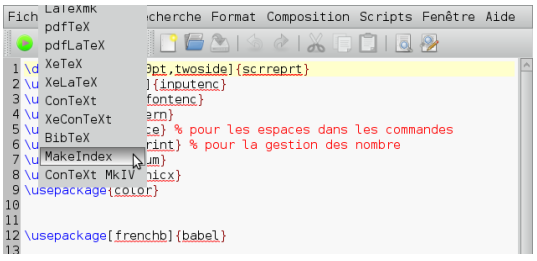


Figure 12.2 : Dans la liste des compilateurs, tu choisis MakeIndex, puis tu lances la compilation par la flèche verte ou Ctrl T.

C'est pas fini il faut également compiler l'index. Dans un premier temps tu compiles ton document normalement avec *pdflatex* comme on l'a vu au chapitre 4, page 39. Cette première compilation produit un fichier `.idx` qui contient la liste brut des index. La seconde étape consiste à compiler le document maître avec le compilateur *makeindex*. L'illustration ci-contre, te montre comment choisir le compilateur sous TeXworks. À l'aide du fichier `.idx` le compilateur *makeindex* produit un fichier `.ind` dans lequel les entrées d'index sont placé dans l'ordre. Une troisième compilation, à nouveau avec *pdflatex* mettra en page ton index.

12.8 Mlle Michu confronté à l'angoisse de la page blanche

Si pour une raison ou une autre tu veuilles insérer une page blanche dans ton document, il te faut connaître deux trois commandes et une explication. Tout d'abord `\newpage` qui permet de passer sur une nouvelle page et la commande `\strut` qui ne fait rien !

Tu te souviens du comportement de LaTeX avec les blancs et les sauts de ligne. Si tu en places plusieurs à la suite LaTeX les ignores (section 3.7, page 35). Et bien pour les pages c'est pareil, plusieurs `\newpage` à la suite seront ignorés et ne conteront que pour un. Tu ne peux donc pas obtenir de page blanche par ce système. Il faut donc trouver un autre truc et c'est là qu'intervient notre commande `\strut` qui ne fait rien. Regarde le code ci-dessous. Les mot1 et mot2 sont séparés par trois lignes. Les mot3 et mot4 sont également séparés par trois lignes mais entre lesquels j'ai intercalé une commande `\strut` qui ne fait rien mais, qui en insérant un objet vide, fait que ces trois lignes, ne sont plus accolées et donc ignorées.

Code

```
mot1



mot2
mot3

\strut

\strut

mot4
```

Résultat

mot1
mot2
mot3

mot4

C'est le même principe pour obtenir une page blanche, j'ajoute simplement `\thispagestyle{empty}` pour ne pas avoir de numéro de page.

Code

```
\newpage
\thispagestyle{empty}
\strut
\newpage
```

12.9 M^lle Michu numérologue

Même s'il n'apparaît pas, dans le document que tu as dans les mains, la page de couverture porte len° 1. Si tu préfères que la numérotation commence à une autre page, au niveau de la table des matière par exemple, il te faut re-numéroter les pages. Cela se fait très facilement en modifiant le compteur de page par la commande `\setcounter{page}{numéro}`. Tu places cette commande à l'endroit où tu veux voir démarrer la numérotation. Tu n'es pas obligé de commencer la numérotation à 1 tu choisis ce que tu veux. Voici le code pour faire démarrer la numérotation des pages au n° 1 au début de la table des matières :

Code

```
...
\begin{document}
\maketitle
\setcounter{page}{1}
\tableofcontents \input{introduction}
\input{historique}
\input{methode}
...
```

Si tu veux que certaines pages aient une numérotation en chiffre romain tu utilises : `\pagenumbering{type}`, où type peux prendre les valeurs suivantes :

alph : pour une numérotation en lettres minuscules : a, b, c, d, e,...

Alph : pour une numérotation en lettres majuscules : A, B, C, D, E,...

arabic : pour une numérotation en chiffres arabes : 1,2 ,3 ,4 ,5,...

roman : pour une numérotation en chiffres romains minuscules : i, ii, iii, iv, v,...

