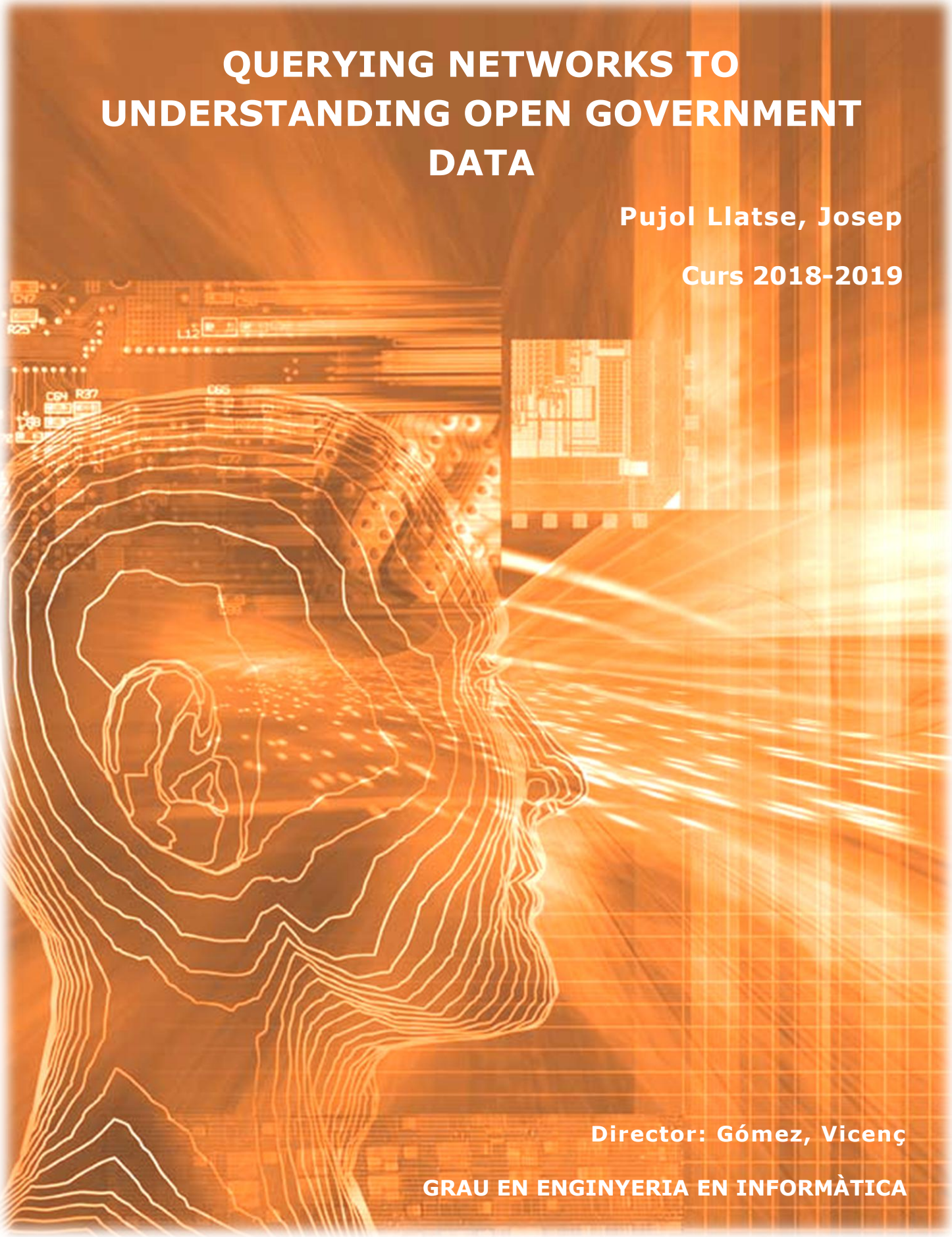
QUERYING NETWORKS TO UNDERSTANDING OPEN GOVERNMENT DATA

Pujol Llatse, Josep

Curs 2018-2019



Director: Gómez, Vicenç

GRAU EN ENGINYERIA EN INFORMÀTICA

upf. Universitat Pompeu Fabra Barcelona
Escola Superior Politècnica

Treball de Fi de Grau

QUERYING NETWORKS TO UNDERSTANDING OPEN GOVERNMENT DATA

Developing a tool that uses networks to help citizens understand and use the Catalan Open Government Data

Josep Pujol Llatse

Treball de Fi de Grau / 2018-2019

Tutor: Vicenç Gómez Cerdà

Universitat Pompeu Fabra Barcelona

ACKNOWLEDGMENTS

To my family and all the people that supported me, I could not thank you enough. You all know who you are. Thanks.

Thanks to the two persons that gave me legal council.

PROLOGUE

How many grate initiatives, ideas, inventions and tools have been forgotten throughout history because of poor implementation, difficult usage, deficient documentation or plain miscommunication of the benefits.

Open Government Data Initiatives (OGD) have grown during the past decade, but their potential usage has not been achieved, due to lack of effort or lack of real intention, in the humble opinion of the author. As will be shown during this work, OGD portals have technical problems, and the Catalan version does as well. One of the most important ones is the difficult access, that don't allow for a quick and simple browsing nor a easy extraction of patterns and conclusions.

So, a great idea is turning up to be a powerful tool, but remains accessible to very few specific people, namely, private organizations and some journalists. This accessibility is not mainly determined by economic factors, but also by technical ones. Citizens do not have the technical skills to use and analyze the OGD.

Our society nowadays is full of this type of imbalances that lead to an incomplete and unequal access to valuable resources like information and technical means.

This work is an initial attempt to improve the usage of OGD, first step towards a practical tool that can be improved in the future, because more data is not always more information.

ABSTRACT

The publication of data by governments is increasing yearly with instauration of Open Government Data (OGD) initiatives around the world. These have materialized into public policies and their objective is to allow citizens and other collectives to access the data that previously were only accessible by public administrations. Most of the times, this data is released in a portal with a huge amount of separated tables with no explicit relations between them nor with a way to integrate them with the everyday work of the public institutions. This makes them very complicated for the extraction of processed information to interested citizens and other interested parties.

The objective of this project is to improve the OGD by creating a new layer on top of these isolated resources. In particular, two sources of data have been considered: the Catalan OGD and the Diari Oficial de la Generalitat de Catalunya (DOGC). After extracting and integrating these data, a software tool has been developed that makes possible for a user to query the data according to different criteria. The result of a query is an interactive network visualization composed of nodes that correspond to entities and edges that represent some form of relation between the entities. Finally, a user case of this tool is described in which the software can be used to extract useful knowledge.

RESUM

La publicació de dades per part dels governs està creixent any rere any gràcies a la instauració de les iniciatives de Dades Obertes. L'objectiu d'aquestes polítiques és permetre als ciutadans I altres col·lectius l'accés a unes dades que fins aquell moment només eren accessibles per a les institucions. Molts cops, aquestes dades tenen format de portal en els que es posen gran quantitat de taules separades I sense relació explicita entre elles ni cap manera d'integrarles amb la feina que es fa dia a dia a les institucions.

Això fa que sigui molt complicat extreure informació processada als ciutadans estàndard I a altres parts interessades.

L'objectiu d'aquest treball és el de millorar les Dades Obertes Catalanes creant una capa per sobre d'aquestes taules. S'han considerat dues fonts, les Dades Obertes Catalanes I el Diari Oficial de la Generalitat de Catalunya (DOGC). Després d'extraure I integrar aquestes dades s'ha desenvolupat una eina de software que permet fer consultes segons diferents criteris. El resultat d'aquesta consulta serà la visualització d'una xarxa interactiva composta de nodes que correspondran a les institucions I els vèrtexs ho faran amb les relacions entre aquestes. Finalment , un cas d'ús en què aquesta eina pot ser usat per a extreure informació serà explicat.

INTRODUCTION

Motivation

For the first time, the work done with the OGD[123] by the public administrations during these last years starts to be acknowledged by the interested citizen as a good policy of governments, but the usage has not taken over yet, even more, as can be observed in the Illustration 1 , the visits and downloads in the portals are marginal.

Illustration 1: Evolution of the usage of the data from the Catalan OGD portal [4]

Probably the habitual citizen will never use OGD sporadically nor for a few times, but this doesn't release governments from the responsibility of facilitating the access and improving the quality of their OGD.

One of the problems of these portals is that all the information provided by institutions is in form of huge databases, with the only interactivity tool of word search engine and, in some cases a sandbox to create charts, leaving the linkage between the data to the manual work of users and their capacity, ensuring a tedious and inefficient experience.

[1] http://governobert.gencat.cat/ca/dades_obertes/
[2] https://datos.madrid.es/portal/site/egob/
[3] https://datos.gob.es/
[4] http://governobert.gencat.cat/ca/dades_obertes/indicadors_acces_cataleg_dades_obertes/

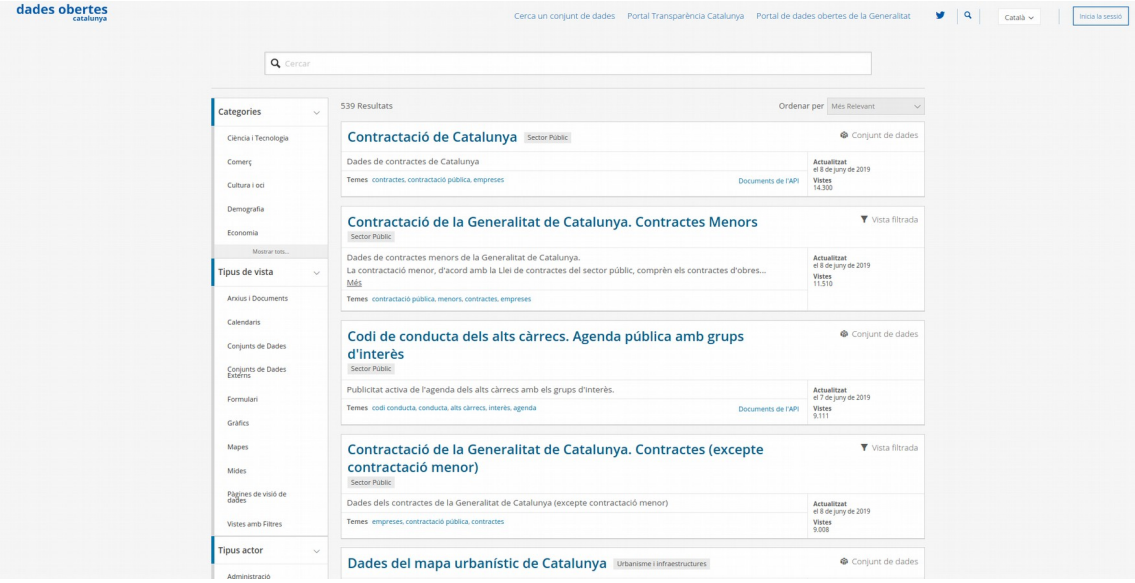


Illustration 2: List of tables of the Catalan OGD portal

When the relations of data have to be extracted manually, the richness of the information obtainable by the common user (standard citizen with user level of computer usage knowledge) is limited to the fields concerning a certain entity in each database e.g. only usable as a source of information to solve a precise ad-hoc question about an entity (for example, knowing the amount of money given by the public entities in a priory known public contract) , by thus letting more wide questions out of the focus.

