

Description du langage de markdown à utiliser avec le programme

lmd

Simon Gustin

Le 22 juin 2016

Première partie

Programme *lmd*

1 Présentation du programme

Le programme **lmd** est un compilateur qui transforme un fichier d'un langage de markdown en code *LaTeX*, et ensuite à le compiler en *pdf* à l'aide de `pdflatex`.

Il n'est disponible que sous *Linux* (et éventuellement sur d'autres *OS Unix-like*)

2 Utilisation

Ce programme s'utilise en ligne de commande, de la façon suivante :

```
CHEMIN_VERS_LE_PROGRAMME [OPTIONS] CHEMIN_VERS_LE_FICHIER_A_TRADUIRE
```

Le chemin vers le programme dépend de l'ordinateur, l'exécutable s'appelant par défaut *lmd*.

Il est possible d'ajouter le répertoire dans lequel se trouve cet exécutable à la variable d'environnement `PATH` (voir ci-dessous pour savoir comment faire cela).

Le chemin vers le fichier à traduire est le chemin (absolu ou relatif) vers le fichier `.lmd` à traduire. Le programme s'exécutera alors avec le répertoire qui contient ce fichier comme répertoire de travail (*working directory*).

Les différentes options sont :

- `--help`, à utiliser seul, permet d'afficher une aide à l'utilisation du programme (la même que celle disponible ici) ;
- `-o` ou `--output-name`, permet de spécifier le nom des fichiers (*pdf* ou *tex*) qui seront produits par le programme ;
- `-t` ou `--output-type`, permet de spécifier le type de fichiers que le programme doit outputer (*pdf*, *tex* ou *tex-pdf*) ;
- `-K` ou `--keep-tmp-files`, demande au programme de n'effacer aucun fichier temporaire (donc le fichier `LMD-xxxxxxxxx.tmp` créé par le compilateur et tous les fichiers créés par `pdflatex` : *file.log*, *file.aux*, etc) ;
- `-k` ou `--keep-pdflatex-tmp-files`, même chose mais uniquement pour les fichiers produits par `pdflatex` ;
- `--keep-lmd-tmp-files`, même chose, uniquement pour le fichier temporaire créé par le compilateur ;
- `--print-pdflatex-stdout`, demande à `pdflatex` d'afficher sa sortie sur la sortie standard (comme si `pdflatex` avait été appelé en ligne de commande).

Pour les options qui ont besoin d'un argument, elles s'utilisent de la même façon qu'une commande habituelle sous *Unix-like*. On a donc 2 cas :

1. Les options à une lettre demande que l'argument suivant du programme soit l'argument demandé par cette option. Par exemple, `-o file`.
2. Pour les options à plusieurs lettres, il faut séparer le nom de l'option et l'argument qu'elle demande par le signe '=', et ce sans espaces. Par exemple, `--output-type=pdf`.

Si chacune des options n'est pas spécifiée, les valeurs par défaut sont les suivantes :

- Le fichier d'*output* aura le même nom que le fichier d'*input*, avec l'extension adéquate.
- Le type de fichier qui sera produit est par défaut un fichier *tex* et un fichier *pdf* (option `tex-pdf` ou `pdf-tex`).
- Aucun fichier temporaire n'est gardé par défaut.
- La sortie standard n'affiche pas ce que `pdflatex` sort sur sa sortie standard par défaut.

Une fois la commande exécutée, le fichier `lmd` est traduit en fichier `tex`, puis ce dernier est compilé en `pdf`, et les fichiers qui doivent être effacés sont effacés. Le fichier demandé est produit dans le même répertoire que le fichier `lmd` (ou dans un sous-répertoire dans le cas où ça a été demandé).

3 Ajouter *lmd* au PATH

Pour pouvoir utiliser *lmd* en appelant simplement `lmd` en ligne de commande, on peut ajouter le répertoire qui le contient dans la variable d'environnement `PATH` ou le mettre dans un répertoire qui est déjà inclus dans `PATH` (taper `$PATH` dans un terminal pour voir la valeur de cette variable).

Pour ajouter un répertoire spécifique à la variable `PATH` et ce de façon permanente, on va procéder comme suit :

- Ajouter la ligne `export PATH=$PATH:RÉPERTOIRE_À_AJOUTER` au fichier `~/ .bashrc`.
- Pour que la modification prenne effet :
 - Soit on redémarre la session ;
 - Soit on exécute `source ~/ .bashrc`.

Deuxième partie

Description du langage de markdown