

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

Sergiu Ivanov

`sergiu.ivanov@univ-evry.fr`

<https://www.ibisc.univ-evry.fr/~sivanov/fr/latex-intro.html>

# Comment prononcer « L<sup>A</sup>T<sub>E</sub>X »

[latɛk]

comme « pastèque »



On peut aussi prononcer [latɛx], où [x] se lit comme le **ch** dans l'allemand Buch.

- ▶ c'est la prononciation de L<sup>A</sup>T<sub>E</sub>X en anglais

Qu'est-ce que  $\text{\LaTeX}$ ??

# Qu'est-ce que $\text{\LaTeX}$ ?

Un langage de balisage (markup)

- ▶ environnements + commandes
  - ▶ un peu comme HTML et CSS

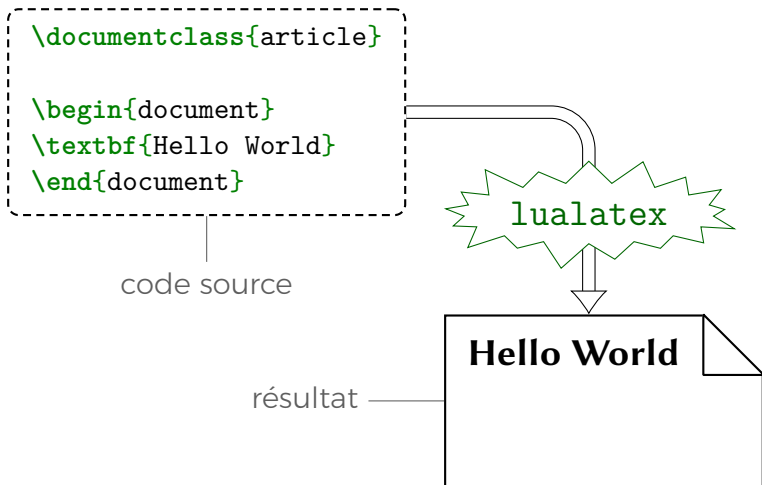
Un langage spécialisé dans la mise en page.

# L<sup>A</sup>T<sub>E</sub>X : la mise en page

Les cas d'usage typiques de L<sup>A</sup>T<sub>E</sub>X :

- ▶ documents techniques/scientifiques
- ▶ présentations techniques/scientifiques
- ▶ affichettes, présentations, lettres, etc.
  - ▶ L<sup>A</sup>T<sub>E</sub>X n'est pas toujours le meilleur outil

# Schéma de fonctionnement



# WYSIWIG et la mise page explicite

WYSIWIG = What You See Is What Your Get

WYSIWIG = l'utilisateur travaille sur la représentation finale du document

- ▶ LibreOffice Writer, Google Docs, Microsoft Word

- + facile à utiliser
- la structure du document n'est pas bien visible
- difficile de changer de format

# Structure et présentation

La philosophie de  $\text{\LaTeX}$  :

1. l'utilisateur définit la **structure** du document
2.  $\text{\LaTeX}$  met en page

L'utilisateur peut ajuster très finement les paramètres de mise en page.



# Comment installer $\text{\LaTeX}$

`https://www.latex-project.org/get/`

Sous Linux  $\text{\LaTeX}$  est souvent disponible via le gestionnaire de paquets/applications.

---

Éditeurs en ligne :

- ▶ `https://www.authorea.com/`
- ▶ `https://www.overleaf.com/`
- ▶ `https://www.sharelatex.com/`
- ▶ etc.

+ faciles à utiliser

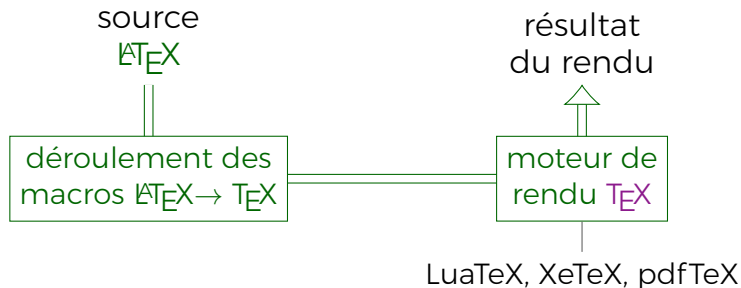
+ écriture collaborative

— éditer dans un navigateur n'est pas pratique

— de gros documents peuvent alourdir l'usage

# $\text{\LaTeX}$ et $\text{\TeX}$

$\text{\LaTeX}$  = une librairie de macros écrits en  $\text{\TeX}$



Les commandes qui effectuent les deux étapes comportent la particule «  $\text{\La}$  » dans leurs noms :

