

# Tableau 9.1 Web Data Connector (WDC) for Neo4j

---

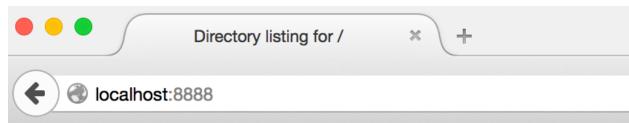
## Prerequisites

Tableau 9.1 includes the Web Data Connector SDK, which lets you connect to any web data accessible over HTTP. You can follow the WDC Tutorial to read about the details: <http://onlinehelp.tableau.com/current/api/wdc/en-us/help.htm>

The WDC for Neo4j consists of a HTML page including a form and JavaScript code to leverage the Tableau WDC API and to process Cypher queries against Neo4j Tx API via Ajax call. The result set (JSON) will be converted into a table and then passed into the Tableau workbook.

To get this working the HTML page Neo4jWdc.html and its linked content (scripts, styles etc.) needs to be published to a web server. For a local use you can just start a simple HTTP server with Python (<https://www.python.org/>) from the same folder:

```
python -m SimpleHTTPServer 8888      // for Python 2.7  
python3 -m http.server 8888          // for Python 3 and above
```

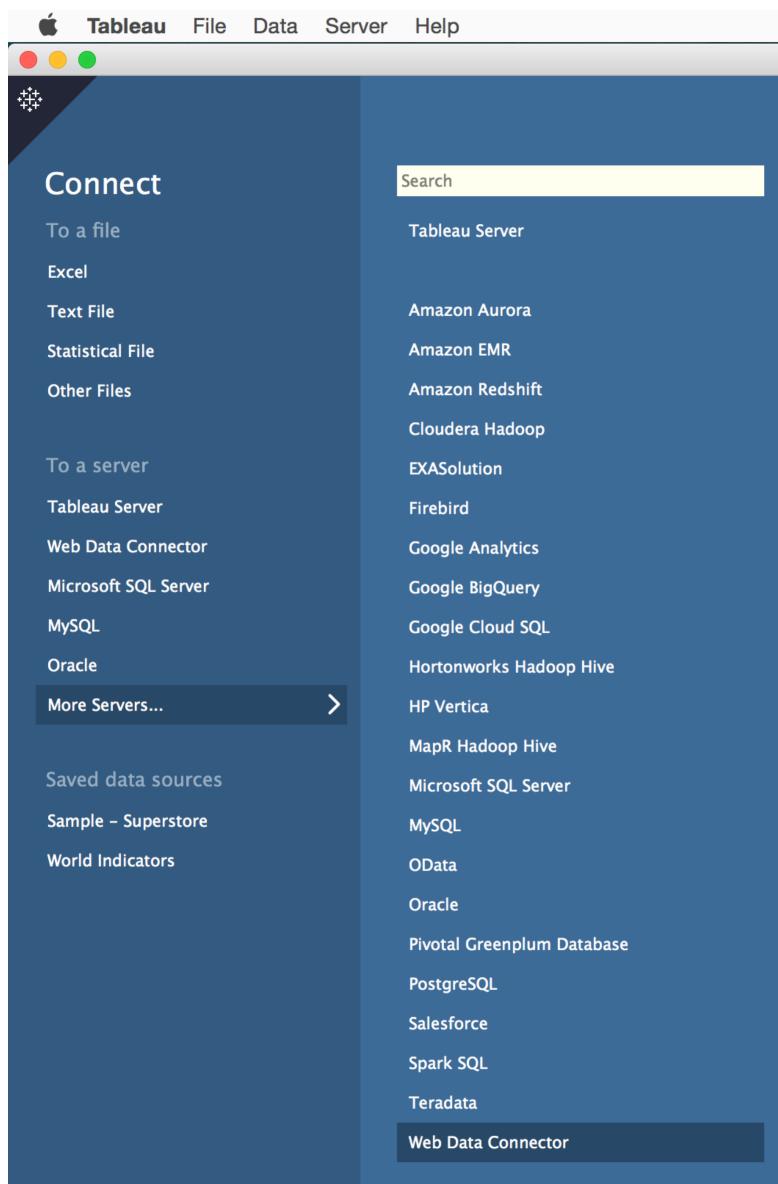


## Directory listing for /

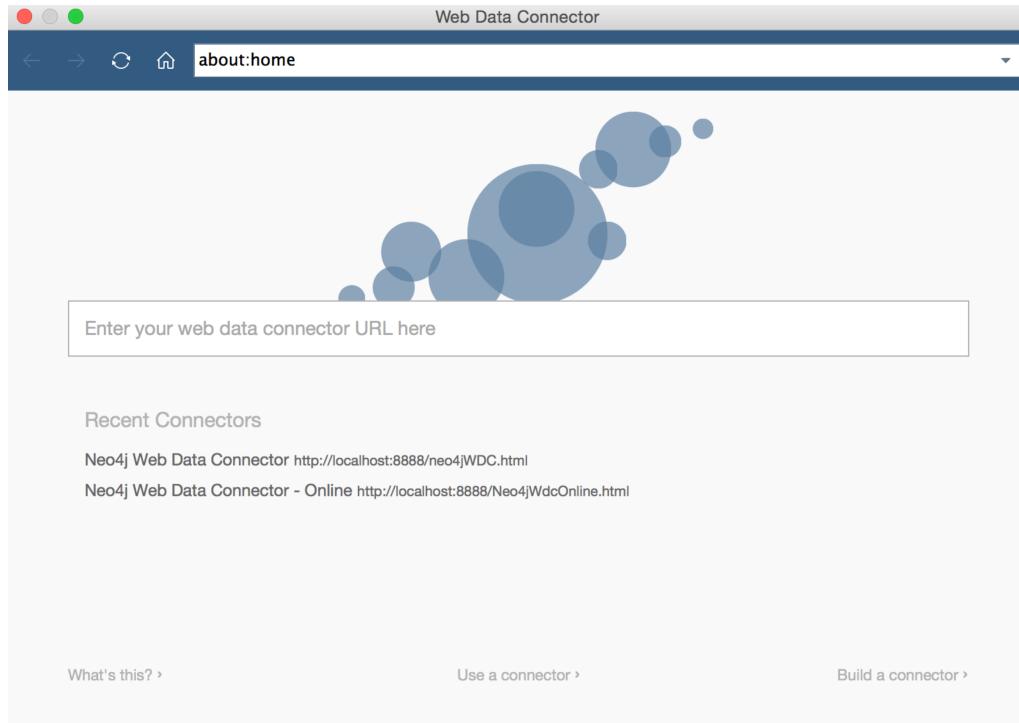
- 
- [images/](#)
  - [Neo4jWdc.html](#)
  - [scripts/](#)
  - [startHttpServer.bat](#)
  - [startHttpServer.sh](#)
  - [styles/](#)
-

## Usage

Start Tableau Desktop 9.1 and choose to connect a WDC:



The WDC selection popup window appears. Enter the URL of your WDC. It lists previously used WDC URLs below you can choose by click:



Enter the URL for Neo4jWdc: <http://localhost:8888/Neo4jWdc.html>

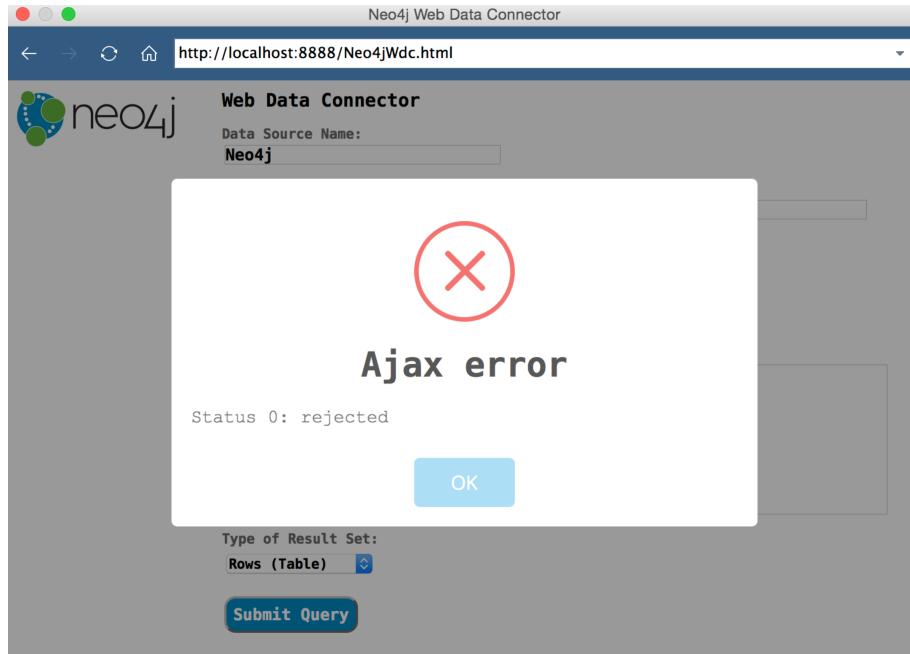
Now the **Neo4jWdc Form** appears. Enter your data source name, Neo4j URL, login credentials and Cypher Query to process (optional choose resultset type):

A screenshot of a web browser window titled "Neo4j Web Data Connector" with the URL "http://localhost:8888/Neo4jWdc.html" in the address bar. The page features a "neo4j" logo. It contains several input fields: "Data Source Name" with "Neo4j" typed in, "Neo4j URL" with "http://localhost:7474" typed in, "Username" with "neo4j" typed in, and "Password" with "..." (three dots). Below these is a "Cypher Query" text area containing the following code:

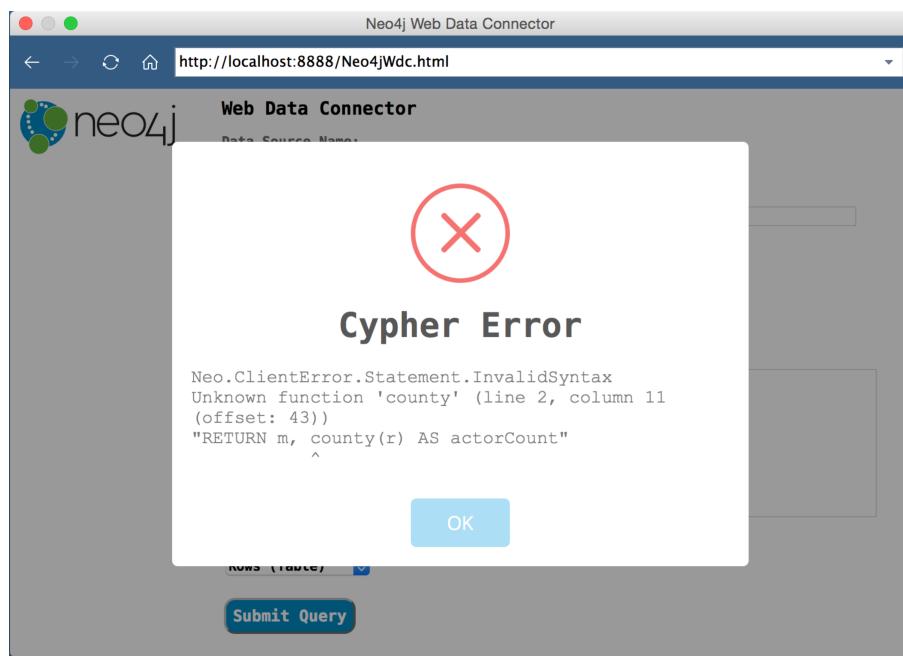
```
MATCH (m:Movie)<-[r:ACTED_IN]-()
RETURN m, count(r) AS actorCount
```

A "Type of Result Set" dropdown menu is open, showing "Rows (Table)" as the selected option. At the bottom is a blue "Submit Query" button.

