

# Premiers pas avec $\text{\LaTeX} 2_{\epsilon}$ sous Windows XP : installation et compilation

Benoît-Xavier LAMBERT

19 septembre 2008

*Version 2.1.3*

Ce document simpliste a pour but de vous aider dans vos premiers pas avec  $\text{\LaTeX} 2_{\epsilon}$ , sur un ordinateur équipé de Windows XP. Ce document explique comment installer  $\text{\LaTeX}$ , éditer un document basique avec ou sans logiciel d'édition « orienté  $\text{\LaTeX}$  ». Quelques exemples sont là pour montrer une partie de la puissance et de la rigueur typographique de  $\text{\LaTeX}$ ...

En guise de premier exemple, voici un sudoku, qu'il est très facile de générer avec  $\text{\LaTeX}$ . Pareil pour en éditer sa solution.

		9					6	4
4								
1			3	6			7	2
		4	6					9
			9		3			
2					5	4		
9	2			5	7			8
								5
3	4					6		

*Une grille de difficulté moyenne*

On remarque que la qualité typographique est au rendez-vous, non ? Pour celles et ceux que ça intéresse, la solution du sudoku se trouve à la page suivante.

Bon amusement avec  $\text{\LaTeX} 2_{\epsilon}$

---

# Table des matières

---

<b>1</b>	<b>Téléchargement et installation</b>	<b>1</b>
1.1	Téléchargement . . . . .	1
1.2	Installation . . . . .	1
1.3	Qu'avez-vous vraiment installé? . . . . .	2
<b>2</b>	<b>Quelques exemples</b>	<b>2</b>
2.1	L <sup>A</sup> T <sub>E</sub> X pour écrire des mathématiques . . . . .	2
2.2	L <sup>A</sup> T <sub>E</sub> X pour dessiner des graphiques en tout genre . . . . .	3
2.3	L <sup>A</sup> T <sub>E</sub> X pour écrire des parties d'échecs . . . . .	4
2.4	L <sup>A</sup> T <sub>E</sub> X... pour plein d'autres choses encore! . . . . .	5
<b>3</b>	<b>Créer son premier document</b>	<b>5</b>
<b>4</b>	<b>Choisir un environnement pour éditer</b>	<b>6</b>
4.1	Comment faire pour compiler, d'abord? . . . . .	6
4.2	Travailler avec un éditeur « orienté L <sup>A</sup> T <sub>E</sub> X » . . . . .	7
4.2.1	Quel éditeur « orienté L <sup>A</sup> T <sub>E</sub> X » convient le mieux? . . . . .	7
4.3	Travailler en mode « MS-DOS » . . . . .	8
4.3.1	L'« invite de commande », dépassée? . . . . .	8
4.3.2	Compiler avec l'« invite de commande » . . . . .	9
4.4	Les commandes de compilation . . . . .	10
<b>5</b>	<b>Petit résumé pour gens pressés</b>	<b>12</b>

## Solution du sudoku :

		9					6	4
4								
1			3	6			7	2
		4	6					9
			9		3			
2					5	4		
9	2			5	7			8
								5
3	4					6		

*Grille à résoudre*

7	3	9	5	2	1	8	6	4
4	6	2	8	7	9	1	5	3
1	5	8	3	6	4	9	7	2
5	1	4	6	8	2	7	3	9
6	8	7	9	4	3	5	2	1
2	9	3	7	1	5	4	8	6
9	2	6	1	5	7	3	4	8
8	7	1	4	3	6	2	9	5
3	4	5	2	9	8	6	1	7

*Solution de la grille*

---

# 1 Téléchargement et installation

---

## 1.1 Téléchargement

L<sup>A</sup>T<sub>E</sub>X est « traitement de texte » un peu particulier. Sa puissance est (en partie) due à son fonctionnement non WYSIWYG<sup>1</sup>. L<sup>A</sup>T<sub>E</sub>X est gratuit, mais pas seulement : il est surtout libre ! C'est ce qui fait qu'il existe des tas de « distributions », contenant diverses compilations de programmes utiles à la compilation de fichier contenant du code L<sup>A</sup>T<sub>E</sub>X.

La distribution la plus simple, et la plus courante, pour Windows est MiKTeX. Cette distribution est gratuitement téléchargeable à l'adresse suivante :

<http://www.MiKTeX.org>.

La dernière version stable est la 2.7. C'est celle-là qu'il faut télécharger. Pour ce faire, allez dans le menu (à gauche sur l'écran), et cliquez sur

« MiKTeX 2.7 ».

Ici, deux possibilités s'offrent à vous :

- Si vous avez un accès régulier à internet, je vous conseille de télécharger MiKTeX 2.7 en version « basique » : vous aurez alors une distribution MiKTeX complète, mais chaque fois que vous compilerez des documents utilisant des packages « peu courants », vous devrez les télécharger. Pour télécharger cette version :

« Basic MiKTeX 2.7 Installer ».

Cela vous permet aussi d'économiser beaucoup de place sur votre disque dur.

- Sinon (ou si vous n'aimez pas la première solution), vous pouvez télécharger la version

« MiKTeX 2.7 Net Installer ».

Dans ce cas, tous les packages existants à la date de téléchargement (et il y en a beaucoup...) seront aussi présents sur votre disque dur. Attention, ça prend de la place !

Dans ce document, je vais expliquer (mais y-a-t'il quelque chose à expliquer ?) le téléchargement et l'installation de la version « basique ». Je suppose que l'installation de l'autre version suit un processus similaire. Descendez donc l'ascenseur et cliquez sur

« Basic MiKTeX 2.7 Installer »;

et lancez le téléchargement (le lien direct vers le téléchargement se trouve à la fin de ce document).

