

GOLLUM: Serveur PlantUML auto-hébergé

Table of Contents

- [GOLLUM: Serveur PlantUML auto-hébergé](#)
 - [Installation et configuration de Maven](#)
 - [Installer le serveur PlantUML local](#)
 - [Exécuter PlantUML avec Tomcat](#)
 - [Configuration du serveur PlantUML](#)
 - [SSL](#)

Installation et configuration de Maven

Apache Maven est un outil de gestion et d'automatisation de production des projets logiciels Java en général et Java EE en particulier. Il est utilisé pour automatiser l'intégration continue lors d'un développement de logiciel.

```
sudo apt-get install maven graphviz openjdk-7-jdk git-core
```

Configurer un proxy à utiliser pour tout ou partie des requêtes HTTP avec Maven. Le nom d'utilisateur et le mot de passe ne sont requis que si le proxy nécessite une authentification de base (les versions ultérieures peuvent prendre en charge le stockage des mots de passe dans un magasin de clés sécurisé - en attendant, il faut s'assurer que le fichier **settings.xml** (généralement `${user.home}/.m2/settings.xml`) est sécurisé avec les autorisations appropriées au système d'exploitation).

Le paramètre **nonProxyHosts** accepte les caractères génériques et chaque hôte à ne pas proxy est séparé par le | . Cela correspond à l'équivalent de configuration JDK.

```
<settings>
  <proxies>
    <proxy>
      <id>example-proxy</id>
      <active>true</active>
      <protocol>http</protocol>
      <host>proxy.example.com</host>
      <port>8080</port>
      <username>proxyuser</username>
      <password>somepassword</password>
      <nonProxyHosts>www.google.com|*.example.com</nonProxyHosts>
    </proxy>
  </proxies>
</settings>
```



es proxys NTLM ne sont actuellement pas pris en charge car ils n'ont pas été testés, utiliser les propriétés système pertinentes sur JDK 1.4+ pour que cela fonctionne.

Installer le serveur PlantUML local

Ces instructions ont été testées sur les distributions Debian Jessie et Ubuntu 14.04 LTS.

```
git clone https://github.com/plantuml/plantuml-server.git
cd plantuml-server
mvn package
```

Une fois le package créé, on peut démarrer un serveur local à l'aide du serveur Jetty intégré:

```
mvn jetty:run
```

Le serveur sera alors disponible sur <http://localhost:8080/plantuml>.

Exécuter PlantUML avec Tomcat

Créer un fichier plantuml.war à partir du code source:

```
sudo apt-get install tomcat7
git clone https://github.com/plantuml/plantuml-server.git
cd plantuml-server
mvn package
```

Copier le fichier plantuml.war généré lors de la construction du package (package mvn) dans le dossier des applications Tomcat:

```
sudo cp plantuml.war /var/lib/tomcat7/webapps/plantuml.war
sudo chown tomcat7:tomcat7 /usr/share/jetty/webapps/plantuml.war
sudo service restart tomcat7
```

Après avoir redémarré Tomcat, l'application sera exécutée sur le port du serveur **8080**. On peut modifier le port et d'autres paramètres dans le fichier **/etc/tomcat7/server.xml**.

Configuration du serveur PlantUML

Par défaut, gollum-lib s'attend à ce que le serveur PlantUML soit situé sur

<http://localhost:8080/plantuml/png>. Si le serveur PlantUML se trouve à un autre emplacement, On peut modifier la valeur par défaut en ajoutant l'option de configuration suivante à l'objet **wiki_options** dans le fichier **config.rb** pour gollum (utiliser l'option **-config** lors du démarrage de gollum!):

```
wiki_options = {  
  :plantuml_url => "http://yourplantumlserver.com/plantuml/png"  
}
```

<L'URL du serveur ne doit pas comporter de barre oblique

SSL

On peut utiliser un serveur PlantUML https simplement en définissant **config.url** dans l'extrait de code ci-dessus sur une URL "https" valide. Cependant, cela peut échouer si gollum ne peut pas vérifier le certificat SSL du serveur PlantUML. Pour contourner ce problème, soit:

- (Recommandé) Définir les fichiers d'autorité de certification que la bibliothèque Net/HTTP de ruby doit utiliser depuis le fichier **config.rb**.
- Désactiver la vérification du certificat SSL en ajoutant ce qui suit au fichier **config.rb**:

```
Gollum::Filter::PlantUML.configure do |config|  
  config.url = "http://my-own-plantuml.com:8080/plantuml/png"  
  config.verify_ssl = false  
end
```

From:

<http://www.ouarte.garden/> - **dwndoc**

Permanent link:

<http://www.ouarte.garden/doku.php?id=outils:gollum-plantuml-server>

Last update: **2025/02/19 10:59**