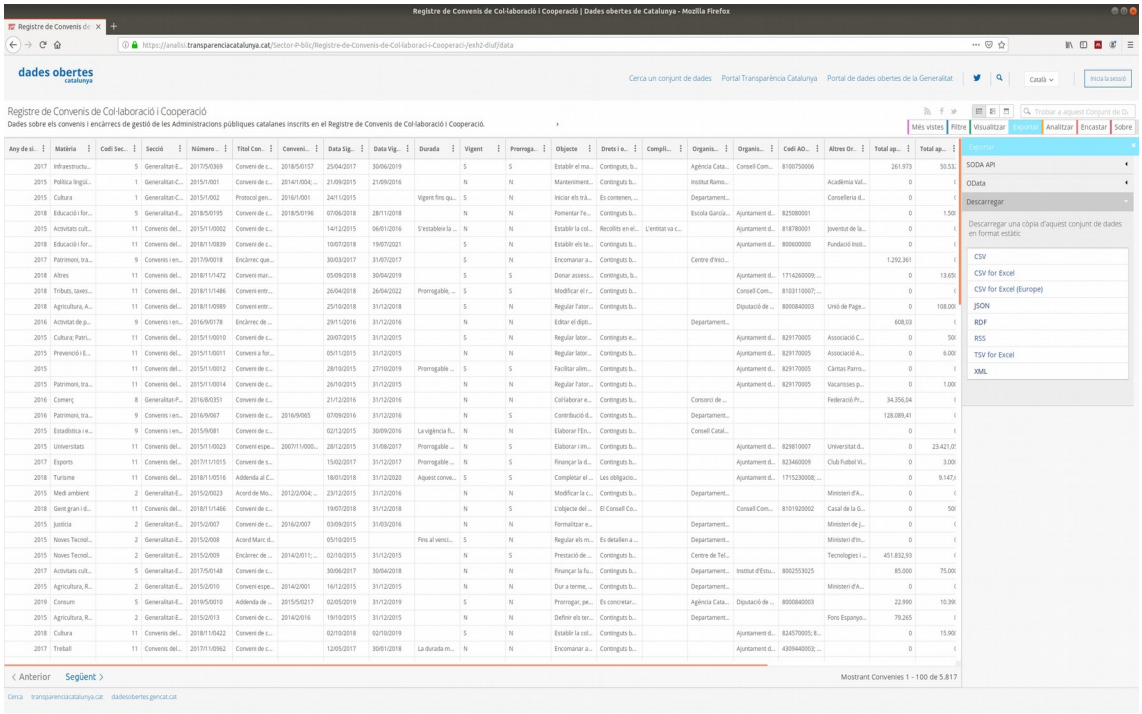


Illustration 3: Table from the Catalan OGD Portal

The same thing is applicable to other official publications like the DOGC, the Catalan official journal, that includes in a multiple hundred page document the daily activity of the public institutions. In both cases, a great source of information is desestimated by the general public as unusable because of its format.

This situation has to be addressed , as information as important as the public administration's operations must not be buried below mountains of data, instead should be offered to the public in a much more approachable way.

Objectives

The idea of this work is to provide a better way with which the standard and specialized users can navigate through the information about the administrations in a more easy way, through a superior layer that doesn't contain as much of information about entities, but represents a linkage of those across the OGD and the DOGC. To this end :

1. A general study will be done about the current state of the art of the resources used by the OGD platforms around the world together with a explanation of the general concepts of networks and they capabilities when applied to other fields.
2. A system that extracts data from the DOGC and OGD portals should be created. It should extract the entities in the public administrations that appear on them so that can be presented to the user in a more compressible way.

Table of Contents

Illustration Index

BACKGROUND

Basic Graph Concepts

Network science is a field that studies complex systems by looking at them as networks, formed by entities that are represented as nodes, and their relations, represented by edges between these nodes. Network science lies at the intersection of graph theory and other fields such as physics, statistics, machine learning and data mining, and many application domains.

Graph Theory

Graphs are the fundamental mathematical tools in network science. Their origins date back to 1735, in Eastern Prussia by the hand of Leonhard Euler, in the form of a solution to a urban famous planning problem named *The Bridges of Konigsberg* [4]. Graph theory provides the formal structure and the mathematical tools used in this work.

A *graph* (or network) $G=V,E>¿$ is defined as a set of *vertices* (or nodes) $V \in \{1,..,N\}$ and *edges* (or links) $E \in V^2$ that connect two vertices. If the order or directionality of an edge is irrelevant, i.e., an edge $e=(v_1,v_2)=(v_2,v_1)$, we talk about an *undirected* graph. Otherwise, the graph is *directed*.

A *path* between two vertices in a graph is a sequence of edges which connects a sequence of vertices that are distinct. The length of a path is *n* when there are *n* edges between the two connected nodes.

[4] https://www.maa.org/press/periodicals/convergence/leonard-eulers-solution-to-the-konigsberg-bridge-problem

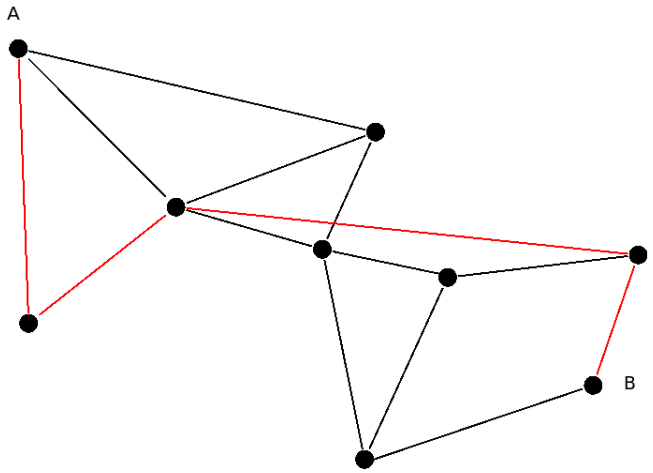


Illustration 4: Path from A to B

A fundamental property of a vertex is the *degree*, defined as the number of edges it has directly linking to other vertices. The *degree distribution* is the probability distribution of these degrees over the whole network. The *average degree* is computed as

$$\langle k \rangle = \frac{1}{N}\sum_{i=1}^{N} k_i = \frac{2L}{N}$$

N is the number of nodes

L is the number of links

k_i is the degree of the node i

For a given graph, the probability distribution of the degrees can be estimated by $P_k = \frac{N_k}{N}$, where N_k is the number of nodes with degree equal to k.

Traversing a graph: Breadth First Search

Several algorithms exist for traversing or searching over a graph depending on the needs. In this work, one has been used to find all the nodes related to a given one depending on the length of the path. This algorithm is named Breadth First Search (BFS) and it is used to walk the nodes of a network in a way that nodes are visited focusing in the current depth and after that incrementing the depth.

Note that the number of nodes visited grows exponentially at each increment in depth. For huge graphs that cannot be fully traversed due to computational limitations, a maximum depth is typically used. This limitations will be important at the end of the implementation of the system.

In pseudo-code , the algorithm would be:

```
BFS (graph, v , goal)
     let Q be a queue
     set v as discovered
     Q.enqueue(v)
     while Q is not empty:
           v=Q.dequeue()
           if v is the goal:
                 return v
           for all edges adjacent to v to I:
                 if I is not set as discovered:
                       set I as discovered
                       I.parent=v
                       Q.enqueue(w)
```

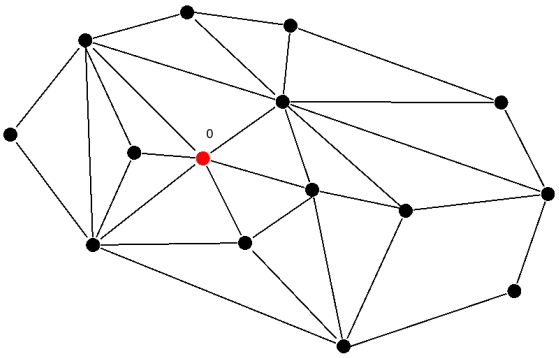


Illustration 5: Stage #0 of BFS algorithm

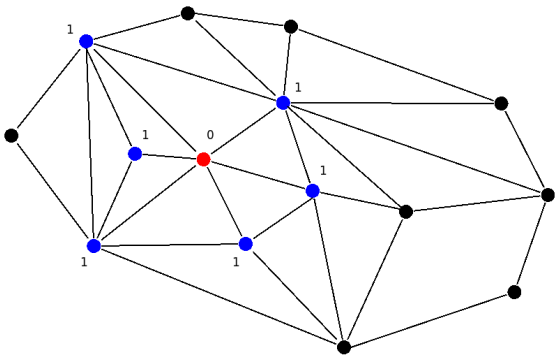


Illustration 6: Stage #1of BFS algorithm

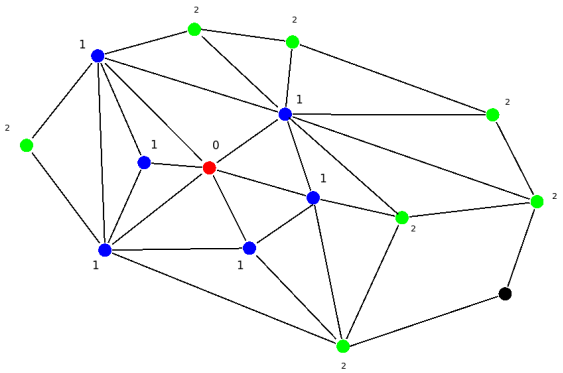


Illustration 7: Stage #2 of BFS algorithm

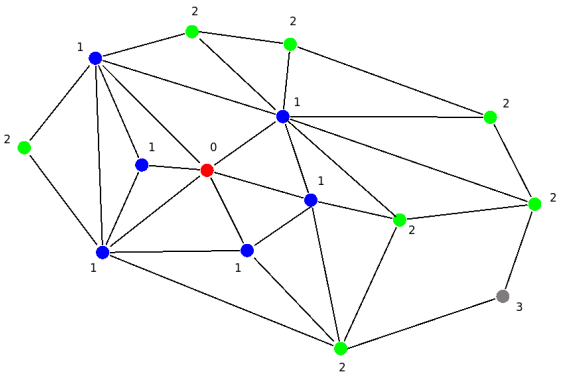


Illustration 8: Stage #3 of BFS algorithm

Network Science and Visualization

Networks and graphs have for long been used for the analysis of fields completely different from computer science or mathematics. The elements in this list can be very different from one to each other, and the only thing in common is the fact that someone has found a way to represent their elements in a network-like structure. The first of this various fields may be the Internet, that by definition is a network. In [1] the authors define the world-wide web as a scale-free network, in which the degree distribution follows a power law, result that is used to study the properties of the HTML connections and the hierarchy in the WWW. In [2] the authors propose the mechanism of preferential attachment that has into account the scale-free topology observed in the Internet.

Another field into which network science has been applied is social sciences, which are composed of lots of fields that have aspects that can be modeled in a network-like structure [3]:

"*For social scientists, the theory of networks has been a gold mine, yielding explanations for social phenomena in a wide variety of disciplines from psychology to economics.*"

There are numerous examples of this approach. For example, in [4] the social network emerging from the commenting activity in online discussion threads was analyzed and compared with other social networks such as friendship networks. More recently, the political elections was used to discover communities by analyzing the community structure of retweet networks [5].

During the development of this work, a very similar tool to the system developed in this work it was found [6], but dedicated to the understanding of the public founding of research organizations.

Open Government Data

According to the Organization for Economic Co-operation and Development (OECD) [5]: "Open Government Data (OGD) is a philosophy- and increasingly a set of policies - that promotes transparency, accountability and value creation by making government data available to all. "

It embraces different ways of making internal data of the government or public institution available to a general audience with different goals.

More than 15 years ago, the European Union (EU) set the legal framework to allow the re-usability of the data that encouraged the publication of Government data in the countries by the hand of governments and regional and local institutions. Although this was public data made open, this publications did not have the same objectives as the nowadays OGD, this where mainly economic reasons, but allowed the start of it in the EU. Since that moment, public institutions and third parties have been developing in the idea of OGD and creating new public policies to start offering the data to the citizens and private organizations.

