



**IMT Mines Albi-Carmaux**  
École Mines-Télécom

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

**Paul Gaborit**

IMT Mines Albi  
2006 – 2007, 2011, 2016 – 2024

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

- Où trouver de l'information ou de l'aide ?
- Le fonctionnement de L<sup>A</sup>T<sub>E</sub>X
- Notions de syntaxe : commandes et environnements
- Et en pratique...

## 2 Structure d'un document L<sup>A</sup>T<sub>E</sub>X

- Le document et ses options
- Structurer son document
- Les graphiques et les tableaux
- Les objets flottants

## 3 Mise en page

- La maquette
- La typographie
- La programmation

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

- Où trouver de l'information ou de l'aide?
- Le fonctionnement de L<sup>A</sup>T<sub>E</sub>X
- Notions de syntaxe : commandes et environnements
- Et en pratique...

## 2 Structure d'un document L<sup>A</sup>T<sub>E</sub>X

## 3 Mise en page



## Où trouver de l'information ou de l'aide ?

- Ce document (et tous ceux de la formation L<sup>A</sup>T<sub>E</sub>X) :
  - <https://gaborit.p.mines-albi.fr/initiation-latex/>
- « Une courte introduction à L<sup>A</sup>T<sub>E</sub>X » sur papier ou en ligne :
  - <http://ctan.mines-albi.fr/info/lshort/french/lshort-fr.pdf>
- La commande `texdoc` qui fournit la documentation d'une classe ou d'un package installé sur votre ordinateur. Ex :

```
texdoc latex2e
texdoc xcolor
```
- La liste L<sup>A</sup>T<sub>E</sub>X de l'IMT Mines Albi [latex@listes.mines-albi.fr](mailto:latex@listes.mines-albi.fr). Pour s'y inscrire :
  - site [listes.mines-albi.fr](http://listes.mines-albi.fr), connexion (bouton `IMT-Mines-Albi`), chercher `latex`;
  - ou écrire à [paul.gaborit@mines-albi.fr](mailto:paul.gaborit@mines-albi.fr) ou directement à la liste pour demander son inscription.
- Les forum de discussion L<sup>A</sup>T<sub>E</sub>X francophone et anglophone :
  - [fr.comp.text.tex](http://fr.comp.text.tex)
  - [comp.text.tex](http://comp.text.tex)
- Les sites de questions/réponses TeX.SE en anglais (excellent) et en français :
  - <http://tex.stackexchange.com/>
  - <https://texnique.fr/osqa/>
- Le « L<sup>A</sup>T<sub>E</sub>X Companion » (disponible au centre de documentation en anglais et en français)... et beaucoup d'autres livres.

Les différents rôles :

### 1 L'auteur :

- décrit le contenu du document (textes, graphiques, équations, tableaux...),
- précise, via des commandes et des environnements, la sémantique associée aux différents éléments du contenu : type du document, titres, légendes, références, citation, langue, remarques...

### 2 Le moteur de mise en page (L<sup>A</sup>T<sub>E</sub>X et T<sub>E</sub>X) :

- réalise la mise en page en respectant les indications sémantiques de l'auteur et les règles typographiques en vigueur,
- indique à l'auteur les éventuelles erreurs sur la structuration du document,
- indique à l'auteur les éventuels problèmes rencontrés lors de la mise en page.

### 3 Le graphiste ou maquettiste *programmeur* :

- fixe tous les réglages typographiques spécifiques : marges, numérotation, en-têtes, choix des polices...
- programme (en L<sup>A</sup>T<sub>E</sub>X et/ou en T<sub>E</sub>X) le moteur de mise en page, sous forme de packages complémentaires pour ajouter de nouvelles fonctionnalités.




L'auteur ne devrait *jamais* se préoccuper de la mise en page sauf lors de la mise en page finale du document.




L<sup>A</sup>T<sub>E</sub>X, le moteur de mise en page prend en charge la mise en page si on sait lui demander...





La programmation prend du temps. Et l'apprentissage du rôle de maquettiste programmeur aussi.


\* L'auteur ne devrait *jamais* se préoccuper de la mise en page sauf lors de la mise en page finale du document.

 L<sup>A</sup>T<sub>E</sub>X, le moteur de mise en page prend en charge la mise en page si on sait lui demander...


Pourquoi programmer?

 La programmation prend du temps. Et l'apprentissage du rôle de maquetteste programmeur aussi.

\* L'auteur ne devrait *jamais* se préoccuper de la mise en page sauf lors de la mise en page finale du document.

 L<sup>A</sup>T<sub>E</sub>X, le moteur de mise en page prend en charge la mise en page si on sait lui demander...

Par paresse !

 La programmation prend du temps. Et l'apprentissage du rôle de maquettiste programmeur aussi.





L'auteur ne devrait *jamais* se préoccuper de la mise en page sauf lors de la mise en page finale du document.



L<sup>A</sup>T<sub>E</sub>X, le moteur de mise en page prend en charge la mise en page si on sait lui demander...



La programmation prend du temps. Et l'apprentissage du rôle de maquettiste programmeur aussi.

Par paresse !

