

---

## L'Unité de Recherche Sciences et Technologies de l'Image et des Télécommunications (SETIT)

*en collaboration avec*

### L'INSTITUT SUPÉRIEUR D'INFORMATIQUE ET MULTIMÉDIA DE GABÈS

*Organise:*

**Les 12èmes Journées de l'UR SETIT: Formation  $\LaTeX$**

*Présenté par*

**KABAOU Mohamed Ouais**

---

---

## **LES PREMIERS PAS AVEC $\LaTeX$ : INSTALLATION ET COMPILATION**

---

---

# Préface

Ce document est préparé pour une formation à l'Institut Supérieur d'Informatique et Multimédia de Gabès intitulée "Les 12èmes journées de l'UR SETIT" et organisée par l'Unité de Recherche SETIT. Le poly a pour vocation de s'initier à  $\LaTeX$  sous Windows. En effet, il faut savoir que l'environnement  $\LaTeX$  a énormément évolué depuis peu :  $\TeX$  s'est porté sur tous les OS (Operating System) et n'est plus réservé qu'aux thésards et aux spécialistes. Il y a même des outils de dessin ou de création de courbes sous Java spécialement dédiés à  $\LaTeX$ . Pour ne pas répéter ce que d'autres ont déjà très bien écrit sur le fonctionnement de  $\LaTeX$  (commandes, packages, astuces), nous avons établi une liste assez complète et non exhaustive des documents pdf disponibles sur le net, des sites internet très bien faits, ainsi que la bibliographie détaillée des meilleurs livres traitant du sujet. En effet, notre page d'accueil sur Facebook dédiée spécialement à cet événement comportera tous les liens utiles liés à cet outil de traitement.

**U.R. SETIT**  
**Novembre 2011**

**Kabaou Mohamed Ouweis**

# Liste des tableaux

# Table des figures

|      |  |    |
|------|--|----|
| 3.1  | Premières lignes sur $\text{\LaTeX}$ . . . . .                             | 14 |
| 3.2  | Chapitrage d'un document sous $\text{\LaTeX}$ . . . . .                    | 15 |
| 3.3  | Chapitrage avancé . . . . .  | 15 |
| 3.4  | Saut de numérotation dans un chapitre . . . . .                            | 16 |
| 3.5  | Inclusion d'un fichier dans un document . . . . .                          | 17 |
| 3.6  | Insertion de figures dans un document . . . . .                            | 17 |
| 3.7  | Insertion d'équation dans un paragraphe . . . . .                          | 18 |
| 3.8  | Insertion d'équation dans un environnement spécifié . . . . .              | 18 |
| 3.9  | Principales fonctions mathématiques les plus utilisées . . . . .           | 19 |
| 3.10 | Topologie en anneau . . . . .  | 19 |
| 3.11 | Exemple d'un fichier <code>.bib</code> . . . . .                           | 21 |
| 3.12 | Insertion du fichier <code>.bib</code> dans le <code>.tex</code> . . . . . | 22 |

