

Introduction à \LaTeX

Introduction

Pascal Bessonneau

05/2016

Plan

L'histoire en trois mots...

Qu'est ce que \LaTeX ?

Avantages et inconvénients

Comment faire du \LaTeX ?

Les éditeurs améliorés pour \LaTeX

Compiler un document \LaTeX sur son poste de travail

Les parties du code

Les packages

L'encodage des caractères

L'histoire en trois mots...

A l'origine de \LaTeX , il y a un homme, Donald Knuth.

Mathématicien et informaticien de génie, il est l'auteur de plusieurs ouvrages de référence dont *The Art of Computer Programming*.

Il a conçu non pas \LaTeX mais \TeX . \TeX est venu, à la fin des années 70, de l'amour de la typographie et des beaux documents de Knuth.

Knuth est également le concepteur de *Metafont* : c'est un langage permettant de faire des dessins qui s'incorporent naturellement avec \TeX . Le langage est un peu difficile mais les résultats sont spectaculaires.

L'histoire en trois mots...

Dans un second temps, \TeX étant un peu trop aride, Leslie Lamport a développé \LaTeX qui se superpose à \TeX et qui offre une syntaxe plus conviviale.

Mais en fait \LaTeX n'a pas été la seule extension de \TeX , \TeX a donné naissance à plusieurs bébés... Parmi les plus célèbres :

- \LaTeX , $\text{\LaTeX} 2_{\epsilon}$
- ConTeX^1
- XeTeX^2
- ...

1. <http://www.pragma-ade.nl/>

2. http://scripts.sil.org/cms/scripts/page.php?site_id=nrsi&id=xetex

L'histoire en trois mots...

Ils ont tous en commun \TeX mais leurs langages et leurs possibilités varient un peu.

Le plus portable est \LaTeX . Un concurrent très sérieux est \ConTeX qui permet facilement de faire des choses peu aisées à réaliser en \LaTeX .

\XeTeX quant à lui permet d'utiliser une gestion des caractères et des fontes très avancées (notamment les ligatures).

Un langage de mise en forme...

\TeX est un langage informatique qui permet la mise en page de document.

Par opposition au logiciel « What You See Is What You Get » tels que Word, le principe de \TeX est de fournir à un compilateur un fichier texte contenant le texte et les commandes de mises en forme à exécuter sur ce texte.

Très grossièrement, \LaTeX peut se comparer à de l'HTML : un mélange de texte et de « balises ».

Un petit exemple de code \LaTeX

Code

```
\begin{frame}[containsverbatim]
  \frametitle{Un langage de mise en forme...}

  \TeX est un langage informatique qui permet la mise en page de document.

  Par opposition au logiciel \og~What You See Is What You Get~\fg tels que Word, le principe de
  \TeX est de fournir à un compilateur un fichier texte contenant le texte et les commandes de
  mises en forme à exécuter sur ce texte.

  Très grossièrement, \latex peut se comparer à de l'HTML~: un mélange de texte et de
  \og~balises~\fg.
```

Phases pour la production d'un document

1. L'utilisateur réalise le fichier « .tex »
2. l'utilisateur lance le compilateur sur le fichier « .tex »
3. Le compilateur renvoie le cas échéant les erreurs de compilation
4. En l'absence d'erreurs, le compilateur produit le fichier PDF ou PostScript

Les avantages de \LaTeX

\LaTeX permet de concentrer l'effort du rédacteur sur le contenu et non sur la mise en forme : la mise en forme découle directement du plan du document.

Le rendu est assez unique et permet d'obtenir des documents extrêmement propres notamment pour les fonctions mathématiques.

Les logiciels sont disponibles sur pratiquement toutes les plateformes existantes : *Windows, Mac OS X, Linux, Solaris, ...*

Les avantages de \LaTeX

Comme c'est un langage de programmation, le rendu correspond exactement à ce que l'utilisateur a spécifié.