## 1.2 Installation

Une fois l'exécutable enregistré sur votre disque dur, il vous suffit (de le retrouver et) de double-cliquer dessus. Vous lancez alors l'assistant d'installation qu'il vous suffit de suivre. Rien de particulier<sup>2</sup> pour cela si ce n'est que :

- N'oubliez pas de choisir votre format de papier par défaut ;
- Cochez la case autorisant<sup>3</sup> le téléchargement de packages manquants lorsque vous en avez besoin.

---

<sup>1</sup> « What You See Is What You Get » — « Ce que vous voyez est ce que vous obtiendrez » .

<sup>2</sup> Est-il nécessaire de vous rappeler de lire la licence ?

<sup>3</sup> Et même : imposez-lui de vous en demander l'autorisation en choisissant l'option « Ask me first » .

## 1.3 Qu'avez-vous vraiment installé ?

Vous pouvez chercher après « MiKTeX 2.7 » dans le menu

« Démarrer » — « Tous les programmes ».

Vous verrez, en pointant la souris dessus, qu'il n'y a aucun logiciel permettant d'éditer votre code-source d'installé avec votre distribution. Et pour cause : n'importe quel éditeur de texte<sup>1</sup> vous permet de le faire ; il n'y avait donc pas besoin d'en rajouter un en plus.

Vous avez donc installé tout le matériel nécessaire pour compiler vos documents, ainsi qu'un « previewer » ; dénommé Yap<sup>2</sup>. Ce programme lit les fichiers d'extention « \*.dvi<sup>3</sup> ». C'est avec Yap que vous pourrez visualiser vos documents fraîchement compilés, avant de les compiler une dernière fois au format « \*.pdf », que tout le monde pourra lire.

Vous possédez aussi un gestionnaire de packages qui vous permettra de gérer/télécharger/mettre à jour des packages dont vous pourriez avoir besoin. Vous possédez donc un programme (presque<sup>4</sup>) complet pour travailler avec L<sup>A</sup>T<sub>E</sub>X ; il ne vous reste donc plus qu'à choisir votre façon de travailler. Je propose plusieurs solutions pour choisir un éditeur de texte dans la section 4 (page 6).

---

## 2 Quelques exemples

---

Ce document étant entièrement réalisé avec L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>, j'espère avoir réussi à vous montrer qu'il était possible de réaliser des choses pas trop moches avec cette « bête » là ...

Et pour vous rassurer, sachez que (avec un peu d'habitude) ce n'est ni plus long, ni plus compliqué qu'avec, par exemple, Microsoft Word. Et même, il est IMPOSSIBLE de réaliser ce document juste en utilisant un traitement de texte « WYSIWYG ». J'en veux pour preuve<sup>5</sup> les exemples qui suivent.

### 2.1 L<sup>A</sup>T<sub>E</sub>X pour écrire des mathématiques

Ecrire des mathématiques avec L<sup>A</sup>T<sub>E</sub>X est très simple et très intuitif. Le tout pour un résultat parfait et professionnel. Jugez vous-même :

Soit  $f \in \mathcal{PR} [0, 2\pi]$  une fonction réelle, impaire (voir **Figure 1**, page suivante), définie par :

$$f(t) = 1 \text{ si } t \in ]0, \pi[ \tag{2.1}$$

dont la série de Fourier  $\mathcal{SF}(f)$  vaut :

$$\frac{4}{\pi} \sum_{k=0}^{\infty} \frac{\sin([2k+1]t)}{2k+1} \tag{2.2}$$

---

<sup>1</sup> EDITEUR de texte et non TRAITEMENT de texte (Microsoft Word est un traitement de texte, donc, à proscrire!).

<sup>2</sup> Pour « Yet Another Previewer ».

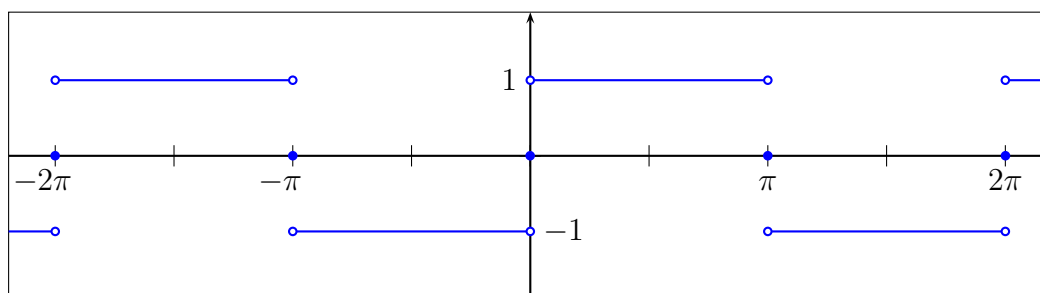
<sup>3</sup> Pour « DeVice Independant ».

<sup>4</sup> Il manque encore les programmes « Ghostscript » et « Ghostview », pour s'amuser avec le format « \*.ps » (utile pour des packages comme « pstricks »). Mais vous n'en êtes pas encore là...

<sup>5</sup> Pas seulement. Vous comprendrez vous-même la puissance de L<sup>A</sup>T<sub>E</sub>X en pratiquant !

