

# Introduction à L<sup>A</sup>T<sub>E</sub>X

J2M

mars 2007

## Introduction

L<sup>A</sup>T<sub>E</sub>X est un langage qui permet de créer des documents écrits (rapports, livres, articles,..) de haute qualité. Même si sa mise en place peut sembler rebutante au début, L<sup>A</sup>T<sub>E</sub>X permet d'obtenir un meilleur rendu final (en terme de mise en page) que les traitements de texte classiques.

## 1 Création d'un fichier L<sup>A</sup>T<sub>E</sub>X

L<sup>A</sup>T<sub>E</sub>X est un langage compilé (on part d'un fichier source, pour arriver à un fichier cible). Pour créer un fichier en L<sup>A</sup>T<sub>E</sub>X on commence par :

- créer un fichier texte (dont l'extension est .tex).
- On place dans ce fichier le texte que l'on désire afficher dans le document final (que l'on va produire) et on y rajoute des commandes L<sup>A</sup>T<sub>E</sub>X afin de préciser l'apparence qu'aura ce texte dans le document final.
- On compile alors ce fichier source (d'extension .tex) afin de produire le document final en utilisant la commande suivante :

```
latex <nom_du_fichier.tex>
```

On obtient alors un fichier `nom_du_fichier.dvi` dont on peut visualiser le contenu avec `xdvi` par exemple.

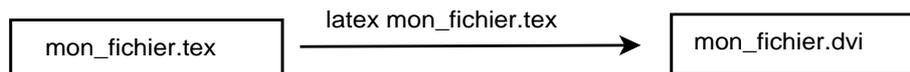


FIG. 1 – Exemple de compilation L<sup>A</sup>T<sub>E</sub>X

## 2 Contenu du fichier .tex

Comme expliqué précédemment, un document .tex contient du texte, ainsi que des commandes L<sup>A</sup>T<sub>E</sub>X qui vont permettre de définir l'apparence de ce texte dans le document final. Une commande L<sup>A</sup>T<sub>E</sub>X est précédée d'un caractère d'échappement (barre oblique inverse "\") afin que le compilateur sache qu'il s'agit d'une commande L<sup>A</sup>T<sub>E</sub>X.

```
\commande [options] {arguments}  
ou encore  
\begin{commande} argument \end{commande}
```

## 2.1 Le préambule

Le fichier .tex commence par un préambule. C'est la partie où l'on indique les packages à utiliser (`\usepackage{nom_du_package}`) et les références à la date, au nom de l'auteur, ainsi que le titre du document. Cette partie se termine avec la balise `\begin{document}`. Pour un exemple de préambule, consultez la figure 2.

C'est ici également que l'on définit le type de document que l'on veut produire (article, report, book, letter, slide), grâce à la commande `\documentclass{type_du_document}`

```
\documentclass[a4paper,10pt]{article}

% Utilisation des accents
\usepackage[latin1]{inputenc}

% Utilisation du français
\usepackage[french]{babel}

% Pour inclure des graphiques
\usepackage{graphicx}

% Pour écrire des formules mathématiques
\usepackage{amsmath,amsthm,amssymb}

% Pour pouvoir écrire sans interprétation de LaTeX
\usepackage{verbatim}

\title{Introduction à LaTeX}
\date{14-11-2005}
\author{J2M}

\begin{document}

% Pour générer la présentation du document
\maketitle

C'est ici que j'écris le texte qui compose mon document.
J'écris en \textbf{gras}, ou en \textit{italique}. C'est comme je veux.

\end{document}
```

FIG. 2 – Exemple de document .tex -

## 2.2 Le corps du document

La deuxième partie, le corps du document est située entre les balises `\begin{document}` et `\end{document}` et contient le texte, les graphiques, etc.. du document final. Pour un exemple de corps de document consultez la figure 2.

On peut noter la présence de la commande `\maketitle` juste au début du corps du document. Sans la présence à cet endroit de cette commande les informations contenues dans le préambule (titre, auteur, date) ne seraient pas affichées dans le document final.

