

Introduction à $\text{\LaTeX} 2_{\epsilon}$

Loïc Jezequel

22 septembre 2010

Qu'est-ce que L^AT_EX ?

- Un éditeur de texte. . .

Qu'est-ce que L^AT_EX ?

- Un éditeur de texte. . .
- . . .adapté à la création de documents scientifiques. . .

Qu'est-ce que L^AT_EX ?

- Un éditeur de texte. . .
- . . .adapté à la création de documents scientifiques. . .
- . . .permettant une mise en page professionnelle. . .

Qu'est-ce que L^AT_EX ?

- Un éditeur de texte. . .
- . . .adapté à la création de documents scientifiques. . .
- . . .permettant une mise en page professionnelle. . .
- . . .sans être trop compliqué à manipuler.

Qu'est-ce que L^AT_EX ?

- Un éditeur de texte. . .
- . . . adapté à la création de documents scientifiques. . .
- . . . permettant une mise en page professionnelle. . .
- . . . sans être trop compliqué à manipuler.

page de votre document. Si vous regardez en haut de cette page, vous verrez un résultat possible de l'utilisation de cette extension.

La difficulté principale pour personnaliser les marges et les pieds de page consiste à mettre à jour le nom de la section ou du chapitre en cours utilisé par ces éléments. L^AT_EX réalise cela en deux étapes. Dans la définition des entêtes et pieds de page les commandes `\leftmark` et `\rightmark` sont utilisées pour désigner respectivement les noms du chapitre et de la section courants. La valeur de ces commandes est redéfinie chaque fois qu'un nouveau chapitre ou une nouvelle section commencent.

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

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

La figure 3.1 montre un exemple de configuration de l'extension `amsmath` qui se rapproche de la présentation utilisée pour ce document. La documentation complète de cette extension se trouve à l'adresse mentionnée dans la note de bas de page.

4.5 L'extension verbatim

Plus haut dans ce document, vous avez appris à utiliser l'environnement `verbatim`. Dans cette section vous allez découvrir l'extension `verbatim`. L'extension `verbatim` est une nouvelle implémentation de l'environnement du même nom qui corrige certaines de ses limitations. En soi cela n'est pas spectaculaire, mais ce package s'ajoute de nouvelles fonctionnalités qui justifient que cette extension soit citée ici. L'extension `verbatim` propose le commande :

```
\verbatiminput{nom de fichier}
```

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

Puisque l'extension `verbatim` fait partie de l'ensemble « outils », elle devrait être déjà disponible sur la plupart des systèmes. Pour en savoir plus au sujet de cette extension, reportez-vous à [1].

4.6 Installation d'extensions

La plupart des installations L^AT_EX fournissent en standard un grand nombre d'extensions, mais il arrive que justement celles dont on aurait be-

Chapitre 3

Formules Mathématiques

Vous êtes prêts ? Dans ce chapitre nous allons aborder l'autre majeur de L^AT_EX : la composition de formules mathématiques. Mais attention, ce chapitre ne fait que décrire les commandes de base. Bien que ce qui est expliqué ici soit suffisant pour la plupart des utilisations, ne désespérez pas si vous n'y trouvez pas la solution à votre problème de mise en forme d'une équation mathématique. Il y a de fortes chances pour que la solution se trouve dans l'extension `amsmath` de L^AT_EX.

3.1 L'ensemble `AMS-LATEX`

Si vous souhaitez saisir des textes mathématiques (amusés), vous devriez utiliser `AMS-LATEX`. Le package `AMS-LATEX` est une collection d'extensions et de classes pour la société mathématique. Nous traitons ici principalement de l'extension `amsmath` qui fait partie de ce package. `AMS-LATEX` est produit par l'*American Mathematical Society*¹ et est utilisé extensivement pour la mise en forme de mathématiques. L^AT_EX seul fournirait bien quelques fonctionnalités et environnements basiques, mais ils sont relativement limités (voici, la logique s'appliquerait plutôt dans l'autre sens : `AMS-LATEX` est éliminé !) et parfois incertaines.