Gain de temps



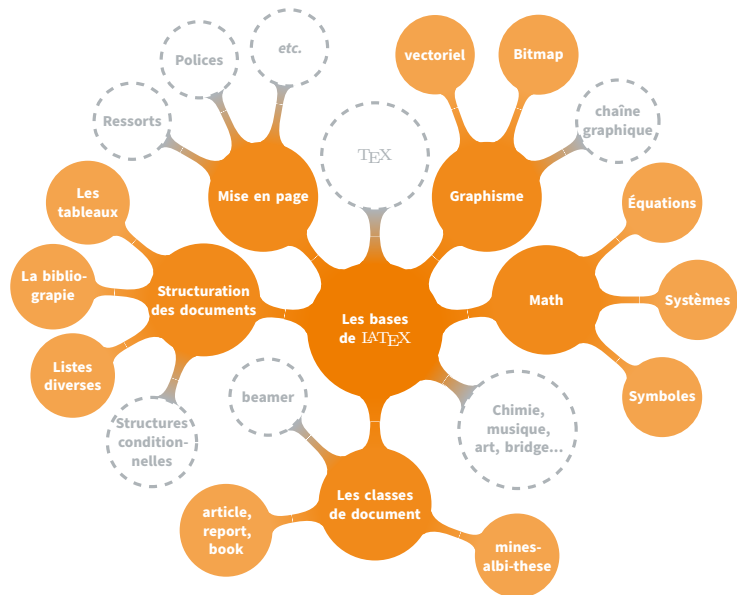
L'auteur ne doit pas se  
préoccuper de la  
lors de la mise  
document.

de temps

On peut quand  
même faire des  
choses **RiGoLoTeS**  
avec L<sup>A</sup>T<sub>E</sub>X !

Par paresse

on prend du  
tissage du rôle  
programmeur aussi.



```
% Mon premier document LaTeX
\documentclass[a4paper]{article}

\usepackage[utf8]{inputenc} % source encodé en UTF-8 (par défaut depuis 2018)
\usepackage[T1]{fontenc}
\usepackage{lmodern} % la police Latin Modern
\usepackage[french]{babel} % document rédigé en français

% Début du document
\begin{document}

Mon premier document \LaTeX{}.

\end{document}
```

 Le fichier L<sup>A</sup>T<sub>E</sub>X `document1.tex` est une pièce-jointe de ce fichier PDF

Comme tous les documents en L<sup>A</sup>T<sub>E</sub>X, cet exemple est structuré en deux parties :

**le préambule** entre `\documentclass` et `\begin{document}` dans lequel on mentionne les éléments (packages, options) qui vont régir la mise en page de l'ensemble du texte;

**le corps du texte** dans lequel on trouvera essentiellement du texte et des commandes L<sup>A</sup>T<sub>E</sub>X. Nous verrons que cette partie est structurée par des chapitres, sections, etc.

Mais que faire de ce document ?

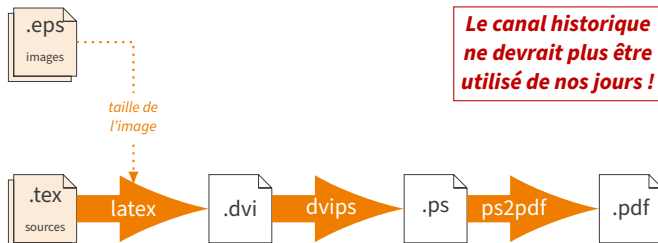
Mais que faire de ce document ?

- ▶ On va le **compiler** pour obtenir le document final dans le format qui nous intéresse (PDF, Postscript...)

***Le canal historique  
ne devrait plus être  
utilisé de nos jours !***

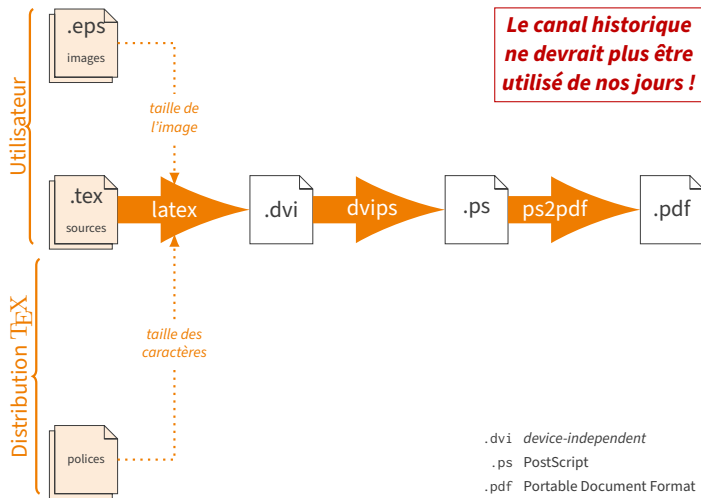


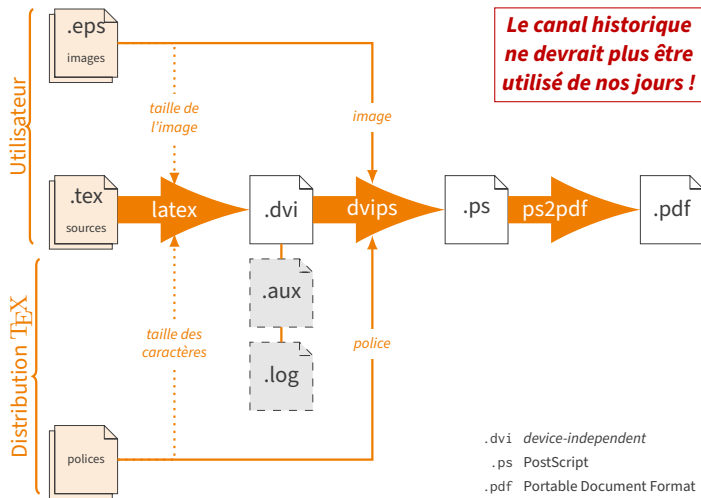
.dvi *device-independent*  
.ps PostScript  
.pdf Portable Document Format

