

8

Étendre Lua avec LuaRocks

On dit souvent que l'une des raisons pour lesquelles Lua est si compact et rapide c'est qu'il est très léger. En fait, la distribution standard ne comprend que le langage de base et les bibliothèques standards essentielles, de sorte qu'il n'est pas surchargé. Cependant, cela signifie que beaucoup de fonctionnalités présentes par défaut dans d'autres langages ne sont simplement pas là : les sockets, le SQL, le support pour divers formats de fichiers... et même la manipulation des répertoires ! À ce propos, nous avons l'habitude de dire ce qui est souvent écrit sur les emballages de gadgets : "piles non incluses". Ainsi, pour être productif avec Lua, vous aurez souvent à inclure les "piles" qui manquent pour obtenir un environnement de travail complet, adapté à votre besoin.

Heureusement, ces "piles" sont disponibles : ce sont des modules d'extension Lua, que vous pouvez facilement télécharger et intégrer à votre environnement avec l'aide de LuaRocks, le gestionnaire de paquets Lua. Une fois LuaRocks installé, toutes les fonctionnalités supplémentaires dont vous avez besoin sont accessibles par une simple commande. Il existe des paquets LuaRocks, appelés *rocks*, pour toutes sortes d'applications : toutes les fonctionnalités mentionnées ci-dessus, mais aussi des *bindings* pour tous les types de bibliothèques (GUI, graphisme, calcul scientifique, web, etc.). De nouveaux rocks sont ajoutés chaque jour dans le dépôt sur luarocks.org. Avec la combinaison de Lua et de LuaRocks, vous pouvez ainsi avoir le meilleur des deux mondes : la variété des fonctionnalités disponibles à partir du dépôt LuaRocks et un système léger contenant uniquement les modules dont vous avez vraiment besoin, dans l'esprit de Lua.

Dans ce chapitre, nous allons voir comment utiliser des modules tiers disponibles dans le dépôt, puis nous aborderons l'une des fonctions les plus puissantes de LuaRocks : construire et installer ses propres modules. Plutôt que de détailler chaque commande et paramètre de LuaRocks, nous développerons un exemple complet de sorte à ce que vous puissiez l'utiliser comme point de départ pour vos propres projets. J'espère que cela vous encouragera non seulement à essayer LuaRocks, mais aussi à développer de nouveaux rocks : nous profitons tous de l'enrichissement de l'écosystème Lua !

8.1. Installer LuaRocks

Pour commencer, voyons comment installer LuaRocks et vérifier que l'installation a réussi. Il y a plusieurs façons de procéder : LuaRocks est disponible dans les dépôts de nombreuses distributions Linux, et il est même inclus dans le paquet Lua dans le dépôt Homebrew pour Mac OS X. Mais, pour nous assurer que nous utilisons la version la plus récente, avec les dernières fonctionnalités et corrections de bugs, nous allons suivre étape par étape le processus d'installation de LuaRocks directement à partir du paquet officiel de luarocks.org.

LuaRocks est une application en Lua pur, sans dépendance à des bibliothèques, il n'y a donc rien à compiler. Mais il y a quand même une étape de construction rapide qui vérifie que tout dans votre environnement est correct, par exemple que votre installation de Lua est trouvée, puis qui écrit la configuration par défaut.

Sur le site de LuaRocks, cliquez sur le lien [Install](#). Vous trouverez deux types d'archives : un fichier tar .gz à destination des systèmes Unix comme Linux et Mac OS X et un fichier .zip pour Windows. Téléchargez et décompressez celui qui correspond à votre plateforme, puis passez à la section appropriée ci-dessous.

Sous Unix (Linux, Mac OS X)

Sous Unix, le système de build de LuaRocks est basé sur un simple script `configure` et un `makefile`. Le script `configure` offre quelques options permettant de personnaliser notre utilisation de LuaRocks. Si nous exécutons `./configure` sans arguments, il va essayer de détecter votre version actuelle de Lua et préparer une installation dans `/usr/local`, qui est le répertoire utilisé par défaut par la distribution Lua elle-même.

À l'étape suivante, l'exécution du `makefile` par la commande `make` propose deux options pour l'installation :

- Si vous tapez `make build` puis `sudo make install` LuaRocks va être compilé et installé normalement. C'est l'option que vous devriez utiliser si vous conditionnez LuaRocks pour une sorte de compilation automatique (par exemple, si vous êtes un gestionnaire de paquets d'une distribution Linux, ou si vous faites des déploiements automatisés avec l'aide de Puppet).
- Si vous tapez `sudo make bootstrap` LuaRocks va s'installer lui-même comme un paquet LuaRocks. De cette façon, la prochaine fois que vous souhaitez mettre à niveau LuaRocks, vous pourrez simplement écrire `luarocks install luarocks` pour installer la nouvelle version ! Notez que si vous choisissez ce mode, vous ne

devez pas exécuter `make install`, car `make bootstrap` compile et installe en une seule étape.