The objectives that had lead the implantation of the OGD policies are various :

[5]https://www.oecd.org/gov/digital-government/open-government-data.htm

- To enhance transparency and accountability: by publishing data from the public administrations, citizens can access to a source of information that can be used to hold politicians and administrators accountant with a more reliable information.
- To stimulate of innovation and economic value: as said, from the beginning, institutions and private sector [7] have always had the perspective that the data that is being made available for private consumption too will have a positive effect on the economics of their governed region by providing the private sector of first class information to provide products and more efficient services.

- To promote participation of the public in the policy making: allowing citizens and the private sector to take part in the policies of government has started to be a idea in public institutions. One necessary tool in this initiatives is the access to information not accessible before.

Although the expected benefits are interesting, the Open Government Data faces several challenges, including data quality, data ownership and privacy violation benefits and others [8].

The publication of this data was delayed, but during the last decade the amount of information has increased vastly and with it, the attempts of create evaluations frameworks [9]. But in the last years, the institutions have started to create specialized portals for groups of data attaining a concrete aspect of the public administration and to decentralize the publication of the data [10].

The improvements and advances that have participated in the development of the OGD are basically the same ones that have changed the way to technically express data and to handle it (formats, infrastructure, visualization , etc), but the particular work-flow of this kind of institutions make these changes appear delayed. Being these the technical tools that have been used (or expected to be used) in this project, now will be explained in more depth.

The process of publishing information has evolved greatly from the simple table with data, not only in the matter of just representing it contained in tables, but using formats that allow to provide more information.

This is not a new field by any means and numerous ways to include added value to the data had been appearing through the years, in which has been clear that there is a great need to add information about the data, or as more commonly known metadata. As said in [11]
“Metadata, most usefully, can be defined as descriptive information about digital information objects for access, use, preservation, interoperability, and management purposes. “

One other technology that has been used to improve the quality of the OGD is the Linked Data [12], integrated by methods that try to add metadata with the intention of allowing it to be interconnected by adding semantic meaning and structure to the fields or columns.

Linked data could be especially useful to this work, because knowing the semantic meaning of the data that can be found in a source makes it automatic processing and usage much easy, in our case, this would mean that the hole extraction of the nodes and relations could be automatized. A minimal implementation of this technology can be observed in the Catalan OGD that offers possibility of downloading the databases in RDF [13]. although this being a great improvement, the names of the tags are not standardized and there is no structure, laving this option out of the usable resources in this work. Here can be seen a comparison between a RDF from the Catalan portal [14] and the Madrid OGD portal:

Illustration 9: Madrid RDF on public property

Illustration 10: Catalan RDF on public contracting

In [15][16] the Brazilian and Greek authors have tried to apply linked data to their own OGD.

USE CASES

Before any development takes place, the problem explained in the Introduction section has to be properly understood and the items in the use case have to be examined.
The main objective of creating a new layer between the user and all the open data about the public authorities has to be divided in more than one so it can be properly satisfied.

Three use cases have been studied will be the ones that the tool will try to address and tin the following sections the reader can .

Use case 1: Retrieve public information related to a name

At this moment, if the user wants to retrieve all the information that the public administration publishes about a person or a organization, it is forced to search by the name in all the tables in the Catalan OGD and the multiple portals in which the Generalitat offers information about specific subjects. Even more , it is going to need to do a search on the DOGC portal's finder tool, that works poorly and very inefficiently.

Name	**Retrieve public information related to a name.**
Description	The user pretends to get information about some organization, indifferently from the nature (private or public), without knowing the type of the information, it could be about any topic or source, or could have relation with the public sector or not.
Actors:	- Standard citizen. - Journalist. - Private organization.
Pre-conditions	The user knows the name of the entity from which wants to know information about.
Post-conditions	The user has been informed of all the relations that the given entity has with the Catalan public administration and any

	other that are related to it.
Use case flow	1- The user opens the application and executes the query module. 2- The user selects the main tool in it's browser. 3- Writes the name in the form together with length of the path displayed to the other entities, the far-ness. 4- The user is presented with an interactive network that represents the interaction of the given entity with other institutions and private entities. 5- The other entities represented in the network can be dragged and the information explaining the nature of the relation is displayed in the side of the network. 6- The user it's informed with all the relations that the previously given entitle has with the others that appear in the public data. *7- The user exits the application.* *Alternative Flows* 7B- The user is interested in another entity that has appeared in the network and keeps using the display of the current network to retrieve information *7C The user is interested in another entity that has appeared in the network* *8 The user goes back to the form and inserts the name of the new entity* *1-7 Repeat*

If the tool implemented satisfies this use case, all this information and relations will be presented in only one place through this added layer from which the user will be able to identify the source and visit it to obtain more precise information or the information about persons that may have appeared in the network.

Although some information will be present in the tool, not all the sources that can be found in the portal are included.

This is obviously due to the immense work and time it would require, and understanding that this work is not a final project but a proof of concept explains this situation and because the immense amount of information added into the network would make very difficult to obtain useful information from the queries due to overpopulation of objects in the result networks.

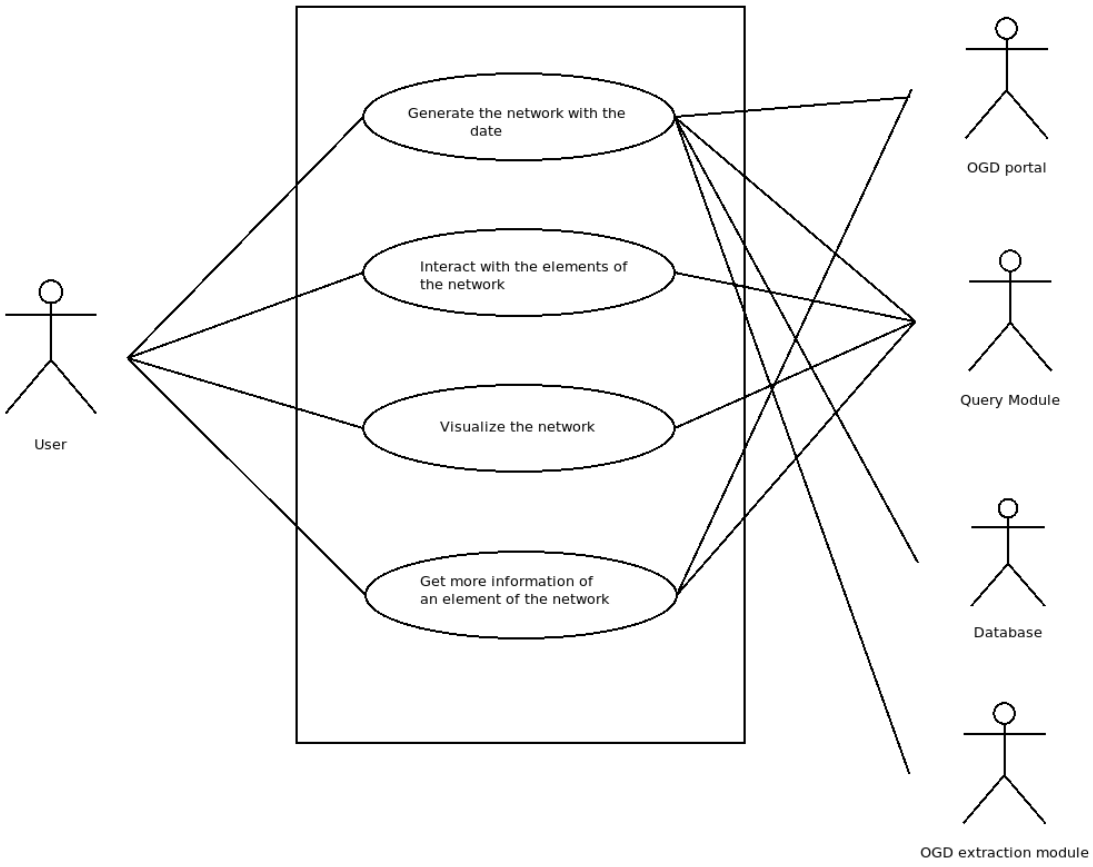


Illustration 11: Use case

Use case 2: Visualize the DOGC journal of today

The other public source of information that is used on this work is the DOGC journal. As said previously, this document is published daily and can achieve several hundred pages. What if there could be a better way to get informed of the contents of the DOGC. Although this use case is focused in the present day of the consultation, the date has to be given to the HTML page in the browser, so the information of any day could be visualized.

Name	***Visualize the DOGC journal of today.***
Description	The user pretends to get informed about what's on the DOGC journal of that day.
Actors:	- Standard citizen. - Journalist. - Private organization.
Pre-conditions	User knows the date. *The database has been loaded with the relations in the DOGC of the said day.*
Post-conditions	The user has been informed of the articles that are published in that date and witch entities participate in them
Use case flow	1- The user opens the application and executes the query module. 2- The user selects the DOGC journal tool in it's browser. 3- The user inserts the date of the day that wants to be informed about. 4- The user is presented with an interactive network that represents the entities that appear in the articles of the imputed day. 5- The other entities represented in the network can be dragged and the information explaining the nature of the relation is displayed in the side of the network, and by so , being able to clicking on the link of the article and use the

	DOGC portal to read the articles . 6- The user is satisfied with all the information provided by the system. *7- The user exits the application.* *Alternative Flows* 7B- The user wants to know the entities involved in the DOGC of another day *1-7 Repeat*

As in the fist use case, the objective is not to offer a substitute of the DOGC document, is to present to the user a way to navigate through its large amount of rows.

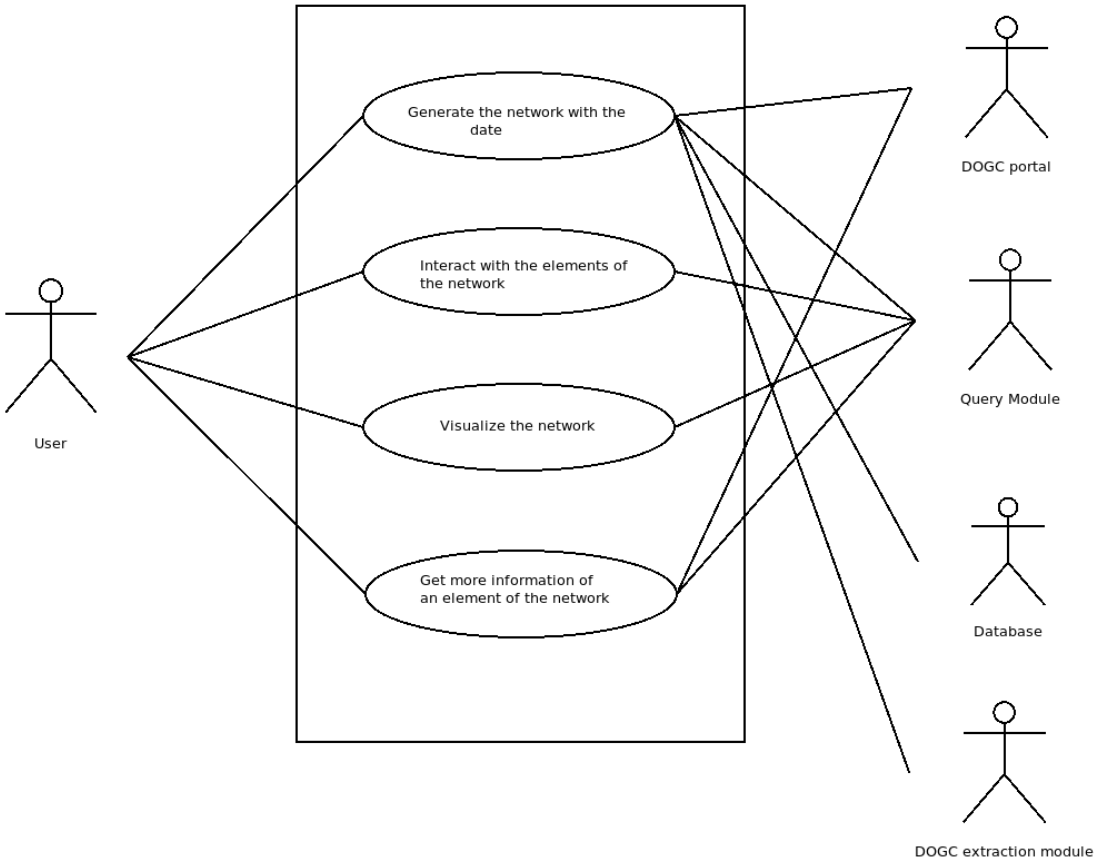


Illustration 12: Diagram for Use Case 2

Use Case 3: Retrieve information concerning a relation attribute

This use case is oriented to a more focused search in the network concerning a particular attribute in the relations.

