

Maven Lite

Version anglaise

- Version : 2.0.0

Table des matières

- [Maven Lite](#)
 - [Table des matières](#)
 - [Compatibilité](#)
 - [Description](#)
 - [Prérequis](#)
 - [Installation et désinstallation](#)
 - [Debian et dérivés](#)
 - [Installation automatique - Debian](#)
 - [Installation manuelle - Debian](#)
 - [Désinstallation automatique - Debian](#)
 - [Désinstallation manuelle - Debian](#)
 - [Autres distributions Linux](#)
 - [Installation automatique - Linux](#)
 - [Installation manuelle - Linux](#)
 - [Désinstallation automatique - Linux](#)
 - [Désinstallation manuelle - Linux](#)
 - [Windows](#)
 - [Installation automatique - Windows](#)
 - [Installation manuelle - Windows](#)
 - [Désinstallation automatique - Windows](#)
 - [Désinstallation manuelle - Windows](#)
 - [MacOs](#)
 - [Installation automatique - MacOS](#)
 - [Installation manuelle - MacOS](#)
 - [Désinstallation automatique - MacOS](#)
 - [Désinstallation manuelle - MacOS](#)
 - [Utilisation](#)
 - [Exemple, fonctionnalités et limites](#)

Compatibilité

- Windows 10 (Non testé)
- Windows 11 (Non testé)
- Linux possédant `bash` (Testé sur Ubuntu 23.10 Mantic Minotaur)
- MacOS Mojave 10.14.6 et inférieur (Non testé)
- MacOS Catalina 10.15 et supérieur à condition d'avoir installé `bash` (Non testé)

Description

Maven Lite est un système de gestion de projet java qui permet de compiler et lancer des projets java très simplement et rapidement.

Le but est de pouvoir compiler et/ou lancer un projet java en une seule courte commande, comme maven mais en plus simple.

Maven Lite est plus simple que Maven car il ne gère pas les dépendances hors local, les plugins, les phases de build, les déploiements automatique, etc. Il est donc adapté pour les projets simples qui ne nécessitent pas de gérer des dépendances externes, cependant Maven Lite gère les dépendance local sous forme de fichier jar. Pour des projets plus complexes, il est conseillé d'utiliser Maven.

Maven lite permet de compiler le projet dans un dossier target, gérer les dépendances locales (fichier jar) et exécuter le projet.

L'intérêt principale de maven lite dans la gestion des dépendances est de fournir un dossier qui contient toutes les dépendances, le script ajoutera automatiquement toutes les dépendances dans le classpath.

Il est possible que cette deuxième version de Maven Lite contienne des bugs. Si vous en trouvez, n'hésitez pas à les signaler.

Prérequis

- Disposer des droits administrateurs si vous voulez utiliser les installations automatiques.

Si vous ne disposez pas des droits administrateurs, vous pouvez installer maven lite manuellement en suivant les instructions de la partie 'Installation manuelle'.

Installation et désinstallation

Debian et dérivés

Installation automatique - Debian

- Téléchargez le [fichier debian](#) dans la langue de votre choix dans la section 'Releases' du dépôt Github
 - [Téléchargez le fichier de la version française.](#)
 - [Téléchargez le fichier de la version anglaise.](#)
- Exécutez le fichier debian

```
sudo dpkg -i maven-lite_<langue>_2.0-1_all.deb
```

- Une fois l'installation terminée, vous pouvez supprimer le fichier `maven-lite_<langue>_2.0-1_all.deb` si vous le souhaitez.