`AMS-LATEX` fait partie de la distribution de base et est fourni avec toutes les distributions récentes de L^AT_EX². Dans ce chapitre nous supposons qu'`amsmath` est chargé en préambule, via `\usepackage{amsmath}`.

3.2 Équations simples

Il y a deux façons de mettre en forme des formules mathématiques : à la fin du texte à l'intérieur d'un paragraphe (ajouté en ligne) ou en interrompant

¹ La Société américaine de mathématiques. (AMSI)

² Si la version ne l'a pas, visitez <http://www.ctan.org/required/amstex>

Pourquoi utiliser \LaTeX plutôt qu'un autre éditeur de texte ?

Principaux arguments

- Mise en page automatique à partir de la structure du document.
- Grande facilité d'écriture de formules mathématiques.
- Gestion automatique des références en avant ou en arrière.
- Possibilité de créer simplement des structures complexes : notes de bas de page, table des matières, références bibliographiques. . .

Pourquoi utiliser \LaTeX plutôt qu'un autre éditeur de texte ?

Principaux arguments

- Mise en page automatique à partir de la structure du document.
- Grande facilité d'écriture de formules mathématiques.
- Gestion automatique des références en avant ou en arrière.
- Possibilité de créer simplement des structures complexes : notes de bas de page, table des matières, références bibliographiques. . .

Et en plus

- De par sa façon de fonctionner \LaTeX encourage la création de documents bien structurés.
- \LaTeX est disponible pour quasiment tous les systèmes d'exploitation existants.

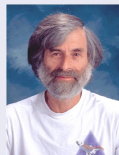
Bref historique.

$\text{T}_{\text{E}}\text{X}$

Composition de textes et d'équations mathématiques. Réponse à la faible qualité visuelle des publications scientifiques de l'époque. Développé par **Donald Knuth** à partir de 1977.



$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$

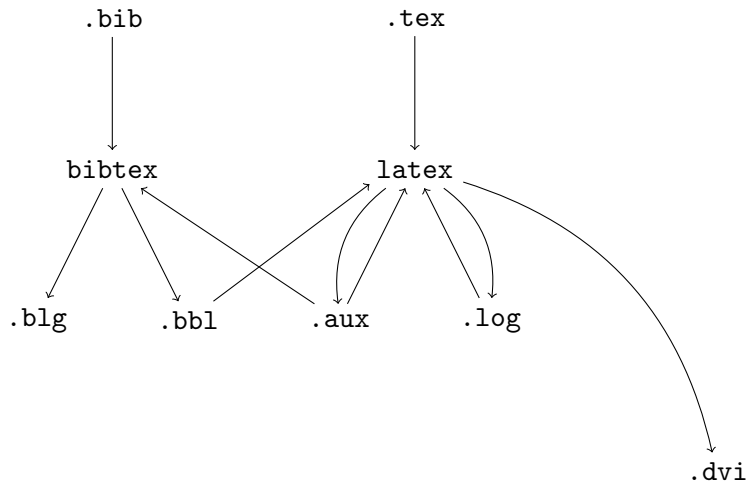


Ensemble de macros permettant de mettre en page des documents. Utilise $\text{T}_{\text{E}}\text{X}$ comme outil de mise en page. Écrit par **Leslie Lamport**.

$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}_{2_{\epsilon}}$

Améliorations de $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ et fusion des différentes variantes existantes en 1994.

Principe de fonctionnement



Plan du cours

- 1 Écrire avec \LaTeX
 - Structure d'un fichier source
 - Étiquettes et références
 - Compilation
- 2 Le corps du document
 - Quelques bases
 - Listes
 - Tableaux
- 3 Formules mathématiques
- 4 Un peu plus loin
 - Objets flottants
 - Bibliographie
 - Extensions
- 5 Macros

Fichier source (1)

Fichier source

Créé avec n'importe quel éditeur de texte (Emacs...), contient le texte du document et les commandes de mise en page.

Commandes

Antislash suivi d'un nom composé uniquement de lettres (avec éventuellement des arguments entre accolades) ou antislash suivi d'exactly un caractère spécial.

