

|  |
| --- |
| Neo4J Teamwork  Documentation  2020 |
|  |
| Due date: 15th of january, 2020  Information Repositories – UniOvi 19-20  Written by: Daniel Finca Martínez (UO264469)  Óscar Sánchez Campo (UO265078) |

Preface

[GRAPHGIST DOC REFERENCE MISSING]

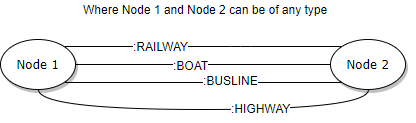
[MENTION TO DELIVERED FOLDER STRUCTURE MISSING]

Contents

# 1.Motivation

Domain: Transport systems across cities, villages and valleys.

This graph database has been created based on the insight of a map. This “map” would display cities, villages and valleys which are interconnected by some conveyance means. These are: buslines, railways, highways and boats.



As it can be seen from above schema, our domain allows nodes to be interrelated using three types of relationships: Railway, Boat, Highway and Busline. Although they are not represented with an arrow in the schema, they will be represented using a directed arrow in our graph. This is no problem, as they can be traversed both ways when querying.

Each relationship has a different set of attributes. Distributed as follows:

1. Railway:
   1. Line: Indicates the line of the linking railway.
   2. Average ride time: The arbitrarily “estimated” time to go through such relationship.
   3. Price: The arbitrarily chosen price to go through this railway relationship.
2. Busline and Boat:
   1. Average ride time: The arbitrarily “estimated” time to go through such relationships.
   2. Price: The arbitrarily chosen price to go through these relationships.
3. Highway:
   1. Average ride time: The arbitrarily “estimated” time to go through such relationship.
   2. Price: The arbitrarily chosen price to go through this highway relationship.

Moreover, nodes also have a defined set of attributes. No discrimination has been applied to different class nodes. In the sense that all of them contain the same attributes. These mentioned classes are: Valley, City and Village.

To continue, the set of attributes set for all the nodes in the domain is this:

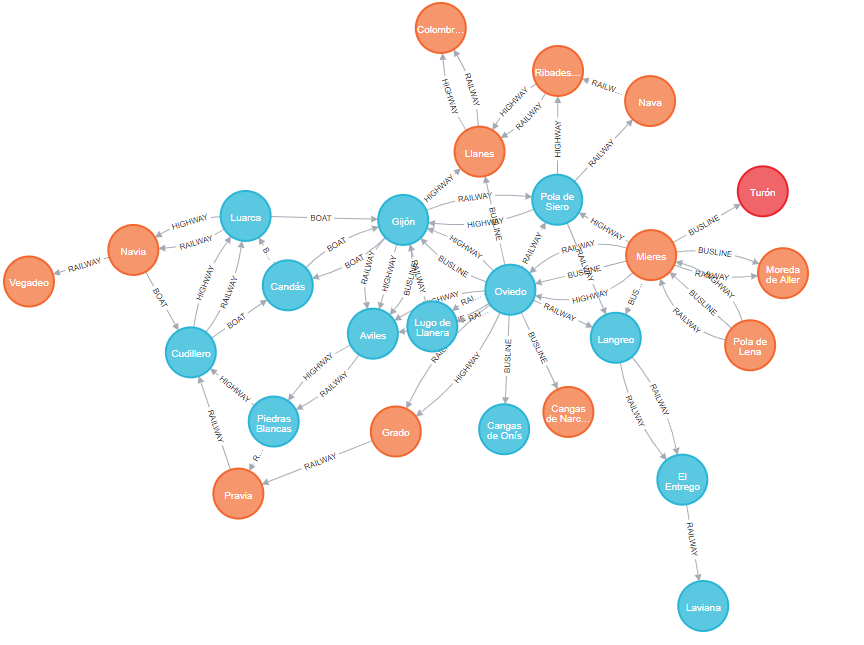
1. City, Village and Valley:
   1. Name: Indicates the name of the place the node represents.
   2. Inhabitants: Stores the number of inhabitants populating the place. (Data taken from the internet. Disclaimer: Could be outdated).

Nodes in our domain might have one class among these: City, Village and Valley. Though our model allows it, some connections would be logically and physically impossible. Not all places represented have a dock. In addition, only some of the possible links have been represented for the sake of simplicity of the database instance.

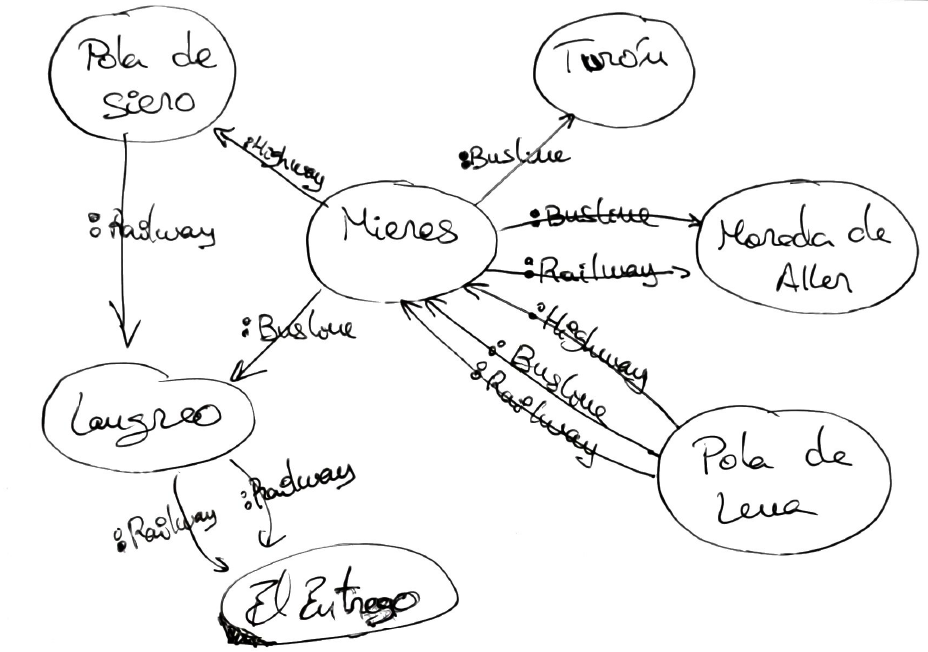
To conclude this summary of the graph database domain, the names and data have been selected using Asturias’s geography. Not every single location of the Principality has been represented because of the same reasons as stated beforehand (Not necessary to be exhaustive).

# 2.Database instance

This database is neither big nor small, but it is still huge when thinking about drawing it by hand. As stated in the task, a subgraph representation is allowed. However, Neo4J provides us with the tools to generate a high-quality image of the complete graph, which will be included down below just in case hand-drawn instance is not clear enough:



Above we saw the complete graph database instance obtained from Neo4J export system. Now we can see the hand-drawn subgraph:



This database instance features a total of:

1. 26 nodes.
2. 59 relationships.
3. 4 types of relation: Railway, Highway, Busline and Boat.

Result obtained after executing the creation script included in the delivery files. Taken from Neo4J browser:



# 3.Database creation script

[GRAPHGIST DOC REFERENCE MISSING]

The script can also be found inside the delivered folder with all the required files. On top of that, the folder “graph.db” has been included as a part of the files delivered just in case it is needed in the marking process of this assignment.