

How to compile and work with `mazhe.tex` ?

Laurent Claessens

September 28, 2015

Contents

1	Pour l'agrégation	1
1.1	Ce qu'il faut télécharger	1
1.2	Compiler tout le document	2
1.3	Compiler seulement une partie	2
	Tout commence par télécharger les sources à l'adresse	

<https://github.com/LaurentClaessens/mazhe>

Pour information le numéro du commit qui correspond à ce qui a été envoyé à l'agrégation en septembre 2015 est `c038d9eee475b3a1d6e5414a698d50007dc69af0`.

1 Pour l'agrégation

Le plus facile est de télécharger la paquet `exocorr.sty` (voir plus bas) puis de compiler avec :

```
pdflatex -synctex=1 -shell-escape Inter_frido-mazhe_pytex.tex
```

et quelque fois `bibtex` et `makeindex`.

Ce faisant vous n'aurez cependant pas les moyens de changer l'ordre des chapitres¹ parce que ce fichier est créé automatiquement par quelque scripts de précompilation. En travaillant de la sorte vous aurez la possibilité d'effectuer des modifications dans les fichiers inclus.

Vu que de mon côté ce fichier est recréé automatiquement à chaque compilation, un `git pull` écrasera les modifications que vous y auriez apporté. Si vous voulez savoir la raison de ce fait, faites

```
pdflatex mazhe.tex
```

Voyant le résultat vous comprendrez pourquoi compiler le document destiné à l'agrégation demande un peu de travail.

1.1 Ce qu'il faut télécharger

Pour avoir une maîtrise plus fine de la compilation, vous devrez télécharger un certain nombre de choses.

Le paquet `exocorr` Vous devez récupérer le paquet `exocorr` à l'adresse

<https://github.com/LaurentClaessens/exocorr>

1. Ou en tout cas pas de me soumettre vos changements parce que c'est dans `mazhe.tex` que je travaille.

Seul le fichier `.sty` vous intéresse a priori. Mettez-le là où vous mettez vos paquets L^AT_EX.

Le module LaTeXparser Vous le téléchargez à l'adresse

<https://github.com/LaurentClaessens/LaTeXparser>

et vous le mettez quelque part là où Python pourra le retrouver.

Le script pytex Il est à l'adresse

<https://github.com/LaurentClaessens/pytex>

1.2 Compiler tout le document

Lorsque tout est téléchargé et correctement configuré (LaTeXparser doit être trouvable par python et pytex trouvable par votre terminal), vous compilez le Frido avec

```
pytex lst_frido.py --lotex
```

Le script s'occupe d'extraire de `mazhe.tex` les choses nécessaires au Frido, crée un fichier intermédiaire et le compile. Des passes de `bibtex` et `makeindex` sont également automatiquement effectuées.

Les `ref` et `eqref` ne correspondant à aucun `label` sont indiqués. Il ne devrait y en avoir aucun.

La compilation produit deux fichiers `pdf`. Le premier est `Inter_frido-mazhe_pytex.pdf` qui est créé par L^AT_EX lui-même durant la compilation. Le second est `0-lefrido.pdf` qui en est une simple copie effectuée après la compilation. Vous devriez ouvrir `0-lefrido.pdf` de façon à éviter que votre lecteur de `pdf` ne se mette en mode «rafraichissement» durant toute la durée de la compilation.

1.3 Compiler seulement une partie

<++>