

# Org-Mode et L<sup>A</sup>T<sub>E</sub>X – Instructions et débogage

Charles-Édouard Lizotte

<2023-03-07 Tue>

## Contents

1	Écrire des équations et ajouter un soupçon de verbatim dans nos vie	1
2	Org-Mode, cases à cocher, énumérations, listes et raccourcis clavier	2
2.1	<b>TODO</b> Ceci est une sous-section avec des cases à cocher avec tag . . . . .	2
2.2	Fonctionnement des listes et documentation de Org-Mode . . . . .	2
3	Une installation sans pareil <span style="float:right">HARDCORE</span>	2
3.1	Installation du module TexLive . . . . .	2
3.2	Installation des modules ( <i>packages</i> ) L <sup>A</sup> T <sub>E</sub> X . . . . .	2
3.3	Solution . . . . .	3
4	Compiler un fichier Org en L <sup>A</sup> T <sub>E</sub> X à l'aide de PDFLatex <span style="float:right">EASY</span>	4
4.1	Compilation . . . . .	4
4.2	Débogage pré-compilation . . . . .	4
5	Org-Cite et compilation L <sup>A</sup> T <sub>E</sub> X	5
5.1	Déroulement d'une compilation et lien indirect avec les références . . . . .	5
6	Formattage des figures en Org-L <sup>A</sup> T <sub>E</sub> X	6
6.1	Une méthode simpliste pour inclure des figures . . . . .	6
6.2	Envelopper une figure dans du texte . . . . .	6
6.3	Erreurs fréquentes sur le positionnement des figures . . . . .	7

## 1 Écrire des équations et ajouter un soupçon de verbatim dans nos vie

On peut écrire un peu n'importe quoi. C'est pas très grave au fond. Un petit truc pour gérer les invite de commande. :

```
M-x comint-clear-buffer
```

permet de tout effacer d'un coup. MAGIQUE et MAGNIFIQUE.

Pour écrire une équation, il est possible de faire du L<sup>A</sup>T<sub>E</sub>X directement, par exemple

$$\int \psi(x, t) dx. \tag{1.1}$$

Mais à l'aide des **sources**, il est possible de mettre le texte en *verbatim*, de sorte que

```
\begin{equation}
\int \psi(x, t) dx,
\end{equation}
```

soit une équation utilisant le fameux package *physics*.

## 2 Org-Mode, cases à cocher, énumérations, listes et raccourcis clavier

### 2.1 TODO Ceci est une sous-section avec des cases à cocher avec tag

- ☐ Ici on va mettre des petits carrés.
- ☒ Ici on va mettre un sous-carré.
  - ☒ Et un autre sous-carré permettant de comprendre comment ça marche.

### 2.2 Fonctionnement des listes et documentation de Org-Mode

N'hésite pas à te référer à la [documentation](#) de Org-Mode. Elle est vraiment *clean* et bien ordonnée et il y a des raccourcis qui en valent vraiment la peine.

## 3 Une installation sans pareil

**HARDCORE**

### 3.1 Installation du module TexLive

En premier lieu, il faut s'assurer que L<sup>A</sup>T<sub>E</sub>X est installé sur l'ordinateur, généralement avec **TeXLive**. Donc,

```
>>> sudo apt install texlive
```

### 3.2 Installation des modules (*packages*) L<sup>A</sup>T<sub>E</sub>X

Ensuite, il risque de manquer quelques *packages*. Pour pallier à ce problème, nous utilisons la commande *tlmgr*. Malheureusement, il y a souvent des problèmes d'installation reliés à *tlmgr*, faut pas hésiter à essayer quand même,

```
>>> tlmgr install physics
(running on Debian, switching to user mode!)
(see /usr/share/doc/texlive-base/README.tlmgr-on-Debian.md)
```

```
tlmgr: Local TeX Live (2021) is older than remote repository (2022).
Cross release updates are only supported with
  update-tlmgr-latest(.sh/.exe) --update
See https://tug.org/texlive/upgrade.html for details.
```

Bref, ça marche pas. La commande

```
>>> latex --version
```

nous montre que la bonne édition de TexLive est installée (2023). Une recherche rapide sur **les internet** nous enseigne qu'il y a souvent des problèmes de versions entre ce que **TeXLive** a installé et **tlmgr** qui est généralement installé en même temps que Ubuntu. Même en installant *tlmgr* avec *apt-get*, on a l'impression que le problème persiste, comme les prochaines commandes nous le démontrerons. Regardons la version,

```
>>> tlmgr --version
tlmgr revision 60693 (2021-10-04 04:24:25 +0200)
tlmgr using installation: /usr/share/texlive
TeX Live (https://tug.org/texlive) version 2021
```