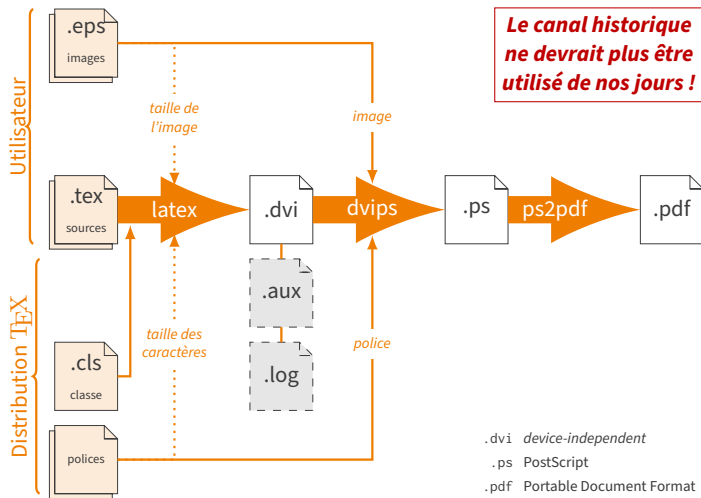


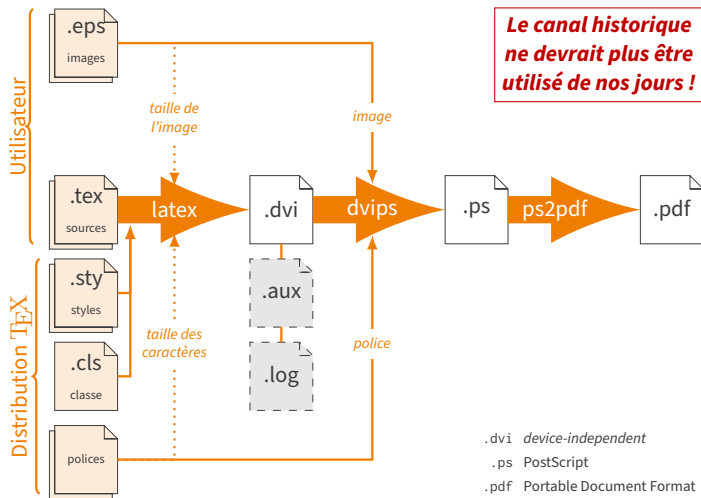
- `.dvi` *device-independent*
- `.ps` PostScript
- `.pdf` Portable Document Format