# Table des matières

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Présentation de L<sup>A</sup>T<sub>E</sub>X</b>                                 | <b>6</b>  |
| 1.1      | Introduction . . . . .   | 6         |
| 1.2      | Historique . . . . .   | 6         |
| 1.3      | Philosophie de L <sup>A</sup> T <sub>E</sub> X . . . . .                           | 7         |
| 1.3.1    | Analogie avec une maison d'édition . . . . .                                       | 7         |
| 1.3.2    | Règles pour une bonne visibilité . . . . .   | 8         |
| 1.3.3    | Principaux avantages et inconvénients de L <sup>A</sup> T <sub>E</sub> X . . . . . | 8         |
| 1.4      | Organisation du manuel . . . . .   | 9         |
| <b>2</b> | <b>Installation de L<sup>A</sup>T<sub>E</sub>X sur les OS</b>                      | <b>10</b> |
| 2.1      | Installation . . . . .   | 10        |
| 2.2      | Cycle de production . . . . .  | 11        |
| 2.2.1    | Édition . . . . .  | 11        |
| 2.2.2    | Compilation . . . . .  | 11        |
| 2.2.3    | Visualisation . . . . .  | 12        |
| 2.2.4    | Impression . . . . .   | 12        |
| 2.3      | Installer L <sup>A</sup> T <sub>E</sub> X sous Windows . . . . .                   | 12        |
| 2.3.1    | Installation de la distribution "MikTeX" . . . . .                                 | 12        |
| 2.3.2    | Installation de l'éditeur "Winedt" . . . . .                                       | 13        |
| 2.3.3    | Enregistrement de "Winedt" . . . . .   | 13        |
| <b>3</b> | <b>Mon premier document L<sup>A</sup>T<sub>E</sub>X</b>                            | <b>14</b> |
| 3.1      | Mes premiers pas . . . . .   | 14        |
| 3.2      | Chapitrage . . . . .   | 15        |
| 3.2.1    | Chapitrage avancé . . . . .  | 15        |
| 3.3      | L'inclusion de fichiers . . . . .  | 16        |
| 3.4      | Inclusion d'images . . . . .   | 17        |
| 3.4.1    | inclusion flottante . . . . .  | 17        |
| 3.5      | Inclusion d'équations mathématiques . . . . .                                      | 18        |
| 3.5.1    | Mode texte . . . . .   | 18        |
| 3.5.2    | Environnement équation . . . . .   | 18        |

|       |  |    |
|-------|--|----|
| 3.5.3 | Fonctions mathématiques couramment utilisées . . . . | 19 |
| 3.6   | Inclusion des tableaux . . . . .                     | 19 |
| 3.7   | Bibliographie avec BibTEX . . . . .                  | 20 |
| 3.7.1 | Fonctionnement . . . . .                             | 20 |
| 3.7.2 | Ecrire un fichier .bib . . . . .                     | 20 |
| 3.7.3 | Inclure le fichier .bib dans le .tex . . . . .       | 21 |
| 3.8   | Conclusion . . . . .                                 | 22 |

# Chapitre 1

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

### 1.1 Introduction

L<sup>A</sup>T<sub>E</sub>X (prononcer Latèk) est un éditeur de textes spécialement adapté pour rédiger des textes en mathématiques créé par Donald E. Knuth dans les années 80. Maintenant, à l'exception de quelques très rares livres écrits avec Word, qui est un traitement de textes dit "WYSWYG" (what you see is what you get), tous les autres ouvrages du supérieur sont écrits en L<sup>A</sup>T<sub>E</sub>X. Cet éditeur est plus complexe, car le texte a besoin d'être écrit dans une syntaxe particulière, puis compilé. Principal avantage : un texte aura toujours la même mise en page quel que soit le format d'impression (A3, A4, ...) et surtout la structure sera intacte, alors qu'avec un traitement de textes classique il faut refaire entièrement la mise en page. En mathématiques par exemple, il offre une quantité incroyable de symboles et connaît leurs conventions de mise en page (taille, fonte, espacement, etc). Ne parlons même pas de la césure française ou de la numérotation automatique des équations que Word ignore totalement. Enfin, cerise sur le gâteau, le texte pourra être compilé au format .pdf utilisé par Adobe et lu par le logiciel "Acrobat Reader" que tout internaute qualifié connaît. Mais le format par défaut est .DVI (pour device independant) que toute la communauté scientifique de maths et de physique utilise ; en particulier les revues spécialisées exigent en général ce format pour une publication.

### 1.2 Historique

T<sub>E</sub>X est un programme écrit par Donald E. Knuth. Il est conçu pour la composition de textes et d'équations mathématiques. Knuth a commencé le développement de T<sub>E</sub>X en 1977 parce qu'il était frustré par l'édition ses

articles, publiés par l’American Mathematical Society. Il avait arrêté de soumettre des articles vers 1974 parce que le résultat final était trop pénible à regarder.  $\text{T}_{\text{E}}\text{X}$ , tel que nous l’utilisons aujourd’hui, est sorti en 1982 et a été amélioré progressivement. Ces dernières années  $\text{T}_{\text{E}}\text{X}$ a atteint une grande stabilité. Aujourd’hui Knuth affirme qu’il n’y a virtuellement plus de bugs . Le numéro de version de  $\text{T}_{\text{E}}\text{X}$ tend vers  $\pi$  et est actuellement 3,14159

$\text{\LaTeX}$ est un ensemble de macros qui permettent à un auteur de mettre en page son travail avec la meilleure qualité typographique en utilisant un format professionnel pré-défini.  $\text{\LaTeX}$ a été écrit par Leslie Lamport. Il utilise  $\text{T}_{\text{E}}\text{X}$ comme outil de mise en page.