Bref, justement. Supprimons complètement TexLive de notre ordinateur (comme suggéré par un des utilisateurs de [Stack Exchange](#)) et on réinstalle littéralement tout (TeXLive et *tlmgr*) avec la commande

```
>>> sudo apt-get purge texlive-*
>>> sudo apt-get autoremove
>>> sudo apt-get autoclean
%
>>> sudo apt-get instal texlive-full
```

Si ça ne marche toujours pas – donc qu’on observe des dépendances non-résolues (*unmet dependencies*) – on peut utiliser le package [aptitude](#). **Aptitude** est une extension de **apt** qui trouve des solutions à l’installation pour les problèmes dépendance. Ça marche généralement bien, comme le confirme plusieurs utilisateurs de **Stack Exchange**.

```
>>> sudo apt-get update
>>> sudo apt-get install aptitude
>>> sudo aptitude install texlive-full
```

Pour confirmer qu’on ne fait pas n’importe quoi, on peut de nouveau essayer de vérifier les versions,

```
>>> latex --version
pdfTeX 3.141592653-2.6-1.40.22 (TeX Live 2022/dev/Debian)
kpathsea version 6.3.4/dev
Copyright 2021 Han The Thanh (pdfTeX) et al.
There is NO warranty. Redistribution of this software is
covered by the terms of both the pdfTeX copyright and
the Lesser GNU General Public License.
For more information about these matters, see the file
named COPYING and the pdfTeX source.
Primary author of pdfTeX: Han The Thanh (pdfTeX) et al.
Compiled with libpng 1.6.37; using libpng 1.6.37
Compiled with zlib 1.2.11; using zlib 1.2.11
Compiled with xpdf version 4.03
```

```
>>> tlmgr --version
tlmgr revision 60693 (2021-10-04 04:24:25 +0200)
tlmgr using installation: /usr/share/texlive
TeX Live (https://tug.org/texlive) version 2021
```

Malheureusement, nous avons toujours le même problème de versions, ce qui confirme que le répertoire (*repository*) est complètement détraqué. On peut tenter de mettre à jour (*update*) le module **tlmgr** à l’aide de la commande

```
>>> sudo tlmgr update --self --all
(running on Debian, switching to user mode!)
(see /usr/share/doc/texlive-base/README.tlmgr-on-Debian.md)
TLPDB: not a directory, not loading: /root/texmf
tlmgr: user mode not initialized, please read the documentation!
```

On constate que ça ne marche toujours pas. On peut quand même *gosser* pour trouver où est situé TLPDB. Un autre utilisateur de [Stack Exchange](#) propose

```
>>> sudo tlmgr init-usertree
>>> sudo tlmgr update --self --all
(running on Debian, switching to user mode!)
(see /usr/share/doc/texlive-base/README.tlmgr-on-Debian.md)

tlmgr: Local TeX Live (2021) is older than remote repository (2022).
Cross release updates are only supported with
  update-tlmgr-latest(.sh/.exe) --update
See https://tug.org/texlive/upgrade.html for details.
```

Bref, peu d’avancement. Au final, impossible de mettre le module à jour, car les versions sont différentes, quelle ironie du sort...

### 3.3 Solution

Après une autre heure de recherche, j’ai trouvé la [solution](#) (qui est en fait le lien proposé par *tlmgr* lui-même).

1. On commence par purger tout ce qui est en lien avec **texlive** (comme on a fait précédemment).
2. On installe le tar.gz comme proposé sur le site – Ça devrait prendre un bon 30 minutes d’installation.
3. On ajoute quelques chemins à notre **PATH** en ajoutant les lignes

```
# >>> Chemin pour TeXLive et tlmgr >>>
export PATH="/usr/local/texlive/2022/bin/x86_64-linux:$PATH"
export MANPATH="/usr/local/texlive/2022/texmf-dist/doc/man:$MANPATH"
export INFOPATH="/usr/local/texlive/2022/texmf-dist/doc/info:$INFOPATH"
```

à notre fichier *.bashrc*. Bref, **maintenant**, tout devrait être en ordre!