Name	Visualize a the network generated by attribute name and value
Description	The user wants to visualize the network that results of only considering the edges that satisfy a logic condition about an

	attribute of them.
Actors:	- Standard citizen. - Journalist. - Private organization.
Pre-conditions	*The user knows which attribute wants to query about and a logic condition that has to be satisfied about it*
Post-conditions	The user has been informed of the relations between edges that satisfy a logic condition of an attribute.
Use case flow	1- The user opens the tool in its browser. 2- The user selects the edge attribute. 3- The user opens the application and executes a query formed by a name of an attribute, a logic operator and a value. 4- The user is presented with an interactive network that represents the entities united by the edges that satisfy the logic condition. 5- The other entities represented in the network can be dragged and the information explaining the nature of the relation is displayed in the side of the network. 6- The user has absorbed all the information about the public authorities that could be interested in. *7- The user exits the application.*

This use case will allow the users to query the network to obtain results about more precise elements on it.

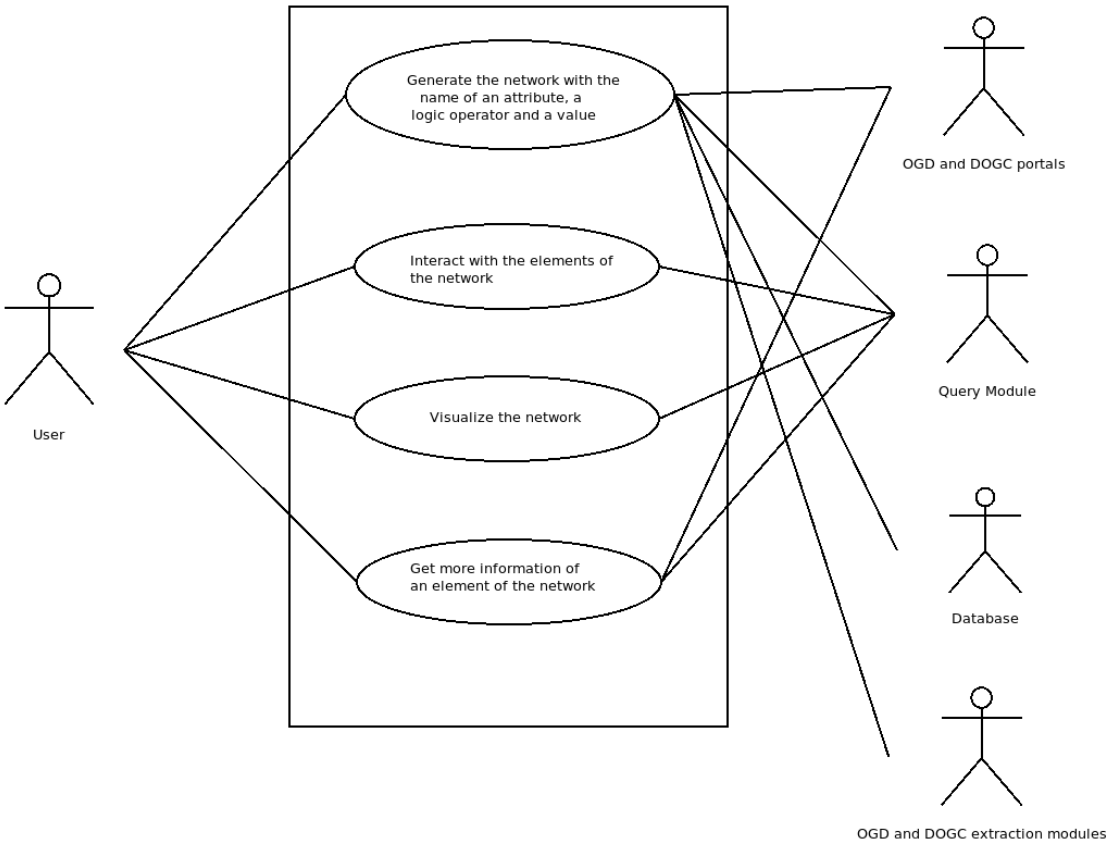


Illustration 13: Diagram for the Use Case 3

DATA MODELING

As said in the objectives, the main purpose of this work is to generate a tool that transforms the entities that appear in the OGD into a network and extracts relations from these sources and the DOGC and relates them by means of a network.

All this information has to be stored in a database in a way that resembles a network, because when the Query Module is executed its needs to fit in a graph structure. For this, the information that is extracted has to be stored in a way that fits an information model that later will allow to use the graph logical structure in the desired way.

In this project , the attributes that define the objects of the network have been chosen so that inside of them all the information of the persons, organizations and their relations can be stored. The modeling became a process in which tables of the OGD Catalan portal have been chosen for it's ability to explain the Catalan public sector and it's relation with the private sector. And from this tables, creating attributes that could store the information in them.

Database	Description	Information extracted
Organization chart of the Generalitat de Catalunya [6]	XML file with the administrative structure of the organizations dependent on the Generalitat.	- Public administrations with locations - Persons with charges - Relations of dependency between organizations Relations of charge between

[6] ttps://analisi.transparenciacatalunya.cat/Sector-P-blic/Organigrama-de-la-Generalitat-de-Catalunya/8s6p-h233h

		persons and organizations
The Generalitat Interest Groups [7]	Table that contains all the interest grups in Catalunya , representatives , its sector and public found	Representatives, private entities, relations with the Generalitat with public found in some cases. Charges of the representatives.
Public licenses of Catalunya [8]	Table with all the contracts between public and other entities, normally private	- Public and private organizations - Relations of contract with amount, description, id code and date
Cooperation and collaboration agreements [9]	Table with all the contracts between public and private entities	- Public and private organizations - Agreements as relations with name, description, amount and id

Having a list of these databases, the necessary attributes were added to the nodes and edges so the information considered important could be retained in the database.

A minimal set of sources have been used, but each one has been chosen with some structural reason in mind and having the objective to resemble how the data about the institutions in the Catalan OGD is organized. This has to be remarkable as there is a huge number of tables that, although related to the public sector, represent data that is not about the institutions, e.g, meteorological archives or names of the missing persons

[7]https://analisi.transparenciacatalunya.cat/Legislaci-just-cia/Registre-de-grups-d-inter-s-de-Catalunya/gwpn-de62

[8]https://analisi.transparenciacatalunya.cat/Sector-P-blic/Contractaci-de-Catalunya/hb6v-jcbf

[9]https://analisi.transparenciacatalunya.cat/Sector-P-blic/Registre-de-Convenis-de-Col-laboraci-i-Cooperaci-/exh2-diuf

during the Spanish Civil War. The sources that fall in this criteria have not been considered into this modeling of the network. On the other hand, there are other sources that have not been added and have been considered in the moment of the modalization.

The objective of this modeling is that a public administrative structure and other satellites can be represented, together with their relations and that new relations from the DOGC can be added afterwards.

Modeling of nodes

After the election of the tables to be added to the database, the following attributes need to be incorporated to represent the information they contain.

Name of the attribute	Description
Name	Name of the entity
Type	This attribute can be four different values: • Person • Public entity • Private entity
Sector	General representative words of the sector to which the entities corresponds to.
Public Found	If applicable, the amount of money the entity receives from the public sector.
Location	Physical location. Address or general location.
Extra	Extra information that can be found in the source database. This attribute is manually and arbitrarily defined at the time of the gathering of the information.

Modeling of the Edges

After considering all the relevant tables in the OGD, this have been the selected attributes to store the information about edges in the table.

Name	Description
Position	Charge that a entity (usually a person) sustains in an organization
Public found	Amount of euros that a public administration has granted to an organization
Id Code	Some rows in the OGD represent objects that have identification number. This is the case of Articles in the DOGC or Cooperation Agreements
Amount	If there is another number that has to be recorded
Parenthood	Some relations represent dependency between organizations.
Title	Some of the rows in the tables represent elements that have a representative title. This is the same case of DOGC articles.

GENERAL STRUCTURE

The code and infrastructure generated for this work has some differentiated elements:

- Extraction module: fetches the data from CSV files and creates structures of the nodes and edges that will be added in the database

- Database: permanent storage system for the network.

- Query module: extracts the elements from the database to generate a Graph with the Graph_Tool [10]and allows the user to query it to visualize the desired sub-graphs.

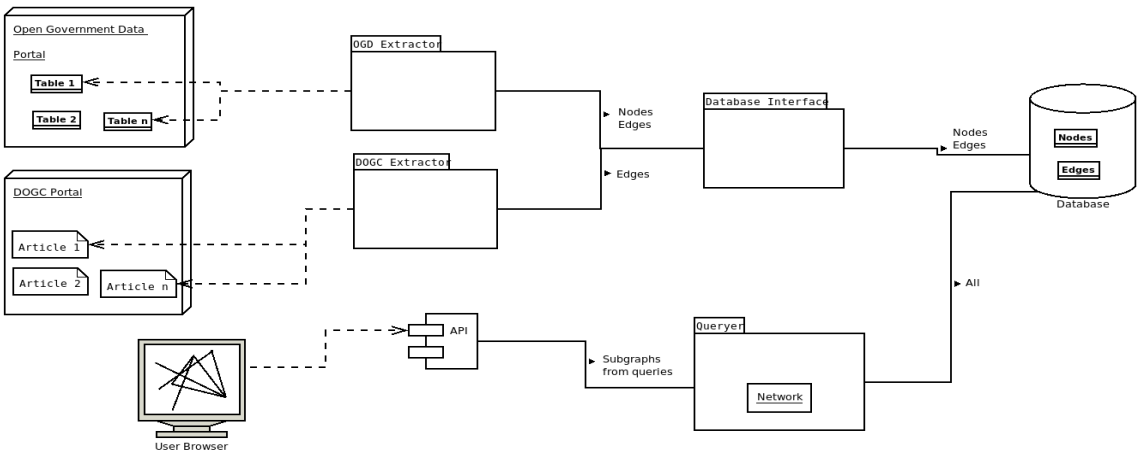


Illustration 14: Structure of the software system

An appropriate explanation of each component although will be explained on the following chapters.