Commentaires

Une ligne précédée du signe % n'est pas prise en compte lors de la compilation du document.

Fichier source (2)

Entête

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

Corps

```
\begin{document}
```

```
...
```

```
\end{document}
```

Fichier source (3)

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

Classes

article pour des documents courts, **report** pour des documents plus longs, découpés en chapitres, **book** pour des livres.

Options

Taille de la police : 10pt, 11pt ou 12pt.

Format du papier : a4paper, letterpaper...

Autres options : twocolumn, titlepage, notitlepage...

Fichier source (4)

Structure du document

```
\begin{document}
  \chapter{...}
  \part{...}
  \section{...}
  \subsection{...}
  \subsubsection{...}
  \paragraph{...}
  \subparagraph{...}
  \appendix{...}
\end{document}
```

Les différentes sections sont automatiquement numérotées.

Table des matières

```
\tableofcontents
```

Fichier source (5)

Titre, auteur, date

```
\title{...}
```

```
\author{...}
```

```
\date{...}
```

```
\begin{document}
```

```
\maketitle
```

```
\end{document}
```


Références

Principe

Il est possible de placer des **étiquettes** dans le fichier source et d'y faire référence par la suite. L'ajout d'une section en milieu de document n'entraîne pas la nécessité de modifier les références à la main.

Étiquettes

```
\label{...}
```

Références

```
\ref{...} ou \pageref{...}
```

Compilation

Produire un document en .dvi

```
latex source.tex
```

Produire directement un document en .pdf

```
pdflatex source.tex
```

Attention !

Il faut compiler deux fois le document lorsque l'on utilise des références.
(ou que l'on veut construire une table des matières)

TP

Plan du cours

- 1 Écrire avec \LaTeX
 - Structure d'un fichier source
 - Étiquettes et références
 - Compilation
- 2 Le corps du document
 - Quelques bases
 - Listes
 - Tableaux
- 3 Formules mathématiques
- 4 Un peu plus loin
 - Objets flottants
 - Bibliographie
 - Extensions
- 5 Macros

Quelques bases (1)

Notes ^a

a. de bas de page

```
\footnote{...}
```

Texte en *italique*

```
\emph{...}
```

Texte en **gras**

```
\bf{...}
```

Souligner du texte

```
\underline{...}
```

Quelques bases (2)

Espaces

Basique : quelle que soit la taille du blanc (tabulations, espaces) entre deux mots dans le fichier source il est toujours considéré comme une seule espace.

Insécable : utiliser le caractère "~".

Commande : espace ignorée si pas d'arguments, ajouter un argument vide pour que l'espace soit prise en compte : `\commande{}`.

Paragraphe, sauts de ligne

Un saut de ligne dans le fichier source définit un nouveau paragraphe dans le document produit.

Listes

```
\begin{itemize}  
\item premier point  
\item deuxime point  
\item troisieme point  
\end{itemize}
```

- premier point
- deuxième point
- troisième point

Énumérations

```
\begin{enumerate}  
\item premier point  
\item deuxime point  
\item troisieme point  
\end{enumerate}
```

- ① premier point
- ② deuxième point
- ③ troisième point

Tableaux

L'environnement à utiliser

`\begin{tabular}{description} ... \end{tabular}`, où *description* permet de définir le nombre et le style des colonnes.

Exemple

colonne 1	colonnes 2 et 3	colonne 4
valeur 1	1,2	commentaire 1
valeur 2	345,6	commentaire 2
valeur 3	7,8910111213	commentaire 3
valeur 4	1415,16	commentaire 4

TP

Plan du cours

- 1 Écrire avec \LaTeX
 - Structure d'un fichier source
 - Étiquettes et références
 - Compilation
- 2 Le corps du document
 - Quelques bases
 - Listes
 - Tableaux
- 3 Formules mathématiques
- 4 Un peu plus loin
 - Objets flottants
 - Bibliographie
 - Extensions
- 5 Macros

Inclure des formules dans du texte

Directement incluses dans le texte

`$formule$` ou `\begin{math}formule\end{math}`

Sur des lignes à part

`$$formule$$` ou `\begin{displaymath}formule\end{displaymath}`

Formules importantes

En utilisant l'environnement `equation`.