## 1.3 Philosophie de $\text{\LaTeX}$

### 1.3.1 Analogie avec une maison d’édition

Pour publier un texte, un auteur confie son manuscrit à une maison d’édition. L’éditeur décide alors de la mise en page du document (largeur des colonnes, polices de caractères, présentation des en-têtes,...). L’éditeur note ses instructions sur le manuscrit et le passe à un technicien typographe qui réalise la mise en page en suivant ses instructions. Un éditeur humain essaie de comprendre ce que l’auteur veut mettre en valeur et décide de la présentation en fonction de son expérience professionnelle et du contenu du manuscrit. Dans un environnement  $\text{\LaTeX}$ , celui-ci joue le rôle de l’éditeur et utilise  $\text{T}_{\text{E}}\text{X}$  comme typographe pour la composition. Mais  $\text{LATEX}$  n’est qu’un programme et a donc besoin de plus de directives. L’auteur doit en particulier lui fournir la structure logique de son document. Cette information est insérée dans le texte sous la forme de commandes  $\text{\LaTeX}$ . Cette approche est totalement différente de l’approche WYSIWYG utilisée par les traitements de texte modernes tels que Microsoft Word, Corel WordPerfect, ou Open Office. Avec ces programmes, l’auteur définit la mise en page du document de manière interactive pendant la saisie du texte. Tout au long de cette opération, il voit à l’écran à quoi ressemblera le document final une fois imprimé. Avec  $\text{\LaTeX}$ , il n’est normalement impossible de voir le résultat final durant la saisie du texte (Sauf avec le programme LyX <sup>1</sup>. Mais celui-ci peut être pré-visualisé après traitement du fichier par  $\text{LATEX}$ . Des corrections peuvent alors être apportées avant d’envoyer la version définitive vers l’imprimante.)

---

<sup>1</sup>dit aussi Latex



### 1.3.2 Règles pour une bonne visibilité

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

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

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

### 1.3.3 Principaux avantages et inconvénients de $\LaTeX$

#### Avantages

1. Mise en page professionnelle qui donne aux documents l'air de sortir de l'atelier d'un imprimeur, la composition des formules mathématiques se fait de manière pratique,
2. Il suffit de connaître quelques commandes de base pour décrire la structure logique du document. Il n'est pas nécessaire de se préoccuper de la mise en page,
3. Des structures complexes telles que des notes de bas de page, des renvois, la table des matières ou les références bibliographiques sont produites facilement,
4. Pour la plupart des tâches de la typographie qui ne sont pas directement gérées par  $\LaTeX$ , il existe des extensions gratuites. Par exemple pour inclure des figures PostScript ou pour formater une bibliographie selon un standard précis,
5.  $\LaTeX$  encourage les auteurs à écrire des documents bien structurés, parce que c'est ainsi qu'il fonctionne (en décrivant la structure) ;  $\TeX$ , l'outil de formatage de  $\LaTeX$ , est réellement portable et gratuit. Ainsi il est disponible sur quasiment toutes les machines existantes.

## Inconvénients

1.  $\text{\LaTeX}$  encourage les auteurs à écrire des documents bien structurés, parce que c'est ainsi qu'il fonctionne (en décrivant la structure);  $\text{\TeX}$ , l'outil de formatage de  $\text{\LaTeX}$ , est réellement portable et gratuit. Ainsi il est disponible sur quasiment toutes les machines existantes,
2. Écrire des documents mal organisés et mal structurés est très difficile.

