

# Neo4j: Búsqueda de caminos mínimos

Big Data Aplicado

Julio Alberto López Gómez  
JulioAlberto.Lopez@uclm.es



**Big\_Data**  
Aplicado



Curso Especialización  
Inteligencia\_Artificial y  
Big\_Data

# Caminos mínimos

## Concepto:

- Obtención del camino mínimo entre un par de nodos de un grafo
  - Número de saltos (no pesado)
  - Mínimo coste (pesado)
- Los algoritmos de caminos mínimos son fundamentales en teoría de grafos
  - Útiles en entornos dinámicos
  - Útiles para aplicaciones que deben dar respuestas en tiempo real

# Caminos mínimos

## Aplicaciones:

- Necesidad de explorar un grafo
  - Nodos y enlaces
- Búsqueda de rutas óptimas entre nodos
- Ejemplos prácticos:
  - Encontrar rutas entre localidades ubicaciones
    - Google Maps
  - Grados de separación entre usuarios de redes sociales
    - LinkedIn

# Caminos mínimos

Un ejemplo:

- Importación de los nodos de un grafo:

```
WITH "https://github.com/neo4j-graph-  
analytics/book/raw/master/data/" AS base  
WITH base + "transport-nodes.csv" AS uri  
  
LOAD CSV WITH HEADERS FROM uri AS row  
  
MERGE (place:Place {id:row.id})  
  
SET place.latitude =  
toFloat(row.latitude),  
place.longitude = toFloat(row.longitude),  
place.population =  
toInteger(row.population)
```




# Caminos mínimos

Un ejemplo:

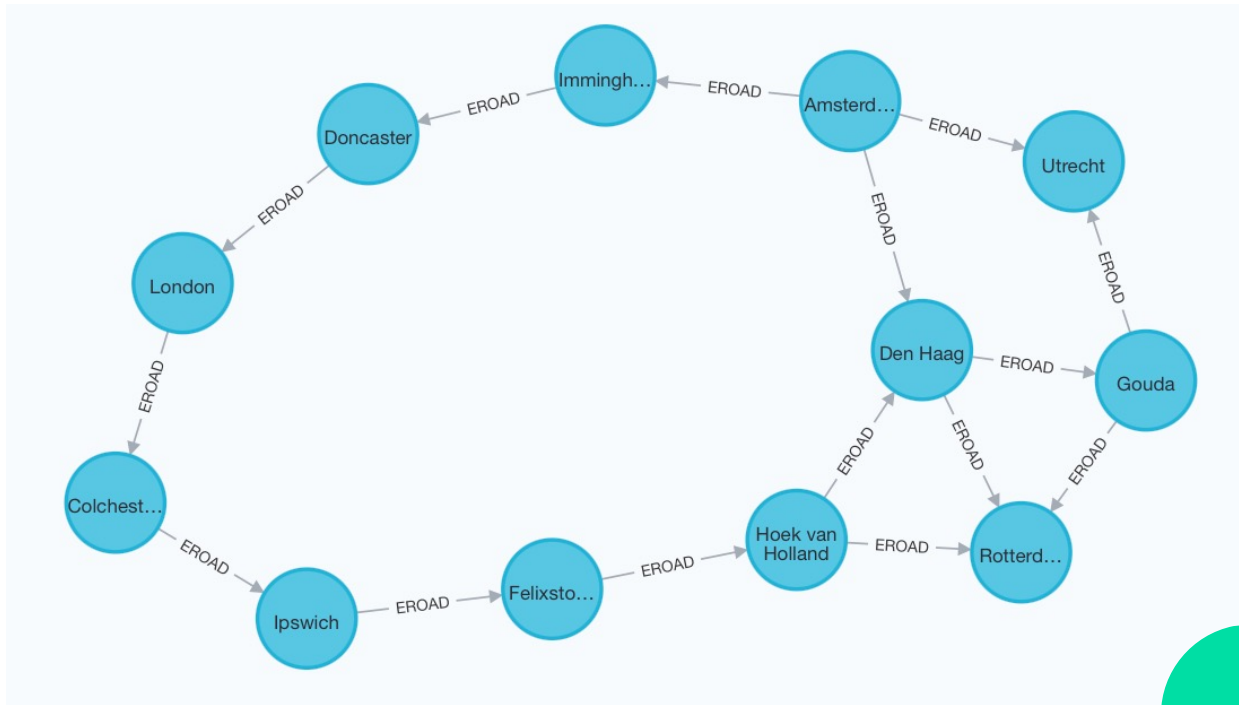
- Importación de las relaciones de un grafo:

```
WITH base + "transport-  
relationships.csv" AS uri  
  
LOAD CSV WITH HEADERS FROM uri AS row  
  
MATCH (origin:Place {id: row.src})  
  
MATCH (destination:Place {id: row.dst})  
  
MERGE (origin)-[:EROAD {distance:  
toInteger(row.cost)}]->(destination)
```