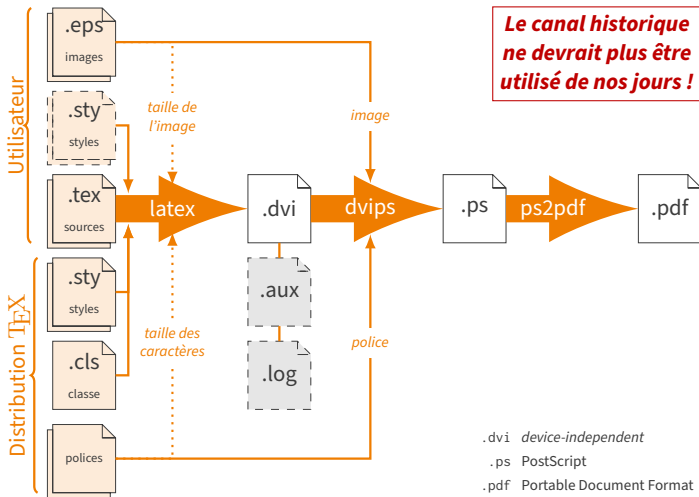




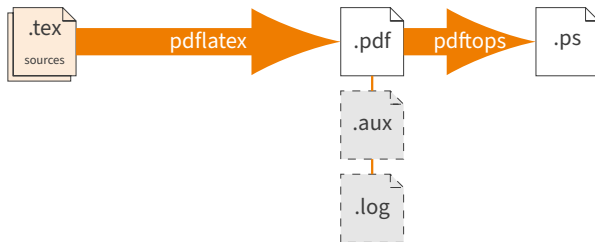






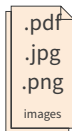


*De nos jours, produire  
un fichier .ps est rare.  
Le fichier .pdf suffit.*

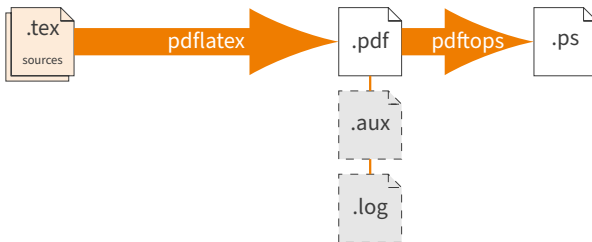


.ps PostScript

.pdf Portable Document Format

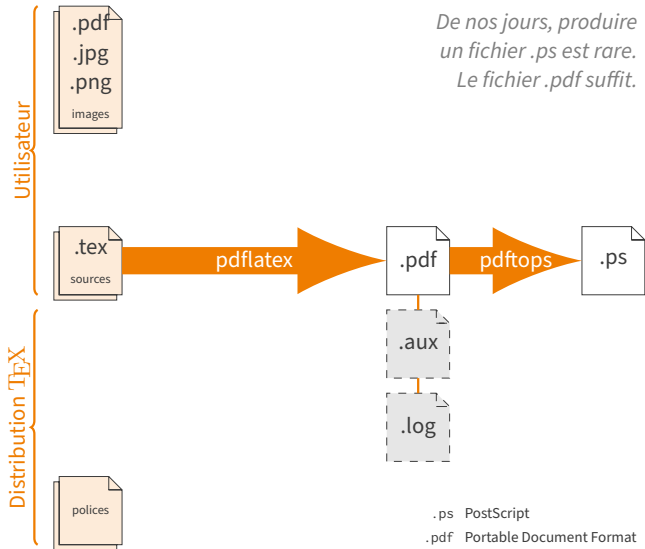


*De nos jours, produire  
un fichier .ps est rare.  
Le fichier .pdf suffit.*

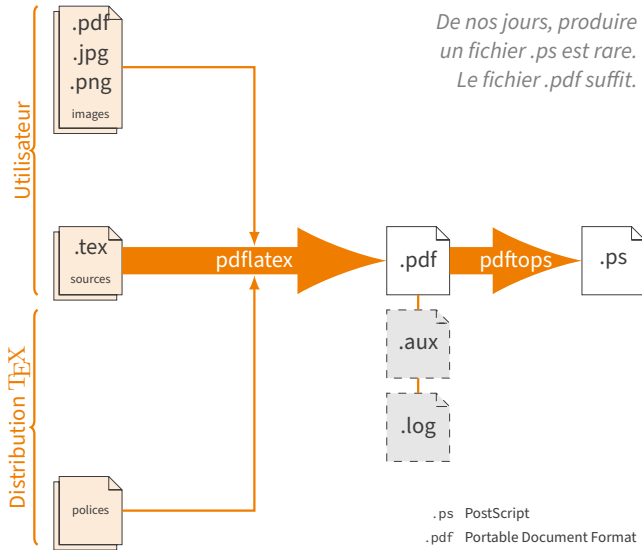


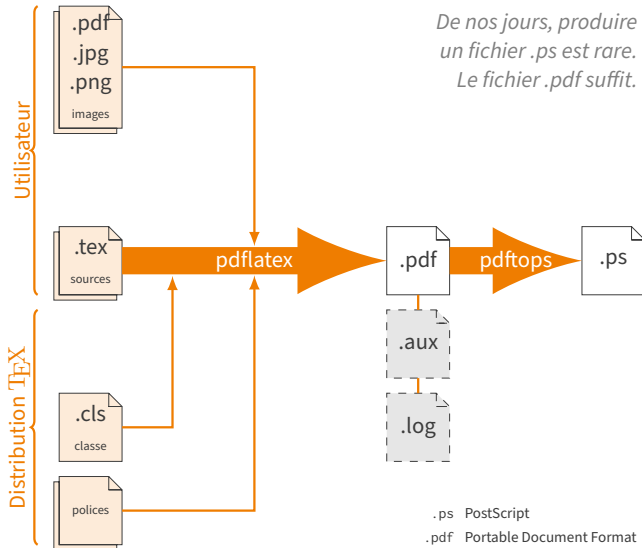
.ps PostScript

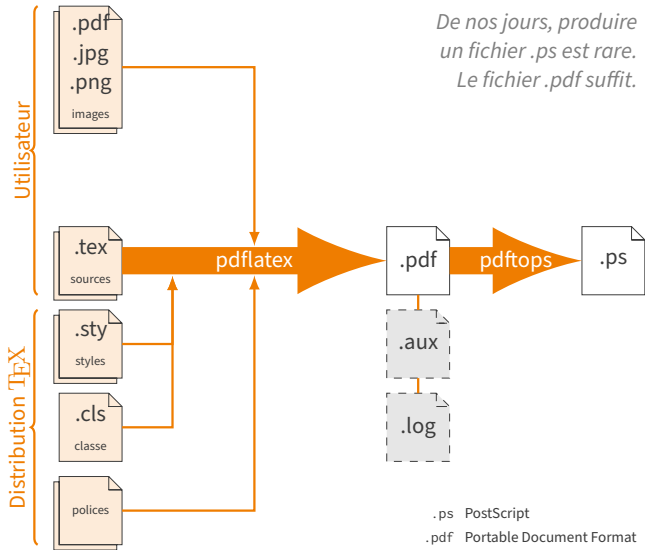
.pdf Portable Document Format

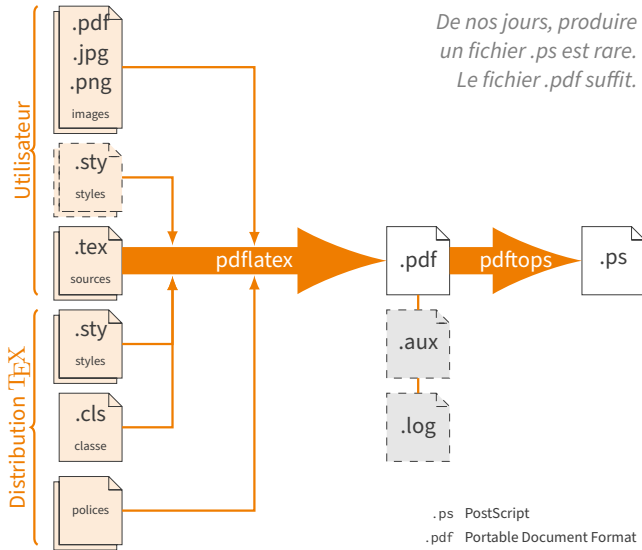












- Il faut parfois *plusieurs* compilations pour obtenir le document final : les références en avant sont passées via le ou les fichier(s) `.aux`.
- Le fichier `.log` contient la trace de toute la compilation (beaucoup plus complet que l’affichage).