## 1.4 Organisation du manuel

Ce manuel est une introduction au traitement de texte à  $\text{\LaTeX}$ ; il ne s'agit pas d'un manuel de référence, mais il a pour but de donner les bases pour utiliser  $\text{\LaTeX}$  et si possible d'y prendre goût. Ainsi trouvera-t-on les informations nécessaires pour commencer en  $\text{\LaTeX}$  et quelques conseils sur la rédaction des documents. Pour votre confort, nous avons eu l'idée lumineuse de diviser ce manuel en chapitres. La première partie présente les bases de  $\text{\LaTeX}$  :

**Présentation de  $\text{\LaTeX}$**  expose les concepts fondamentaux de  $\text{\LaTeX}$ , décrit un bref historique et définit la philosophie de cet outil de traitement de texte;

**Installation de  $\text{\LaTeX}$**  présente les étapes permettant d'installer une distribution correspondant à votre système d'exploitation (MacOS, Windows, Unix).

# Chapitre 2

## Installation de L<sup>A</sup>T<sub>E</sub>X sur les OS

### 2.1 Installation

Afin d'utiliser L<sup>A</sup>T<sub>E</sub>X il vous faudra installer une distribution correspondant à votre système d'exploitation. Les distributions fournissent des programmes permettant d'automatiser la configuration et l'installation de L<sup>A</sup>T<sub>E</sub>X, T<sub>E</sub>X et tous les utilitaires connexes.

**Sous Unix :** on trouve encore la distribution teTeX bien que son développement ait été stoppé en 2006. Aujourd'hui on installe généralement la TEXLive ([www.tug.org/texlive](http://www.tug.org/texlive)) sur un système Unix ;

**Sous MacOS :** la distribution de référence est MacTeX ([www.tug.org/mactex](http://www.tug.org/mactex)) ;

**Sous Windows :** le plus simple est sans doute de choisir proTEXt([tug.org/protext](http://tug.org/protext)) qui installe la distribution MiKTeX([miktex.org](http://miktex.org)) et quelques outils de développement dont un programme de visualisation de fichiers au format PostScript (gsview).

Pour faire du L<sup>A</sup>T<sub>E</sub>X, il faut trois choses :

1. une distribution L<sup>A</sup>T<sub>E</sub>X, qui comporte l'ensemble des compilateurs, visualiseurs, polices, etc... nécessaire au fonctionnement de L<sup>A</sup>T<sub>E</sub>X,
2. un éditeur de texte sachant faire du texte brut (plain text),
3. un visualiseur Postscript (type ghostview) et/ou PDF (Adobe Acrobat Reader )

Vous apprendrez également bien assez vite que la production d'un document avec L<sup>A</sup>T<sub>E</sub>X consiste à traduire (on dit aussi compiler ) un source Ū

donc créé par un éditeur de texte en un format destiné à l’affichage ou à l’impression. Il existe donc, plus ou moins intégrés aux distributions, des outils célèbres pour la visualisation des différents fichiers résultants de la compilation :

**Format DVI :** `xdvi`, `kdvi` sous Unix et `yap` sous Windows font partie des programmes permettant de visualiser le résultat de la compilation d’un fichier  $\text{\LaTeX}$  ;

**Format Postscript :** la suite `ghostscript` disponible sous des noms qui peuvent varier selon la plateforme, permet de visualiser des fichiers au format PostScript ;

**Format PDF :** mis à part le célèbre `acrobat reader`, il existe sous Unix des utilitaires permettant de visualiser le format Pdf : `xpdf`, `evince`, ...

## 2.2 Cycle de production

Même si  $\text{\LaTeX}$ n’est pas à proprement dit un langage de programmation compilé, on peut malgré tout faire une analogie entre le cycle de production d’un document  $\text{\LaTeX}$ et le cycle édition-compilation-exécution d’un développement de logiciel avec un langage de programmation classique.

### 2.2.1 Édition

Un document source  $\text{\LaTeX}$ est un fichier texte<sup>1</sup>. Ainsi la manipulation d’un fichier  $\text{\LaTeX}$ ne demande pas de logiciel particulier, si ce n’est un éditeur de texte classique. Donc, pour manipuler un document  $\text{\LaTeX}$  : nom de fichier.tex devrait suffire pour entrer dans ce monde qu’est la saisie d’un document  $\text{\LaTeX}$ . Sous Windows, on s’équipera d’un éditeur de texte de son choix. Notez qu’il est recommandé de donner l’extension `.tex` aux sources  $\text{\LaTeX}$ .

