



# ANF 2012 - Mathrice

Composants réutilisables dans la pratique

« RubyGems »

David Delavennat



# Agenda de la présentation

- Rubygems en dix secondes
- Fonctionnalités importantes
- Avantages
- Commandes principales
- Documentation
- Compatibilité avec RPM/DEB
- RubyVersionManager



# RubyGems en dix secondes

- Une application ou une bibliothèque est rassemblée dans une gemme, qui est un élément d'installation unique.
- RubyGems utilise exclusivement son propre système de fichier, plutôt que d'installer les fichiers à leurs emplacements « habituels ». Cela permet d'obtenir de nouvelles fonctionnalités et augmente la fiabilité du système.



# Fonctionnalités importantes

- Installation distante de paquets
- Gestion des dépendances
- Désinstallation facile et fiable
- Fonctionnalités de requête, locale et distante
- Possibilité d'avoir plusieurs versions du même paquet installées
- Interface web pour visualiser la documentation de vos gemmes



# Avantages

En utilisant RubyGems, vous pouvez :

- télécharger et installer des librairies Ruby facilement ;
- ne pas vous soucier des dépendances entre telle librairie A et telle autre librairie B par rapport à une librairie C sous différentes versions ;
- retirer facilement des librairies que vous n'utiliseriez plus ;
- avoir un contrôle sur votre plateforme Ruby !



# Lister les gems distants disponibles

```
$ gem list --remote | wc -l
35782
$ gem list --remote | grep sequel_
i18n-sequel_bitemporal (0.4.2)
rspec_sequel_matchers (0.2.0)
sequel_acts_as_cacheable (0.5)
sequel_bitemporal (0.4.9)
sequel_bulk_attributes (0.0.2)
sequel_column_type_array (0.0.4)
sequel_core (3.8.0)
sequel_fast_columns (0.0.2)
sequel_mappable (0.0.1)
sequel_migration_builder (0.3.1)
sequel_model (3.8.0)
sequel_notnaughty (0.6.2)
...
```



# Rechercher des gems distants disponibles

```
$ gem search --remote sequel_

*** REMOTE GEMS ***

i18n-sequel_bitemporal (0.4.2)
rspec_sequel_matchers (0.2.0)
sequel_acts_as_cacheable (0.5)
sequel_bitemporal (0.4.9)
sequel_bulk_attributes (0.0.2)
sequel_column_type_array (0.0.4)
sequel_core (3.8.0)
sequel_fast_columns (0.0.2)
sequel_mappable (0.0.1)
sequel_migration_builder (0.3.1)
sequel_model (3.8.0)
sequel_notnaughty (0.6.2)
sequel_oracle_extensions (0.6.7)
sequel_orderable (0.0.3)
sequel_paperclip (0.5.7)
sequel_pg (1.2.2 ruby x86-mingw32 x86-mswin32-60)
sequel_plus (0.4.1)
...
```



# Installer un gem distant

```
$ gem install progressbar
```

Le gem progressbar sera installé sur votre ordinateur.

Notez que vous n'avez pas besoin de spécifier la version, mais vous pouvez si vous le voulez.

Par défaut, c'est la dernière version qui sera installée.

```
« gem install -r progressbar-0.0.3 » ou « gem install -r progressbar --version '> 0.0.1' »
```

Dans ces deux cas, vous obtiendrez simplement :

```
Attempting remote installation of 'progressbar'
```

```
Successfully installed progressbar, version 0.0.3
```

RubyGems vous autorise à avoir plusieurs versions d'une librairie installées et de choisir dans votre code quelle version vous souhaitez utiliser.

Parmi les options d'installation utiles, on trouve `—gen-rdoc` qui génère la documentation RDoc du gem, et `—run-test` qui lance les unités de test du gem, s'il en existe.

Notez également que lorsque vous installez un gem distant, cela téléchargera et installera également toute dépendance nécessaire.





# Obtenir des informations sur un gem installé

```
$ gem spec uuid
--- !ruby/object:Gem::Specification
name: uuid
version: !ruby/object:Gem::Version
  version: 2.3.5
prerelease:
segments:
- 2
- 3
- 5
platform: ruby
authors:
- Assaf Arkin
- Eric Hodel
autorequire:
bindir: bin
cert_chain: []
date: 2012-01-23 00:00:00.000000000 Z
dependencies:
...
```



# Utiliser un gem dans votre code

```
require 'rubygems'  
require_gem 'progressbar'
```

# Crée une barre de progression qui va de 0 à 100% par palier aléatoires.

```
barre = ProgressBar.new("Programme exemple", 100)
```

```
total = 0
```

```
until total >= 100
```

```
  sleep(rand(2)/2.0)
```

```
  increment = (rand(6) + 3)
```

```
  barre.inc(increment)
```

```
  total += increment
```

```
end
```

Voici un « aperçu » de la barre en cours de progression.