$\text{\texttt{lualatex}}$ ,  $\text{\texttt{xelatex}}$ ,  $\text{\texttt{pdflatex}}$

# LuaTeX ← XeTeX ← pdfTeX

Évolution historique, très grosso modo :

1. pdfTeX : mise en page avancée T<sub>E</sub>X → PDF
2. XeTeX : mise en page avec des caractères Unicode et différentes polices
3. LuaTeX : mise en page avec Unicode et polices, possibilité d'utiliser le langage de scripts Lua pour communiquer avec le moteur de rendu

Consigne : utiliser LuaTeX (la commande `lualatex`).

# LuaTeX ← XeTeX ← pdfTeX

Évolution historique, très grosso modo :

1. pdfTeX : mise en page avancée T<sub>E</sub>X → PDF
2. XeTeX : mise en page avec des caractères Unicode et différentes polices
3. LuaTeX : mise en page avec Unicode et polices, possibilité d'utiliser le langage de scripts Lua pour communiquer avec le moteur de rendu

Consigne : utiliser LuaTeX (la commande `lualatex`).

---

Beaucoup de scientifiques utilisent encore pdfTeX (la commande `pdflatex`).





# Outline

1. Éléments de base
2. Rédaction des documents
3. Tableaux
4. Figures
5. Mathématiques
6. Usages avancés : quelques références


# Outline

1. Éléments de base
2. Rédaction des documents
3. Tableaux
4. Figures
5. Mathématiques
6. Usages avancés : quelques références



## Références générales $\text{\LaTeX}$ :

-  <https://en.wikibooks.org/wiki/LaTeX>
-  <https://fr.wikibooks.org/wiki/LaTeX>
-  <http://smail.yolasite.com/resources/Petit%20manuel%20d'utilisation%20de%20LaTeX.pdf>
-  <https://www.latex-project.org/get/>

## Références TikZ :

-  <http://mirrors.ctan.org/graphics/pgf/base/doc/pgfmanual.pdf>

## Références Beamer :

-  <https://en.wikibooks.org/wiki/LaTeX/Presentations>
-  [https://fr.wikibooks.org/wiki/LaTeX/Faire\\_des\\_pr%C3%A9sentations](https://fr.wikibooks.org/wiki/LaTeX/Faire_des_pr%C3%A9sentations)

# Espaces et retours à la ligne

Code source	Résultat
$\geq 1$ espaces	1 espace
1 retour à la ligne	1 espace
$> 1$ retour à la ligne	1 nouveau paragraphe

Le caractère « `~` » **force** une espace.

Les caractères « `\\` » **forcent** un retour à la ligne.



# Espaces et retours à la ligne

Ces caractères ont des fonctions spéciales et ne seront pas affichés tels quels.

\$	\\$
%	\%
^	\^{}{}
&	\&
_	\_
{	\{
}	\}
~	\~{}{}
\	\textbackslash{}



commandes pour afficher ces caractères spéciaux

# Commandes

arguments optionnels

- ▶ peuvent être omis

`\commande` [`opt1, opt2, ...`] `{arg1}{arg2}...`

The diagram illustrates the syntax of a LaTeX command. It shows the command name, optional arguments in square brackets, and mandatory arguments in curly braces. Arrows indicate the relationship between the text and its labels: a downward arrow from 'arguments optionnels' points to the optional arguments; an upward arrow from 'nom de la commande' points to the command name; and an upward arrow from 'arguments obligatoires' points to the mandatory arguments.

nom de la commande

arguments obligatoires

- ▶ caractères alphabétiques  
uniquement

# Environnements

nom de l'environnement

arguments obligatoires

`\begin{environnement}[opt1,opt2,...]{arg1}{arg2}...`

contenu

`\end{environnement}`


arguments optionnels

# Commentaires

*% Ce texte est un commentaire. Il n'apparaîtra pas  
% dans le résultat final.*

---

Les commentaires peuvent être utilisés pour **supprimer des retours à la ligne** indésirables :

ligne%  
ligne  ligneligne

Cela permet de mieux formater le code source.

- ▶ en évitant des lignes trop longues

# Outline

1. Éléments de base
2. Rédaction des documents
3. Tableaux
4. Figures
5. Mathématiques
6. Usages avancés : quelques références

# Structure d'un document

## 1. Définition de la classe du document

- ▶ `\documentclass{book}`

- ▶ classe = type

## 2. Imports de paquets

- ▶ `\usepackage{fontspec}`

## 3. Définitions de commandes

- ▶ `\newcommand{\hello}{Salut !}`

## 4. Titre, auteur, date, etc.

- ▶ `\title{Mon premier article}`

- ▶ `\author{Moi}`

- ▶ `\date{Aujourd'hui}`

## 5. Contenu

- ▶ sections, exemples, code, etc.