### 2.2.2 Compilation

La compilation génère un jour ou l’autre des erreurs. En tout cas, après suppression des erreurs de compilation, on obtient un fichier portant l’extension "dvi" pour device independant. Ce qui signifie que le fichier contient des informations indépendantes du périphérique de sortie (écran, imprimantes,

---

<sup>1</sup>C’est-à-dire un fichier ne contenant que le code des caractères qui le composent

...). Ce fichier de type binaire contenant une image  $\dot{z}$  du document portable sur tout système  $\text{\TeX}$  quel que soit le système d'exploitation. Il existe ensuite des programmes permettant soit :

- de visualiser le document :  $\text{dvi} \rightarrow \text{bitmap \acute{e}cran}$  ;
- de l'imprimer :  $\text{dvi} \rightarrow \text{langage imprimante}$  ;
- de le convertir :  $\text{dvi} \rightarrow \text{fichier PostScript}$ .

### 2.2.3 Visualisation

La visualisation s'effectue simplement après compilation sans erreur, grâce au programme "xdvi". Il s'agit d'un logiciel tournant sous X Window, très intuitif, qui donne un aperçu très lisible du document. La distribution  $\text{teTeX}$  pour Windows propose un visualiseur nommé yap<sup>2</sup>.

### 2.2.4 Impression

Pour imprimer un document, on utilise le programme dvips (nom de fichier.dvi). Il est aussi possible de générer un fichier PostScript en redirigeant la sortie de dvips "nom de fichier.dvi" au "nom de fichier.ps". Le fichier nom de fichier.ps est un fichier ASCII pur contenant des commandes PostScript.

## 2.3 Installer $\text{\LaTeX}$ sous Windows

### 2.3.1 Installation de la distribution "MikTeX"

Une  $\dot{z}$  distribution est un ensemble de micro-programmes réunis autour d'un seul processus d'installation. Pour Windows, il en existe plusieurs. Une des plus répandues est la distribution existante sur le CD qu'on a préparé. Pour l'installer, il vous suffit de suivre les étapes suivantes :

1. Ouvrir le dossier Latex contenu dans le CD dénommé Les 12èmes journées de l'UR SETIT ;
2. Ouvrir le dossier MiKTeX<sup>3</sup> et installer le en cliquant sur le fichier "setup-2.4.1705".
3. Une boîte de dialogue s'ouvre puis cliquer sur suivant jusqu'à ce qu'une fenêtre de progression de l'état d'installation apparait. L'installation peut prendre plusieurs minutes, veuillez donc vous patientez,

---

<sup>2</sup>yet another previewer, soit encore un visualiseur

<sup>3</sup>distribution TeX libre pour Windows visant à fournir un environnement TeX/LaTeX complet et prêt à utiliser

### 2.3.2 Installation de l'éditeur "Winedt"

Un éditeur est un logiciel qui aide à entrer du code  $\LaTeX$ . Là aussi, il en existe plusieurs. La plupart sont gratuits, tous ont des avantages. Lors de ce séminaire, nous utiliserons l'éditeur Winedt. Pour l'installer, suivre les étapes suivantes :

1. Retourner au dossier parent Latex, et ouvrir le dossier Winedt<sup>4</sup> et cliquer sur le fichier d'exécution "winedt54",
2. Cliquer "Next" puis "I accept the agreement" puis "Next" jusqu'à ce qu'une fenêtre d'installation s'ouvre puis cliquer sur "Install",
3. Une fois Miktex et Winedt ont été installés, dans le menu Démarrer, cliquer sur l'icône Winedit, l'éditeur s'ouvrira en conséquence,

### 2.3.3 Enregistrement de "Winedt"

Le logiciel Winedt nécessite un enregistrement pour vous permettre dans un premier temps de travailler non pas avec une version d'évaluation, mais avec une version complète.