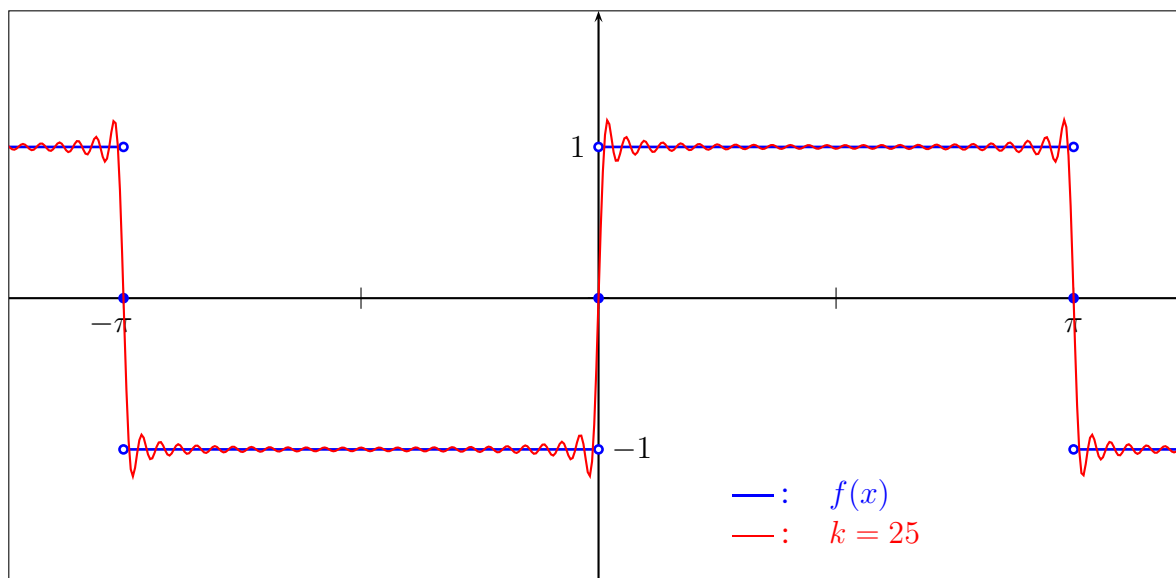
## 2.2 $\LaTeX$ pour dessiner des graphiques en tout genre<sup>1</sup>

Voici trois exemples graphiques : pour commencer, voici le graphe de la fonction 2.1 définie à la page précédente. Rien de bien difficile en soit :



**FIG. 1** – Graphe de la fonction 2.1 ; définie page précédente.

Regardons maintenant à sa série de Fourier : le graphe suivant montre la convergence de la série de de Fourier de la fonction 2.1. Elle permet aussi d’apprécier toute la splendeur du « phénomène de Gibbs<sup>2</sup> » !



**FIG. 2** – Représentation graphique de la série de Fourier 2.2 ; pour  $k = 25$ .

Il va de soi que le graphe a été réalisé avec une boucle et non itéré 25 fois à la main<sup>3</sup>... Un petit package supplémentaire, un petit peu de syntaxe en plus ; et  $\LaTeX$  se tranforme en véritable langage de programmation<sup>4</sup>.

Un dernier petit graphe pour la route : la **Figure 3** de la page suivante montre les étapes de la construction (à la règle et au compas) d’un pentadécagone régulier, à partir d’un pentagone déjà construit. Cette figure est tirée d’un travail de groupe d’Algèbre portant sur les constructions avec règle et compas.

<sup>1</sup> Merci à Marianne LAHAYE qui m’a toujours soutenu quand je me faisais ch\*\*\* (non ; pour être franc, je me suis bien amusé !) à lui réaliser les graphes 1 et 2 pour je-ne-sais quel travail interdisciplinaire...

<sup>2</sup> En gros, il dit (permettez-moi l’expression) que la série de Fourier « perd le nord près des bords »...

<sup>3</sup> Hé non, je ne suis pas (encore) fou à ce point là...

<sup>4</sup> Même si je me fais *descendre* par tout bon informaticien en disant cela...

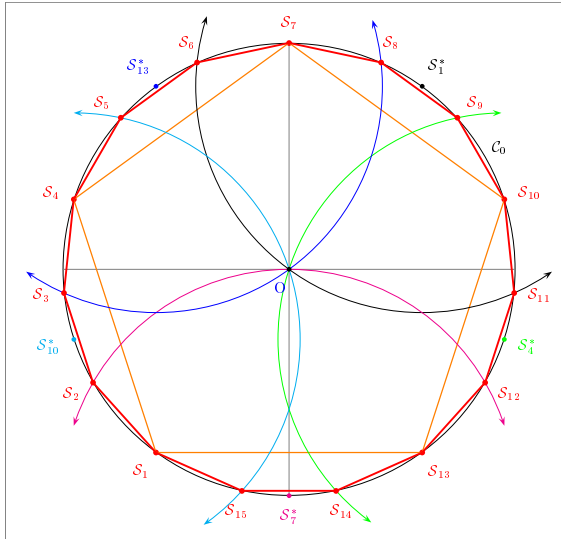


FIG. 3 – Etapes de la construction d'un pentadécagone régulier à partir d'un pentagone régulier.

## 2.3 L<sup>A</sup>T<sub>E</sub>X pour écrire des parties d'échecs

La partie suivante, qui ne devait être qu'une partie amicale entre deux grands joueurs de l'époque, s'est retrouvée dans tous les manuels d'échecs ; grâce à un splendide sacrifice d'attraction...

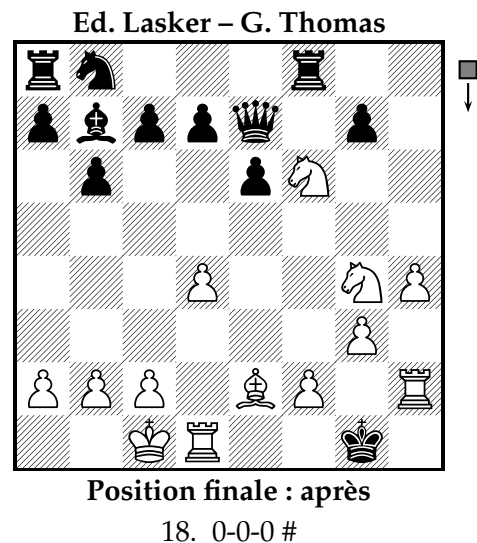
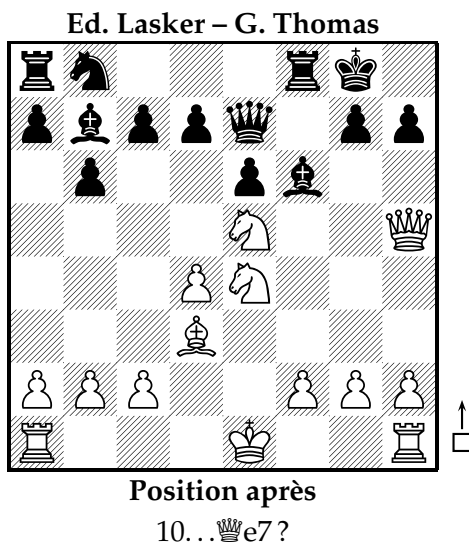
□ Ed. Lasker  
 ■ G. Thomas