Nevertheless this work has been thought conceptually as a complete superior layer between OGD about the public authorities and the citizen, for legal reasons, although the data is public, the GDPR states that creating a system (this software) that stores information of identifiable persons it's forbidden without the authorization of these. In addition the Law that regulates the re-usage of the OGD in Spain states that information extracted from these sources that is about persons cannot be aggregated with

[10] https://graph-tool.skewed.de/

information from other sources (the DOGC articles in this case). Having this restrictions and little inclination to violate them, in this software the nodes representing natural persons have been anonymized. Further explanation about this decision can be found in the chapter named “Legal considerations”. This modification means that when the rows in the tables are fetched, the names (only thing that identificates the natural persons) won’t be recorded and a integer number will be passed as a name.
This anonymization does not interfere much in the usage of the data of the OGD portal , since all the nodes representing persons are terminal, normally connected with the entity, public or private which represent and most of the times these are the ones that are connected to the rest of the graph through the relations with important meaning. Persons in the OGD only hold importance when have charges in the public administrations.

Obviously, the traceability its preserved. If a user is interested in a person node, although the name will not appear in the visualization , the source table will figure as attribute and the user will be able to access it by his/hers own.

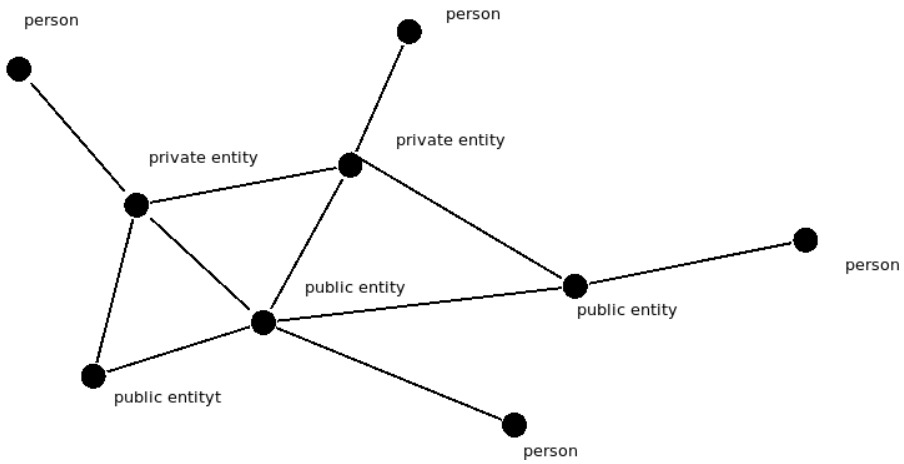


Illustration 15: Example of entity-name structure

When this decision does interfere is in the gathering of relations that come from the DOGC, since one of the elements between which relations are indexed are persons that appear in the database. Since this persons will not appear with the names , the relations wont be linked to these nodes. This would only mean that the relation will be only established between the public and private entities and by thus, the amount of relations created decreases.

EXTRACTION MODULE

This module is concerned with the extraction of the information and its conversion to the network structure. As said this information comes tables of the Catalan OGD portal and from the DOGC journal.

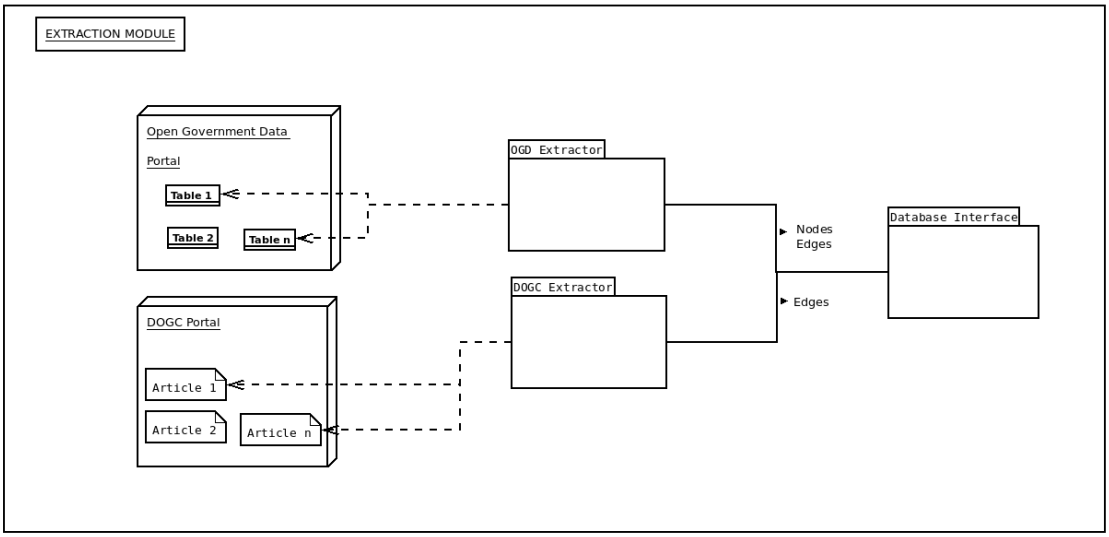


Illustration 16: Structure of the Extraction Module

Extraction from the Catalan OGD

The first thing in the process of generating the network is obtaining the source data, the databases that will be used to extract the entities to be stored in the database.

At the end of this work, this functionality is implemented as a simple reading of CSV files that contain the same rows as in the tables, that have to be previously downloaded manually from the portal. Nevertheless the Catalan portal offers APIs (groups of virtual endpoints on the Internet that can be accessed to obtain data) for the source tables, the manual approach has been chosen for this proof of concept to allow quick debugging.

After having the files opened, the rows with the information are read sequentially. In this process, from each row, preset fields for each specific table that contain important information will be fetched to be structured in a node Python object or a edge. This elements will be added to the database if they not exist already. The criteria of existence in the case of the nodes is that a node with a name that already exists on the database will not be added, and in the case of the edges, all the attributes will be compared.

The decision of the nodes being identified by the name has been error prone, as the in the OGD, an entity can be found under multiple names. Some times including the legal format of the organization (“S.L.”, “S.A.” , etc), that can be even found in multiple forms (“s.a”, “S.A” even “SA”), sometimes the names of organizations can be found in capitalized letters.

	CLINICA DALMASES SL
	Josep Selga, SL
	FACTOR ENERGIA, SA
	Llibreria Adserà, SL
	VERTEX TECNICS
	Edicions del País Valencià, SA
	TACKLEN MEDICAL TECNHOLOGY, S.L.
	ALMIRALL, S.A.

Illustration 17: Some naming policies

If these incoherence wouldn’t had been addressed, multiple nodes would have been generated for the same entities. This has been mostly corrected by the addition of a name normalizer. This piece of software gets all the names written in the same format, with words capitalized only in the first letter and the legal form always in the same format. The parentheses where not dealt with.

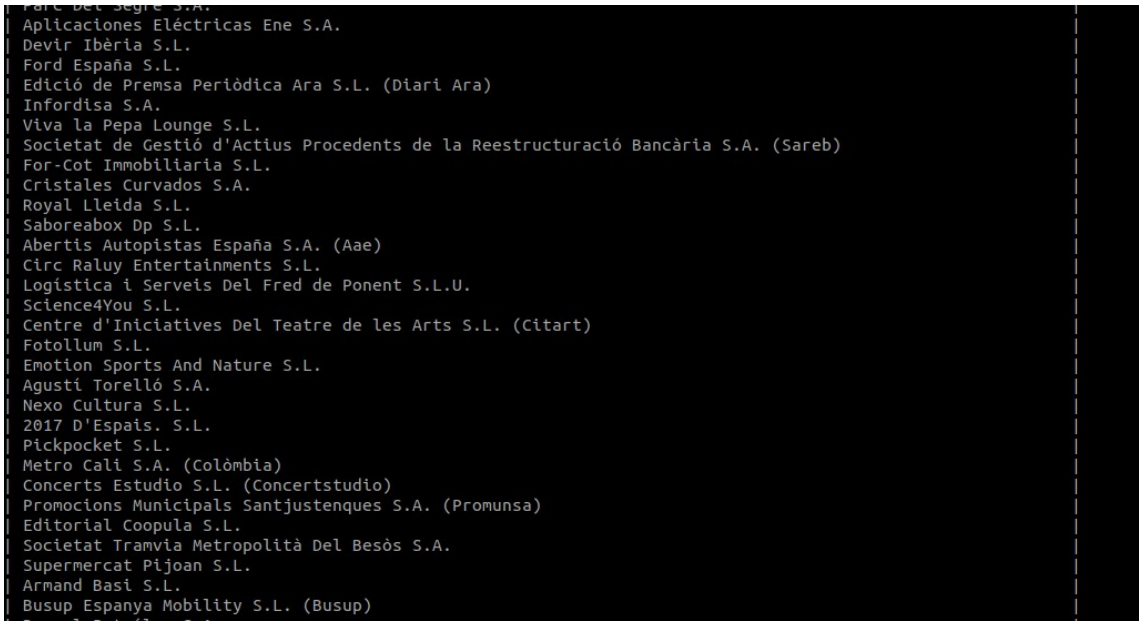


Illustration 18: Names after normalization

Edge and node python objects are only used to store data, do some parsing and checking after they are inserted on the database. In no way, the nodes and edges will be linked (as in the final graph). First the nodes will be added to the database, and then , it is queried to know the unique node ids , generated in the database, that will reference the origin and target of the edges.

At the end of the execution of the script that loads all the elements from the OGD tables, the number of nodes is 26000 and the number of edges is around 29000 instances.

Obtaining data from the DOGC

The Diari Oficial de la Generalitat de Catalunya (DOGC) is the official journal in which the Catalan government, it's administration or any public entity, publish all the texts that have validity in a format of a daily compilation of articles. This includes rules, general provisions, agreements, resolutions, edicts, notifications, announcements, legal resolutions and others. This publication is edited all working days, although if the Catalan government find it suitable can be published in weekend days or festivities and it's published in Catalan and in spanish.

The number of articles vary from day to day, from less than a dozen in some special cases to a hundred or more, in addition, the length of the articles has a big variation as well, ranging from a paragraph to tenths of pages in some cases.

The publication of this journal it's dependent on the Entitat Autònoma del Diari Oficial i de Publicacions (EADOP) [11], that is in charge of the edition and distribution of the journal and other legal texts.

There are multiple reasons to use this documents in this work. The most obvious is that provides a day to day great source of information that reflects the activity of the administrations, from relations with the Spanish government to the little contracts of little local authorities.

Scraping the Articles

The nature of the data to be extracted from the DOGC is very different from the one obtained from the OGD, but even more different is the way to obtain it. The DOGC can be found in two forms in the DOGC portal [12], a PDF document and a HTML version, in the DOGC portal, in which the user can access the journal of a certain day and navigate through the sections, that collect the articles published by administrations of the same nature (local authorities, the Catalan government, justice administration, etc). Inside of this sections is where the articles are organized and the links and PDF can be accessed.

In the development of this project, only the HTML version is used. This is because the ease of automatically scrape the DOGC portal for the individual texts and the metadata by date and because the grate size that the journal some days achieve (of the order of multiple MB).

[11] https://dogc.gencat.cat/ca/pdogc_eadop/
[12] https://dogc.gencat.cat/ca/

The functioning principle of the script is that scraps the individual journals first by date and then by section and article. This is done this way because of the self explainable composition of the portal's URLs, that allow for a very easy serial access of the articles.

Illustration 19: URL format of the DOGC

As can be seen, the URL allows to visit directly a page where all the articles are listed and by so, the link to the concrete articles can be scraped and downloaded for the analysis of the text and metadata.

```
self.url = 'http://dogc.gencat.cat/ca/pdogc_canals_interns/pdogc_sumari_del_dogc/?anexos=1&language=ca_ES&numDOGC='
for n in dogc:
    for section in range(5):
        page = urllib.request.urlopen(self.url + str(number)+'&seccion=' + str(section) + str(n) ).read()
        soup = bs.BeautifulSoup(page, 'lxml')
        for bloc in soup.find_all("div", {"class": "disposicions"}):
            for link in soup.find_all('a'):
                if link.text == 'Text i fitxa':
                    url = link.get('href')
                    page = urllib.request.urlopen(url).read()
```

Illustration 20: Code for extraction of the articles

For number of dogc :

The script loads the pages of the DOGC using the request package and the parsing of the DOM element to get the paragraphs and other information with the help of the Beautiful Soup [13] package with the help of the help of the LXML parser [14].