Maintenant, on teste les versions pour être sur, on obtient

```
>>> tlmgr --version
tlmgr revision 63068 (2022-04-18 07:58:07 +0200)
tlmgr using installation: /usr/local/texlive/2022
TeX Live (https://tug.org/texlive) version 2022
>>> latex --version
pdfTeX 3.141592653-2.6-1.40.24 (TeX Live 2022)
kpathsea version 6.3.4
Copyright 2022 Han The Thanh (pdfTeX) et al.
There is NO warranty. Redistribution of this software is
covered by the terms of both the pdfTeX copyright and
the Lesser GNU General Public License.
For more information about these matters, see the file
named COPYING and the pdfTeX source.
Primary author of pdfTeX: Han The Thanh (pdfTeX) et al.
Compiled with libpng 1.6.37; using libpng 1.6.37
Compiled with zlib 1.2.11; using zlib 1.2.11
Compiled with xpdf version 4.03
```

Par la suite, on peut finalement installer les modules (*packages*) nécessaires à l’aide de la commande

```
>>> sudo $(which tlmgr) install physics
```

## 4 Compiler un fichier Org en L<sup>A</sup>T<sub>E</sub>X à l’aide de PDFLatex

EASY

### 4.1 Compilation

1. **M-X org-export-dispatch** et on gosse dans la section L<sup>A</sup>T<sub>E</sub>X, avec les touches nécessaires montrées à l’écran.

Par exemple, on devrait avoir exporté un nouveau fichier en **tex**.

1. Il est aussi possible de dérouler à travers les anciens **M-x** à l’aide de **M-p**.
2. J’ai oublié. . .

### 4.2 Débogage pré-compilation

Si le fichier *pdf* ou *tex* n’apparaît pas, un buffer emacs **Org PDF L<sup>A</sup>T<sub>E</sub>X Output** devrait déjà être présent sur les lieux du crime. Si ça ne suffit pas, le fichier *tex* dans le répertoire courant devrait nous renseigner sur les erreurs.

## 5 Org-Cite et compilation L<sup>A</sup>T<sub>E</sub>X

### 5.1 Déroulement d’une compilation et lien indirect avec les références

Juste pour se situer dans le schéma narratif de la compilation de fichiers Org-L<sup>A</sup>T<sub>E</sub>X :

- Emacs compile le langage *Org* dans fichier *tex*.
- Ensuite, le compilateur *pdflatex* compile le fichier L<sup>A</sup>T<sub>E</sub>X en *pdf*.

Donc, lorsqu’on crée une ou des citations à l’aide de la commande **M-x org-cite-insert**, Emacs crée un lien symbolique vers une entrée dans un fichier *.bib*. Lors de la compilation en fichier PDF, Emacs exporte le liens dans un formatage compréhensible pour un compilateur L<sup>A</sup>T<sub>E</sub>X. C’est pourquoi il faut signaler le méthode d’exportation des références et citations à l’aide de la commande

```
#+cite_export: bibtex
```

Dans cet exemple, on dit à Emacs que toutes nos références se transformeront en citations compréhensibles pour le module *biblatex*. Personnellement, j’ai l’habitude d’utiliser *natbib*, donc je mentionne

```
#+cite_export: natbib
```

Dans mon préambule, je mentionne donc à mon compilateur L<sup>A</sup>T<sub>E</sub>X d’utiliser le module *natbib* de sorte à savoir comment lire les citations exportées par Emacs, justement sous le format de *natbib* comme désiré, soit

```
/usepackage{natbib}
```

Finalement, *pdflatex* passe au moins 3 fois sur le texte pour, dans un premier temps :

- Ramasser les citation et les références,
- Créer un fichier *bbl* qui contiendra notre bibliographie
- Apposer la bibliographie et les liens sur le texte avec les bonnes références.

C’est d’ailleurs pourquoi on fait mention au nombre de compilations dans le *buffer* **Org PDF L<sup>A</sup>T<sub>E</sub>X Output**.



Figure 1: Ce magnifique chat fera titre d’exemple.

