

Neo4j: Recorrido de grafos

Big Data Aplicado

Julio Alberto López Gómez
JulioAlberto.Lopez@uclm.es



Big_Data
Aplicado



Curso Especialización
Inteligencia_Artificial y
Big_Data

Recorrido de grafos

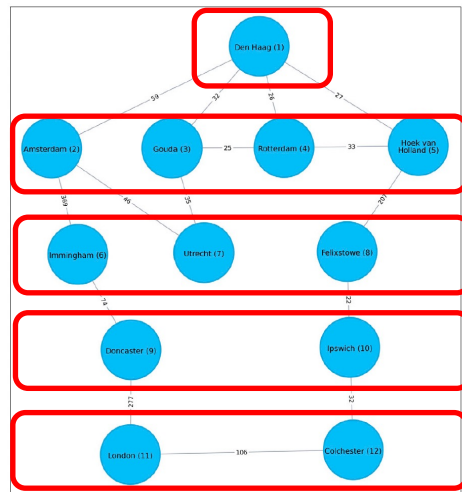
Concepto:

- El recorrido implica visitar todos los nodos de un grafo
 - Recorrido en anchura (**BFS**)
 - Recorrido en profundidad (**DFS**)
- Muy utilizados para realizar consultas y recuperar información en un grafo
- Son la base de algoritmos más avanzados
 - Búsqueda de caminos mismos
 - Componentes conexas
 - Estudio de la centralidad

Recorrido de grafos

BFS:

- **Breadth First Search**
 - Recorrido en anchura
 - Se exploran todos los hijos de un nodo en un salto
 - Una vez visitados todos, se exploran los hijos del siguiente nivel
 - ...



Recorrido de grafos

BFS:

- Creación de un grafo de ejemplo

CREATE

```
(nA:Node {tag: 'a'}) ,  
(nB:Node {tag: 'b'}) ,  
(nC:Node {tag: 'c'}) ,  
(nD:Node {tag: 'd'}) ,  
(nE:Node {tag: 'e'}) ,
```

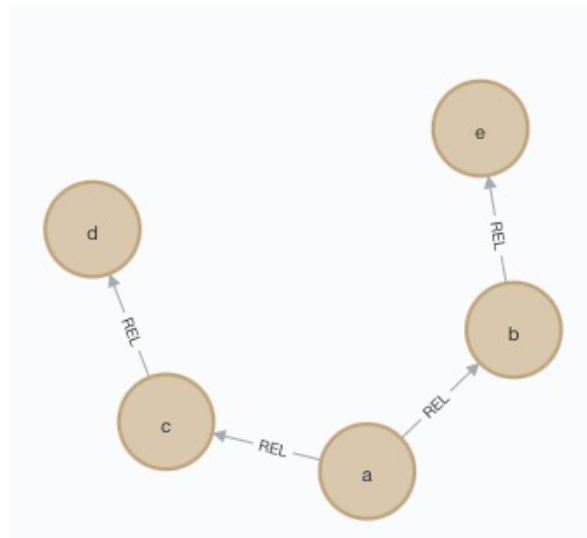
```
(nA) - [ :REL {cost: 8.0} ] -> (nB) ,  
(nA) - [ :REL {cost: 9.0} ] -> (nC) ,  
(nB) - [ :REL {cost: 1.0} ] -> (nE) ,  
(nC) - [ :REL {cost: 5.0} ] -> (nD)
```

Recorrido de grafos

BFS:

- Visualización

```
MATCH (n) RETURN (n)
```



- Se almacena el grafo en el catálogo

```
CALL gds.graph.create('myGraph',  
'Node', 'REL', {  
relationshipProperties: 'cost'})
```

Recorrido de grafos

BFS:

■ Recorrido BFS

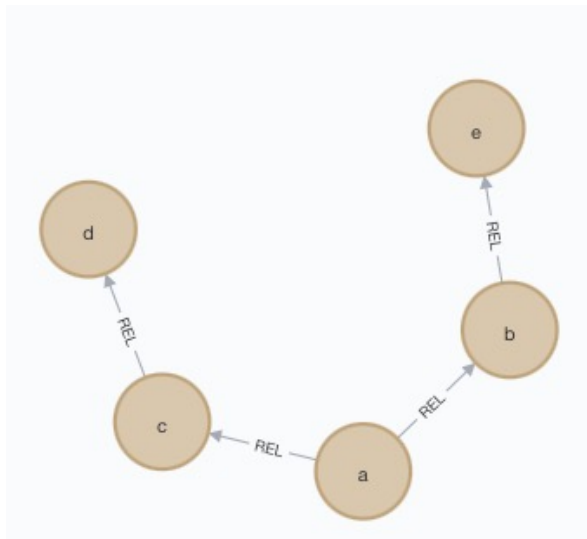
```
MATCH (a:Node{tag:'a'})  
WITH id(a) AS startNode  
  
CALL gds.alpha.bfs.stream('myGraph',  
    {startNode: startNode})  
  
YIELD path  
  
UNWIND [ n in nodes(path) | n.tag ]  
AS tags  
  
RETURN tags
```



Recorrido de grafos

■ Recorrido BFS. Resultado

BFS:

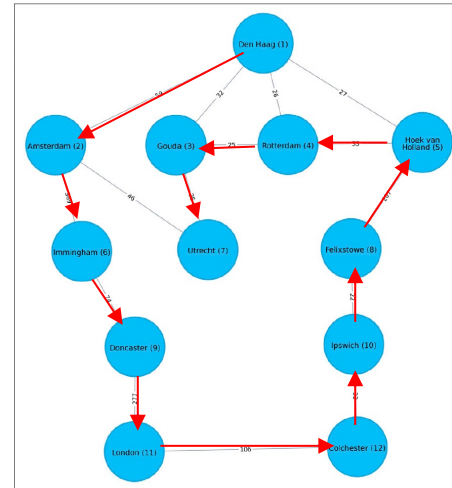


tags	
1	"a"
2	"b"
3	"c"
4	"e"
5	"d"

Recorrido de grafos

DFS:

- Depth First Search (DFS)
 - Recorrido en profundidad
 - Se parte desde un nodo inicial, recorriendo sus hijos hasta el último nivel antes de volver atrás



Recorrido de grafos

DFS:

- Dado el grafo del ejemplo anterior, recorrerlo utilizando DFS

```
MATCH (a:Node{tag:'a'})  
  
WITH id(a) AS startNode  
  
CALL gds.alpha.dfs.stream('myGraph',  
    {startNode: startNode})  
  
YIELD path  
  
UNWIND [ n in nodes(path) | n.tag ]  
AS tags  
  
RETURN tags
```



Recorrido de grafos

■ Recorrido DFS. Resultado

DFS:

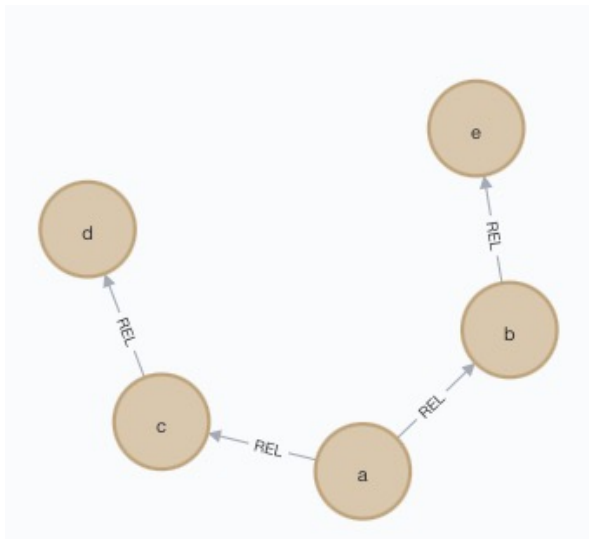


Table	tags
1	"a"
2	"c"
3	"d"
4	"b"
5	"e"

Recorrido de grafos

DFS:

- Otro recorrido, especificando nodos destino

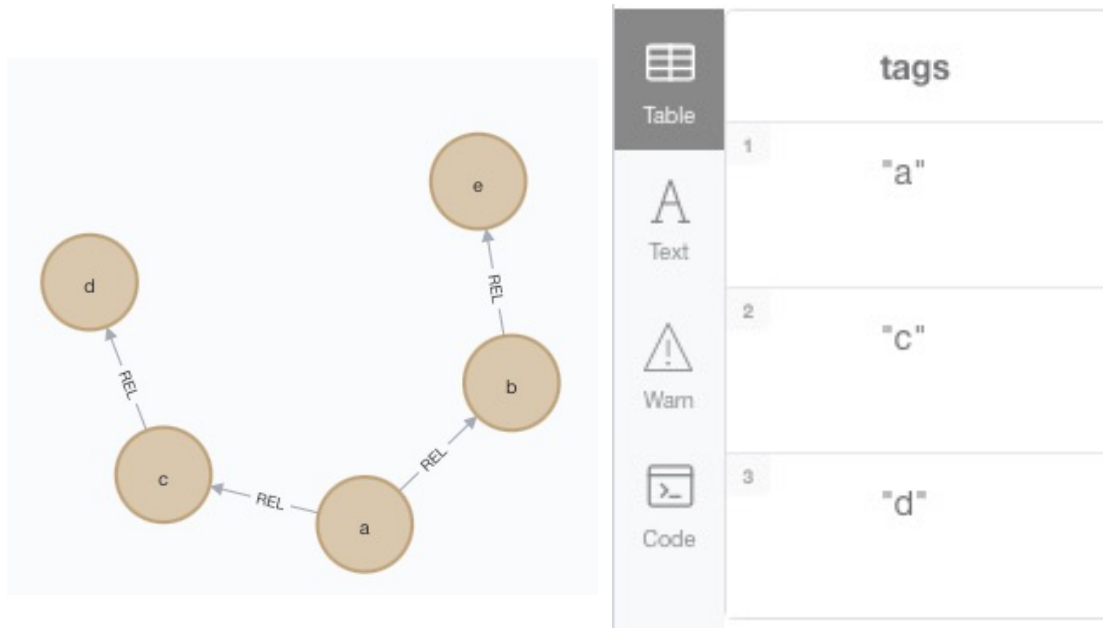
```
MATCH (a:Node{tag:'a'}),  
      (d:Node{tag:'d'}), (e:Node{tag:'e'})  
  
WITH id(a) AS startNode, [id(d),  
id(e)] AS targetNodes  
  
CALL gds.alpha.dfs.stream('myGraph',  
{startNode: startNode, targetNodes:  
targetNodes})  
  
YIELD path  
  
UNWIND [ n in nodes(path) | n.tag ]  
AS tags  
  
RETURN tags
```



Recorrido de grafos

- Recorrido DFS con nodos destino. Resultado.

DFS:



Neo4j: Recorrido de grafos

Big Data Aplicado

Julio Alberto López Gómez
JulioAlberto.Lopez@uclm.es



Big_Data
Aplicado



Curso Especialización
Inteligencia_Artificial y
Big_Data