Pour enregistrer votre version Winedt, vous devez disposer d'une clé. En effet, dans le CD fourni, vous disposez déjà d'un "keygen". Pour enregistrer votre logiciel, il suffit de suivre les étapes suivantes :

1. Dans le même dossier parent "Winedt", vous trouverez un fichier zipper sous le nom de "Winedt 5.4 Keygen". Ce dernier vous permet d'enregistrer la version installée,
2. Cliquer sur le fichier keygen.rar, puis sur le fichier "wet54802.exe",
3. Faire copier le nom "Hard Wisdom" et le code "1135362106278309830",
4. Dans la barre d'outils de Winedit, aller à "Help", puis "Registration". Une fenêtre s'ouvre dans laquelle vous devez faire entrer le nom et le code dans vous disposez,
5. Enfin vous devez refermer Winedit, faire reculer la date de votre PC puis réouvrir encore une fois Winedt.
6. Félicitations, commencer à créer votre premier document  $\LaTeX$  et dénommé le "Mon premier document sur  $\LaTeX$ ".

---

<sup>4</sup>éditeur de texte TeX/LaTeX similaire à Kile. Il fonctionne sous Windows

# Chapitre 3

## Mon premier document L<sup>A</sup>T<sub>E</sub>X

### 3.1 Mes premiers pas

Comme dans tous les manuels, dans toutes les documentations, il faut bien commencer par le commencement. Ouvrez un éditeur de document et tapez ceci :

```
1
2 \documentclass[12pt , a4paper]{report}
3 \usepackage[french]{babel}
4 \usepackage[T1]{fontenc}
5
6 \title{Mon premier document}
7 \author{Votre nom}
8 \date{\today}
9
10 \begin{document}
11
12 \maketitle
13
14 Hello World
15
16 \end{document}
17
```

FIG. 3.1 – Premières lignes sur L<sup>A</sup>T<sub>E</sub>X

Enregistrez votre document en extension `.tex` et compilez une fois avec votre compileur. La sortie se fait en `dvi`.

## 3.2 Chapitrage

Rajoutez maintenant du texte entre `\begin{document}` et `\end{document}` :

```
1 \chapter{chapitre}
2 J'aime \TeX{}
3
4 \section{sous chapitre}
5
6 \begin{center}
7 Le texte du sous chapitre en centré
8 \end{center}
9
10 \subsection{sous partie}
11 Et bien voila comment marche le chapitrage
12
13
```

FIG. 3.2 – Chapitrage d'un document sous L<sup>A</sup>T<sub>E</sub>X

Compilez et regardez ce que ça donne. Changez `{report}` par `{article}` en enlevant `\chapter{chapitre}`<sup>1</sup> et voyez la différence de mise en page.

### 3.2.1 Chapitrage avancé

L<sup>A</sup>T<sub>E</sub>X numérote automatiquement les chapitres et sous chapitre. Mais comment faire pour, par exemple, insérer une "Introduction" sans qu'elle soit numérotée mais qu'elle apparaisse dans la table des matières? Essayez la commande suivante :

```
1 \chapter*{Introduction}
2 \addcontentsline{toc}{chapter}{Introduction}
```

FIG. 3.3 – Chapitrage avancé

---

<sup>1</sup>la classe article ne gère pas les chapter



Le `*` permet de faire sauter la numérotation, tandis que le `\addcontentsline` permet de faire apparaître dans le `toc`<sup>2</sup> au niveau chapter l'Introduction. On peut s'amuser à tester.

```
1 \section*{Introduction}
2 \addcontentsline{tof}{section}{Introduction}
```

FIG. 3.4 – Saut de numérotation dans un chapitre

### 3.3 L'inclusion de fichiers

La commande `\input{toto}` permet d'importer le fichier "toto.tex" comme s'il était dans le texte<sup>3</sup>, cependant le fichier "toto.tex" doit être absolument dans le même dossier que votre fichier-mère.

`\input` est une commande précieuse. En effet, elle permet de simplifier l'écriture de gros fichiers  $\text{\LaTeX}$  en les découpant en petits bouts<sup>4</sup>.

#### Exemple pratique

Vous décidez de faire un rapport de thèse. Pour cela vous allez créer :

1. Votre fichier principale thèse.tex dans `d:\thèse\` avec dedans les appels aux packages classiques,
2. Vos fichiers `intro.tex`, `chap1.tex`, `chap2.tex`, `chap3.tex`,
3. Vos figures `fig1.tex` `fig2.tex` dans `d:\elec\fig`,
4. Votre tableau `tab.tex` : `\elec\tab`.