After hitting „Submit Query“ button a connect to Neo4j is made and the Cypher query will be executed. If something fails you will get an error message:



An **Ajax Error** occurs if URL or credentials are wrong or something happened during the network communication with the database.

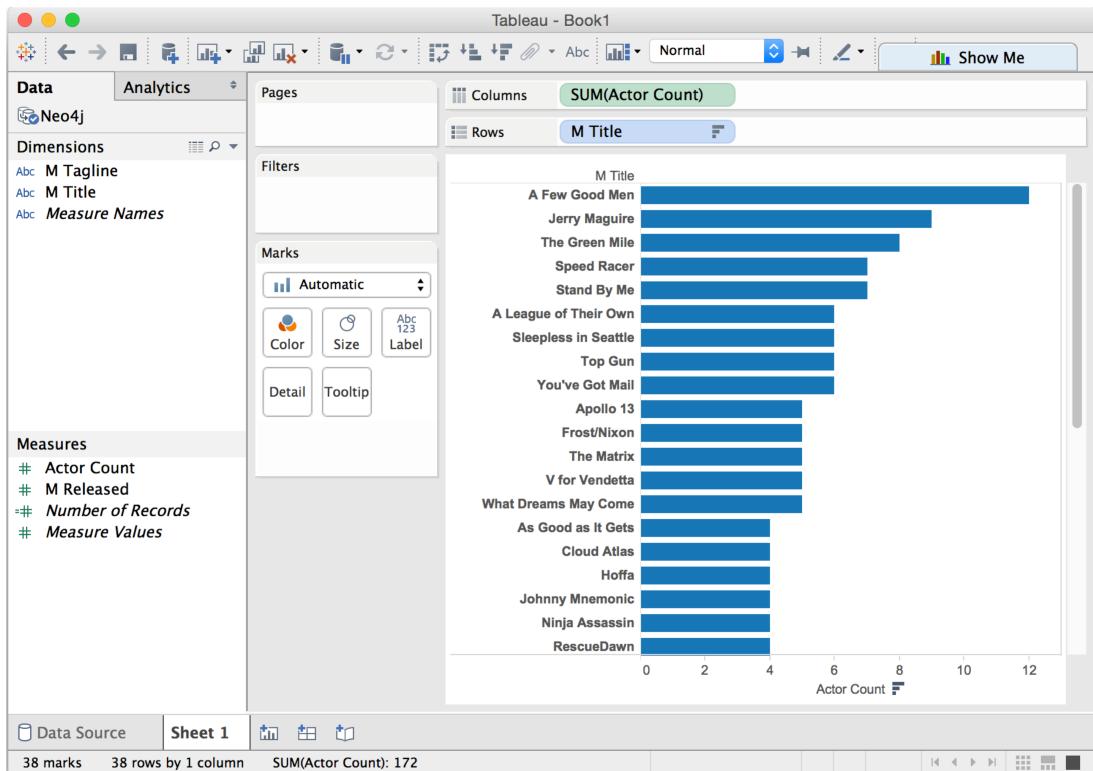


A **Cypher Error** if your query syntax is wrong.

After query execution the created and loaded **Tableau Workbook** will appear and shows the data source or extract: Neo4j, the name we've entered, and the dimensions and measures from the Cypher query result set. Please note that JSON properties are unnested into fields and numeral values are already casted to measures out of the JSON from Neo4j result.