Partie amicale (1912)  
 Défense Hollandaise – A83

1. d4 e6 2. ♘f3 f5 3. ♘c3 ♘f6 4. ♙g5 ♙e7 5. ♙×f6 ♙×f6 6. e4 f×e4 7. ♘×e4 b6 8. ♘e5 0–0  
 9. ♙d3 ♙b7 10. ♚h5 ♚e7? (D)

Les noirs voulaient parer la menace sur le Roi. Malheureusement pour eux, ce coup est catastrophique et permet un sacrifice d'une rare violence : les blancs font mat en 8 coups :

11. ♚×h7!! Le sacrifice. 11...♙×h7 12. ♘×f6++ ♙h6 [12...♙h8? 13. ♘g6 #] 13. ♘eg4 ♙g5  
 14. h4+ ♙f4 15. g3+ ♙f3 16. ♙e2+ ♙g2 17. ♚h2+ ♙g1 18. 0–0–0 # (D)



La position finale, magnifique, voit les blancs mater leur adversaire *en roquant* !  
 Il faut remarquer que, à part la variante qui mène au mat en 1 au 12<sup>e</sup> coup, tous les coups noirs étaient forcés !

## 2.4 L<sup>A</sup>T<sub>E</sub>X... pour plein d'autres choses encore !

Evidemment, les capacités de L<sup>A</sup>T<sub>E</sub>X ne s'arrêtent pas là. En fait, elles sont infinies. Les seules limites sont votre imagination et vos limites à vous ! Vous pouvez notamment éditer à la perfection vos partitions de musiques avec L<sup>A</sup>T<sub>E</sub>X ; ou faire des dessins en 3D, ou des diagrammes de Feynmann; ou encore taper des formules chimiques compliquées ou des schémas électroniques (ou des sudokus...) ! Et ce n'est que le côté graphique.

Plus « sérieusement » , L<sup>A</sup>T<sub>E</sub>X vous permet d'éditer n'importe quel document (avec une prédilection pour les documents scientifiques) de manière professionnelle, et ce rien qu'en utilisant les styles par défaut. Et il est bien sûr possible de redéfinir ces styles pour vos propres besoins<sup>1</sup>. C'est d'ailleurs ce que j'ai fait (modestement, mais bon) pour ce document.

De plus, L<sup>A</sup>T<sub>E</sub>X vous force à vous concentrer sur le fond et non sur la forme lorsque vous éditez, ce qui accroît votre efficacité et la qualité des documents réalisés<sup>2</sup>.

### Pourquoi je vous montre tout ça ?

Non non, ce n'est pas pour faire étalage de mes (maigres<sup>3</sup>) connaissances en L<sup>A</sup>T<sub>E</sub>X. On pourrait croire que c'est très difficile à faire, toutes ces figures et diagrammes. En réalité, il existe, dans la plupart des situations, des « packages » fort bien faits qui facilitent grandement ces constructions a priori compliquées, en mettant (librement) à disposition de l'utilisateur intéressé des commandes efficaces. Le tout consiste à trouver ces packages.

Je vous montre donc tout ceci parce que je pense que, comme moi lorsque j'ai découvert L<sup>A</sup>T<sub>E</sub>X, la plupart d'entre vous ne croit pas aux exceptionnelles qualités de ce logiciel libre. J'espère vivement me tromper...

---

## 3 Créer son premier document

---

Après vous avoir mis l'eau à la bouche<sup>4</sup>, il est temps de passer à la phase pratique... A partir d'ici, je vais vous expliquer en détails<sup>5</sup> la création et la compilation d'un document en L<sup>A</sup>T<sub>E</sub>X.

Commençons par créer un premier document en L<sup>A</sup>T<sub>E</sub>X, intitulé `Premier.tex`. Pour ce faire, créez un dossier dans lequel vous placerez tous vos (premiers) documents réalisés avec L<sup>A</sup>T<sub>E</sub>X.

Puis, ouvrez un éditeur de texte, n'importe lequel. Pour votre premier document, n'en installez pas de nouveaux; le bloc note; ou textpad si vous avez, suffisent. La section 4 ( page suivante) vous proposera divers possibilités pour choisir un éditeur de code L<sup>A</sup>T<sub>E</sub>X adapté.

Ouvrez donc votre éditeur de texte, et tapez-y le petit code source suivant :

---

<sup>1</sup> Ou selon les exigences de votre maison d'édition, si un jour vous publiez. Et c'est 1000 fois plus facile, efficace... que, par exemple, dans Microsoft Word ; le tout pour un meilleur rendu, et pour infiniment moins cher !

<sup>2</sup> Et ça, ce n'est pas moi qui le dit !

<sup>3</sup> En effet : je ne possède qu'une relativement bonne connaissance de l'interface de L<sup>A</sup>T<sub>E</sub>X, et de quelques packages qui m'intéressent ; mais ça s'arrête là. Je connais des personnes qui sont capables de faire des choses infiniment plus compliquées ; et que je ne saurai sans doute jamais faire.

<sup>4</sup> J'espère, en tout cas ; parce que ça m'a pris du temps !

<sup>5</sup> D'ailleurs, ça me paraît même trop détaillé...

```

1 \documentclass[a4paper, 12pt]{article} % Préambule.
3 \usepackage[frenchb]{babel}           % Pour gérer la typographie française.
  \usepackage[latin1]{inputenc}        % Pour utiliser les caractères accentués.
5
7 \begin{document}                       % Document.
  Ceci est un document \LaTeX\ !
9 \end{document}

```

LISTING 1 – Code-source de « Premier.tex ».