# Caminos mínimos

Un ejemplo:



# Caminos mínimos

Un ejemplo:

- Camino mínimo entre Amsterdam y Londres (no pesado)

```
MATCH (source:Place {id: "Amsterdam"}),  
      (destination:Place {id: "London"})
```

```
CALL algo.shortestPath.stream(source,  
destination, null)
```

```
YIELD nodeId, cost
```

```
RETURN algo.getNodeById(nodeId).id AS  
place, cost
```



# Caminos mínimos

Un ejemplo:

- Resultado
  - El coste del camino se mide por el número de saltos

place	cost
Amsterdam	0.0
Immingham	1.0
Doncaster	2.0
London	3.0



# Caminos mínimos

Un ejemplo:

- Camino mínimo entre Amsterdam y Londres (pesado)

```
MATCH (source:Place {id: "Amsterdam"}),  
      (destination:Place {id: "London"})
```

```
CALL algo.shortestPath.stream(source,  
                              destination, "distance")
```

```
YIELD nodeId, cost
```

```
RETURN algo.getNodeById(nodeId).id AS  
place, cost
```



# Caminos mínimos

Un ejemplo:

- Resultado
  - El coste del camino se mide por coste de los enlaces.

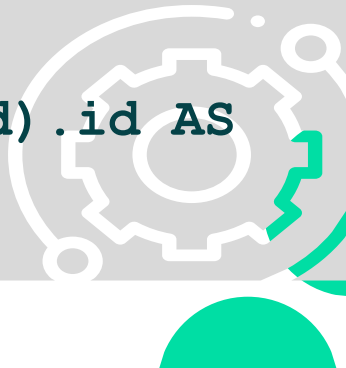
place	cost
Den Haag	59.0
Hoek van Holland	86.0
Felixstowe	293.0
Ipswich	315.0
Colchester	347.0
London	453.0

# Caminos mínimos

Un ejemplo:

- Camino mínimo entre Amsterdam y Londres (A\*)

```
MATCH (source:Place {id: "Den Haag"}),  
      (destination:Place {id: "London"})  
  
CALL  
  algo.shortestPath.astar.stream(source,  
    destination, "distance", "latitude",  
    "longitude")  
  
YIELD nodeId, cost  
  
RETURN algo.getNodeById(nodeId).id AS  
  place, cost
```



# Caminos mínimos

Un ejemplo:

- Resultado

place	cost
Den Haag	0.0
Hoek van Holland	27.0
Felixstowe	234.0
Ipswich	256.0
Colchester	288.0
London	394.0

---

# Neo4j: Búsqueda de caminos mínimos

Big Data Aplicado

Julio Alberto López Gómez  
JulioAlberto.Lopez@uclm.es



**Big\_Data**  
Aplicado



Curso Especialización  
Inteligencia\_Artificial y  
Big\_Data