[13] https://www.crummy.com/software/BeautifulSoup/
[14] https://lxml.de/

Entity Extraction and Relation Creation

The entities have to be extracted from the body of the text so that a relation between them can be added to the database, but first the tool must know which names appear in the article , so the relations can be created between them. A naive Named Entity Recognizer (NER) has been created with the intention only to extract this names that are capitalized with only articles in between words e.g: “Ajuntament de Barcelona”, “Vilafranca del Penedés”.

The nature of the articles that appear each day on the journal make of a great source of relations, even more , in lots of them the quantity of named entities can be of the order of tens or even hundreds. This new candidate entities most of the time won’t be interesting for the scope of this project and because of this not all of them will be inserted in the database because of this, only relations between entities that have been extracted from the tables in the OGD and already appear in the database will be inserted in it, and by thus , no new entities. The process after obtaining the text of the article will follow like this:

1. The text will be tokenized using the NLTK [15] word tokenizer and converted to a list of words
2. The list obtained will be given to the NER, that will run over all the words and extract all the candidate entities. This list will include lots of other grups of words, obtained because the simplicity of the NER e.g. all the words capitalized that are not names like words after points and identification numbers that start with letters.

Although the number of false positives at this stage is great, the number of false negatives is near to zero. This is because of the official nature of the documents, that oblige the submitter of the article (administrators in the institutions) to follow a strict guides [17] the names always appear well written and at least one time written with the name and surnames, like they appear in the database.

[15] https://www.nltk.org/

3. All the groups of words obtained in the previous part are contrasted with a corpus to obtain only the ones that appear in the database. This corpus has been generated quarrying all the names of entities that can be found in the name column in the nodes table.
4. After obtaining all the named entities in the article that appear in the database, a list of all unique combinations of the entities is computed e.g. for ne1, ne2, ne3 the couples [e1,e2], [e1,e3], [e2,e3] are computed.
5. A new edge is added in the database between the entities and the id of the article and the date of the journal is recorded as extra information to allow traceability with the original article and future work with time.

The new edges extracted from the DOGC journal represent new relations that are formed in the day-to-day activity of all the public administration and easily inform of a interaction that without this tool would be buried in hundreds of pages of a administrative document than by any means allow for easy and quick retrieval information.

As maybe the reader has thought, the amount of computation required to extract the couples of names in the articles is great, and comprises several steps, from the download of the multiple pages, from the DOM tree parsing, passing through the tokenized, the NER, the comparison with the strings in the corpus (huge cost as the false positives of the NER are lots) and finally the composition of the couples to be translated to relations in the database. This amount of calculations for just a few relations makes this process inefficient and slow. If more time could be invested in the developing and the usage of a more advanced NER, the results could improve dramatically, but as a prove of concept the results obtained are enough.

DATABASE

The data has to be stored in a database so that all the attributes are preserved and the network structure is easily represented. Technically , the server used is a habitual MySQL that is accessed by the Python scripts with the help of a package made by Oracle called “mysql-python-connector”[16].

The need of storing the information and not retrieving the specific data every time a query is done comes from the high computational needs to generate the relation between the nodes. The data retrieved from the CSV files has the same structure as the tables in the OGD portal and by thus it’s not linked. During the processing of these files and the articles of the DOGC the relations are created. This process of generating the links between the entities (the edges) is computationally intensive, specially in the case of the DOGC articles (the specific reasons can be found in the Extraction Module chapter).

Doing this processing at the start of the execution would mean at least five minute wait only to create the relations concerning the OGD entities, and multiple hours just for ten DOGC journals.

The portal of OGD offers APIs that allow to query the tables by name and this could mean that the tools created in this work could generate sub-networks by name during the execution of the query module, by calling this API end-points with this names and by thus not making necessary to store the information nor even the generation of the hole network. But with this method the integration of relations found in the DOGC articles would be impossible.

The database will be made of only two tables, one for nodes and the other fot edges.

The attributes are trivial to be represented, they will be stored mainly in VARCHAR columns of various sizes, having in mind that some fields on the tables on the OGD are unnecessarily big.

The fields that will be added that have not been obvious in the modeling of the network are the following:

[16] https://dev.mysql.com/downloads/connector/python/

- Sources: in each table, a column will be added so the source of the individual items can be stored. To this field, each time that entity or relation appears on a different source table, a two letter code will be added to this field. If in the adding process of a node it's found that a node with that name happens to already exist, the two code letter will be added to the source field if that two letters don't appear already.
- Ids: Identification numbers have been added so the names have not been used to represent the relations and a non string efficient index can be used. Each node and edge will have a unique identifier and the source and target columns in the edges table will be foreign keys of the nodes table.

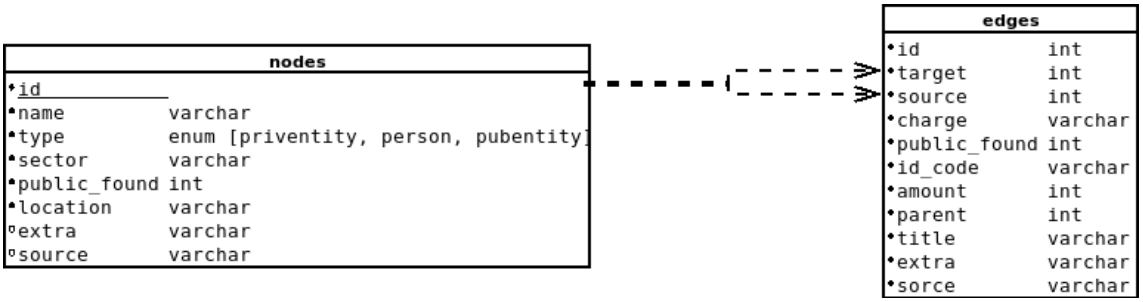


Illustration 21: Database structure

QUERY MODULE

The final element of the development project and the software tool is the one that is related to satisfy the inquiries of the end users of the OGD. Until now the data from the OGD and the relations of the DOGC have been extracted and archived in the database, but this data has to be re-fetched , organized in a graph like logic structure, queried upon and shown to the user in a better way and maybe more important in a interactive way via any internet browser.

The structure of this last part is separated in two parts :

- Backend: The queries of the user have to be received and the responses to this queries have to be generated and sent. This will be done with the help of the Graph_Tool [17] package to manage great dimension networks and a API-like resource named Flask[18] that will allow to receive requests and send responses to the user browser.
- Frontend: Concerned with the presentation of the results generated for the queries sent to the backend in a graphical and interactive way. This is done with simple HTTP requests and the D3 [19] JavaScript library to communicate information graphically.

The design of this last part of the tool has been designed in a distributed pattern after having the following ideas into account:

- The distributed design allows to use technologies developed in different languages and platforms allowing to get the best of each one. In this case the Javascript language and the D3 for visualization have been gaining the reference in the interface with the user and the visualization of information. In the other hand the backend development in python has been gaining adepts since more and more new packages like Graph_Tool are implemented with the computation weight placed in a C++ implementation back the curtains that allows for a heavier usage, although it’s interpreted nature wouldn’t make python for a great option for heavy used servers.

[17] https://graph-tool.skewed.de/
[18] http://flask.pocoo.org/
[19] https://d3js.org/

- The distributed design allows to change the local only usage (although the interaction with the tool is done via browser, it's not accessible from internet) for a one done via Internet with just a few infrastructural changes.
- Javascript, as python some years ago, has been growing immensely, but even more important it can be executed natively without the need of special set-up like the installation of packages in any browser. This wide implantation has been further reinforced by the Javascript native applications, JS written apps that run on a disguised Chrome browser that looks like a native Windows, Mac or Linux application, one example of this trend are the Electron apps. The usage of this technologies would mean a great portability for the application. This has been a strong point for the Javascript frontend.
- A huge positive aspect of this technologies is that where mostly already known to the developer of this project.

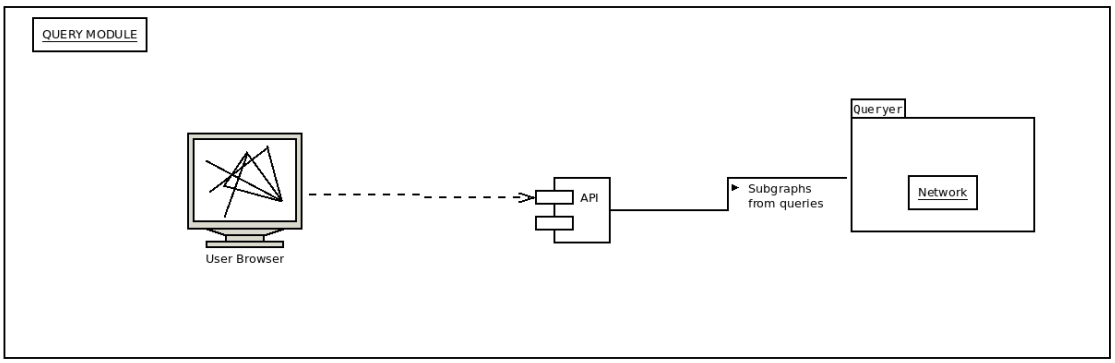


Illustration 22: Structure of the Query Module

In the image , the backend is represented by the API and the Queryer, and the frontend would be placed inside the User Browser

Backend

The backend has the format of a monolithic software program written in python that does the function of re-fetching all the nodes and edges and organizing them in the Graph_Tool Graph structure and the PropertyMaps that will hold the values of the attributes.

Network Querying

The network querying module is in charge of loaging the graph from the database and depending on the queries done to the API, generate the appropriate subgraph.

First, the module communicates with the database to obtain all the nodes and edges and then stores them in the logical structure of the network using the Graph_Tool. An important design decision of this library is that the attributes of the nodes and edges are not stored inside these, but in separated Map structures called PropertyMap.
This PropertyMaps are stored as attributes in the graph, but are being kept independent of the node and edge structures, that aren't much more than sole integers. When the nodes are imputed in one of this maps, the value of the attribute for that attribute and node or edge is outputted.