# Paquets et styles

**Paquet** = une librairie de commandes et de définitions

**Style** = un **paquet** qui redéfinit des paramètres de style globaux du document (police, couleurs, décoration de la page, etc.)

Un paquet est défini dans un fichier `.sty`.

Un paquet est utilisé avec la commande

```
\usepackage{paquet}
```

Comment afficher les caractères français ?

---

LuaTeX : Ça marche tout seul.

Rien que pour ça il faut l'utiliser!

---

pdfTeX : Utiliser le paquet `inputenc` :

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



```
\documentclass{article}
\begin{document}
1   \title{C'est le titre, il apparaîtra en gros}
2   \author{Le nom de l'auteur + détails}
3   \date{La date}
4   \maketitle

\end{document}
```

1. `\title` : obligatoire pour avoir un titre
2. `\author` : optionnel
3. `\date` : optionnelle ; si omise, la date de la compilation du document sera affichée
4. `\maketitle` : affiche le titre, l'auteur et la date (si définis)

# Sections

```
\section{Première section}
```

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

```
\subsubsection{Une sous-sous-section}
```

```
\section{Deuxième section}
```

Les sections sont **numérotées** par défaut.

Utiliser **\*** pour rajouter une section **non numérotée** :

```
\section*{Section non numérotée}
```

# Table des matières

Afficher la table des matières :

```
\tableofcontents
```

Afficher la liste des figures :

```
\listoffigures
```

Afficher la liste des figures :

```
\listoftables
```

# La langue du document

La table des matières s'appelle « Contents » par défaut.

Pour changer de langue pour le document :

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

# Listes

Liste numérotée :

<code>\begin{enumerate}</code>	
<code>\item élément1</code>	1. élément1
<code>\item élément2</code>	2. élément2
<code>\item élément3</code>	3. élément3
<code>\end{enumerate}</code>	

Liste **non** numérotée :

<code>\begin{itemize}</code>	
<code>\item élément1</code>	▶ élément1
<code>\item élément2</code>	▶ élément2
<code>\item élément3</code>	▶ élément3
<code>\end{itemize}</code>	

# Format du paragraphe

Les paragraphes sont **justifiés** par défaut.

- ▶ `\raggedright` pour aligner à gauche
- ▶ `\raggedleft` pour aligner à droite
- ▶ `\centering` pour centrer

Ces commandes changent le style globalement. Pour localiser leur effet, utiliser `{ }` :

```
{\raggedleft Texte à droite}
```

---

Pour modifier l'interligne, utiliser `\usepackage{setspace}` :

- ▶ `\singlespace` : interligne normal
- ▶ `\onehalfspace` : interligne 1.5 fois plus grand
- ▶ `\doublespace` : interligne double

## Tailles des caractères

<code>\tiny{Gérard !}</code>	⇒	Gérard !
<code>\scriptsize{Gérard !}</code>	⇒	Gérard !
<code>\footnotesize{Gérard !}</code>	⇒	Gérard !
<code>\small{Gérard !}</code>	⇒	Gérard !
<code>\normalsize{Gérard !}</code>	⇒	Gérard !
<code>\large{Gérard !}</code>	⇒	Gérard !
<code>\Large{Gérard !}</code>	⇒	Gérard !
<code>\LARGE{Gérard !}</code>	⇒	Gérard !
<code>\huge{Gérard !}</code>	⇒	Gérard !
<code>\Huge{Gérard !}</code>	⇒	Gérard !

# Styles de polices

- ▶ `\textbf{en gras}`  $\implies$  **en gras**
  - ▶ `\textit{italique}`  $\implies$  *italique*
  - ▶ `\texttt{monospace}`  $\implies$  monospace
  - ▶ `\textsc{Petites Capitales}`  $\implies$  PETITES CAPITALES
- 

Pour mettre un fragment de texte en emphase :

`\emph{emphase}`

L'emphase est souvent rendue en italique, mais le rendu peut varier en fonction du contexte.



# Couleurs

```
\usepackage{xcolor}
...
\textcolor{red}{Text en rouge}
```

La commande `\color` modifie la couleur actuelle :

Ce texte est noir `\color{red}` mais celui-ci est rouge.