Installation manuelle - Debian

- Téléchargez le [Fichier compressé](#) contenant les fichiers de l'applications dans la langue de votre choix dans la section 'Releases' du dépôt Github
 - [Téléchargez le fichier de la version française.](#)
 - [Téléchargez le fichier de la version anglaise.](#)
- Décompressez le fichier compressé
- Déplacez le dossier `maven-lite` dans le dossier `/etc`

```
sudo mv maven-lite /etc/
```

- Déplacez les fichiers shell dans le dossier `/usr/local/bin`

```
sudo mv mvnl /usr/local/bin/  
sudo mv mvnl-uninstall /usr/local/bin/
```

- Ajoutez les droits d'exécution au fichier `mvnl` et `mvnl-uninstall`

```
sudo chmod +x /usr/local/bin/mvnl*
```

- Déplacez le dossier `man` dans le dossier `/usr/local/share` pour avoir les pages de manuel

```
sudo mv man /usr/local/share/
```

- Vous pouvez supprimer le reste des fichiers inutilisés.

Désinstallation automatique - Debian

- Exécutez la commande suivante

```
mvnl-uninstall
```

Désinstallation manuelle - Debian

- Supprimez le dossier `/etc/maven-lite`

```
sudo rm -r /etc/maven-lite
```

- Supprimez les fichiers shell

```
sudo rm /usr/local/bin/mvnl /usr/local/bin/mvnl-uninstall
```

- Supprimez les pages de manuel

```
sudo rm /usr/local/share/man/fr/man1/mvnl.1.gz /usr/local/share/man/en/man1/mvnl.1.gz
```

Autres distributions Linux

Installation automatique - Linux

- Téléchargez le [Fichier compressé](#) contenant les fichiers de l'application dans la langue de votre choix dans la section 'Releases' du dépôt Github
 - [Téléchargez le fichier de la version française.](#)
 - [Téléchargez le fichier de la version anglaise.](#)
- Décompressez le fichier compressé
- Exécutez le script d'installation

```
sudo ./installer.sh
```

- Vous pouvez supprimer le reste des fichiers inutilisés.

Installation manuelle - Linux

- Téléchargez le [Fichier compressé](#) contenant les fichiers de l'application dans la langue de votre choix dans la section 'Releases' du dépôt Github
 - [Téléchargez le fichier de la version française.](#)
 - [Téléchargez le fichier de la version anglaise.](#)
- Décompressez le fichier compressé
- Déplacez le dossier `maven-lite` dans le dossier `/etc`

```
sudo mv maven-lite /etc/
```

- Ajoutez les droits d'exécution au fichier `mvnl` et `mvnl-uninstall`

```
sudo chmod +x /usr/local/bin/mvnl*
```

- Déplacez le dossier `man` dans le dossier `/usr/local/share` pour avoir les pages de manuel

```
sudo mv man /usr/local/share/
```

- Vous pouvez supprimer le reste des fichiers inutilisés.

Désinstallation automatique - Linux

- Exécutez la commande suivante

```
mvnl-uninstall
```

Désinstallation manuelle - Linux

- Supprimez le dossier `/etc/maven-lite`

```
sudo rm -r /etc/maven-lite
```

- Supprimez les fichiers shell

```
sudo rm /usr/local/bin/mvnl /usr/local/bin/mvnl-uninstall
```

- Supprimez les pages de manuel

```
sudo rm /usr/local/share/man/fr/man1/mvnl.1.gz /usr/local/share/man/en/man1/mvnl.1.gz
```

Windows

Installation automatique - Windows

Installation manuelle - Windows

Désinstallation automatique - Windows

Désinstallation manuelle - Windows

MacOs

Installation automatique - MacOS

- Téléchargez le [Fichier compressé](#) contenant les fichiers de l'application dans la langue de votre choix dans la section 'Releases' du dépôt Github
 - [Téléchargez le fichier de la version française.](#)
 - [Téléchargez le fichier de la version anglaise.](#)

- Décompressez le fichier compressé
- Exécuter le script d'installation

```
sudo ./installer.sh
```

- Vous pouvez supprimer le reste des fichiers inutilisés.