Une **commande** L<sup>A</sup>T<sub>E</sub>X :

- est préfixée par le caractère `\` (*barre oblique inversée* ou *backslash*).
- a un **nom** constitué :
  - d'une ou plusieurs lettres (les 52 lettres majuscules ou minuscules, **pas de chiffres...**) :  
`\textbf` `\emph` `\oe` `\section`
  - ou d'un seul symbole :  
`\'` `\,` `\{` `\}` `\%` `\&`
- et utilise généralement un ou plusieurs arguments :
  - entourés d'accolades `{...}`
  - ou de crochets `[...]` pour les arguments optionnels

- Exemples de commandes nommées :

```
\section{Le titre un peu longuet d'une section}
\section[Le titre court]{Le titre un peu longuet d'une section}
Le b\oe{}uf est \emph{petit}...
\frac{1}{4}
```

- Exemples de commandes décrites par un symbole :

```
La distance est de 4\,400\,cm.
L'\et\e est chaud !
L'\{e}t\{e} est chaud !
L'été est chaud !
L'art \& la manière...
L'ensemble \{1, 2, 3\}
\\
\\[4cm]
```

## Les environnements

Un **environnement** L<sup>A</sup>T<sub>E</sub>X :

- porte un **nom** composé d'une ou plusieurs lettres majuscules et/ou minuscules
- commence par `\begin{nom}` et se termine par `\end{nom}` et tout le code placé entre ces deux commandes s'appelle le *corps* (*body*) de l'environnement.
- peut utiliser un ou plusieurs arguments
  - entourés d'accolades
  - entourés de crochets pour les arguments optionnels
- peut être inclus dans d'autres environnements et en contenir lui-même.

Exemple d'utilisation d'un environnement nommé **exemple** :

```
\begin{exemple}[argument optionnel]{argument}
contenu de l'environnement 'exemple'
éventuellement sur plusieurs lignes
\end{exemple}
```

L'environnement le plus externe d'un document L<sup>A</sup>T<sub>E</sub>X est toujours l'environnement **document**.



Parfois (selon le contexte), certains caractères sont actifs. Par exemple :

- le caractère `$` pour débiter une formule mathématique dans le texte ainsi que les caractères `^` et `_` (en mode mathématique) pour indiquer un exposant ou un indice :

La formule `$E = mc^{2}$` est célèbre.

La formule  $E = mc^2$  est célèbre.

- les caractères `!`, `?`, `:`, etc. (toutes les ponctuations doubles) en français pour insérer l'espace insécable qui précède :

La typographie française: facile?!?

La typographie française : facile?!?

- Le caractère `%` pour débiter un commentaire.

```
% ceci est un commentaire
```

Pour obtenir un `.dvi` puis un `.ps` (canal historique – ne plus utiliser)

<i>action</i>	<i>commande</i>
compiler un fichier <code>.tex</code> en <code>.dvi</code>	<code>latex document.tex</code>
voir un fichier <code>.dvi</code>	<code>xdvi document.dvi</code>
convertir un fichier <code>.dvi</code> en <code>.ps</code>	<code>dvips -o document.ps document.dvi</code>
voir un fichier <code>.ps</code>	<code>gv document.ps</code>

Pour obtenir un `.pdf`

<i>action</i>	<i>commande</i>
compiler un fichier <code>.tex</code> en <code>.pdf</code>	<code>pdflatex document.tex</code>
voir un fichier <code>.pdf</code>	<code>acroread document.pdf</code> <code>xpdf document.pdf</code> <code>evince document.pdf</code>

👉 En cas d'erreur, `(pdf)latex` indique l'erreur puis le numéro de la ligne erronée et même l'endroit concerné :

```
! Undefined control sequence.
l.11 Mon \premier
      document \LaTeX{}
```

Pour arrêter `(pdf)latex`, taper `Ctrl-d` (`Ctrl-z` sur PC). Pour lui demander de continuer sans tenir compte des erreurs, taper `R` (majuscule). Parfois d'autres choix sont proposés...

## Quel éditeur de texte utiliser ?

Vous pouvez utiliser n'importe quel éditeur permettant d'écrire des programmes (donc pas Word ni LibreOffice 😊). Certains sont cependant plus adaptés à l'écriture de textes en L<sup>A</sup>T<sub>E</sub>X, voire même ne servent qu'à cela.

Parmi les plus utilisés (et recommandés), on peut citer :

<code>emacs</code>	disponible sur toutes les bonnes machines,
<code>vim</code>	éternel rival du précédent (ou le contraire),
Visual Studio Code	IDE générique multi-plateformes,
TeXmaker	IDE L <sup>A</sup> T <sub>E</sub> X multi-plateformes,
TeXstudio	IDE L <sup>A</sup> T <sub>E</sub> X multi-plateformes (clone du précédent),
TeXShop	pour Macintosh,
TeXnic Center	pour Windows.

Ces éditeurs permettent généralement un accès aisé aux commandes qui permettent de créer un document complet en L<sup>A</sup>T<sub>E</sub>X. Tous offrent la coloration syntaxique, très utile pour les débutants aussi bien que les experts.

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

## 2 Structure d'un document L<sup>A</sup>T<sub>E</sub>X

- Le document et ses options
- Structurer son document
- Les graphiques et les tableaux
- Les objets flottants

## 3 Mise en page



## Un exemple plus complet

```
\documentclass[a4paper]{article}
\usepackage[utf8]{inputenc} % inutile depuis 2018
\usepackage[T1]{fontenc}
\usepackage{lmodern} % la police Latin Modern
\usepackage[french]{babel}
```

```
\title{Un second document \LaTeX}
\author{P.Gaborit}
\date{9 décembre 2003}
```

```
\begin{document}
```

```
\maketitle
```

```
\section{Introduction}
```

Cette section permet de présenter l'article. Elle ne devrait servir que d'introduction.

```
\section{Conclusion}
```

C'est ici que s'arrête notre second exemple.