Comme c'est un langage il est possible de :

- reproduire fidèlement un document (depuis un modèle de document par exemple)
- transmettre le document brut qui se compilera sur n'importe quelle plateforme de la même façon
- automatiser les processus
- ...

Les avantages de \LaTeX

Et il y'a les maths... Écrit par un mathématicien, écrire des maths en \LaTeX est rapide et confortable.

Et c'est un logiciel libre :) Donc gratuit, maintenu par une communauté de programmeurs actives, documentation abondante sur Internet, ...

Les inconvénients de \LaTeX

- \LaTeX est un langage et nécessite donc un apprentissage
- changer radicalement la présentation par rapport aux présentations par défaut est une tâche parfois lourde
- \LaTeX repose sur des fichiers texte ce qui suppose quelques acrobaties entre les formats de fichiers texte (Latin1, ISO, UTF-8, ...)

Sur le principe...

Le fichier texte peut être créé et/ou modifié par n'importe quel éditeur de texte.

Puis dans l'interpréteur de ligne de commande, vous appelez un des compilateurs :

`latex,dvips` pour passer du fichier texte au fichier PostScript

`pdflatex` pour passer du fichier texte au fichier PDF

Le nom du compilateur peut varier selon les plates-formes et la version de \LaTeX utilisée.

Sur le principe...

Pour un fichier texte *monfichier.tex*, cela donnerait dans l'interpréteur ligne de commande sous Windows (Mik \TeX) :

Code

```
pdflatex monfichier.tex
```

pour obtenir depuis le fichier « monfichier.tex » le fichier PDF correspondant.

En vrai...

Les fichiers \LaTeX ne s'éditent pas avec le bloc-notes de *Windows*... Des éditeurs beaucoup plus conviviaux sont disponibles pour créer et modifier des fichiers \LaTeX . Ils permettent d'éditer confortablement les fichiers et de compiler les fichiers sans passer par un interpréteur de commande.

Les éditeurs de texte intelligents

Pourquoi utiliser ces éditeurs plutôt que le bloc-note ?

- Ils permettent de différencier visuellement les commandes \LaTeX du texte « brut »
- Ils permettent de ne pas passer par la console pour compiler
- Ils apportent une aide (raccourcis, génération de code \LaTeX , ...) lors de l'édition
- Beaucoup d'entre d'eux permettent d'utiliser et/ou de modifier l'encodage

Des éditeurs beaucoup plus conviviaux sont disponibles pour créer et modifier des fichiers \LaTeX .

En vrai...

Les éditeurs sont parfois appelés également environnements de développement intégrés (IDE).

Quelques exemples...

- \TeX Maker
- \TeX Works (toutes plates formes)
- \TeX nicCenter Windows
- Emacs (toutes plates formes)
- ...

\TeX Maker

\TeX Maker est très bien. Lorsque vous compilez de \LaTeX en Postscript, il fait apparaître le log et surtout les erreurs de compilation de façon très claire.

Il supporte les formats Latin1 et Unicode ce qui est plutôt bien pour *Windows* et *Mac OS X*. En effet, dans les préférences, on peut définir le type d'encodage que l'on souhaite utiliser.

\TeX Works

\TeX Works est similaire à \TeX Maker avec quelques fonctionnalités supplémentaires.

Une version portable, sans installation est disponible sur le site.

\TeX nicCenter

\TeX nicCenter a un gros avantage c'est qu'il compile la totalité du fichier \LaTeX même en présence d'erreurs... C'est le plus riche en matière de personnalisation de la compilation.

Il ne corrige pas vos fautes, mais vous permet de regarder en une seule passe s'il y a une ou plusieurs erreurs dans votre fichier \LaTeX : il suffit d'appuyer sur $F9$ et vous passez d'une erreur à l'autre tout au long du fichier log. C'est très pratique.

En revanche, il ne supporte pas les fichiers Unicode ce qui peut poser problème.

Oui mais le compilateur ?

Pour compiler un document \LaTeX sur votre poste de travail, vous avez la possibilité d'utiliser une version portable ou, si vous avez les droits administrateurs sur votre poste, d'installer une « distribution ».