Il en vous reste plus qu'à ajouter dans le fichier `elec.tex` entre `\begin{document}` et `\end{document}` :

---

<sup>2</sup>table of contents, table des matières ; tof=table of figures etc.

<sup>3</sup>Cela sert également lorsque l'on veut importer ses propres bibliothèques de macros.

<sup>4</sup>je vous laisse imaginer le foutoir que ça aurait été de faire ce poly sans la commande `input`

```

1 \input{intro}
2 \input{chap1}
3 \input{chap2}
4 \input{chap3}
5 \input{fig/fig1}
6 \input{fig/fig2}
7 \input{tab/tab}

```

FIG. 3.5 – Inclusion d'un fichier dans un document

## 3.4 Inclusion d'images

### 3.4.1 inclusion flottante

Différents environnements sont proposés dans L<sup>A</sup>T<sub>E</sub>X, mais on peut les diviser en deux groupes : les statiques et les flottants. La différence : on peut y inclure des références et des étiquettes (captions). Cela s'applique particulièrement aux images et aux tableaux.

Voici le code pour l'inclusion flottantes :

```

1
2 \begin{figure}[H]
3   \includegraphics[width=votre taille]{nom_du_fichier}
4   \caption{l'image représente ...}
5   \label{mon_image}
6 \end{figure}

```

FIG. 3.6 – Insertion de figures dans un document

**Nota :** le [H] désigne l'endroit où vous voulez inclure l'image. Il y a plusieurs possibilités donc [h] pour le mettre en haut de la page, [b] pour le bas etc.

Dans cette étape, il est préférable d'installer un logiciel dédié aux illustrations tel que Adobe Illustrator CS dont vous avez une version sur le CD. En effet, ce logiciel vous aidera à faire créer des illustrations vectorielles et les sauvegarder sous format "eps" (Encapsulated PostScript).

## 3.5 Inclusion d'équations mathématiques

Comme nous l'avons déjà dit en introduction,  $\text{\TeX}$  a été conçu pour produire des documents avec un beau environnement mathématique. Il est donc très aisé d'afficher des formules mathématiques d'une beauté inégalée.

### 3.5.1 Mode texte

Essayez d'écrire les formules ci-dessous et regardez attentivement la différence entre elles.

```
1 On sait que :  $\sum_{i=1}^n x_i = x_1 + x_2 + \dots + x_n$ , cela
2 nous permet de conclure que...
```

FIG. 3.7 – Insertion d'équation dans un paragraphe

### 3.5.2 Environnement équation

Pour mieux saisir et afin de simplifier l'insertion des équations mathématiques dans vos documents sur  $\text{\LaTeX}$ , il vaut mieux installer l'éditeur d'équation *Mathtype*, éditer l'équation puis faire un copier coller sur *Winedt* entre  $\backslash\text{begin}\{equation\}$  et  $\backslash\text{end}\{equation\}$ .

```
1 \begin{equation}
2   x = \sqrt{\delta_\Omega} \\
3   y = \int_0^x f(y) \, dy
4 \end{equation}
```

FIG. 3.8 – Insertion d'équation dans un environnement spécifié

### 3.5.3 Fonctions mathématiques couramment utilisées

|   |  |
|---|--|
| <code>x^2</code>                          | $x^2$                                  |
| <code>x_2</code>                          | $x_2$                                  |
| <code>\sqrt{x}</code>                     | $\sqrt{x}$                             |
| <code>\int_0^x f(y)\,dy</code>            | $\int_0^x f(y) dy$                     |
| <code>\frac{x}{y}</code>                  | $\frac{x}{y}$                          |
| <code>\sum_{i=1}^n x</code>               | $\sum_{i=1}^n x$                       |
| <code>\lim_{n \to +\infty} x_n = 0</code> | $\lim_{n \rightarrow +\infty} x_n = 0$ |

FIG. 3.9 – Principales fonctions mathématiques les plus utilisées

## 3.6 Inclusion des tableaux

Dans la barre d'outils de WinEdt, aller à "Insert", puis "Tabular (n\*m)". Cliquer la dessus et choisir le nombre de lignes et le nombre de colonnes désiré.

```

1 \begin{table}[H]
2 \begin{tabular}....
3 ...votre tableau...
4 \end{tabular}
5 \caption{vos commentaires}
6 \label{votre reference}
7 \end{table}

