

Visualizar Dependencias

Para visualizar el gráfico de dependencias de Maven generado por el script de Python proporcionado, puedes usar **Graphviz**, una herramienta gráfica ampliamente utilizada y sencilla que representa gráficos descritos en el lenguaje DOT. El script genera un gráfico de dependencias en formato DOT, representando las dependencias entre módulos dentro de tu proyecto Maven como un gráfico dirigido, con nodos etiquetados por `artifactIds` y bordes que muestran las relaciones de dependencia. A continuación se presentan los pasos para dibujar y visualizar este gráfico utilizando Graphviz, junto con menciones de herramientas alternativas como Polyglot o otras si prefieres diferentes opciones.

Usando Graphviz (Enfoque Recomendado)

Graphviz es ideal aquí porque el script genera directamente una salida en formato DOT, que Graphviz soporta nativamente. Sigue estos pasos:

1. **Ejecutar el Script** Ejecuta el script de Python, proporcionando el directorio raíz de tu proyecto Maven como argumento. Esto genera la salida DOT para el gráfico de dependencias.

```
python script.py /ruta/a/proyecto/maven
```

2. **Guardar la Salida DOT en un Archivo** Redirige la salida del script a un archivo, por ejemplo, `dependencies.dot`. Este archivo contendrá la descripción del gráfico en formato DOT.

```
python script.py /ruta/a/proyecto/maven > dependencies.dot
```

3. **Instalar Graphviz (si no está ya instalado)** Graphviz está disponible para Windows, macOS y Linux. Instálalo usando tu gestor de paquetes:

- **Ubuntu/Debian:**

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

- **macOS (con Homebrew):**

```
brew install graphviz
```

- **Windows:** Descarga e instala desde el sitio web de Graphviz.

4. **Generar una Imagen Visual** Usa el comando `dot` de Graphviz para convertir el archivo DOT en una imagen. Por ejemplo, para crear un archivo PNG:

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

- Puedes reemplazar `-Tpng` con otros formatos como `-Tsvg` para SVG o `-Tpdf` para PDF, dependiendo de tu preferencia.

5. **Ver el Gráfico** Abre el archivo generado `dependencies.png` con cualquier visor de imágenes para ver el gráfico de dependencias. Cada nodo representará el `artifactId` de un módulo, y las flechas indicarán las dependencias entre módulos.

Herramientas Alternativas

Si prefieres no usar Graphviz o quieres explorar otras herramientas gráficas comunes, aquí tienes algunas opciones:

Polyglot Notebooks (por ejemplo, con Jupyter) Los Polyglot Notebooks no visualizan directamente archivos DOT, pero puedes integrar Graphviz dentro de un entorno de notebooks Jupyter: - **Pasos:** 1. Instala Graphviz y el paquete de Python graphviz: `bash pip install graphviz sudo apt-get install graphviz` # Para Ubuntu, ajusta para tu SO 2. Modifica el script para usar la biblioteca de Python graphviz en lugar de imprimir DOT en bruto. Añade esto al final de tu 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. Ejecuta el script modificado para generar y mostrar `dependencies.png` directamente.

- **Nota:** Esto aún depende de Graphviz en el fondo, por lo que no es una herramienta completamente separada.

Gephi Gephi es una herramienta de visualización de redes de código abierto que puede importar archivos DOT: - **Pasos:** 1. Descarga e instala Gephi desde gephi.org. 2. Ejecuta el script y guarda la salida DOT en `dependencies.dot`. 3. Abre Gephi, ve a File > Import > Graph File, y selecciona `dependencies.dot`. 4. Ajusta el diseño (por ejemplo, ForceAtlas 2) y visualiza de manera interactiva. - **Pros:** Excelente para gráficos grandes con opciones de diseño avanzadas. - **Contras:** Requiere importación y configuración manual.

Herramientas en Línea de Graphviz Para una opción rápida sin instalación: - **Pasos:** 1. Ejecuta el script y copia la salida DOT (desde `digraph G {` hasta `}`). 2. Visita una herramienta en línea como GraphvizOnline. 3. Pega el contenido DOT y haz clic para renderizar el gráfico. - **Pros:** No se necesita instalación. - **Contras:** Requiere acceso a internet; menos control sobre el formato de salida.

yEd yEd es un editor de gráficos gratuito, pero no soporta DOT de manera nativa: - **Pasos:** 1. Convierte el archivo DOT a GraphML u otro formato compatible usando una herramienta como dot (por ejemplo, dot -Tgraphml dependencies.dot -o dependencies.graphml). 2. Abre yEd, importa el archivo GraphML y ajusta el diseño. - **Contras:** Paso de conversión adicional requerido.

Por Qué Graphviz es la Mejor Opción Aquí

La salida del script está adaptada para DOT, lo que hace que Graphviz sea la opción más directa y eficiente. Es ligero, basado en línea de comandos y ampliamente soportado. El gráfico generado se verá algo así (conceptualmente): - Nodos: artifactId de cada módulo (por ejemplo, "module-a", "module-b"). - Bordes: Flechas como "module-a" -> "module-b", mostrando que module-a depende de module-b.

Si tu gráfico es grande, puedes ajustar el diseño con opciones de Graphviz (por ejemplo, dot -Kneato para un arreglo diferente), pero los pasos básicos anteriores deberían ser suficientes.

Recomendación Final

Para visualizar el gráfico de dependencias de Maven desde tu script: 1. Ejecuta: `python script.py /ruta/a/proyecto/maven > dependencies.dot` 2. Instala Graphviz si es necesario. 3. Convierte a una imagen: `dot -Tpng dependencies.dot -o dependencies.png` 4. Abre dependencies.png para ver tu gráfico.

Este enfoque es sencillo, usa una herramienta común (Graphviz) y se alinea perfectamente con la salida de tu script. Si necesitas interactividad o formatos alternativos, prueba Gephi o una herramienta en línea en su lugar.