---

Pour définir ses propres couleurs en système RGB :

```
% 100% rouge, 50% vert, 0% bleu
\definecolor{orange}{rgb}{1, 0.5, 0}
```

Pour définir ses propres couleurs à base d'autres couleurs :

```
% 80% rouge, 20% vert
\colorlet{monrouge}{red!80!yellow}
```

# Outline

1. Éléments de base
2. Rédaction des documents
3. Tableaux
4. Figures
5. Mathématiques
6. Usages avancés : quelques références

# Tableaux

alignement des colonnes :  
à gauche, centré, à droite

```
\begin{tabular}{l c r}  
un & deux & trois \\ quatre & cinq & six  
\end{tabular}
```

séparateur des lignes

séparateur des colonnes

Ce tableau sera rendu ainsi :

un	deux	trois
quatre	cinq	six

`\\` est optionnel après la dernière ligne.

# Lignes dans les tableaux

ligne verticale



```
\begin{tabular}{l | c r}
```

```
un & deux & trois \\
```

```
\hline ← ligne horizontale
```

```
quatre & cinq & six
```

```
\end{tabular}
```

Ce tableau sera rendu ainsi :

un	deux	trois
quatre	cinq	six

## Espaces entre les lignes et les colonnes

Régler l'espace entre les colonnes à 4mm :

```
\tabcolsep=4mm
```

Multiplier par 2 l'espace entre les lignes :

```
\renewcommand{\arraystretch}{2}
```

---

Ces commandes modifient les paramètres de rendu globalement. Pour localiser leur effet, utiliser `{ }` :

```
{\tabcolsep=4mm  
  \begin{tabular}{l c r}  
  ...  
  \end{tabular}  
}
```

## Texte multiligne dans les cellules

Le format de colonne `p{largeur}` permet de fixer la largeur d'une colonne :

```
\begin{tabular}{|c|p{5cm}|}  
  \hline  
  une ligne & un texte très long qui  
              ne tient pas en une ligne \\  
  \hline  
  une autre ligne & un texte plus court \\  
  \hline  
\end{tabular}
```

une ligne	un texte très long qui ne tient pas en une ligne
une autre ligne	un texte plus court

## Styles des colonnes

Le paquet `array` permet de définir des **commandes** à rajouter **en début de chaque cellule** d'une colonne.

```
\usepackage{array}
...
\begin{tabular}{ >{\Large}l l}
  un & deux \\
  trois & quatre
\end{tabular}
```

Les cellules de la première colonne auront des caractères plus grands :

un	deux
trois	quatre

# Outline

1. Éléments de base
2. Rédaction des documents
3. Tableaux
4. Figures
5. Mathématiques
6. Usages avancés : quelques références



# Images

Le paquet `graphicx` permet d'importer des **images** en formats PNG, JPG et PDF :

```
\usepackage{graphicx}
```

```
...
```

```
% Chercher cat.png, cat.jpg ou cat.pdf
```

```
\includegraphics{cat}
```

L'extension du fichier d'image doit être **omise**.

---

Réduire de 10 fois la taille de l'image à l'affichage :

```
\includegraphics[scale=0.1]{cat}
```

Afficher l'image dans un rectangle 30mm de largeur :

```
\includegraphics[height=30mm]{cat}
```

Comment associer un titre à l'image ?

Comment positionner l'image ?

Élément flottant = un groupe de contenu avec les propriétés suivantes :

- ▶ est positionné automatiquement
- ▶ a un titre
- ▶ peut être référencé par une étiquette

---

```
\begin{figure}[t] ← positionner la figure plutôt  
  % Image ou tableau en haut de page  
  \caption{Le titre qui tue}  
  \label{fig:mafigure} ← étiquette de référence  
\end{figure}
```

# Étiquettes et références

```
\begin{figure}[t]  
  ...  
  \label{fig:mafigure}  
\end{figure}
```

Pour faire référence à cette figure :

On voit dans la figure~\ref{fig:mafigure} que ...

---

On peut référencer d'autres éléments :

```
\section{Ma section}\label{sec:masection}  
...
```

Dans la section~\ref{sec:masection} on a dit que ...

---

Pas de contraintes pour les noms d'étiquette, mais on préfixe habituellement par le type de l'objet référencé.

# Outline

1. Éléments de base
2. Rédaction des documents
3. Tableaux
4. Figures
5. Mathématiques
6. Usages avancés : quelques références

# Mise en page d'expressions mathématiques

Les notations mathématiques ont été l'une des motivations initiales pour la création de T<sub>E</sub>X.