Il ne vous reste plus qu'à l'enregistrer dans le dossier que vous venez de créer (c'est TRÈS IMPORTANT si vous compilez en mode MS-DOS<sup>1</sup> !). Une petite subtilité s'impose, pour l'enregistrement : il ne faut pas enregistrer le fichier « normalement », c'est-à-dire :

« Fichier » — « Enregistrer sous... » — Premier.tex ;

mais comme ceci (AVEC les guillemets<sup>2</sup> rouges) :

« Fichier » — « Enregistrer sous... » — "Premier.tex".

Je le répète, n'oubliez-pas de taper les guillemets lors de l'enregistrement du document. Une fois le document correctement enregistré, il ne vous reste plus qu'à le compiler !

Et ne vous inquiétez pas si lorsque vous réessayerez d'ouvrir votre code-source, Windows ne reconnaît pas votre fichier : c'est normal, vous en avez changé l'extension. Pour l'ouvrir, il suffit de faire :

« clic droit » sur le fichier — « Ouvrir avec... »

et de choisir votre éditeur de texte !

## 4 Choisir un environnement pour éditer

Vous pouvez donc choisir entre une multitude d'éditeurs de texte, allant du bloc note à emacs, en passant par textpad, notepad++... Vous pouvez aussi choisir d'installer un éditeur « spécifique » à L<sup>A</sup>T<sub>E</sub>X, qui vous donnera plus de « facilités ». Chacun de ces « éditeurs », spécifique ou non, possède ses propres avantages et inconvénients. C'est à vous de voir...

### 4.1 Comment faire pour compiler, d'abord ?

Comme vous l'avez sûrement remarqué dans le menu Démarrer, il n'y a pas trace de compilateur téléchargé avec MiKTeX<sup>3</sup>... Et pour cause, tous les ordinateurs disposent déjà d'un « compilateur » incorporé : avec Windows, il s'agit de l'« invite de commande<sup>4</sup> ».

C'est de cette manière qu'il vous faudra compiler vos documents si vous les éditez avec un éditeur de texte « simple » (Bloc note, notepad, textpad, notepad 2, notepad++...).

Par contre, si vous optez pour la « facilité », à savoir, travailler avec un éditeur plus complet; qui « gère L<sup>A</sup>T<sub>E</sub>X », vous pourrez (la plupart du temps) compiler vos documents directement depuis l'interface du programme que vous choisirez d'utiliser.

<sup>1</sup> Explications à la section 4.3.2, page 9.

<sup>2</sup> Simplement parce que l'extension par défaut des fichiers dans un éditeur de texte est « \*.txt ». Si vous enregistrez le fichier « normalement », l'éditeur de texte renommera votre fichier en « Premier.tex.txt ». Ce sera donc un « bête » fichier de texte et vous ne pourrez pas le compiler avec L<sup>A</sup>T<sub>E</sub>X...

<sup>3</sup> Evidemment qu'un compilateur a été téléchargé. Mais pas d'« interface » permettant de l'utiliser.

<sup>4</sup> C'est l'« équivalent », mais en « moins puissant », de la console sous Linux.



## 4.2 Travailler avec un éditeur « orienté L<sup>A</sup>T<sub>E</sub>X »

C'est le choix de la facilité. L<sup>A</sup>T<sub>E</sub>X étant libre, la plupart des éditeurs qui gèrent L<sup>A</sup>T<sub>E</sub>X le sont aussi ! La plupart du temps, un éditeur « orienté L<sup>A</sup>T<sub>E</sub>X » vous propose un environnement pour taper votre code-source, gère la coloration syntaxique<sup>1</sup>, vous propose toute sorte de menus pour vous faciliter la saisie de commandes, procure des facilités pour gérer la structure du document, ainsi que les références...

Bien entendu, cela ne vous aidera pas à retenir les commandes qu'il vous faudra pourtant mémoriser. Et puis, l'« invite de commande », ce n'est pas si barbare que ça... D'ailleurs, tous mes documents ont été réalisés avec textpad et l'« invite de commande ».

### 4.2.1 Quel éditeur « orienté L<sup>A</sup>T<sub>E</sub>X » convient le mieux ?

C'est une bonne question, ça... Vu que moi, j'utilise l'« invite de commande », je ne me suis pas intéressé beaucoup aux éditeurs « spécifiques ». Cependant, trois peuvent certainement convenir<sup>2</sup> :

- Le complet(?) T<sub>E</sub>XnicCenter;
- l'agréable T<sub>E</sub>XMaker;
- l'indémorable Emacs.

T<sub>E</sub>XnicCenter, je ne l'ai jamais essayé, mais il paraît qu'il est très complet et très populaire. Je pense que ce ne doit pas être un mauvais choix, mais je ne connais pas. Pour le télécharger :

<http://www.toolscenter.org/>.

Il est gratuit, mais en anglais, et vous devez spécifier vous-même le chemin de certains dossiers de votre installation.

T<sub>E</sub>XMaker, quant à lui, possède une interface très agréable, gère les projets, permet l'insertion facile de la plupart des symboles (mathématiques ou autres), est libre, disponible pour Windows, MacOS X et Linux ; et disponible en français à l'adresse suivante :

[http://www.xmlmath.net/texmaker/index\\_fr.html](http://www.xmlmath.net/texmaker/index_fr.html).

Il est personnalisable (mais pas beaucoup), gère la coloration syntaxique (mais pas parfaitement), procure des raccourcis pour les diverses compilations possibles. Pour moi, ce serait mon deuxième choix<sup>3</sup>

Emacs, enfin, possède un environnement « basique » pour éditer avec L<sup>A</sup>T<sub>E</sub>X. Mais il existe une extension à Emacs, appelée « AUC-T<sub>E</sub>X », qui améliore considérablement les capacités d'Emacs pour éditer en L<sup>A</sup>T<sub>E</sub>X ; disponible à cette adresse :

<http://www.gnu.org/software/auctex/>.