Autres environnement utiles

`eqnarray` et `eqnarray*`

Écrire des formules (1)

Exposant

a^b pour a^b et $a^{\{bc\}}$ pour a^{bc}

Indice

a_b pour a_b et $a_{\{bc\}}$ pour a_{bc}

Exposant et indice

On peut cumuler exposant et indice, $a^{\{bc\}}_d$ donne a_d^{bc} , ou les imbriquer, $a^{\{bc_d\}}$ donne a^{bc_d}

Attention

$\min_{x \rightarrow \infty}$ s'écrit exactement de la même façon que

$$\min_{x \rightarrow \infty}$$

mais en utilisant un environnement différent.

Écrire des formules (2)

Racines

`\sqrt{x}` donne \sqrt{x} et `\sqrt[n]{x}` donne $\sqrt[n]{x}$

Fractions

`\frac{x}{y}` donne $\frac{x}{y}$

On peut évidemment composer le tout :

$$\sqrt[3]{\frac{\sqrt{x^2 + y + 2}}{a_1 + a_2 + a_3}}$$

Écrire des formules (3)

Vecteurs

`\vec a` donne \vec{a} et `\overrightarrow{ab}` donne \overrightarrow{ab}

Coefficients binomiaux

`\binom{n}{k}` donne $\binom{n}{k}$

Écrire des formules (4)

Parenthèses : taille automatique

On utilise `\left(` et `\right)`, par exemple : $\left(\frac{1}{x^2+1}\right)$

Parenthèses : taille manuelle

On dispose des commandes `\big(`, `\Big(`, `\bigg(` et `\Bigg(`, ce qui donne :

$$\left(\left(\left(\left(\right)\right)\right)\right)$$

Autres délimiteurs

$$\left\{ \left\{ \left\{ \left\{ \left[\left[\left[\left[\right] \right] \right] \right] \right. \right. \right. \right.$$

Écrire des formules (5)

Lettres grecques minuscules

Il suffit de taper le nom de la lettre précédé d'un antislash, par exemple `\gamma` pour γ .

$\alpha, \beta, \gamma, \delta, \epsilon, \zeta, \eta, \theta, \iota, \kappa, \lambda, \mu, \nu, \xi, \omicron, \pi, \rho, \sigma, \tau, \upsilon, \phi, \chi, \psi, \omega$

Lettres grecques majuscules

Même méthode mais avec une majuscule au début du nom, par exemple `\Gamma` pour Γ .

$\Gamma, \Delta, \Theta, \Lambda, \Xi, \Pi, \Sigma, \Upsilon, \Phi, \Psi, \Omega$

Écrire des formules (6)

Matrices

On peut utiliser des tableaux dans les formules (avec l'environnement `array`, qui fonctionne comme `tabular`) ce qui permet de construire, par exemple, des matrices :

$$M = \begin{pmatrix} M_{11} & M_{12} & \dots & M_{1n} \\ M_{21} & M_{22} & \dots & M_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ M_{m1} & M_{m2} & \dots & M_{mn} \end{pmatrix}$$

TP

Plan du cours

- 1 Écrire avec \LaTeX
 - Structure d'un fichier source
 - Étiquettes et références
 - Compilation
- 2 Le corps du document
 - Quelques bases
 - Listes
 - Tableaux
- 3 Formules mathématiques
- 4 Un peu plus loin
 - Objets flottants
 - Bibliographie
 - Extensions
- 5 Macros

Objets flottants (1)

Le principe

Lorsque l'on veut inclure un objet de grande taille (image, tableau, ...) dans un document \LaTeX peut se charger de le positionner automatiquement.