---

Expressions de base : prévues dans le langage

- ▶ pas besoin d'inclure de paquets

Expressions avancées :

```
\usepackage{mathtools}
```

ou alors

```
\usepackage{amsmath}
```

# Environnements pour les mathématiques

Pour afficher des maths dans le corps du texte :

Le résultat de  $1+2$  est 3.

Pour afficher des maths hors du paragraphe, centrées :

L'équation suivante est vraie :

$[$

$$1 + 2 = 3.$$

$]$

# Notations de base

Puissances et indices :

$$\text{\$}x_1 + y^2 + x_1^2\text{\$} \implies x_1 + y^2 + x_1^2$$

Opérateurs :

$$\text{\$}\cos(2x) + \sin^2 x\text{\$} \implies \cos(2x) + \sin^2 x$$

Caractères grecs :

$$\text{\$}\alpha^3 + \Gamma_1\text{\$} \implies \alpha^3 + \Gamma_1$$

Symboles mathématiques :

$$\text{\$}\forall x \in A \text{\$} \implies \forall x \in A$$



# Notations avancées

## Fractions

$$\text{\$}\text{\frac{1 + \frac{1}{x}}{2}}\text{\$} \implies \frac{1 + \frac{1}{x}}{2}$$

## Racines

$$\text{\$}\text{\sqrt{\frac{1}{2}}}\text{\$} \implies \sqrt{\frac{1}{2}}$$

$$\text{\$}\text{\sqrt[3]{\sin^2 \alpha}}\text{\$} \implies \sqrt[3]{\sin^2 \alpha}$$

## Matrices

$$\begin{array}{l} \text{\$} \\ \text{\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}} \\ \text{\end{pmatrix}} \\ \text{\$} \end{array} \implies \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}$$

# Théorèmes

```
\usepackage{amsthm}
\newtheorem{theorem}{Théorème}
\newtheorem{definition}{Définition}
```

Souvent on utilise des fichiers de style qui font déjà ces définitions.

```
\begin{theorem}
  Mon joli théorème.
\end{theorem}
\begin{proof}
  La preuve que tout ce que je dis est vrai.
\end{proof}
```

# Outline

1. Éléments de base
2. Rédaction des documents
3. Tableaux
4. Figures
5. Mathématiques
6. Usages avancés : quelques références

TikZ = une librairie pour programmer des  
dessins en L<sup>A</sup>T<sub>E</sub>X

Un manuel très complet + 5 tutoriels :

[http://mirrors.ctan.org/graphics/pgf/base/doc/  
pgfmanual.pdf](http://mirrors.ctan.org/graphics/pgf/base/doc/pgfmanual.pdf)



# TikZ vs. solutions externes à $\text{\LaTeX}$

- + **Intégration continue** avec le moteur de mise en page
  - ▶ rendu fidèle de polices
  - ▶ pas de distorsions dues au redimensionnement
- + Langage **performant** pour les images vectorielles
- Langage **difficile** à maîtriser

# Beamer

Beamer = une librairie pour préparer des  
transparents avec  $\text{\LaTeX}$

Un transparent  $\frac{1}{1}$  un environnement frame

<https://en.wikibooks.org/wiki/LaTeX/Presentations>

[https://fr.wikibooks.org/wiki/LaTeX/Faire\\_des\\_pr%C3%A9sentations](https://fr.wikibooks.org/wiki/LaTeX/Faire_des_pr%C3%A9sentations)

# Beamer vs. d'autres solutions





- + Préparation rapide de **transparents techniques**
- + Séparation entre le **contenu** et la **présentation**
- **Peu** d'animations
- **Pas** d'interactivité

# Beamer vs. d'autres solutions


- + Préparation rapide de **transparents techniques**
- + Séparation entre le **contenu** et la **présentation**
- **Peu** d'animations
  - ▶ il y a quand même les overlays!
- **Pas** d'interactivité





## Références générales $\text{\LaTeX}$ :

-  <https://en.wikibooks.org/wiki/LaTeX>
-  <https://fr.wikibooks.org/wiki/LaTeX>
-  <http://smayolasite.com/resources/Petit%20manuel%20d'utilisation%20de%20LaTeX.pdf>
-  <https://www.latex-project.org/get/>

## Références TikZ :

-  <http://mirrors.ctan.org/graphics/pgf/base/doc/pgfmanual.pdf>

## Références Beamer :

-  <https://en.wikibooks.org/wiki/LaTeX/Presentations>
-  [https://fr.wikibooks.org/wiki/LaTeX/Faire\\_des\\_pr%C3%A9sentations](https://fr.wikibooks.org/wiki/LaTeX/Faire_des_pr%C3%A9sentations)