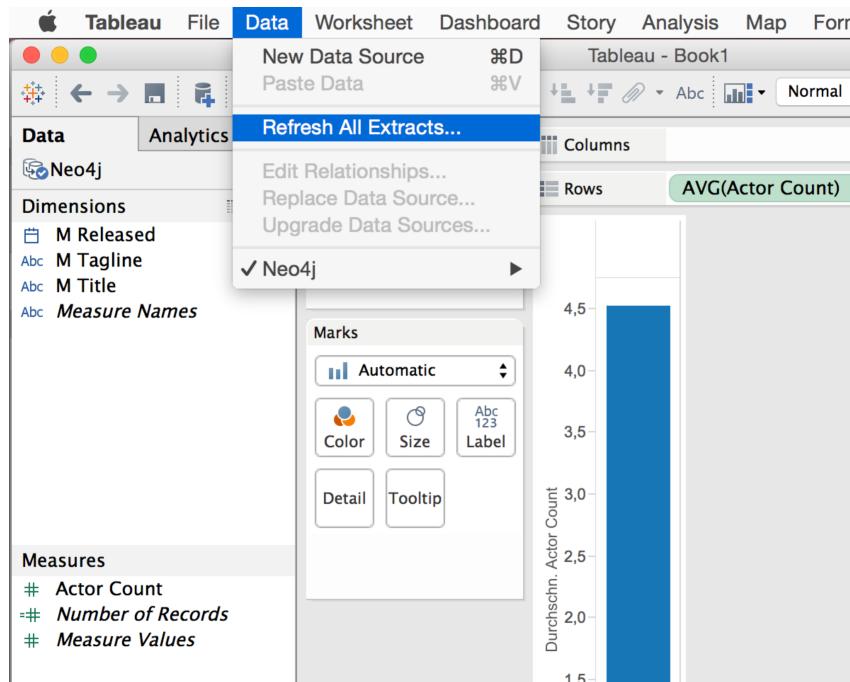
The screenshot shows the Tableau Data Source interface. In the top left, under the 'Data' tab, 'Neo4j' is selected. In the bottom left, under the 'Measures' tab, '# Actor Count' and '# M Released' are selected. The main workspace is empty, with placeholder text 'Drop field here' in the Columns and Rows sections. A 'Show Me' panel on the right displays various chart and visualization options.

Now we can easily start analyzing the Data from Neo4j:

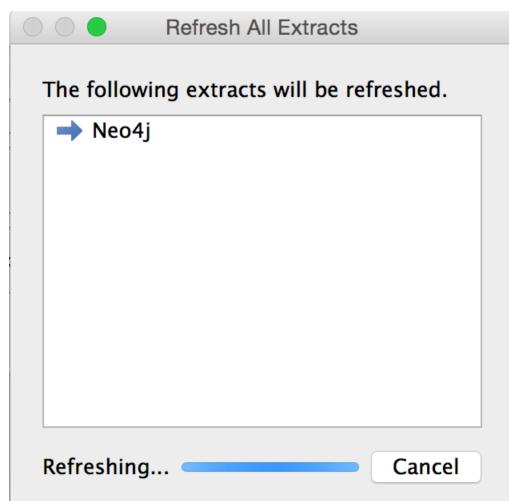


## Refreshing Workbook Data

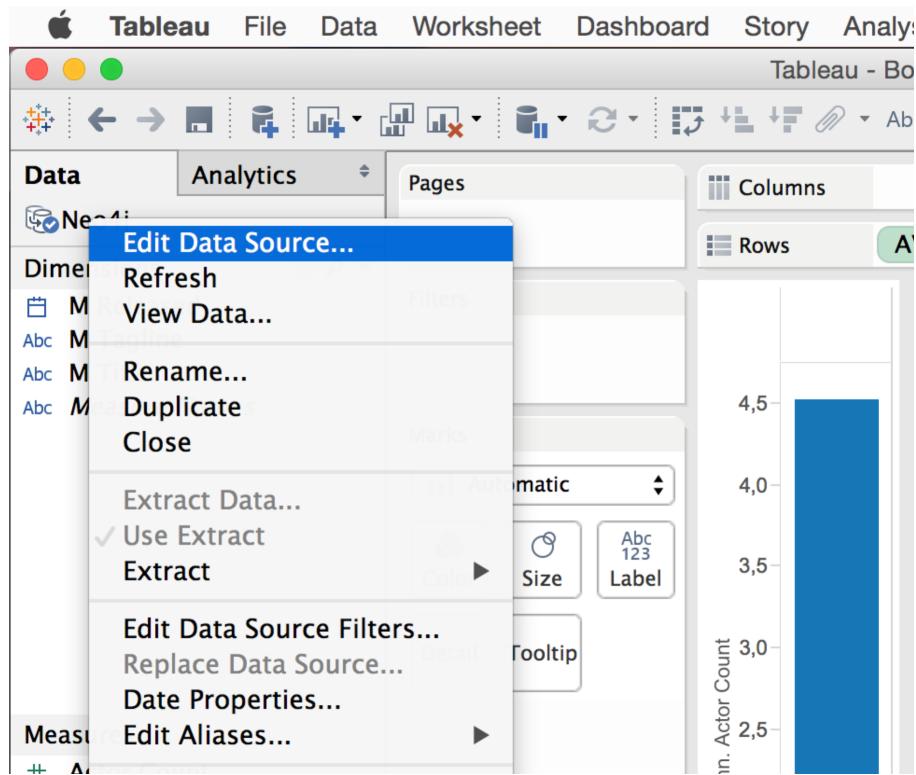
If data has been changed in the Neo4j database you might want to update your Tableau Workbook. In this case go to the Menu *Data/Refresh All Extracts*:



Then click the *Refresh* button. The data refreshing will be executed like this:



If you save and close the **Tableau Workbook** all connection details except the login credentials will be stored inside. If you re-open the Workbook and then want to refresh the data you have to re-enter your credentials! Do right-click the **data source** (Neo4j in this case) and choose *Edit Data Source*.



Then, the Tableau **Import Wizard** appears and shows the **link** to the Neo4jWdc. Click the URL and the Neo4jWdc Form will re-appear with all stored properties. Then you can re-enter username/password and click *Submit Query* to retrieve the new data:

The screenshot shows the 'Data Source' tab of the Tableau Import Wizard. It displays a table of fields from a 'webdata' table, mapping local field names to remote field names. The 'Table' column lists fields like '# Actor Count', 'M Title', 'M Released', and 'M Tagline'. The 'Remote Field Name' column lists corresponding Neo4j field names. At the top, it shows the connection is 'Connected to Web Data' at the URL <http://localhost:8888/Neo4jWdc.html>. The 'Connection' section has 'Extract' selected. A message at the bottom says 'Extract Required. 9/20/15 11:17:33 PM'.

Field Name	Table	Remote Field Name
# Actor Count	webdata.42267.937494594909	actorCount
M Title	webdata.42267.937494594909	m_title
M Released	webdata.42267.937494594909	m_released
M Tagline	webdata.42267.937494594909	m_tagline

## Run Neo4j WDC as Server Extension

Copy the extension library `neo4j-web-extension-tableau-wdc-1.0.0.jar` into the server's plugin directory:

```
cp neo4j-web-extension-tableau-wdc-1.0.0.jar /path/to/neo/plugins/
```

Edit `conf/neo4j-server.properties` to register the package name with an endpoint:

```
org.neo4j.server.thirdparty_jaxrs_classes=
    org.neo4j.tableau.wdc=/tableau/wdc
```

For the moment, the extension works only with disabled authentication:

```
dbms.security.auth_enabled=false
```

Restart the server:

```
/path/to/neo/bin/neo4j restart
```

Then navigate in browser to the extension URL to start the Neo4j WDC Form:

```
http://localhost:7474/tableau/wdc/
```