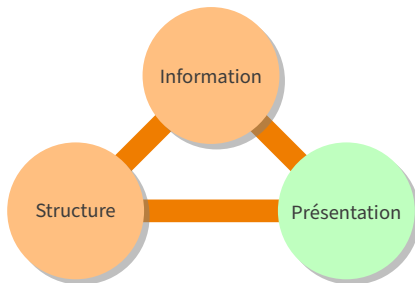
```
\tableofcontents
```

```
\end{document}
```

 Le fichier L<sup>A</sup>T<sub>E</sub>X `document2.tex` est une pièce-jointe de ce fichier PDF

## Les règles de base

- 1 On tape son texte au fil de l'eau sans se préoccuper des fins des lignes et de la mise en page (les différents paragraphes sont séparés par une ligne vide) (**Information** dans le document).
- 2 On ajoute des commandes et des environnements afin de structurer et préciser le sens (la sémantique) du texte (**Structure** dans le document).
- 3 Les commandes ou les environnements de présentation ne devraient servir qu'à définir des nouvelles commandes ou environnements sémantiques (**Présentation** dans le préambule).



► On ne dit pas « **Met ça en gras** » !

► On dit « **Ceci est important** » et au début du document, on ajoute « **Ce qui est important sera en gras** ».

► On ne dit pas « **Ce paragraphe doit être en italique avec une marge de 3 cm** » !

► On dit « **Ce paragraphe est une citation** » et au début du document, on ajoute « **Les citations seront présentées en italique avec une marge de 3 cm** ».

► On ne dit pas « **Ce mot doit être rouge et en courier** » !

► On dit « **Ce mot est une commande** » et au début du document, on ajoute « **Les commandes seront présentées en courier et en rouge** ».

Les arguments possibles pour la commande `\documentclass` :

- article** pour écrire des articles dans des revues ou des congrès. Un **article** est découpé en sections, sous-sections (`\subsection`), sous-sous-sections (`\subsubsection`)...
- report, book** pour écrire des livres complets (une thèse par exemple). Un **book** ou un **report** est découpé en chapitres (`\chapter`) éventuellement regroupés par parties (`\part`). Les chapitres sont découpés en sections, sous-sections... Le document peut contenir des annexes (`\appendix`).
- letter** pour écrire des lettres. Adapté aux règles de présentation anglo-saxonne. On peut trouver une classe **lettre** pour les lettres en français.
- beamer** pour faire des présentations (lire la documentation fournie).
- ... Il existe de nombreux autres classes pour tout faire. Elles sont parfois développées spécifiquement pour des journaux ou des conférences (IEEE, Elsevier...) ou pour certaines institutions (`mines-albi-these`,...).



Les options de `documentclass` sont les réglages généraux du document :

- `10pt, 11pt, 12pt` : choix de la taille des caractères;
- `a4paper, letterpaper, legalpaper...` : indique le format du support;
- `portrait, landscape` : indique l'orientation du support;
- `oneside, twoside` : recto simple ou recto/verso;
- `draft, final` : version brouillon ou version finale;
- `onecolumn, twocolumn` : choix du nombre de colonnes.

Exemples :

- Un rapport sur papier américain en recto/verso et en version finale :

```
\documentclass[letterpaper, twoside, final]{report}
```

- un article en deux colonnes sur papier A4, version brouillon :

```
\documentclass[a4paper, draft, twocolumn]{article}
```

- Les paquetages (*packages*) sont des extensions permettant de paramétrer le document ou d'ajouter de nouvelles fonctionnalités :

```
\usepackage[option1,option2...]{nomdupaquetage}
```

- Très souvent, il existe un (ou plusieurs) paquetage installé en standard permettant de faire *ce que vous voulez!*
- N'hésitez pas à demander plutôt que de réinventer la roue. Pour des besoins plus spécifiques (chimie, musique...), il existe parfois des paquetages spécialisés de très haute qualité.
- Le contenu du document lui-même est dans l'environnement `document` :

```
\begin{document}  
...  
\end{document}
```

Exemples standard (avec numérotation) :

```
\part{Une première partie}
\chapter{Un premier chapitre}
\section{Une première section}
\subsection{Une première sous-section}
\subsubsection{Une première sous-sous-section}
\paragraph{Un premier paragraphe}
\subparagraph{Un premier sous-paragraphe}
```

Exemples sans numérotation :

```
\part*{Une partie sans numérotation}
\chapter*{Un chapitre sans numérotation}
\section*{Une section sans numérotation}
\subsection*{Une sous-section sans numérotation}
\subsubsection*{Une sous-sous-section sans numérotation}
\paragraph*{Un paragraphe sans numérotation}
\subparagraph*{Un sous-paragraphe sans numérotation}
```

 le fichier L<sup>A</sup>T<sub>E</sub>X `document3.tex` est une pièce-jointe de ce fichier PDF

- La commande `\tableofcontents` permet de créer la table des matières. Plusieurs compilations sont nécessaires pour que la table des matières soit correcte.

