

Babel

Michel Casabianca
michel.casabianca@gmail.com

Babel est une boîte à outils permettant de gérer vos langages de programmation favoris.

Par exemple, pour builder et installer GO en version *1.5.1*, on tapera :

```
go-build 1.5.1
```

Pour utiliser cette version, lancer le script *version* qui liste les versions installées et permet de choisir celle à utiliser :

```
$ version go
Please choose a version:
0: Default
1: 1.0.3
2: 1.1.2
3: 1.2.2
4: 1.3.3
5: 1.4.2 *
6: 1.5.1
6
Selecting go version '1.5.1'
$ go version
go version go1.5.1 linux/amd64
```

Cet outil est capable de builder et sélectionner les langages suivants :

- Go
- Python
- Ruby
- Java
- Lua
- Rust

Comment ça marche

Les script de build compilent et installent les VMs ou compilateurs dans le répertoire */opt/<lang>/<version>*. Donc GO *1.5.1* sera installé dans le répertoire */opt/go/1.5.1*.

Le script *version* sélectionne une version en effectuant un lien symbolique entre */opt/<lang>/<version>* et */opt/<lang>/current*.

Par conséquent, si vous placez le répertoire `/opt/<lang>/current/bin` dans votre `PATH`, vous lancerez la version sélectionnée.

Ce système est simple et fonctionne parfaitement.

Installation

Pour installer Babel :

- Placez les scripts quelque part dans votre `PATH` (dans `/opt/bin/` par exemple).
- Créez les répertoires `/opt/go`, `/opt/lua`, `/opt/python`, `/opt/ruby` et/ou `/opt/java` suivant les langages que vous souhaitez gérer avec Babel.

Installez vos VMs avec le script `<lang>-build version` (excepté pour Java que l'on installera déjà compilé dans le répertoire `/opt/java/<version>`).

Langages

GO

GO n'a pas de VM, donc :

- Le script `go-build` va installer le compilateur et les outils associés.
- Le script `version go` permettra de sélectionner la version à utiliser.

Vous devez définir les variables d'environnement suivantes :

- `PATH` : doit inclure `/opt/go/current/bin`.
- `GOROOT` : doit être `/opt/go/current`.
- `GOPATH` : vous pouvez lui affecter la valeur que vous voulez (`~/go` for instance).

Note

Même si les tests échouent lors de l'installation (ce qui se produit pour la version `1.1` par exemple), le compilateur est installé.

Python

- Le script `python-build` va installer la VM pour la version passée en argument.
- Le script `version python` permettra de sélectionner la version à utiliser.

Pour installer un VM Python, vous devez installer les bibliothèques de développement suivantes :

- `tetex-nox` et `latex2html` pour générer la documentation des versions 2.6 et antérieures.
- `libreadline-dev` pour permettre la complétion et l'édition des lignes dans l'interpréteur.
- `libssl-dev` pour activer HTTPS.
- `libzip-dev`, `libtar-dev`, `libbz2-dev` et `zlib1g-dev` pour permettre le support de ZIP et TAR.
- `libmysqld-dev`, `libmysqlclient-dev` et `python-dev` pour installer le driver MySQL.

- *libsqlite3-dev* pour activer le support de *sqlite3*.

Ruby

- Le script *ruby-build* va installer la VM pour la version passée en argument.
- Le script *version ruby* permettra de sélectionner la version à utiliser.

Java

- Il n'y a pas de script de build : installer le contenu de l'archive dans le répertoire */opt/java/<version>*.
- Le script *version java* permettra de sélectionner la version à utiliser.

Vous devez définir les variables d'environnement suivantes :

- *PATH* : doit inclure */opt/go/current/bin*.
- *JAVA_HOME* : doit être */opt/java/current*.

Lua

- Le script *lua-build* va installer la VM pour la version passée en argument.
- Le script *version lua* permettra de sélectionner la version à utiliser.

Rust

- Le script *rust-build* va installer la VM pour la version passée en argument (ce qui est très long).
- Le script *version rust* permettra de sélectionner la version à utiliser.

Téléchargement

Vous pouvez télécharger Babel à l'adresse suivante : <https://github.com/c4s4/babel/releases>.

Enjoy!