Roman : pour une numérotation en chiffres romains majuscules : I, II, III, IV,...

`\pagenumbering` ré-initialise le compteur de page à 1 automatiquement donc, inutile de jouer avec `\setcounter{page}`. Voici le code pour mettre la numérotation des pages de la table des matière, en chiffres romains et en chiffres arabes le reste du document.

Code

```
...
\begin{document}
\maketitle
\pagenumbering{Roman}
\tableofcontents
\pagenumbering{arabic}
\input{introduction}
\input{historique}
\input{methode}
...
```

12.10 M^lle Michu et les *Overfull* et *Underfull*

12.10.1 *Overfull*

Les *Overfull* sont des erreurs qui interviennent quand LaTeX ne sait pas comment couper une fin de ligne et qu'il laisse dépasser un peu le dernier mot. Voici

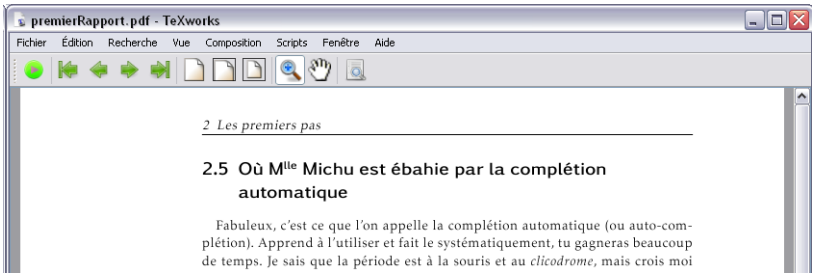
un exemple d'erreur affichée par LaTeX :

```
Overfull \hbox (4.10168pt too wide) in paragraph at line 262--263
```

Tout d'abord pourquoi LaTeX parle de boite horizontale (\hbox) ? Et bien parce que pour LaTeX tout est boite. LaTeX ne compose pas des mots avec des lettres mais manipule des boites qui contiennent des objets. Un peu comme les ouvriers typographes et leurs caractères en plomb. On a donc des boites qui contiennent des lettres (en vert dans le dessin ci-dessous). Puis des boites de mots (en bleu) contenant des boites de lettres et enfin des boites de phrases (en orange) contenant des boites de mots. La ligne rouge représente la ligne de base sur laquelle LaTeX dispose les boites.

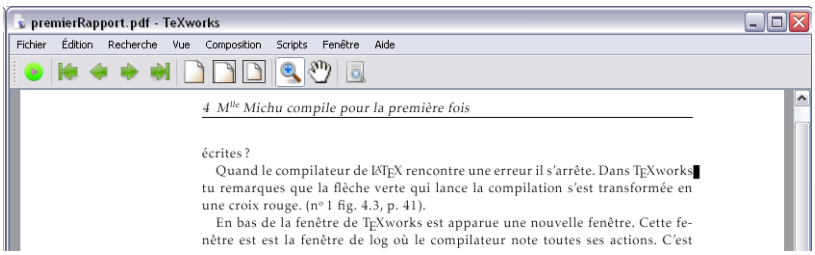
Donc le message d'erreur ci-dessus nous indique que dans la ligne 262, une boite déborde de 4,10168 pt. Si tu es en mode brouillon (*draft*), TeXworks indique les débordements en faisant apparaitre une barre noire (▮) dans la marge. Cette barre à toujours la même largeur, indépendante de la taille du débordement. L'illustration ci-dessous te montre un exemple de débordement. Il correspond à l'erreur mentionnée plus haut.

Pour supprimer ce genre d'erreur il faut indiquer à LaTeX qu'il peut couper le mot auto-complétion entre *com* et *plétion* de cette façon auto-com\-plétion (voir la section 12.6, page 125 sur les césures).



Tu n'es pas obligé de corriger toutes ces erreurs. Dans l'exemple ci-après, LaTeX nous indique : `Overfull \hbox (0.25597pt too wide) ...` .

Le débordement est très minime et je n'ai pas envie de couper TeXworks, donc je l'ai laissé (tu peux le constater page 46) .