Installation manuelle - MacOS

- Téléchargez le [Fichier compressé](#) contenant les fichiers de l'application dans la langue de votre choix dans la section 'Releases' du dépôt Github
 - [Téléchargez le fichier de la version française.](#)
 - [Téléchargez le fichier de la version anglaise.](#)
- Décompressez le fichier compressé
- Déplacez le dossier `maven-lite` dans le dossier `/etc`

```
sudo mv maven-lite /etc/
```

- Ajoutez les droits d'exécution au fichier `mvnl` et `mvnl-uninstall`

```
sudo chmod +x /usr/local/bin/mvnl*
```

- Déplacez le dossier `man` dans le dossier `/usr/local/share` pour avoir les pages de manuel

```
sudo mv man /usr/local/share/
```

- Vous pouvez supprimer le reste des fichiers inutilisés.

Désinstallation automatique - MacOS

- Exécutez la commande suivante

```
mvnl-uninstall
```

Désinstallation manuelle - MacOS

- Supprimez le dossier `/etc/maven-lite`

```
sudo rm -r /etc/maven-lite
```

- Supprimez les fichiers shell

```
sudo rm /usr/local/bin/mvnl /usr/local/bin/mvnl-uninstall
```

- Supprimez les pages de manuel

```
sudo rm /usr/local/share/man/fr/man1/mvnl.1.gz /usr/local/share/man/en/man1/mvnl.1.gz
```

Utilisation

- placer vous dans le dossier de votre projet java et exécuter la commande suivante :

```
mvnl [options] [arguments]
```

Exemple, fonctionnalisés et limites

Vous avez un projet java dans le dossier 'C:\Users\user\Documents\MonProjetJava' Tout le code source se trouve dans le dossier 'src' et les fichiers .class doivent être compilés dans le dossier 'bin'. Votre projet contient des dépendances sous forme de fichier jar qui se trouve dans le dossier 'lib'.

Pour compiler le projet, vous devez ouvrir un terminal, vous placer dans le dossier 'C:\Users\user\Documents\MonProjetJava' puis exécuter la commande suivante :

```
mvnl -s src -o bin -d lib -c
```

Maintenant que votre projet est compilé, vous voulez lancer la classe principal 'Main.java' dans le package 'com.exemple'. Pour cela, vous devez exécuter la commande suivante :

```
mvnl -cp bin -d lib -m com.exemple.Main -l
```

Il est possible de combiner les deux commandes en une seule :

```
mvnl -s src -o bin -d lib -cp bin -m com.exemple.Main -cl
```

Comme vous pouvez le voir, ça fait une commande assez longue à taper. C'est pour cela que vous pouvez créer un fichier de configuration, par exemple 'config.txt' puis mettre les arguments dedans.

Le fichier de configuration doit contenir les options et argument de la commande. Soit une option avec son argument par ligne, soit plusieurs options et arguments sur la même ligne séparé par un espace.

Les options du fichier de configuration prévalent sur les options de la ligne de commande, il n'est donc pas possible de modifier une option dans la ligne de commande si elle est déjà présente dans le fichier de configuration. Cependant, il est possible de mettre une partie des options dans le fichier de configuration et le reste dans la ligne de commande.

Exemple de fichier de configuration :

```
-s src -o bin  
-d lib -cp bin -m com.exemple.Main
```

- Une fois que vous avez créé le fichier de configuration, vous pouvez exécuter la commande suivante pour compiler et lancer le projet :

```
mvnl -f config.txt -cl
```

Je vous recommande de ne pas mettre les options '-c' et '-l' dans le fichier de configuration car vous n'avez pas besoin de compiler et lancer le projet à chaque fois. Cela vous permettra de les mettre dans la commande quand vous en aurez besoin. Malgré tout, vous pouvez les mettre dans le fichier de configuration si vous le souhaitez.

- Pour compiler le projet, vous pouvez exécuter la commande suivante :

```
mvnl -f config.txt -c
```

- Pour lancer le projet, vous pouvez exécuter la commande suivante :

```
mvnl -f config.txt -l
```

Les options inutiles comme l'option -s pour lancer le projet seront ignorées, donc pas de problème si vous les laissez dans le fichier de configuration.

L'ordre des options n'a pas d'importance sauf les options -f, -h et -v qui doivent être la première option si elles sont utilisées.