Programme exemple : 29% |oooooooooooo

| ETA: 00:00:04



## Spécifier une version

Imaginons qu'à moyen terme, l'auteur de `progressbar` mette à disposition la version 1.0 avec une API qui serait incompatible avec ce que nous aurions codé auparavant. Ce que nous pouvons faire, pour nous assurer que notre code sera toujours fonctionnel, c'est de remplacer la ligne `require_gem` par celle-ci :

```
require_gem 'progressbar', '< 1.0'
```

Ainsi, si nous mettons à disposition notre code sous la forme d'un gem, nous pouvons spécifier explicitement quelle version de `progressbar` sera requise, de sorte que les utilisateurs qui installeront notre gem installent dans le même temps une version appropriée de `progressbar`.



## Désinstaller un gem

```
$ gem uninstall progressbar
```

```
Successfully uninstalled progressbar version 0.0.3
```

S'il y a plus d'une version du gem installées, la commande gem vous demandera quelle version supprimer.



# Navigation dans les gems installés et leurs documentation

Usage: `gem server [options]`

## Options:

<code>-p, --port=PORT</code>	port to listen on
<code>-d, --dir=GEMDIR</code>	directories from which to serve gems
	multiple directories may be provided
<code>--[no-]daemon</code>	run as a daemon
<code>-b, --bind=HOST,HOST</code>	addresses to bind
<code>-l, --launch[=COMMAND]</code>	launches a browser window
	COMMAND defaults to 'start' on Windows and 'open' on all other platforms

## Common Options:

<code>-h, --help</code>	Get help on this command
<code>-V, --[no-]verbose</code>	Set the verbose level of output
<code>-q, --quiet</code>	Silence commands
<code>--config-file FILE</code>	Use this config file instead of default
<code>--backtrace</code>	Show stack backtrace on errors
<code>--debug</code>	Turn on Ruby debugging

## Summary:

Documentation and gem repository HTTP server

## Description:

The server command starts up a web server that hosts the RDoc for your installed gems and can operate as a server for installation of gems on other machines.

The cache files for installed gems must exist to use the server as a source for gem installation.

To install gems from a running server, use `gem install GEMNAME --source http://gem_server_host:8808``

You can set up a shortcut to gem server documentation using the URL:

`http://localhost:8808/rdoc?q=%s` - Firefox  
`http://localhost:8808/rdoc?q=*` - LaunchBar



# gem2rpm / gem2deb

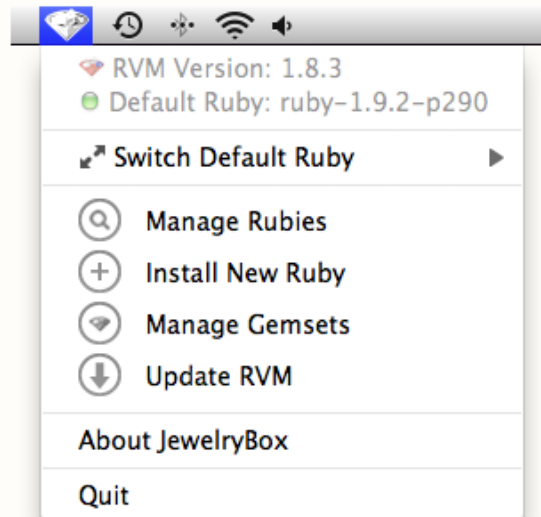
## Qu'est ce que c'est?

- gem2rpm / gem2deb est un utilitaire qui transforme un package au format RubyGems en un package équivalent au format RPM / DEB.

## Pourquoi utiliser gem2rpm / gem2deb?

- Vous gérez des serveurs Linux et vous passez déjà beaucoup de temps à vous assurer que vos machines ont un serveur de mise-à-jour local, font automatiquement leur mises-à-jour, sont monitorées, etc.
- Le système utilise déjà yum/rpm ou apt/deb.
- Vous rencontrez des problèmes pour spécifier les dépendances entre des logiciels Ruby et non-Ruby
- En installant les gems via rpm/deb le problème est résolu.

# Ruby Version Manager (RVM)





# Références

<http://www.rubyfrance.org/documentations/rubygem---introduction/>

<http://unfiniti.com/software/mac/jewelrybox>

<http://beginrescueend.com/>