Les lignes qui commencent par `%` sont des lignes de commentaires et ne sont pas interprétées par le compilateur `LATEX` et ne seront pas visibles dans le document final.

### 2.2.1 chapitre, section, ...

Voici les commandes utilisées pour séparer logiquement les différentes parties d'un document :

- `\chapter{nom du chapitre}`
- `\section{nom de la section}`
- `\subsection{nom de la sous-section}`
- `\subsubsection{nom de la sous-sous-section}`
- `\paragraph{nom du paragraphe}`

### 2.2.2 Passages à la ligne

Le passage à la ligne forcé se fait en utilisant le mot-clé `\\`. `LATEX` se chargeant lui-même de la mise en page et de la mise en forme il se peut qu'il coupe un mot en deux dans une phrase trop longue en passant à la ligne suivante. Si l'on désire éviter qu'un mot ou groupe de mot soit automatiquement coupé par `LATEX`, on l'englobera avec la commande `\mbox{texte}`.

### 2.2.3 Gras, italique, ...

Pour modifier les caractéristiques de la police, il faut utiliser les commandes suivantes :

- `\textbf{le texte}` pour mettre en gras
- `\textit{le texte}` pour mettre en italique
- `\texttt{le texte}` pour mettre en police fixe (machine à écrire)
- `\textsc{le texte}` pour utiliser les petites majuscules

### 2.2.4 Listes

Les commandes les plus fréquemment utilisées pour créer des listes sont `itemize` et `enumerate`. On utilise également parfois `description`.

Voici un exemple :

```
\begin{itemize}
\item{Un premier item}
\item{Un autre item}
\item{Mon dernier item}
\end{itemize}

\begin{enumerate}
\item{Un premier item}
```

```
\item{Un autre item}
\item{Mon dernier item}
\end{enumerate}
```

Ce qui produit :

- Un premier item
  - Un autre item
  - Mon dernier item
1. Un premier item
  2. Un autre item
  3. Mon dernier item

### 2.2.5 Mathématiques

Voici les principales fonctions utilisées en mode mathématique. Dans ce cas, il ne faut pas oublier d'ajouter la ligne `\usepackage{amsmath}` dans le préambule. Un exemple d'utilisation de cet environnement est :

```
\begin{math}
a^{i+1}=a.a^i\ \
\sqrt{\frac{a^{i+1}}{a}}=b \quad (a \neq 0)
\end{math}
```

Qui produit :

$$a^{i+1} = a.a^i$$

$$\sqrt{\frac{a^{i+1}}{a}} = b \quad (a \neq 0)$$

remarques :

- `\frac{Dénominateur}{Numérateur}` permet d'écrire une fraction
- `\sqrt` permet de définir une racine
- `\quad` permet d'ajouter un espace, on peut aussi utiliser `\qquad`, `\Quad`, `"\,"`, `\;` ; et d'autres

### 2.2.6 Ne pas mettre en forme du texte que l'on a écrit

Dans un rapport concernant un projet informatique on peut être amené à citer un morceau du code de programme. Les langages de programmation ayant une syntaxe qui leur est propre, on ne veut pas que L<sup>A</sup>T<sub>E</sub>X modifie le code que l'on incorpore dans le document .tex. Il existe deux façons d'introduire ce code source dans le fichier L<sup>A</sup>T<sub>E</sub>X, la première consiste à utiliser l'environnement *verbatim* et à écrire le texte dans celui-ci. La seconde solution consiste à utiliser la commande `\inputverbatim{mon_fichier}`, où `mon_fichier` contient le code à laisser inchangé. Mais dans ce cas, il ne faut pas oublier d'importer le package *verbatim*.

### 3 Exercice

Produire un article de titre "Exercice", daté d'aujourd'hui et à votre nom. Cet article contiendra deux chapitres, correspondant aux paragraphes suivants :

**paragraphe n°1**

– article n°1

1. article n°1

2. article n°2

3. article n°3

**pim** article n°1

**pam** article n°2

**poum** article n°3

– article n°2

– article n°3

**paragraphe n°2**

$$\sqrt{1 + \sqrt{1 + \sqrt{1 + x^k}}}$$