Tu peux voir apparaître ces *overfull* au niveau d'illustrations ou de tableaux, dans ce cas il te suffit de diminuer la taille de l'image ou d'une colonne.

12.10.2 *underfull*

Quand LaTeX assemble ses boites de lettres, de mots et de phrases, il attribue une valeur de laideur à la ligne (badness), de 0 pour une ligne parfaite à 10 000 pour la plus laide. Cela lui permet, pour simplifier, de trouver en fonction des possibilités à sa disposition, le meilleur arrangement. Donc quand il n'est pas satisfait de son travail, il te prévient en indiquant le niveau de laideur de la ligne. Dans ce document, en raison de mes images qui sont à cheval dans le texte et la marge, LaTeX affiche de nombreux *underfull*. Par exemple pour la page 9, reproduite ci-dessous, LaTeX trouve la ligne, commençant par « Sans entrer dans le détail... », trop lâche et lui attribue une valeur de laideur de 1975. `Underfull \hbox (badness 1975) in paragraph at line 137--138`

c:\texlive\2010\bin\win32.

Tu dois trouver le chemin indiqué dans la fenêtre n° 1 fig 1.3. Si ce n'est pas le cas tu modifie en cliquant sur le signe plus pour ajouter un chemin et sur le signe moins pour supprimer un chemin erroné.

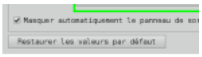
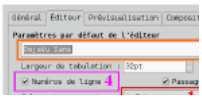


FIGURE 1.3 : Le men
tion ».

Sans entrer dans le détail, il faut que tu saches qu'il existe plusieurs compilateurs, en fonction du type de document que tu veux obtenir, par exemple le compilateur *latex* qui produit un fichier au format `.dvi` (le format d'origine des documents TeX). Le but étant de réaliser un fichier pdf tu vas utiliser *pdflatex* comme compilateur. Tout en bas de la fenêtre, tu choisis pdflatex dans le menu déroulant (n° 2 fig 1.3).

Maintenant tu ouvres l'onglet « Éditeur » et tu vas régler la coloration syntaxique. Tu choisis LaTeX dans le menu déroulant n° 1 de l'image 1.4. Ceci permet d'avoir un source (le document `.tex` s'appelle un source) coloriés ou les commandes LaTeX apparaissent en couleur, pour bien les repérer dans le document.



Conclusion

Il est bien évident que l'on est loin d'avoir fait le tour des potentialités de LaTeX, mais le but de ce document est de te mettre le pied à l'étrier. J'espère que ce tutoriel aura aplanie tes premières difficultés et t'aura donné le goût d'utiliser ce merveilleux logiciel.

Index

\', 15
\,, 33
—, 33
..., 15, 31

\addchap{}, 29
\addpart{}, 29
\addsec{}, 29
\addtocounter, 30
annexe, 103
appeler un fichier, 25
\appendix , 103
article, 12
article (classe), 11
\author{}, 106
auto-complétion, 20–22

babel, 23
\begin{}
    center, 17
    document, 24
\bigskip, 107
book (classe), 11
brouillon, 119
\bsc{}, 34

center, 17
centrer, 17
chapitre, 27
\chapter{}, 27
classe
    article, 11
    book, 11
    KOMA-Script, 11
    report, 11
    scrartcl, 11, 12
    scrbook, 11, 12
    scrreprt, 11, 12, 20
\color{}, 18, 19, 113
\colorbox{}{}, 115
commande, 12, 15
    à argument, 17, 119
    contraindre, 19
    creer, 119
    déclaration, 18
    environnement, 17
    sans argument, 15
complétion automatique, 20–22
compteur, 30
Computer Modern, 21
couleur, 113
    feuille, 113
    fond du texte, 115
    intensité, 115
    texte, 113
couleur (écrire en), 18, 19
couleurs prédéfinies
    dvipsnames, 113
    svgnames, 113
    x11names, 113

déclaration, 18, 19
\date{}, 107
division (du texte), voir sectionnement
\documentclass{}, 11, 20
\documentclass, 20
\dots, 15, 31
draft, 119
dvipsnames, 113