Sous Windows

Le paquet Windows de LuaRocks comprend Lua, LuaRocks et un ensemble de programmes complémentaires qui constituent un environnement de base pour que LuaRocks puisse fonctionner. Pour l'installer, décompressez le fichier `.zip` dans un dossier et, à l'aide de l'invite de commande, exécutez le fichier batch `INSTALL.BAT`. Ce script d'installation propose un certain nombre d'options, que vous pouvez voir en tapant :

```
INSTALL.BAT /?
```

Ces options vous permettront notamment d'indiquer l'emplacement de l'installation, si vous voulez utiliser la version de Lua distribuée ou la vôtre avec LuaRocks et le compilateur C à utiliser (*Microsoft Visual Studio* et *MinGW* sont pris en charge). Pour une première installation, les valeurs par défaut conviennent généralement. Si vous faites une installation pour l'ensemble du système, assurez-vous juste que le terminal de commande que vous utilisez dispose des autorisations administratives appropriées.

Note > Les rocks installés à l'aide de LuaRocks peuvent lancer votre compilateur C, qui doit être correctement configuré dans votre variable PATH. Visual Studio installe une série de raccourcis vers des invites de commande correctement configurées qui prédéfinissent les variables de compilation dans diverses configurations ; vous devriez probablement exécuter LuaRocks en utilisant l'une de ces invites de commande.

8.2. Utiliser LuaRocks

Vous avez suivi les instructions d'installation (sur Unix ou sur Windows), et LuaRocks est désormais installé sur votre machine. Maintenant, vous voulez probablement installer des rocks (paquets contenant des modules Lua) et les utiliser dans votre code Lua.

LuaRocks installe des outils en ligne de commande qui vont vous permettre de gérer ces rocks : `luarocks` et `luarocks-admin`. Nous ne nous intéresserons ici qu'à `luarocks`¹. Assurez-vous que le répertoire où il se trouve est dans votre `PATH` – l'emplacement exact dépend des options que vous avez utilisées lors de l'installation de LuaRocks. Exécutez `luarocks` pour voir les commandes disponibles :

```
luarocks
```

¹`luarocks-admin` n'est utilisé que pour la maintenance de serveurs personnalisés de rocks. Les utilisateurs pouvant désormais héberger leur propre ensemble de rocks sur `luarocks.org`, cet outil s'avère moins utile.

Vous pouvez obtenir de l'aide sur une commande en utilisant la commande `help` :

```
luarocks help install
```

L'installation se fait en utilisant la commande `install`. Par exemple, supposons que nous avons besoin d'un parseur de JSON et que nous avons opté pour `dkjson`, qui est écrit en pur Lua. Pour l'installer, nous exécuterons :

```
luarocks install dkjson
```

À la première exécution de cette commande, il est courant d'obtenir le message d'erreur suivant :

```
Error: Your user does not have write permissions
in /usr/local/lib/luarocks/rocks -- you may want
to run as a privileged user or use your local
tree with --local.
```

Cela signifie que vous avez essayé de l'installer à un endroit du système où vous n'avez pas la permission d'écriture. Les paramètres par défaut sont prévus pour une installation de LuaRocks sur l'ensemble de votre système, tout en laissant la possibilité à chaque utilisateur d'installer ses propres rocks depuis son compte dans son répertoire utilisateur. Seuls les rocks installés par l'administrateur depuis son compte seront disponibles pour tout le monde. Toutefois, vous pouvez aussi installer LuaRocks dans votre propre répertoire home en utilisant l'option `--local`. En revanche, si vous voulez vraiment procéder à une installation globale sur votre système, la solution sur Unix est d'élever vos privilèges :

```
sudo luarocks install dkjson
```

Sur Windows, vous pouvez obtenir le même effet en exécutant `luarocks` à partir d'une invite de commandes administrateur. Cela permet d'installer le rock dans l'arborescence système de Windows.

Mais qu'est-ce qui a vraiment été installé ? Nous pouvons interroger LuaRocks pour voir ce qui a été installé en utilisant la commande `show` :

```
luarocks show dkjson
```

Si l'installation précédente s'est déroulée correctement, vous obtiendrez quelque chose comme ceci :