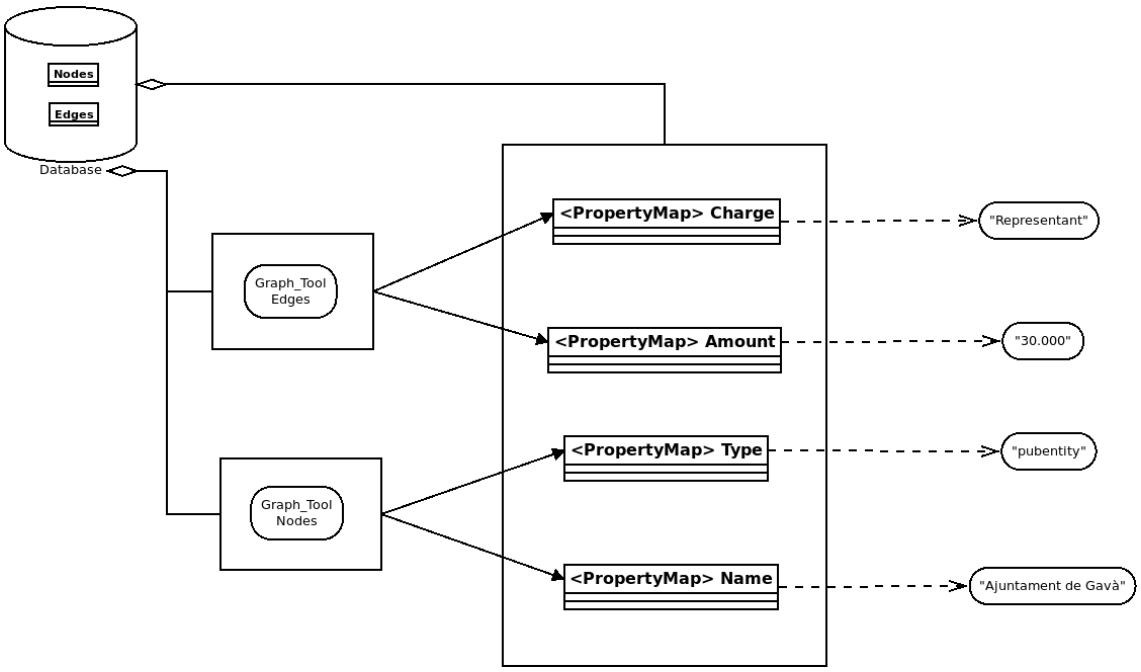


Illustration 23: Working principle of the PropertyMap

When a edge/node is given to a ProperyMap of an attribute, the value of this attribute for this edge/node is outputted.

This tool provides a data structure to represent networks that uses a Numpy array-like interaction [18] with the data in them. It's usage is different from what could be expected in a resource for python all to increase efficiency.

After the graph is loaded the backend awaits for the Flask API to be called and execute one of this querying methods:

BfsForName

Arguments: Name , N
This function will generate a new graph with the node corresponding to the Name as the root and all the nodes and relations that can be found in a path of lenght N, allowing the user to visualize all the relations and entities arrount the entity desired by the user.
In the backgroung, this module uses a modified Breath First Search to identify all the nodes in a less than N distance so that they can be displayed

DogcDay

Arguments: Date

This functionality will search through all the edges of the graph and adding the edges in a JSON that satisfy that represent a DOGC article (in the "source" PropertyMap the string "DOGC" is present) and that the article is from the inputted Date (the "extra" PropertyMap contains the Date). When a edge that fulfills these conditions, is added to the JSON and the nodes too if they weren't already on it. At the end, the resultant JSON is returned. The resultant graph will be constituted of multiple connex components (all the nodes inside this components can be connected by paths, but not with other connex components) representing the entities that appear in the individual articles.

graphByEdgeAttr

Arguments: Attribute Name, Logical Operator, Value

As the last functionality, the graphByEdgeAttr runs over the edges of the graph selecting the ones that satisfy the logical expression generated by composing the Attribute Name + Logical Operator + Value. All the edges that satisfy this expression will be added to the graph together with the attached nodes.

API

As said, the API has been implemented in Flask, a package allows python to become a application with methods that are published and called from 0.0.0.0 in a asynchronous way.
Via Flask the tool will publish all the querying methods that will allow the user to interact with the backend.

All Calls work in the same way, the flask methods are called from the Javascript elements in the various tool pages. This calls will be done against the Flask endpoints with the required elements to generate the queries, that will be imputed by the user through HTML forms. When the endpoints are called asynchronously with the parameters, the functions that do the computation over the network are called.
This are the calls that can be done to the Flask API:

- /bfs: corresponds to the bfsForName functionality
    Arguments: Name , N
- /daydogc: corresponds to the dogcDay functionalities
    Arguments: Date
- /edgeAttr: corresponds to the graphByEdgeAttr functionality.
    Arguments: Attribute Name, Logical Operator, Value

Other endpoints can be found in the API but are related to plain HTML page serving.

Frontend

After all the data scraping, processing, storing and network walking , the final problem is how to properly generate a graphical representation of a network that is interactive and easy to read and interpret independently (or not) of the functionality chosen by the user. As previously said, This has been done using the D3 [20]visualization tool, that allow to generate data-driven visualizations and DOM-modifications based on it.

The frontend of the application is what the user will be able to see, and by thus , would need to be the part in which more efforts would be invested. This have not been the case for the following reasons:
- The developer knowledge in User Experience or graphical design was not enough
- The amount of time required to design a proper user experience and interaction would had taken much more that the one reasonable for a proof of concept.

Visualization

All the displays in the functionality work the same way. After he user has filled the appropriate forms and sent them to the Flask API via a “Submit” button, will be generated and given to the frontend via JSON. This JSON is parsed and used to generate the network.
The generation and representation is completely made with the help of the D3 interface.
There are three elements in the page that have to be explained:

Network display: This area is where the network is represented, and will be bigger than the actual screen of the browser and the user will be able to scroll around. The option of setting a bigger space to represent the network than the browser has been chosen to allow a more sparse representation of the graph, and improving the interaction by doing so.
Node information area: When the network is displayed the user can click on the nodes to view the information in the bottom right side of the screen (rectangle in blue in the

[20] https://d3js.org/

next Illustration). This space will display name , sector, type, information and the source of the information in a two letter code.

Relation area: all the relations that involve the previously selected node that are present on the visualization are explained on the bottom of the page (the user will need to scroll down), together with all the entities that is related to.

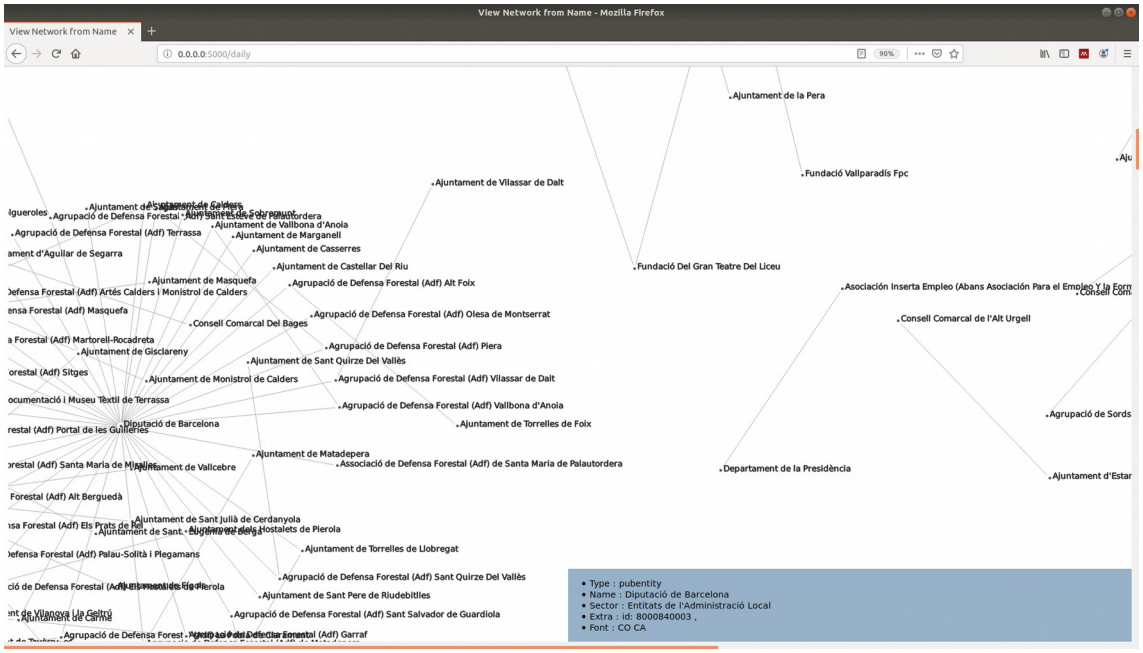


Illustration 24: Image of the browser displaying one DOGC journal

Illustration 25: Information with the relations of a certain node

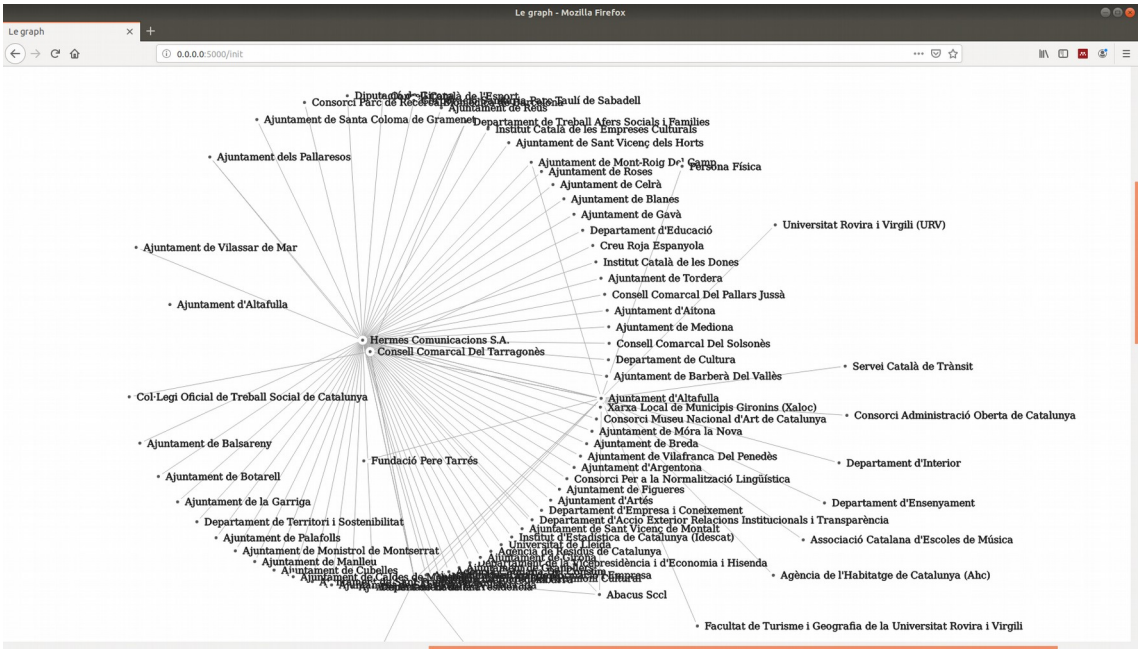


Illustration 26: Image of the browser displaying the relations with "Ajuntament de Torredembarra"

Traceability

The user must be allowed to follow the data in a edge or a node to the source of the information, where more data can be given, this is mandatory, as this tool was not conceived as a substitution of the OGD portal and the DOGC journal, but as a interface between these two and the user.

When the user is interested in any edge or node has to check the Source tag in the HTML page. In the case of the nodes, this one needs to be selected and the “Font” tag in the blue area in the bottom right corner has to be checked.
This two letters correspond to one of the four databases of the Catalan OGD portal :

CO: Public Contracting [21]
CA: Cooperation Agreements of Collaboration and Cooperation [22]
OG: Organisms of the Generalitat [23]
IN: Interest Groups of the Generalitat[24]

The name of the entity will have to be copied and searched by in the table of the Catalan OGD that is pointed by the two letter code.
In the case that the node is a person the name will be a number that does not identify the individual. In this case , the name of the entity that is related to will be copied to be searched in the appropriate database.
The usage of this organization name could lead to a miss-identification of the person.
This process is the same for relations that have the two letter code in the “Font” label but any of the two names of the nodes at the end of the relation can be used.