Avec cette extension, Emacs colorie les commandes L<sup>A</sup>T<sub>E</sub>X, indente automatiquement le code pour en augmenter la lisibilité, et si L<sup>A</sup>T<sub>E</sub>X détecte des erreurs lors de la compilation, Emacs positionne automatiquement le curseur sur les lignes problématiques<sup>4</sup>. Cela étant, je ne connais pas du tout Emacs.

Il existe aussi beaucoup d'autres éditeurs qui sont à même de gérer L<sup>A</sup>T<sub>E</sub>X, et ce n'est pas parce que je ne les connais pas qu'ils sont mauvais... Par exemple, L<sup>A</sup>T<sub>E</sub>X-Editor (pour Windows), disponible à l'adresse suivante ; et Kile (pour Linux (bibliothèque KDE)).

<http://www.latexeditor.org/index.html>.

<sup>1</sup> Plus ou moins correctement...

<sup>2</sup> Au moins. A cela s'ajoute ceux que je ne connais pas : je ne suis pas logiciel-scient !

<sup>3</sup> Après un éditeur « simple » combiné à l'« invite de commande », bien entendu.

<sup>4</sup> Déjà vu, mais jamais utilisé...

### 4.3 Travailler en mode « MS-DOS<sup>1</sup> »

Vous savez, cette affreuse petite fenêtre noire, dans laquelle on tape des commandes « sauvages » ; dans laquelle vous ne pouvez pas utiliser la souris... Oui oui, c'est bien avec ça que je veux vous faire compiler ! Mais rassurez-vous, il n'y a rien de compliqué : juste quelques commandes de compilation à retenir.

Avant de passer aux choses sérieuses, laissez-moi vous proposer deux éditeurs de textes simples, légers, libres et portables<sup>2</sup> :

Le premier s'appelle « [Notepad2](#) ». Il ne gère pas la coloration synthaxique pour L<sup>A</sup>T<sub>E</sub>X, mais pour d'autres langages (C, C++, Pascal, Java, PHP, XML...), oui. Voici un lien vers le site du développeur :

<http://www.flos-freeware.ch/notepad2.html>.

Ses avantages viennent du fait qu'il est très léger, qu'il possède une interface simple et agréable ; et qu'il existe en français ; il ferait un excellent remplacement du bloc note livré par défaut avec Windows !

Un éditeur un peu plus complet est « [Notepad++](#) ». L'interface est moins agréable (c'est sans doute une question de goût), mais il gère la coloration synthaxique pour L<sup>A</sup>T<sub>E</sub>X, en plus d'autres langages (plus que notepad2). Vous pouvez le trouver à :

<http://notepad-plus.sourceforge.net/fr/site.htm>.

Il est moins léger, mais fonctionne avec des onglets, ce qui peut-être pratique, et est aussi disponible en français.

Je pense qu'ils valent la peine d'être essayés tous les deux. Mais l'absence de coloration synthaxique n'est pas un problème : moi, j'utilise Textpad, qui ne gère pas la coloration synthaxique, et je m'en sors très bien. Le tout (comme pour un « vrai » programme), c'est de BIEN COMMENTER son code, et de l'aérer suffisamment.

#### 4.3.1 L'« invite de commande », dépassée ?

Non, bien sûr. Ni ringarde, ni dépassée. Malgré son aspect rebutant, elle peut se révéler bien pratique pour certaines opérations sur vos disques. Par exemple, j'ai notamment fait un programme batch (suite de commandes DOS) qui me permet de supprimer tous les fichiers auxiliaires créés par la compilation lorsque mon projet est terminé.

Mais vous pouvez aussi vous servir de DOS pour programmer des back-up de votre machine ; le plus simplement du monde. Ou accéder à un répertoire partagé d'une machine distante avec une seule ligne de code ! Ou compiler des programmes écrits dans n'importe quel langage de programmation, tant qu'un compilateur est installé sur votre machine. Mais cela n'a plus rien à voir avec L<sup>A</sup>T<sub>E</sub>X...

L'« invite de commande » va vous permettre, après quelques petites manipulations de départ<sup>3</sup> de compiler vos documents rédigés en L<sup>A</sup>T<sub>E</sub>X simplement avec quelques commandes de compilation ; commandes qu'il vous faudra de toute façon connaître<sup>4</sup>.

<sup>1</sup> MicroSoft - Disk Operating System — Très ancien système d'exploitation de Windows.

<sup>2</sup> On ne sait jamais que ça intéresserait quelqu'un...

<sup>3</sup> Explications à la section 4.3.2 qui suit.

<sup>4</sup> Il vous reste juste à ne pas être déstabilisé par l'apparence un peu rustre de cette petite fenêtre noire...

### 4.3.2 Compiler avec l'« invite de commande »

Pour ouvrir l'invite de commande, vous pouvez cliquer sur :

« Démarrer » — « Exécuter ».

Et dans la fenêtre qui s'ouvre, taper « cmd », et validez. Cette affreuse petite fenêtre noire apparaît alors. Mais par défaut, l'invite de commande s'ouvre dans le dossier

« C:\Documents and Settings\Nom\_du\_PC ».

Vous pourrez donc compiler UNIQUEMENT<sup>1</sup> vos documents qui se trouvent dans ce dossier (et même pas dans les sous-dossiers). Vos documents ne se trouvant vraisemblablement PAS dans ce dossier (sinon, c'est qu'il y a un problème), vous devez changer le chemin de démarrage de l'invite de commande.

Cela peut s'effectuer manuellement via la commande « cd » (pour remonter jusqu'à la racine du disque « C:\ » puis, passer sur le disque « D:\ » , puis, sélectionner le dossier dans lequel est enregistré « Premier.tex »).

L'inconvénient de cette méthode... est qu'il n'y a aucun avantage ! En effet, chaque fois qu'il vous faudra ré-ouvrir l'invite de commande, vous devrez re-changer le chemin de départ. Vous en conviendrez, c'est assez lourd...