Une distribution installe de nombreux compilateurs, logiciels, packages et modifie les valeurs système pour permettre une utilisation optimale de \LaTeX .

FramaKey, Paquet \LaTeX

La première possibilité est la FramaKey³. C'est une version portable. Elle ne nécessite aucune installation et n'exige pas que vous soyez administrateur de votre poste.

Il suffit de dé-zipper le fichier téléchargé sur le site de Framasoft. Les utilitaires et les compilateurs minimaux sont présents et correctement configurés.

La Framakey \LaTeX n'est pas très bien mise à jour. Le résultat est qu'il est difficile de s'en servir car les paquets correspondant à la version de \LaTeX sur la clef n'existe plus en ligne.

La FramaKey est donc à éviter pour l'instant ...

3. <http://framakey.org/Pack/PackLatex>

MiKTeX , version portable

La version portable de MiKTeX permet de réaliser des documents \LaTeX . Elle ne nécessite ni droits administrateurs ni installation comme la FramaKey.

Il suffit de dé-zipper le fichier téléchargé sur le site de Framasoft. Les utilitaires et les compilateurs minimaux sont présents et correctement configurés. La page indique comment utiliser la version portable. Notamment le lancement de l'environnement avec la petite application dans la barre des tâches. Le fichier est téléchargeable [ici](#).

Les distributions

La seconde possibilité est d'installer une distribution. Les distributions les plus communes par plate-formes sont :

- *Windows* MiKTeX
- *Mac OS X* MacTeX
- *Linux* TeXLive

MiKTeX

L'installation de MiKTeX , réservée aux administrateurs de leur poste. Par conséquent elle ne sera pas décrite ici. Sur demande, un document d'aide est toutefois fourni avec les documents du cours.

Structure d'un fichier \LaTeX

\LaTeX est un langage à balise qui n'est pas sans rappeler HTML. Comme HTML, il y a une première partie, en amont du texte qu'on appelle *preamble*. Il correspond au *header* de l'HTML. Après une commande \LaTeX spécifique, la seconde partie du document commence : c'est le corps du document contenant ce qui va être lisible dans le document final. Puis une commande \LaTeX termine le document.

Structure d'un fichier \LaTeX

La définition minimale d'un document \LaTeX est :

Code

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

Le préambule

Code

```
\documentclass{article}
\usepackage[latin1]{inputenc}
\usepackage[french]{babel}
\title{Introduction à \text{\LaTeX}}
\author{Pascal Bessonneau}
\date{DEPP, 12/2010.}
\definecolor{myblue}{rgb}{0,0.31,0.64}
\renewcommand\section{\@startsection {section}{1}{\z@}%
    {-3.5ex \@plus -1ex \@minus -.2ex}%
    {2.3ex \@plus .2ex}}
```

Le préambule

Le préambule va contenir :

1. La liste des paquets que doit charger \LaTeX
2. La définition ou la redéfinition des couleurs
3. La définition ou la redéfinition des commandes
4. Les informations concernant le titre, l'ouvrage, ...
5. ...

Le corps du document

Dans cette partie, sauf exception, le texte va être compilé et va apparaître dans le document.

Cette partie est définie par les commandes :

Code

```
...  
\begin{document}  
\maketitle  
...  
La preuve qu'il y a des êtres intelligents...  
essayés de nous contacter.  
...  
\end{document}
```

\LaTeX est un langage...

\LaTeX est un langage proche d'un langage de programmation par conséquent il faut respecter des conventions simples :

- pour que le compilateur \LaTeX puisse interpréter correctement les commandes
- pour vous faciliter le travail de recherche et corrections des erreurs
- pour faciliter la vie du(des) relecteur(s) du code

\LaTeX est un langage...