- On peut fixer le niveau de numérotation des sections :

```
\setcounter{secnumdepth}{5}
```

- On peut fixer le niveau de détail de la table des matières :

```
\setcounter{tocdepth}{5}
```


- Les niveaux sont définis de la manière suivante :

book ou report		article	
<code>\part</code>	niveau -1	<code>\part</code>	niveau 0
<code>\chapter</code>	niveau 0		
<code>\section</code>	niveau 1	<code>\section</code>	niveau 1
<code>\subsection</code>	niveau 2	<code>\subsection</code>	niveau 2
<code>\subsubsection</code>	niveau 3	<code>\subsubsection</code>	niveau 3
<code>\paragraph</code>	niveau 4	<code>\paragraph</code>	niveau 4
<code>\subparagraph</code>	niveau 5	<code>\subparagraph</code>	niveau 5

Quelques commandes parmi les plus communes :

- `\emph` pour mettre en évidence (habituellement par de *l'italique*) ou pour présenter des citations. Supporte plusieurs niveaux d'imbrication.
- `\textbf` pour montrer quelque chose d'important (en **gras**).
- `\texttt` pour afficher du code ou des commandes à saisir. Se présente avec une police type machine à écrire (*chasse fixe* ou *espacement non proportionnel*).
- `\textsc` pour afficher en petites capitales. Souvent utilisé pour les noms ou acronymes : J.CHIRAC, EMAC...
- `\og, \fg` permet d'ouvrir et fermer les guillemets **à la française** : « *ceci est une citation* »

Quelques environnements parmi les plus communs :

- itemize** pour faire des listes. Supporte plusieurs niveaux d'imbrication. Chaque item de la liste débute par la commande `\item`.
- enumerate** pour faire des énumérations (une liste numérotée). Supporte plusieurs niveaux d'imbrication. Chaque item de la liste débute par la commande `\item`.
- description** pour faire une liste de termes avec leur description (comme la liste actuelle par exemple). Chaque item de la liste débute par la commande `\item[texte]` (où `texte` est le terme à décrire).
- equation** pour saisir des équations.
- verbatim** pour afficher du texte présenté tel que saisi.
  -  le fichier L<sup>A</sup>T<sub>E</sub>X `document4.tex` est une pièce-jointe de ce fichier PDF

## Les graphiques

- Dans le préambule du document, utiliser le paquetage (*package*) `graphicx` :

```
\usepackage[final]{graphicx} % pour voir les graphiques inclus
\usepackage[draft]{graphicx} % pour ne voir que les cadres
```

- Pour inclure un graphique quelque part (avec le nom de l'image *sans* l'extension) :

```
\includegraphics{nom-fichier-image}
```

On peut préciser la largeur (`width`), la hauteur (`height`) ou l'échelle (`scale`) et l'orientation (`angle`) :

```
\includegraphics[width=5cm,angle=90]{nom-fichier-image}
```

- Les fichiers images sont de types différents selon le type de compilation choisie :
  - `latex` et `dvips` fichiers images en EPS (Encapsulated PostScript).
  - `pdflatex` fichiers images en PDF, JPEG ou PNG.

 le fichier L<sup>A</sup>T<sub>E</sub>X `document5.tex` est une pièce-jointe de ce fichier PDF

 le graphique `graphique.pdf` est une pièce-jointe de ce fichier PDF

## Les tableaux

On peut créer un tableau grâce à l'environnement `tabular` :

```
\begin{tabular}{r|l}
  \emph{Titre 1} & \emph{Titre 2} \\
  \hline
  Texte court & Petite case \\
  Autre texte & Contenu... \\
\end{tabular}
```

<i>Titre 1</i>	<i>Titre 2</i>
Texte court	Petite case
Autre texte	Contenu...

L'argument de l'environnement `tabular` permet de préciser la forme du tableau avec les correspondances suivantes :

<code>l</code>	une colonne alignée à gauche ( <i>left</i> )
<code>r</code>	une colonne alignée à droite ( <i>right</i> )
<code>c</code>	une colonne centrée
<code>p{3cm}</code>	une colonne de 3 cm de large sur plusieurs lignes
<code>\$\$</code>	une ligne de séparation verticale
<code>@{...}</code>	pour insérer ... entre deux colonnes

 le fichier L<sup>A</sup>T<sub>E</sub>X `document6.tex` est une pièce-jointe de ce fichier PDF



- Un maquettiste n'aime pas insérer des figures ou des tableaux au milieu du texte.
- Il préfère les positionner en haut ou en bas de page ou les regrouper sur des pages spécifiques.
- C'est ce que propose L<sup>A</sup>T<sub>E</sub>X avec les objets flottants créés par les environnements `figure` ou `table`.
- Dans le texte, on fait référence au numéro de la figure ou de la table (puisqu'elle n'apparaissent pas obligatoirement là où on les crée).

```
\begin{figure}[tbp]
% l'image
\centering
\includegraphics[width=2cm]{graphique}
% la légende et le label
\caption{Le graphique d'exemple}
\label{fig-premiere}
\end{figure}
```

La figure `\ref{fig-premiere}` et la table `\ref{tab-premiere}` ne représentent rien : ce sont juste des exemples !

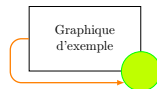


Figure 1 – Le graphique d'exemple

La figure 1 et la table 1 ne représentent rien : ce sont juste des exemples !

## Objets flottants : exemple de table

```

\begin{table}[tbp]
% la légende et le label
\caption{La table d'exemple}
\label{tab-premiere}
% le tableau
\centering
\begin{tabular}{l|l}
1 & 2 \\ \hline
3 & 4
\end{tabular}
\end{table}

```


La figure `\ref{fig-premiere}` et la table `\ref{tab-premiere}` ne représentent rien : ce ne sont que des exemples !

Le paramètre optionnel (par défaut `tbp`) règle le placement : **h**(ere) sur place — **t**(op) haut de page — **b**(ottom) bas de page — **p**(age) page seule.