HEUREUSEMENT, il existe une alternative à cette méthode barbare : créer un raccourci de l'invite de commande dans le dossier dans lequel est enregistré « PREMIER.TEX », et d'en changer une bonne fois pour toute le chemin de démarrage du raccourci<sup>2</sup> !

Supposons que votre fichier « Premier.tex » soit enregistré à l'emplacement suivant :

« D:\Mes Documents\LaTeX\Premier.tex ».

Voici les deux étapes à respecter pour créer pour compiler décemment avec l'invite de commande. Vous pouvez les répéter pour chaque dossier contenant des codes-source de documents réalisés avec L<sup>A</sup>T<sub>E</sub>X.

#### Créez un raccourci de l'invite de commande :

1. Commencez par cliquer sur :

« Démarrer » — « Tous les programmes... » — « Accessoires »;

2. Cliquez-droit sur

« Invite de commandes »;

3. Sélectionnez ensuite :

« Envoyer vers » — « Bureau (créer un raccourci) ».

4. Il ne vous reste plus qu'à « couper – coller » votre raccourci dans dans le dossier dont le chemin est mentionné plus haut.

Puisque ce raccourci ne vous servira qu'à compiler des documents produits avec L<sup>A</sup>T<sub>E</sub>X, renommez le : appelez-le par exemple « compilateur L<sup>A</sup>T<sub>E</sub>X ».

<sup>1</sup> Ce qui justifie la note 1 du bas de la page 6.

<sup>2</sup> Vous pourrez même « personnaliser » votre fenêtre. Super, non ?

## Changez le chemin de démarrage par défaut du raccourci :

Ce n'est pas parce que vous avez créé un raccourci que le dossier de démarrage par défaut de ce raccourci a changé... Il va donc vous falloir le changer vous même, une bonne fois pour toutes ! Pour ce faire :

1. « Cliquez-droit » sur votre raccourci, et sélectionnez « Propriétés ».
2. Dans la fenêtre qui s'ouvre, sélectionnez l'onglet « Démarrer dans... ».
3. Remplacez le chemin par défaut (« C:\Documents and Settings\Nom\_du\_PC ») par celui spécifié plus haut (et valider) ; à savoir :

`« D:\Mes Documents\LaTeX »`<sup>1</sup>.

4. Double-cliquez sur votre « Compilateur L<sup>A</sup>T<sub>E</sub>X » pour vérifier que le chemin de démarrage a bien été correctement changé.

Vous êtes maintenant prêts à compiler tous vos documents rédigés avec L<sup>A</sup>T<sub>E</sub>X, se trouvant dans votre « dossier spécifique » ; mais uniquement ceux-là.

Lorsque vous travaillerez plus souvent avec L<sup>A</sup>T<sub>E</sub>X, que vous aurez beaucoup de documents, rien ne vous empêche de subdiviser vos dossiers, ou de les éparpiller dans vos documents. La seule restriction est de refaire les 2 étapes sus-mentionnées pour chaque dossier contenant des codes-source à compiler.

## 4.4 Les commandes de compilation

Que vous travaillez sous DOS ou avec un éditeur « orienté L<sup>A</sup>T<sub>E</sub>X », il est toujours utile de connaître les quelques commandes de base<sup>2</sup> pour compiler vos documents.

Les principales commandes de base (celle que vous utiliserez le plus souvent) sont les suivantes (les deux premières seront plus utiles que les deux suivantes pour commencer) :

1. `LaTeX nom_du_fichier(.tex)` ;
2. `pdfLaTeX nom_du_fichier(.tex)` ;
3. `dvipdfm nom_du_fichier(.dvi)` ;
4. `dvips nom_du_fichier(.dvi)`.

Vous n'êtes pas obligés d'écrire les extensions des fichiers lorsque vous compilez. C'est pourquoi ils sont écrits entre parenthèses.

Utiliser la commande « `LaTeX nom_du_fichier(.tex)` » permet de créer, à partir d'un code-source écrit avec L<sup>A</sup>T<sub>E</sub>X, un fichier « `nom_du_fichier.dvi` ». Vous pouvez ouvrir ce nouveau fichier simplement en double-cliquant dessus. Vous l'ouvrirez alors dans votre previewer « Yap ». « `dvi` » signifiant « DeVice Independant », l'aperçu que vous verrez dans « Yap » sera EXACTEMENT ce que vous imprimerez, quel que soit l'imprimante utilisée.

C'est la commande la plus utile. Elle fonctionnera toujours, sauf si vous incluez dans votre document des images d'un format autre que « `*.eps` »<sup>3</sup> (c'est-à-dire, les formats les plus courants tels que « `jpg` » ; « `gif` » ; « `bmp` » ...).

<sup>1</sup> Si vous avez choisi un autre dossier et que vous ne voyez pas apparaître le chemin complet vers ce dossier dans la barre des titres, vous pouvez cliquer sur « Outil » — « Options des dossiers... » — « Affichage », et cocher la 3<sup>e</sup> case : « afficher le chemin complet dans la barre des titres ».

<sup>2</sup> Qui peuvent légèrement varier si l'on est sous Linux ou sous Windows.

<sup>3</sup> Et sauf si vous faites des erreurs, bien entendu...

La commande « `pdfLaTeX nom_du_fichier(.tex)` » permet, quant à elle, de compiler directement votre code-source au format « `*.pdf` », format que tout le monde pourra lire. Cette commande n'est pas utilisable si vous incluez des images au format « `*.eps` » dans votre document.

Comme vous transformerez tous vos codes-source au format « `*.pdf` », on peut se demander à quoi sert la première commande...