Par conséquent on retiendra les règles suivantes :

- toutes les commandes \LaTeX sont à taper en minuscules (en fait c'est une obligation \LaTeX est sensible à la casse)
- à l'intérieur et à l'extérieur du document, ne pas hésiter à laisser des espaces ou des tabulations pour augmenter la lisibilité
- ajouter des commentaires pertinents pour la relecture ou le débogage

Les packages

Les paquets (ou *packages*) sont un ensemble de macro-commandes. Ils sont un peu à l'image des macros INSEE de SAS ou des paquets de R.

Ces macros permettent de réaliser des mises en pages avancées, de changer la langue du document, de créer des tableaux personnalisés, ...

Ces ensembles de commandes \TeX et/ou \LaTeX permettent de réaliser des opérations sur le document ou les données associées. Vous devez les appeler dans le préambule.

Les packages

Les paquets sont au nombre de quelques milliers. Le problème est souvent de trouver le paquet qui fait ce que l'on souhaite. Les paquets officiels (stables) sont disponibles sur le site Comprehensive \TeX Archive Network.

Les packages

Lors de l'installation sous *Windows* avec \MikTeX , on peut installer (presque) l'ensemble des paquets (ce qui représente 1,2 Go). Sinon on peut les installer à la demande... Si vous utilisez un paquet non présent sur l'ordinateur, l'éditeur \LaTeX vous proposera automatiquement (avec le bon réglage) de l'installer.

Le paquet *inputenc*

A retenir et à placer dans tous vos documents :

Code

```
\usepackage[latin1]{inputenc}
```

Code

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

Le paquet *inputenc* permet de définir le format de fichier texte (Latin1, UTF-8, ...). Il est important car il permettra à \LaTeX de reconnaître notamment les caractères accentués.

Le paquet *babel*

A retenir et à placer dans tous vos documents rédigés en français :

Code

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

Le paquet Babel permet de *franciser* les guillemets, les césures, les accents, ...

Un package tout à fait inutile

Nous verrons des packages utiles mais voilà un exemple de package tout à fait inutile donc indispensable: \LaTeX Coffee Stains



L'encodage des caractères en quelques mots...

L'encodage des caractères provient du conflit entre les limites qui étaient imposées par les ordinateurs et la diversité des alphabets et des glyphes utilisés dans chaque langue.

Ainsi, les premiers jeux de caractères codés sur peu de bits ne contenaient pas les caractères accentués. Puis le codage des caractères a été réalisé sur un plus grand nombre de bits, offrant la possibilité de coder (presque tous) les caractères d'une langue.

Et ainsi de suite jusqu'à parvenir aux systèmes actuels qui permettent le codage d'un grand nombre de glyphes.

L'encodage des caractères en quelques mots...

ANSI, ISO-8859-x, Unicode... sont autant de normes utilisées pour représenter les caractères d'une langue.

Aujourd'hui, si *Windows* XP et certains logiciels utilisent la norme ISO *Western character*, ie. ISO-8859-1, la plupart des systèmes récents utilisent l'Unicode.

Unicode, plus récent et offrant un très large panel de caractères, est à préférer. Seul bémol, Unicode est dérivé en 8bits, 16bits et 32bits.

L'encodage des caractères en quelques mots...

Pourquoi parler de l'encodage? La raison est qu'il faudra lors des échanges de fichiers s'assurer d'utiliser le même encodage sous peine de voir des caractères exotiques ou des erreurs de compilation apparaître.

L'autre exemple type est le copier-coller. Le copier-coller de Word dans un éditeur de texte peut s'avérer problématique.

Indiquer à \LaTeX d'utiliser un autre format d'encodage

Il suffit de préciser l'encodage en début de document (le préambule) :

_____ Code _____
`\usepackage[latin1]{inputenc}`

Ici l'encodage est en *latin1* (ISO-8859-1). *utf8* correspond à Unicode (UTF8).

La conversion de fichier texte

Des utilitaires comme Notepad++ peuvent vous permettre de convertir le texte d'un format à l'autre.

Sinon des langages de programmation comme R, Perl et Python permettent de le faire également.

Pour faire de l'aide sur ce point, le terme exacte est translittération plutôt que conversion.