Environnements possibles

- `\begin{figure}[placement]` pour une figure
- `\begin{table}[placement]` pour un tableau

Objets flottants (2)

Placement

h *here*

t *top*

b *bottom*

p *page*

! sans prendre en compte certains paramètres pouvant empêcher le placement

Légende

`\caption{légende}`

Objets flottants (3)

Exemple

```
\begin{table}[htbp]
  \begin{tabular}{ccc}
    case1 & case2 & case3 \\
    \hline
    case4 & case5 & case6
  \end{tabular}
  \caption{1\'egende}\label{tableau1}
\end{table}
```

case1	case2	case3
case4	case5	case6

TABLE: légende

Bibliographie (1)

À la main

Utiliser l'environnement `thebibliography`. Chaque référence est précédée de `\bibitem{étiquette}`. On cite les références grâce à la commande `\cite{étiquette}`.

Exemple

On peut citer la première référence `\cite{ref1}`,
et la deuxième `\cite{ref2}`.

```
\begin{thebibliography}{99}  
\bibitem{ref1} première référence  
\bibitem{ref2} seconde référence  
\end{thebibliography}
```

On peut citer la première
référence [1], et la
deuxième [2].



première référence



seconde référence

Bibliographie (2)

Avec *bibtex*

```
\bibliographystyle{style}  
\bibliography{fichier_biblio}
```

Citer

On utilise toujours la commande `\cite{...}`. Seules les références citées apparaissent dans le document final. On peut cependant utiliser la commande `\nocite{...}`.

Compilation

```
pdflatex fichier.tex  
bibtex fichier  
pdflatex fichier.tex
```


Bibliographie (3)

fichier_biblio.bib

```
@article{ref,  
  author = {l'auteur},  
  title = {le titre},  
  journal = {le nom du journal},  
  volume = {...},  
  number = {...},  
  month = {...},  
  year = {l'annee de parution}  
}
```

Extensions

Ajouter une extension

On utilise la commande `\usepackage [options] {nom_extension}` dans l'entête du fichier source.

Quelques exemples

babel, pgf, tikz, algorithmic, graphicx, amsmath, amssymb, amsthm, fullpage, geometry, subfig, picins, bussproof, ntheorem, fncychapter, fncyhdr, fontenc, inputenc, varioref, url, verbatim, xspace, enumitem, needspace, hyperref, showkeys. . .

Graphicx

Ajouter l'extension

```
\usepackage{graphicx}
```

Inclure une image

```
\includegraphics [options] {image}
```

Options

width, *height* et *angle*

Quels types d'images ?

PDF, JPEG, PNG et TIFF si on compile avec `pdflatex`.

Algorithmic

Ajouter l'extension

```
\usepackage{algorithmic}
```

Exemple

```
\begin{algorithmic}
\IF{...}
  \STATE ...
\ELSE
  \FOR{...}
    \STATE ...
  \ENDFOR
\ENDIF
\end{algorithmic}
```

```
if ... then
...
else
  for ... do
  ...
end for
end if
```

Algorithmes « flotants »

```
\usepackage{algorithm}
\begin{algorithm}...\end{algorithm}
```

TP

Plan du cours

- 1 Écrire avec \LaTeX
 - Structure d'un fichier source
 - Étiquettes et références
 - Compilation
- 2 Le corps du document
 - Quelques bases
 - Listes
 - Tableaux
- 3 Formules mathématiques
- 4 Un peu plus loin
 - Objets flottants
 - Bibliographie
 - Extensions
- 5 Macros

Définir de nouvelles commandes

Sans arguments

```
\newcommand{\maCommande}{...}
```

Avec arguments

```
\newcommand{\maCommande}[nb_args]{...#i...}
```

Redéfinir une commande existante

```
\renewcommand{\maCommande}{...}
```

Utiliser la nouvelle commande

```
\maCommande{...}
```

Définir de nouveaux environnements

Commande à utiliser

```
\newenvironment{monEnvironnement}{début}{fin}
```

Utiliser le nouvel environnement

```
\begin{monEnvironnement}... \end{monEnvironnement}
```


Définir des « théorèmes »

Commande à utiliser

```
\newtheorem{nomCourt}{nomComplet}
```

Utiliser le nouveau théorème

```
\begin{nomCourt}... \end{nomCourt}
```

TP