## 6 Formattage des figures en Org-L<sup>A</sup>T<sub>E</sub>X

### 6.1 Une méthode simpliste pour inclure des figures



Figure 2: *Ce pauvre chat est enrobé par le texte.*

Il est raisonnablement simple d'ajouter des figures à notre fichier L<sup>A</sup>T<sub>E</sub>X en Org-Mode. La méthodologie est légèrement différente qu'en compilation L<sup>A</sup>T<sub>E</sub>X. Sur Overleaf, la complétion de texte rend le travail assez simple. Mais en absence de cette fonctionnalité sur Emacs (en date d'écriture de ce texte), je serai honnêtement prêt à dire que c'est plus simple en Org qu'en L<sup>A</sup>T<sub>E</sub>X. Pour ajouter une figure, il n'y a qu'à suivre la méthodologie suivante :

- On procède de la même manière que pour l'ajout d'un lien symbolique. On appuie donc sur les touches **Ctrl-x Ctrl-l**.
- Ensuite, on entre la commande « *file* » pour mentionner qu'on recherche un fichier et on appuie sur **Ret**.
- Le mini-buffer nous permet ensuite de choisir un fichier quelconque. Il faut juste s'assurer de ne pas entrer de description à ce lien symbolique.
- Une fois achevé, en Org-Mode, on devrait voir notre fichier entre doubles crochets.

Par la suite, il est extrêmement pertinent de :

- Nommer notre figure à l'aide de la notation Org,
- Ajouter une description à notre figure,
- De dicter la taille de la figure, du même usage qu'on ferait dans un compilateur L<sup>A</sup>T<sub>E</sub>X.

Le tout devrait s'harmoniser dans un arrangement de quatre courtes lignes; une victoire en comparaison de tous les environnements L<sup>A</sup>T<sub>E</sub>X qu'il aurait fallu appeler, soit

```
#+NAME: fig:chat1
#+CAPTION: Ce magnifique chat fera titre d'exemple.
#+ATTR_LATEX: :width 0.4\textwidth
[[file:figures/pexels-photo-4587959.jpeg]]
```

Le résultat direct de l'exécution des lignes précédentes par le compilateur L<sup>A</sup>T<sub>E</sub>X devrait se retrouver à la figure précédent le début de cette section. Dinalement, à l'aide de la variable NAME, il est facile de créer une référence direct à cette image (voir figure 1).

### 6.2 Envelopper une figure dans du texte

Il est assez simple Il est possible de faire des *wrapfig* en Org-Mode aussi (Voir [documentation](#)). Pour réaliser cette action, on suit la précédente méthodologie, mais on se permet de jouer sur les attribus L<sup>A</sup>T<sub>E</sub>X, comme illustré à la figure 2.

```
#+NAME: fig:chat2
#+CAPTION: Ce pauvre chat est enrobé par le texte.
#+ATTR_LATEX: :float wrap :width 0.37\textwidth :placement [20]{r}{0.38\textwidth} \vspace{-\baselineskip}
[[file:figures/pexels-photo-4587955.jpeg]]
```

Observons le placement des attribus que nous avons ajouté à la figure 2, soit

```
#+ATTR_LATEX: :width 0.23\textwidth :float wrap :placement {r}{0.25\textwidth} \vspace{-\baselineskip}
```

La commande *vspace* est utilisée pour **mettre à niveau** notre figure en relation avec le début du paragraphe et la commande *baselineskip* est associée à la distance (Voir [StackOverflow](#)). Comme la commande est ajoutée à droite du *ATTR-LATEX*, cette dernière sera ajoutée à l'intérieur de l'environnement *figure* dans notre fichier L<sup>A</sup>T<sub>E</sub>X, comme en font état les lignes suivantes :

```
\begin{wrapfigure}[12]{r}{0.25\textwidth} \vspace{-\baselineskip}
\centering
\includegraphics[width=0.23\textwidth]{figures/pexels-photo-4587955.jpeg}
\caption{\label{fig:orgb0e0af3}Ce pauvre chat est enrobé par le texte.}
\end{wrapfigure}
```

### 6.3 Erreurs fréquentes sur le positionnement des figures

Pour optimiser le placement des figures – surtout lorsqu'elles sont enveloppées par du texte, il est fortement conseillé de **ne jamais utiliser la commande**

`\`

Car c'est une commande de type « caméléon » qui tente au mieux de s'adapter à n'importe quelle situation (texmode, mathmode, etc.). Cette dernière est donc redéfini dans beaucoup de modules et à toutes les sauces et son [utilisation](#) induit des erreurs de type

```
[2] Underfull \hbox (badness 10000) in paragraph at lines 258--260
```

qui se résument par une justification forcée du texte, ce qui est esthétiquement catastrophique. Il est donc préférable de toujours utiliser les commandes

```
\newline
\bigskip
```

pour séparer des paragraphes.