

Graphical Representation

- Neo4j is a native graph database, built from the ground up to leverage not only data but also data relationships.
- Neo4j has a flexible structure defined by stored relationships between data records.
- To generate the dynamic relationship between the nodes we are adding a plug-in to our Neo4j environment called as APOC.
- APOC is an add-on library for Neo4j that provides hundreds of procedures and functions adding a lot of useful functionality.
- It can be installed with a single click in Neo4j Desktop.

We loaded the triples dataset into the Graph database and performed the following steps to derive the knowledge graph.

Commands to Generate Graph in Neo4j

- **To load the data**

```
LOAD CSV FROM 'file:///article0_triples_simple.csv' AS row
WITH row[0] AS ID, row[1] AS Subject, row[2] AS Verb, row[3] AS Predicate
RETURN ID, Subject, Verb, Predicate
```

- **To create the nodes and dynamic relationship between them**

```
LOAD CSV FROM 'file:///article0_triples_simple.csv' AS row
WITH row[0] AS ID, row[1] AS Subject, row[2] AS Verb, row[3] AS Predicate
RETURN ID, Subject, Verb, Predicate
```

- **To create the nodes and dynamic relationship between them**

```
LOAD CSV WITH HEADERS FROM 'file:///article0_triples_simple.csv' as row
MERGE (s:MyGraph {Name: row.Subject})
MERGE (p:MyGraph {Name: row.Predicate})
with s,p,row
CALL apoc.create.relationship(s,row.Verb,{},p) yield rel
return rel
```

- **To display the graph**

MATCH (s:MyGraph) RETURN s

