

視覺化依賴

要可視化由提供的 Python 腳本生成的 Maven 依賴圖，可以使用 **Graphviz**，這是一個廣泛使用且簡單易用的圖形工具，能夠渲染用 DOT 語言描述的圖形。腳本會輸出一個 DOT 格式的依賴圖，表示 Maven 專案中模組之間的依賴關係，以有向圖的形式呈現，節點標籤為 artifactId，邊表示依賴關係。以下是使用 Graphviz 繪製和可視化這個圖的步驟，並提及其他替代工具如 Polyglot 或其他選項，如果你更喜歡不同的選項。

使用 Graphviz (推薦方法)

Graphviz 非常適合這個用途，因為腳本直接生成 DOT 格式的輸出，這是 Graphviz 原生支持的。請按照以下步驟進行：

1. **執行腳本** 執行 Python 腳本，並將 Maven 專案的根目錄作為參數。這將生成依賴圖的 DOT 輸出。

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

2. **將 DOT 輸出保存到文件** 將腳本的輸出重定向到一個文件，例如 dependencies.dot。這個文件將包含 DOT 格式的圖形描述。

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

3. **安裝 Graphviz (如果尚未安裝)** Graphviz 適用於 Windows、macOS 和 Linux。使用您的包管理器安裝它：

- **Ubuntu/Debian**：

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

- **macOS (使用 Homebrew)**：

```
brew install graphviz
```

- **Windows**：從 Graphviz 網站下載並安裝。

4. **生成視覺圖像** 使用 Graphviz 的 dot 命令將 DOT 文件轉換為圖像。例如，創建一個 PNG 文件：

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

- 您可以將 -Tpng 替換為其他格式，如 -Tsvg 為 SVG 或 -Tpdf 為 PDF，根據您的偏好。

5. **查看圖表** 使用任何圖像查看器打開生成的 dependencies.png 文件，以查看依賴圖。每個節點將表示模組的 artifactId，箭頭將表示模組之間的依賴關係。

替代工具

如果您不想使用 Graphviz 或想探索其他常見的圖形工具，這裡有一些選項：

Polyglot Notebooks (例如, 使用 Jupyter) Polyglot Notebooks 不會直接可視化 DOT 文件, 但您可以在 Jupyter Notebook 環境中集成 Graphviz: - **步驟**: 1. 安裝 Graphviz 和 graphviz Python 包: `bash pip install graphviz sudo apt-get install graphviz # 適用於 Ubuntu, 根據您的操作系統進行調整` 2. 修改腳本以使用 Python 的 graphviz 庫, 而不是打印原始 DOT。在腳本結尾添加以下內容: “ `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. 執行修改後的腳本以直接生成和顯示 dependencies.png。

- **注意**: 這仍然依賴於 Graphviz, 因此它並不是完全獨立的工具。

Gephi Gephi 是一個開源的網絡可視化工具, 可以導入 DOT 文件: - **步驟**: 1. 從 gephi.org 下載並安裝 Gephi。2. 執行腳本並將 DOT 輸出保存到 dependencies.dot。3. 打開 Gephi, 轉到 File > Import > Graph File, 然後選擇 dependencies.dot。4. 調整佈局 (例如, ForceAtlas 2) 並進行互動式可視化。- **優點**: 適合大圖表, 具有高級佈局選項。- **缺點**: 需要手動導入和設置。

在線 Graphviz 工具 快速無需安裝的選項: - **步驟**: 1. 執行腳本並複製 DOT 輸出 (從 digraph G {到})。2. 訪問在線工具, 例如 GraphvizOnline。3. 粘貼 DOT 內容並點擊渲染圖表。- **優點**: 無需安裝。- **缺點**: 需要互聯網訪問; 輸出格式控制較少。

yEd yEd 是一個免費的圖形編輯器, 但它不原生支持 DOT: - **步驟**: 1. 使用工具 (例如 dot) 將 DOT 文件轉換為 GraphML 或其他支持的格式 (例如 `dot -Tgraphml dependencies.dot -o dependencies.graphml`)。2. 打開 yEd, 導入 GraphML 文件並調整佈局。- **缺點**: 需要額外的轉換步驟。

為什麼 Graphviz 是最佳選擇

腳本的輸出是針對 DOT 設計的, 使 Graphviz 成為最直接和高效的選擇。它輕量級、命令行驅動且廣泛支持。生成的圖表看起來會像這樣 (概念上): - 節點: 每個模組的 artifactId (例如 "module-a", "module-b")。- 邊: 箭頭如 "module-a" -> "module-b", 表示 module-a 依賴於 module-b。

如果您的圖表很大, 可以使用 Graphviz 選項 (例如 `dot -Kneato`) 調整佈局, 但基本步驟應該足夠。

最後推薦

要可視化來自腳本的 Maven 依賴圖：1. 執行：`python script.py /path/to/maven/project > dependencies.dot`
2. 如果需要，安裝 Graphviz。3. 轉換為圖像：`dot -Tpng dependencies.dot -o dependencies.png` 4. 打開 `dependencies.png` 查看您的圖表。

這種方法簡單，使用的是常見工具（Graphviz），並與腳本的輸出完全匹配。如果您需要互動性或替代格式，可以嘗試 Gephi 或在線工具。