Eh bien d'abord, vous compilez vos codes-source en « `*.pdf` » uniquement parce que vous êtes certains que, de cette façon, tout le monde pourra lire vos documents. Mais vous n'êtes pas obligés : vous pourriez les compiler simplement en « `*.dvi` », ou encore, au format « `*.ps` ». Par exemple, les 3 figures de la section 2.2 (voir page 3) ont été générées chacune séparément (avec  $\text{\LaTeX}$ , bien entendu) au format « `*.dvi` », puis « `*.ps` ». De là, en utilisant le programme « `Ghostview` », transformées au format « `*.eps` » ; et enfin, insérées dans le présent document.

De plus, lorsque vous éditez un document d'au moins quelques pages, vous aurez besoin de le compiler beaucoup de fois avant d'avoir supprimé toutes les erreurs de syntaxe, d'orthographe, de distraction... L'avantage, avec la première commande, c'est que vous pouvez laisser Yap ouvert pendant que vous re-compilez ; alors que vous ne pouvez pas compiler au format « `*.pdf` » un document dont la version « `*.pdf` » est ouverte : vous ne pourrez pas « ré-écrire dessus ». Par contre, lorsque vous aurez corrigé toutes les erreurs de frappe et autres, vous pourrez re-compiler votre document avec la deuxième commande.

La troisième commande (« `dvipdfm nom_du_fichier(.dvi)` ») permet de transformer un document d'extension « `*.dvi` » au format « `*.pdf` » (utile lorsque vous incluez des images au format « `*.eps` » (comme pour ce document) et que, par conséquent, vous ne pouvez pas utiliser la deuxième commande).

Enfin, la commande « `dvips nom_du_fichier(.dvi)` » permet de transformer un fichier d'extension « `*.dvi` » au format « `*.eps` ». Cela est utile lorsque vous voulez (comme expliqué plus haut) générer des images au format « `*.eps` » : vous devez absolument d'abord passer par le format « `*.ps` ».

Quelques « subtilités » quand même, en compilant : parfois, vous aurez besoin de compiler votre document 2 fois de suite (voire 3 fois) avant d'avoir un document correct. Cela se passe si vous utilisez des références croisées, une table des matières...

Si vous incluez un index dans votre document, il vous faudra intercaler une commande générant l'index entre deux compilations successives... Tout ça sont des détails que vous apprendrez en pratiquant.

Lorsque vous compilez avec l'invite de commande<sup>1</sup>, et que le chemin indiqué par l'invite de commande pointe dans le bon répertoire, il vous suffit simplement de taper la commande suivie du nom du document à compiler ; et de valider. Par exemple :

```
D:\Mes Documents\LaTeX>LaTeX Premier
```

En cas d'erreur, la compilation s'arrête à l'endroit où le compilateur pense que l'erreur se trouve. Dans l'invite de commande, vous verrez apparaître le numéro de la ligne où l'erreur s'est produite, puis, la ligne en question jusqu'à l'erreur ; un saut de ligne et la suite de l'erreur. Vous pouvez « sauter » l'erreur en appuyant sur « Enter »<sup>2</sup> ; continuer la compilation en tenant compte de l'erreur, en appuyant sur « x » ; ou arrêter la compilation en appuyant sur « Ctrl+c ».

<sup>1</sup> La casse n'a aucune importance en mode DOS : taper « `LaTeX` » ou « `latex` », par exemple, aura le même effet.

<sup>2</sup> Mais gare au résultat...

Si vous voyez votre invite de commande s'arrêter longtemps sans indiquer d'erreur, c'est probablement parce vous êtes en train de télécharger un package que vous ne possédez pas. C'est pour cette raison aussi que je préfère, lors de l'installation, forcer « MiKTeX » à me demander pour installer de nouveaux packages : de cette façon, je suis au courant de ce qu'il installe sur mon PC.

---

## 5 Petit résumé pour gens pressés<sup>1</sup>

---

Voici, succinctement, les étapes pour installer et la compiler des documents avec L<sup>A</sup>T<sub>E</sub>X, dans le cas d'un éditeur de texte « simple » :

1. Téléchargez le logiciel en utilisant le lien suivant (pour plus de détails, voir la [section 1.1 page 1](#)) :  
<http://miktex.org/2.7/Setup.aspx> .
2. Installez le logiciel en suivant l'assistant d'installation ([section 1.2 page 1](#)) ; je vous conseille d'autoriser « MiKTeX » à vous demander pour télécharger les packages manquants en cas de besoin (est inclus dans ce téléchargement un visualisateur de fichiers « \*.dvi »).
3. Créer un premier document ([section 3 page 5](#)) (avec un éditeur de texte) dans un dossier (spécifique à L<sup>A</sup>T<sub>E</sub>X , de préférence). N'oubliez pas de changer l'extension dudit fichier...
4. Créez un raccourci de l'invite de commande ([section 4.3.2 page 9](#)) ; depuis :  

« Démarrer » — « Tous les programmes... » — « Accessoires »
---

 ;  
vers ce dossier « spécifique » .
5. Modifiez l'emplacement de démarrage de ce raccourci ([section 4.3.2 page 9](#)) : cliquez-droit sur l'icône du raccourci, sélectionnez « Propriétés » , et modifiez le champ « démarrer dans » en y rentrant le chemin complet du « dossier spécifique » .
6. Compilez votre document ([section 4.4 page 10](#)) : Ouvrez votre « compilateur » , et entrez les commandes nécessaires à la compilation :
  - **LaTeX** nom\_du\_fichier(.tex)  
(du format « \*.tex » vers le format « \*.dvi »),
  - **pdfLaTeX** nom\_du\_fichier(.tex)  
(du format « \*.tex » vers le format « \*.pdf »),
  - **dvipdfm** nom\_du\_fichier(.dvi)  
(du format « \*.dvi » vers le format « \*.pdf »),
  - **dvips** nom\_du\_fichier(.dvi)  
(du format « \*.dvi » vers le format « \*.ps »).

Bon amusement<sup>2</sup> avec L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>

---

<sup>1</sup> Et pour les autres aussi, bien entendu !

<sup>2</sup> Oui, encore !