éditeur, 6
\emph{}, 34
emphase, 34
empty, 105
encodage, 21
entête, 105
environnement, 17, 19
espace, 35
    fine, 33
    fine insécable, 23, 33
    insécable, 23
espace vertical, 107

feuille
    mettre en couleur, 113
\fg, 15
fichier
    appeler, 25
    inserrer, 25
    .lof, 105
    .toc, 105
fonte
    changer de, 34
    gras, 34
    italique, 34
    médium, 34
    machine à écrire, 34
    penchée, 34
    petites capitales, 34
    roman, 34
    sans serif, 34
fontec
    \usepackage[T1]{fontenc}, 21
\footnote{}, 31
fourier, 22
french, 23

gras, 34
gras (écrire en), 17, 19, 20
guillemet, 15

headings, 105
\headings{}
    big, 27
    medium, 27
    small, 27
helvet, 23

\includegraphics{}, 17
index, 125
\index, 125
\input{}, 25
inputenc
    \usepackage[utf8]{inputenc}, 21
italique, 34
italique (écrire en), 17

KOMA-Script, 11
\KOMAoptions{chapterprefix}, 103
kpfonts, 23

\large, 18
\large, 18

LaTeX
    distribution, 5, 6
liste des figures, 105
\listeoffigures, 105
listes des tableaux, 105
\listoftables, 105
livre, 12
lmodern, 21

machine à écrire, 34
\makeindex, 126
\maketitle, 106
mathpazo, 23
mathptmx, 23
\medskip, 107
minipage, 111
    en couleur, 116
mode brouillon, 119

\command, 119
\no, 15
nombre (écriture des), 33
note en bas de page, 31
\numprint{}, 33

\oe, 15, 16
\fg, 31
\og, 15, 31
overfull, 128

page
    format, 20
\pagecolor{}, 113
\pagestyle{}, 105
\par, 34, 35
\paragraph{}, 27
paragraphe, 35
    changer de, 34, 35
\part{}, 27
penchée (écriture), 34
petites capitales, 34
pied de page, 105
plain, 105
police de caractère, 22
    kpfonts, 23
police de caractères
    Computer Modern, 21
    fourier, 23
    helvet, 23
    lmodern, 21
    mathpazo, 23

mathptmx, 23
préambule, 23
\printindex, 126
profondeur de la numérotation, 27, 29

rapport, 12
recto-verso, 20
report (classe), 11
roman, 34
\rotatebox{}{}, 17

sans serif, 34
scrartcl, 11, 12
scrbook, 11, 12
scrreprt, 11, 12, 20
\section{}, 27
sectionnement, 26
profondeur de la numérotation, 27, 29
sans numéro, 29
taille des titres, 27
titre trop long, 28
titres avec note de bas de page, 28
titres courts, 28
titres sur deux lignes, 28
\setcounter, 30
\smallskip, 107
\subparagraph{}, 27
\subsection{}, 27
\subsubsection{}, 27
\subtitle{}, 106
svgnames, 113

table des matières, 105
titre trop long, 28
\tableofcontents, 105
TEXLive, 6
\textbf{}, 17, 19, 20
\textcolor{}{}, 113
texte
fond en couleur, 115
mettre en couleur, 113
taille, 20
\textesc{}, 34
\textit{}, 17
TEXworks, 6
complétion automatique, 22
configuration, 9
chemin de la TexLive, 9
coloration syntaxique, 9
compilateur, 9
correction orthographique, 9
encodage, 9
numérotation des lignes, 10
nouveau fichier, 26
raccourci, 22
sauvegarde, 26
utilisation, 8
TEXworks
makeindex, 126
\thispagestyle{}, 105
\tiny, 18
\tiny, 18
tiret, 33
\title{}, 106
titre (fabriquer), 106
.lof, 105
.toc, 105
trois petits points, 15
twoside, 21

underfull, 130
unité (écriture des), 33
utf8, 21

\vspace{}, 107

x11names, 113
xcolor, 113