```

FIG. 3.10 – Topologie en anneau

## 3.7 Bibliographie avec BibTEX

Un des problèmes que trouvent les étudiants est bien sûr la partie Bibliographique (ou Webographique). Ne serait-ce que pour la présenter correctement. Avec L<sup>A</sup>T<sub>E</sub>X plus besoin de se creuser la tête, tout est automatique. Que ce soit les références croisées dans le document ou l'impression de la bibliographie correctement typographiée.

### 3.7.1 Fonctionnement

Nous avons vu que la source d'un document TEX est un .tex qui peut être édité par n'importe quel éditeur de texte (notepad) ou un éditeur plus spécialisé. Le compilateur permet ensuite de mouliner le .tex en .dvi ou .ps ou .pdf, suivant le compilateur utilisé. Cependant pour créer une table des matières, une table des figures, ou un table de tableaux, nous avons besoin de compiler deux fois. Tout simplement parce qu'à la première compilation le compilateur génère un fichier .dvi, .ps ou .pdf avec un fichier .toc, .tof ou .tot comprenant respectivement les informations du chapitrage, des images ou des tables. En le recompilant une seconde fois le compilateur prend les fichiers .tex, .toc, .tof, .tot et les compile en un fichier de sortie .dvi, .ps, .pdf comprenant la table des matières, la table des figures et la table des tableaux.

Le procédé est le même pour la bibliographie, à une différence près :

- nous devons nous même écrire le fichier .bib (logique),
- compiler le fichier .bib avec le compilateur bibtex,
- indiquer dans le fichier .tex que nous avons une bibliographie (localiser le fichier .bib),
- compiler le fichier .tex qui prendra en compte notre .bib compilé

### 3.7.2 Ecrire un fichier .bib

Il faut comprendre le .bib comme un base de donnée :

```

1
2 @BOOK{latex1,
3
4 author="\textsc{Rolland}",
5
6 title="LaTeX par lapratique CD-Rom inclus",
7
8 publisher="\textbf{O'Reilly}",
9
10 year="1999",
11
12 note="Des méthodes pour préparer des documents à partir de commandes
13 macros, en utilisant LaTeX. Résumé des commandes en annexe.", }

```

FIG. 3.11 – Exemple d'un fichier .bib

- latex1 représente l'ID que nous avons choisi
- BOOK précise le type de document (ici un livre). Mais il y en a plusieurs au total :
  - article,
  - book,
  - conference,
  - inproceedings
  - phdthesis,
  - techreport
  - etc...

Avec pour chacun d'eux, des champs spécifiques à entrer ou non. Sauvegardez le fichier .bib, il ne reste plus qu'à le compiler.

### 3.7.3 Inclure le fichier .bib dans le .tex

Pour afficher la bibliographie dans le fichier de sortie, il faut inclure le fichier .bib dans le .tex. Pour cela taper à la fin de votre fichier TEX [1].

```
1 \bibliographystyle{plain}  
2 \bibliography{nom_de_fichier}
```

FIG. 3.12 – Insertion du fichier .bib dans le .tex

## 3.8 Conclusion

À ce stade, vous pouvez réaliser tranquillement un mémoire de mastère, préparer votre TD ou écrire votre thèse. Il faudrait une bonne dizaine de millier de pages pour expliciter toutes les possibilités de  $\text{\LaTeX}$ . Sachez qu'absolument tout est possible avec ce logiciel. Le plus important est d'apprendre à chercher par soi-même ce dont on a besoin à un moment précis. Trouver le bon package, qui vous permettra de faire ce que vous souhaitez. Un conseil : Prenez un bouquin de référence sur  $\text{\LaTeX}$  et l'ayez toujours à côté de vous lorsque vous faites du TEX.

# Bibliographie

- [1] ROLLAND. *LaTeX par lapratique CD-Rom inclus*. **O'Reilly**, 1999. Des méthodes pour préparer des documents à partir de commandes macros, en utilisant LaTeX. Résumé des commandes en annexe.