

# Visualiser les dépendances

Pour visualiser le graphe de dépendances Maven généré par le script Python fourni, vous pouvez utiliser **Graphviz**, un outil graphique largement utilisé et simple qui rend les graphes décrits dans le langage DOT. Le script produit un graphe de dépendances au format DOT, représentant les dépendances entre les modules de votre projet Maven sous forme de graphe dirigé, avec des nœuds étiquetés par les `artifactId` et des arêtes montrant les relations de dépendance. Voici les étapes pour dessiner et visualiser ce graphe en utilisant Graphviz, ainsi que des mentions d'outils alternatifs comme Polyglot ou d'autres si vous préférez des options différentes.

---

## Utilisation de Graphviz (Approche Recommandée)

Graphviz est idéal ici car le script génère directement une sortie au format DOT, que Graphviz prend en charge de manière native. Suivez ces étapes :

1. **Exécuter le Script** Exécutez le script Python, en fournissant le répertoire racine de votre projet Maven en argument. Cela génère la sortie DOT pour le graphe de dépendances.

```
python script.py /path/to/maven/project
```

2. **Enregistrer la Sortie DOT dans un Fichier** Redirigez la sortie du script vers un fichier, par exemple, `dependencies.dot`. Ce fichier contiendra la description du graphe au format DOT.

```
python script.py /path/to/maven/project > dependencies.dot
```

3. **Installer Graphviz (si ce n'est pas déjà fait)** Graphviz est disponible pour Windows, macOS et Linux. Installez-le en utilisant votre gestionnaire de paquets :

- **Ubuntu/Debian :**

```
sudo apt-get install graphviz
```

- **macOS (avec Homebrew) :**

```
brew install graphviz
```

- **Windows :** Téléchargez et installez à partir du site web de Graphviz.

4. **Générer une Image Visuelle** Utilisez la commande `dot` de Graphviz pour convertir le fichier DOT en une image. Par exemple, pour créer un fichier PNG :

```
dot -Tpng dependencies.dot -o dependencies.png
```

- Vous pouvez remplacer `-Tpng` par d'autres formats comme `-Tsvg` pour SVG ou `-Tpdf` pour PDF, selon vos préférences.

5. **Visualiser le Graphe** Ouvrez le fichier `dependencies.png` généré avec n'importe quel visualiseur d'images pour voir le graphe de dépendances. Chaque nœud représentera un `artifactId` de module, et les flèches indiqueront les dépendances entre les modules.
- 

## Outils Alternatifs

Si vous ne souhaitez pas utiliser Graphviz ou si vous souhaitez explorer d'autres outils graphiques courants, voici quelques options :

**Polyglot Notebooks (par exemple, avec Jupyter)** Les Polyglot Notebooks ne visualisent pas directement les fichiers DOT, mais vous pouvez intégrer Graphviz dans un environnement de notebook Jupyter :

- **Étapes** : 1. Installez Graphviz et le package Python `graphviz` : `bash pip install graphviz sudo apt-get install graphviz` # Pour Ubuntu, ajustez selon votre OS 2. Modifiez le script pour utiliser la bibliothèque Python `graphviz` au lieu d'imprimer du DOT brut. Ajoutez ceci à la fin de votre script : `python from graphviz import Digraph`

```
dot = Digraph()
for from_module, to_module in sorted(dependencies):
    dot.edge(from_module, to_module)
dot.render('dependencies', format='png', view=True)
...
```

3. Exécutez le script modifié pour générer et afficher `dependencies.png` directement.

- **Note** : Cela repose toujours sur Graphviz sous le capot, donc ce n'est pas un outil complètement séparé.

**Gephi** Gephi est un outil de visualisation de réseau open-source qui peut importer des fichiers DOT :

- **Étapes** : 1. Téléchargez et installez Gephi depuis [gephi.org](http://gephi.org). 2. Exécutez le script et enregistrez la sortie DOT dans `dependencies.dot`. 3. Ouvrez Gephi, allez dans Fichier > Importer > Fichier de graphe, et sélectionnez `dependencies.dot`. 4. Ajustez la mise en page (par exemple, ForceAtlas 2) et visualisez de manière interactive. - **Avantages** : Idéal pour les grands graphes avec des options de mise en page avancées. - **Inconvénients** : Nécessite une importation et une configuration manuelles.

**Outils Graphviz en Ligne** Pour une option rapide sans installation : - **Étapes** : 1. Exécutez le script et copiez la sortie DOT (de `digraph G { à }`). 2. Visitez un outil en ligne comme GraphvizOnline. 3. Collez le contenu DOT et cliquez pour rendre le graphe. - **Avantages** : Pas besoin d'installation. - **Inconvénients** : Nécessite un accès à Internet ; moins de contrôle sur le format de sortie.

**yEd** yEd est un éditeur de graphes gratuit, mais il ne prend pas en charge le DOT de manière native : - **Étapes** : 1. Convertissez le fichier DOT en GraphML ou un autre format pris en charge en utilisant un outil comme dot (par exemple, `dot -Tgraphml dependencies.dot -o dependencies.graphml`). 2. Ouvrez yEd, importez le fichier GraphML et ajustez la mise en page. - **Inconvénients** : Étape de conversion supplémentaire requise.

---

## Pourquoi Graphviz est le Meilleur Ici

La sortie du script est conçue pour DOT, ce qui fait de Graphviz le choix le plus direct et efficace. Il est léger, commandé par ligne de commande et largement pris en charge. Le graphe généré ressemblera conceptuellement à ceci : - Nœuds : `artifactId` de chaque module (par exemple, `"module-a"`, `"module-b"`). - Arêtes : Flèches comme `"module-a" -> "module-b"`, montrant que `module-a` dépend de `module-b`.

Si votre graphe est grand, vous pouvez ajuster la mise en page avec des options Graphviz (par exemple, `dot -Kneato` pour un autre agencement), mais les étapes de base ci-dessus devraient suffire.

---

## Recommandation Finale

Pour visualiser le graphe de dépendances Maven à partir de votre script : 1. Exécutez : `python script.py /path/to/maven/project > dependencies.dot` 2. Installez Graphviz si nécessaire. 3. Convertissez en image : `dot -Tpng dependencies.dot -o dependencies.png` 4. Ouvrez `dependencies.png` pour voir votre graphe.

Cette approche est simple, utilise un outil courant (Graphviz) et s'aligne parfaitement avec la sortie de votre script. Si vous avez besoin d'interactivité ou de formats alternatifs, essayez Gephi ou un outil en ligne à la place.