Table 1 – La table d'exemple

1	2
3	4

La figure 1 et la table 1 ne représentent rien : ce ne sont que des exemples !

 le fichier L<sup>A</sup>T<sub>E</sub>X final `document7.tex` est une pièce-jointe de ce fichier PDF

- La légende d'un objet flottant se déclare via la commande `\caption`.  
**Remarque :** en typographie française, la légende d'un graphique se place en-dessous et celle d'un tableau au-dessus ! La commande `\caption` doit donc être placée avant ou après le contenu du flottant selon sa nature.
- Pour pouvoir faire référence à un objet flottant par son numéro, il faut tout d'abord lui donner une étiquette (pour le nommer) via la commande `\label`. **Attention :** cette commande doit toujours apparaître **après** la légende (`\caption`). Puis dans le texte, on utilise la commande `\ref` avec comme paramètre l'étiquette pour récupérer automatiquement le numéro de la figure correspondante.
- Les commandes `\listoffigures` ou `\listoftables` fabriquent toutes seules la liste des figures ou la liste des tables (avec leur numéro, leur légende et leur numéro de page).

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

2 Structure d'un document  
L<sup>A</sup>T<sub>E</sub>X

- 3 Mise en page
- La maquette
  - La typographie
  - La programmation



## Le travail de maquettiste

Quelques commandes et extensions pratiques pour régler des paramètres de mise en page :

- pour régler les marges (une ligne au choix) :

```
\usepackage[margin=2cm]{geometry}
\usepackage[vmargin=3cm,hmargin=1.5cm]{geometry}
\usepackage[left=1.5cm,right=1.5cm,top=3cm,bottom=3cm]{geometry}
\usepackage[inner=1.5cm,outer=3cm,vmargin=3cm]{geometry}
```

- pour régler la présentation des paragraphes (retrait et espacement) :

```
\setlength{\parindent}{0cm} % pas de retrait
\setlength{\parskip}{12pt plus 5pt minus 5pt} % espacement extensible
```

On peut aussi commencer un paragraphe par la commande `\noindent` pour ne pas prendre en compte le retrait.

- pour changer l'interligne (très demandé par les thésards) :

```
\usepackage{setspace}
% puis au choix (avant de commencer le document)
\singlespacing % pour un interligne standard
\onehalfspacing % pour un interligne 1.5
\doublespacing % pour un interligne double
```

- Les commandes suivantes permettent de personnaliser les entêtes et pieds de page de votre document :

```
\usepackage{fancyhdr} % à mettre dans le préambule  
\pagestyle{fancy}
```

- Chaque entête (head) et pied de page (foot) est divisé en trois parties : gauche (L), centre (C), droit (R) pour les pages paires (E) et impaires (O).

```
\fancyhead{} % Pour effacer l'entête par défaut  
\fancyfoot{} % Pour effacer le pied de page par défaut
```

```
% Un entête à gauche sur les pages paires (LE)  
% et à droite sur les pages impaires (RO)
```

```
\fancyhead[LE,RO]{\textbf{Utilisation de fancyhdr}}
```

```
% Le numéro de page au centre sur toutes les pages
```

```
\fancyfoot[C]{\thepage}
```

```
% L'épaisseur de la ligne de séparation
```

```
\renewcommand{\headrulewidth}{0.pt}
```

```
\renewcommand{\footrulewidth}{0.pt}
```

- Pour produire une page sans entête ni pied de page (une page de titre par exemple) :

```
\thispagestyle{empty}
```

## Le travail du typographe

Quelques commandes pratiques pour choisir des variantes de polices :

*commande avec argument*

*commande à portée longue*

```
\textnormal{...}
\textrm{...}
\textsf{...}
\texttt{...}
\textbf{...}
\textmd{...}
\textit{...}
\textsl{...}
\textsc{...}
```

```
{\normalfont ...}
{\rmfamily ...}
{\sffamily ...}
{\ttfamily ...}
{\bfseries ...}
{\mdseries ...}
{\itshape ...}
{\slshape ...}
{\scshape ...}
```

police du document

serif

sans-serif

machine à écrire

**gras**

medium

*italique*

*incliné*

PETITE CAPITALE



```
{\tiny ...}
{\scriptsize ...}
{\footnotesize ...}
{\small ...}
{\normalsize ...}
{\large ...}
{\Large ...}
{\LARGE ...}
{\huge ...}
{\Huge ...}
```

minuscule

extrêmement petit

très petit

petit

normal

large

très large

extrêmement large

énorme

**GIGANTESQUE**



- Le programmeur fait le lien entre la structure et la mise en page. Il crée de nouvelles commandes et de nouveaux environnements pour mieux structurer et présenter l'information.

- Pour créer la nouvelle commande `\important`, on place dans le préambule du document :

```
\newcommand{\important}[1]{\textbf{#1}}
```

- Ensuite on l'utilise dans le document :

Ce `\important{truc}` est le  
sujet principal de ma thèse...

Ce **truc** est le sujet principal de ma  
thèse...

- Ce qui sépare le **fond** (la sémantique voulue par l'auteur dans le texte) de la **forme** (le rendu visuel choisi par l'éditeur dans le préambule) :

```
\newcommand{\important}[1]{\textcolor{red}{\emph{#1}}}
```

Ce `\important{truc}` est le  
sujet principal de ma thèse...

Ce *truc* est le sujet principal de ma  
thèse...

- Pour créer le nouvel environnement `extrait`, on place dans l'en-tête du document :

```
\newenvironment{extrait}
% début
{\begin{quote}\color{lime!50!black}\rmfamily\itshape\bfseries\og{}}
% fin
{\fg{}\end{quote}\par}
```

- Ensuite on l'utilise dans le document :

Confucius a dit :

```
\begin{extrait}
```

Quand le Sage montre la Lune,  
le Sot regarde le doigt.

```
\end{extrait}
```

Et il avait raison !

Confucius a dit :

*« Quand le Sage montre  
la Lune, le Sot regarde le  
doigt. »*

Et il avait raison !

# FIN

Merci de votre attention.