CO2 Atmosphérique

Manuel d'installation

24/02/2015

Université de Bretagne Occidentale

Marie-Josée Tetchi, Jean-Sébastien Rolland

[I. Qu'est ce que Co2Atm ? 3](#_Toc412543485)

[II. Pré-requis pour l'installation 3](#_Toc412543486)

[a. cURL 3](#_Toc412543487)

[b. SSH et Git 3](#_Toc412543488)

[III. Installation de Co2Atm 4](#_Toc412543489)

[a. Ruby 4](#_Toc412543490)

[b. Rails 4](#_Toc412543491)

[c. Récupération du dépôt Git 4](#_Toc412543492)

[IV. Configuration 5](#_Toc412543493)

# Qu'est ce que Co2Atm ?

Le projet Co2Atm se base sur un projet existant de récolte de données atmosphériques sur le site de St Anne-du-Portic. Dans le but de pérenniser les données et d'y accéder plus facilement, nous avons développé un site Internet en Ruby on Rails. Voici le manuel permettant l'intégration du projet à un environnement de production.

# Pré-requis pour l'installation

Afin d'installer les différents programmes qu'utilise notre application, il est nécessaire de disposer d'un serveur. Ici, nous nous appuierons sur l'environnement de l'IUEM, sous Unix, où un serveur Apache est déjà installé.

Par ailleurs une connexion Internet sera obligatoire pour l'ensemble de l'installation.

 [schéma]

## cURL

L'installation de Ruby nécessite l'utilisation de Curl.

Dans le cas d'un système Ubuntu, tapez

 sudo apt-get install curl libcurl3 libcurl3-dev php5-curl

et si vous êtes sur Debian

 sudo apt-get update && sudo apt-get install php5-curl

Pensez à redémarrer votre serveur Apache après cette manipulation.

## SSH et Git

Afin de transférer l'application de l'environnement de développement à celle de production, nous utiliserons SSH et Git. Si vous avez besoin de les installer, référez-vous aux étapes décrites par la documentation :

 - SSH pour Ubuntu [http://doc.ubuntu-fr.org/ssh] ou

Debian [http://openclassrooms.com/courses/installation-et-utilisation-d-un-serveur-ssh-sous-debian-etch]

 - Git : http://git-scm.com/book/fr/v1/Démarrage-rapide-Installation-de-Git

Ici, nous supposons qu'il est déjà installé et configuré [https://www-iuem.univ-brest.fr/feiri/soutiens-scientifique/calcul-scientifique/wiki/utilisation-de-git ] sur le poste de l'IUEM, sur lequel nous effectuerons notre transfert de fichiers.

# Installation de Co2Atm

L'installation du projet se décompose en trois parties. Tout d'abord il est nécessaire d'installer Ruby et Rails. Ensuite nous déplaceront le projet sur ce nouvel environnement grâce à Git.

## Ruby

Pour l'installation de Ruby, nous avons fait le choix d'utiliser Ruby Version Manager qui permet de choisir automatiquement la dernière version stable. Un guide est disponible à l'URL suivante : https://rvm.io/rvm/install

 \curl -sSL https://get.rvm.io | bash -s stable --ruby

## Rails

Rails pourrait également être installé par RVM mais nous avons simplement choisi d'utiliser les gem, comme le propose le site de Ruby on Rails [http://rubyonrails.org/download/]. Lancez l'installation par la commande

 gem install rails

Il ne reste plus qu'à créer une application Rails pour y transférer notre dépôt de développement. Vous pouvez également démarrer rails pour être certain que l'installation s'est correctement effectuée.

 rails new path/to/your/new/application
 cd path/to/your/new/application
 rails server

## Récupération du dépôt Git

Pour envoyer les fichiers du dépôt de développement à celui de production, nous devons d'abord nous connecter en ssh à l'host de l'IUEM. Attention cela présuppose qu'il est préalablement défini dans ~/.ssh/config. Ensuite nous clonons le dépôt existant et faisons un pull afin de récupérer les fichiers.

 ssh gitiuem

 git clone git://tucuxi.univ-brest.fr/co2atm

 git pull

# Configuration

Avant de pouvoir utiliser l'application dans son nouvel environnement, nous devons la configurer. En développement, nous travaillions en localhost grâce à une base de données sqlite3. Il faut donc modifier les différentes variables host ainsi que la configuration de la database.

## Base de données

Modifier la base de données dans config/database.yml

 production:

   adapter: postgresql

   encoding: unicode

   database: blog\_prod

   pool: 5

   username: blog

   password:

## Gem Devise

La gem Devise gère l'authentification. Il est nécessaire de modifier l'host dans config/environnements/production.rb, actuellement :

config.action\_mailer.default\_url\_options = { host: 'localhost', port: 3000}