[21]https://analisi.transparenciacatalunya.cat/Sector-P-blic/Contractaci-de-Catalunya/hb6v-jcbf
[22]https://analisi.transparenciacatalunya.cat/Sector-P-blic/Registre-de-Convenis-de-Col-laboraci-i-Cooperaci-/exh2-diuf
[23]ttps://analisi.transparenciacatalunya.cat/Sector-P-blic/Organigrama-de-la-Generalitat-de-Catalunya/8s6p-h233h
[24]https://analisi.transparenciacatalunya.cat/Legislaci-just-cia/Registre-de-grups-d-inter-s-6
de-Catalunya/gwpn-de62

In the case that the relation contains “DOGC” in the “Font” label, the title of the article, that can be found in the “Title” label of the relation have to be searched by in the DOGC portal [25].

[25] https://dogc.gencat.cat/ca/

EXAMPLE OF USAGE: DayDogc

In the case of the tool DayDogc, the user tries to retrieve all the public, private organizations and persons that appear in the DOGC journal of a certain day.
This process can be extrapolated to all the tools developed in this work, with the minor differences of the strings that have to be inserted in the HTML forms.

Pre-requisites

- Have previously extracted the entities and relations of the given day using the Extraction Module for the DOGC journal.
At the end of the development the way to retrieve the DOGC is giving the id to the module (that has to be searched in the DOGC portal).

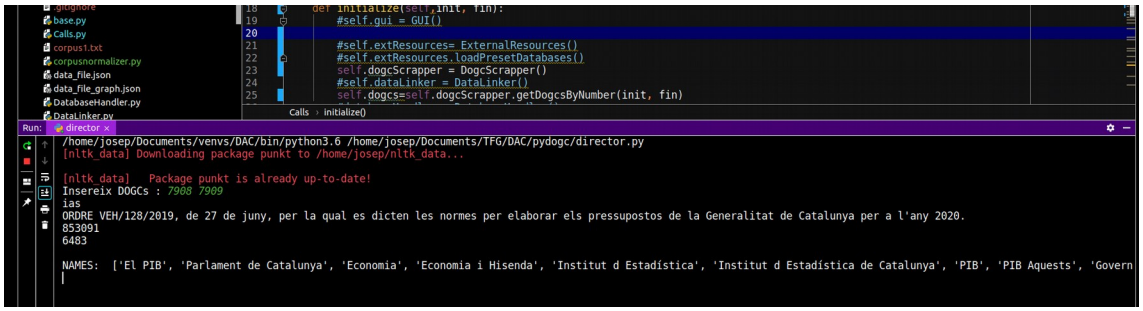


Illustration 27: Example of execution of the DOGC Extraction Module

Step 1 Execute the Query Module

The Query Module has to be executed via command line.

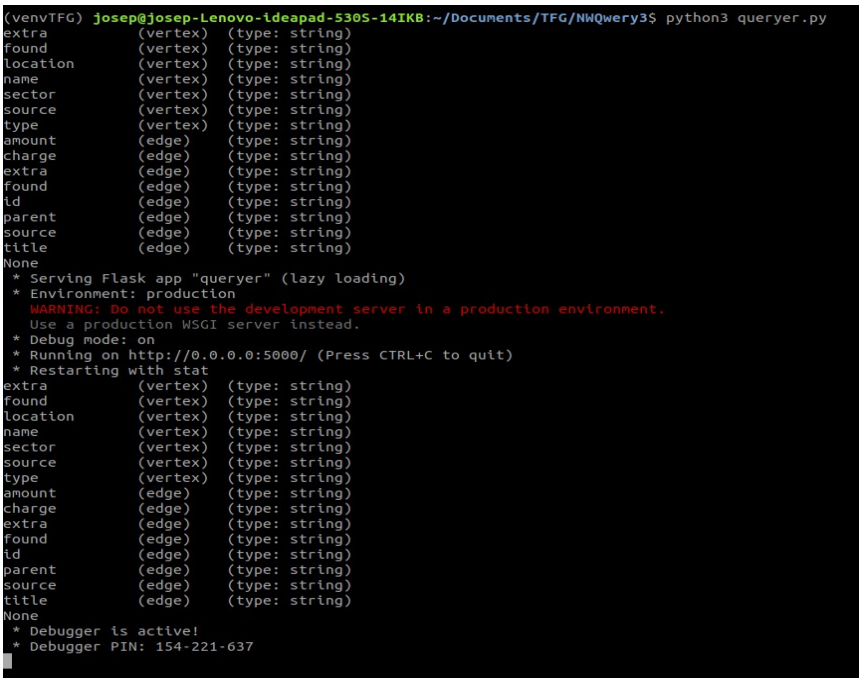


Illustration 28: Execution of the Query Module

After the execution, the flask API is published and requests can be sent.

Step 2: Using the Frontend

In the URL "http://0.0.0.0:5000/daily" the user will be able to interact with the system. In this case , the DOGC from the day 30th of May of 2019 will be requested. As said in the Query Module chapter, in this case this is done by inserting the string "30/05/2019" in the form displayed.

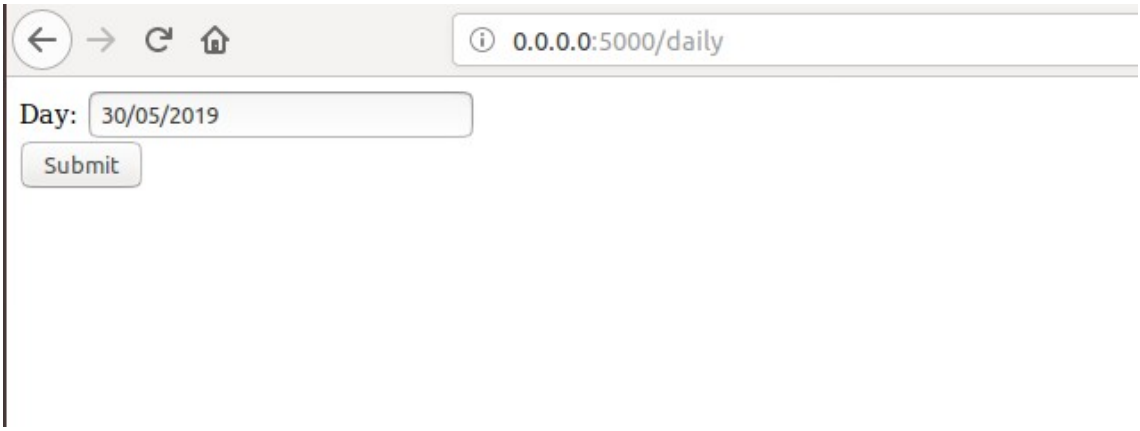


Illustration 29: Example of interaction with the DayDogc tool

After this interaction and some seconds, the user will be able to interact with the recently displayed network as shown in the next Illustrations.

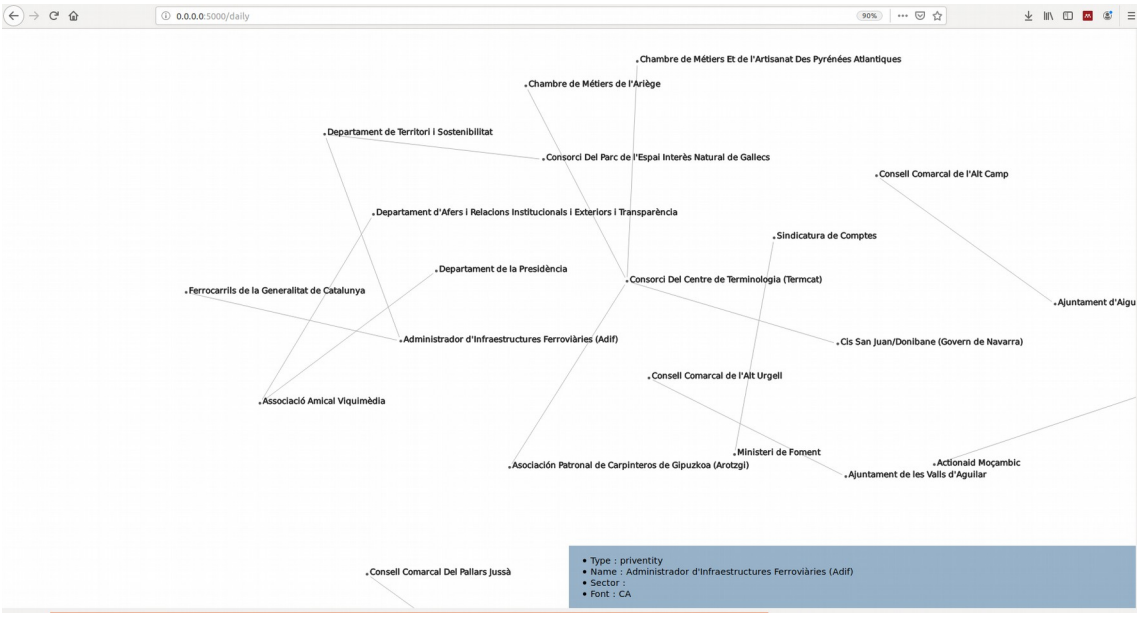


Illustration 30: Example of the network displayed by the DayDogc tool

Illustration 31: Example of the explanation of the relations between the entities

LEGAL POINTS

As said in the Objectives section, the idea behind this new tool is to become a new layer that could be placed on top of the sources of information in the Catalan portal of OGD and the DOGC, but at this moment it has been developed as a isolated application that doesn't have any political or administrative relation with the Catalan administrations, that is saying that has been developed by a external actor (the author). Having this clear, anyone that uses the open data published on the portal will be tied to some regulations that specify how to re-use this sources. Another important law that affects some parts and decisions in this work is the GDPR, the European law that protects the personal data of citizens of the Union.

To respect this laws, some modifications in the initial idea of the tool have been done before the development, provably in a more restricting way that would be strictly needed, due to the little inclination of the author to test the limits and consequences of this laws.

Nevertheless, all legal sources consulted and the laws itself do remark the necessity of preponderate in the application of these. The intentions of the re-usage have great importance in whether it conflicts with the legislation, and the author has to remember to any reader that the sole intention of this work is to create a proof of concept as proposal of a tool so that it could be adopted by the public institutions to improve their transparency, without having any monetary or professional intentions.

Spanish OGD Laws

The Catalan portal contains guidelines, inherited from the Spanish law, about how the external developers and citizens can re-use the data from portal and the information that can be found on the DOGC, specifically in the OGD Catalan portal [26] the administration explains to which laws and documents the open data is tied to.

[26] http://governobert.gencat.cat/ca/dades_obertes/informacio-per-a-desenvolupadors/

In general, it can be said that the information published in the tables used is not considered to contain works nor other content made by third parts, so it is allowed the usage without more restrictions in terms of licenses, of the ones imposed in the article 8 of the Lei 37/2007 and modified by the Law 18/2015.

But one part of this laws that does greatly affect this work are some of the conditions of re-use, concretely:

" - When the information contains information of personal nature, the concrete purpose or purposes for which are possible re-usage in the future"
" - When the information, even dough is given in a dissociated manner, contains enough elements that allow for the identification of the interested ones, the prohibition to revert the process of dissociation with the addition of new data obtained from other sources."

The first one is referred to personal information and it's general usage, and will be addressed in the GDPR section. The second announces that when information about natural persons is used and comes from the OGD tables, it is forbidden to aggregate new information about this persons that come from other sources.

This statement applies directly to the intentions and the initial ambitions of this work. When the structure and relations are extracted from the tables from the portal, names and charges inside these are used, and when the new relations that come from the DOGC articles are added , often involve this persons too. Although both sources are public, open and meant to be used, they constitute different sources and if data about persons would be aggregated this would conflict with the laws Lei 37/2007 and the modifications in the Lei 18/2015 from the Spanish country. Data from different sources about other entities that do not refer to person can be aggregated.

The conflicts with this Laws finally have been solved by the solution found for not contradicting the laws of the GDPR, although not very complicated formulas could have been found to not violate this law, in the following section will be explained that the European law is more strict, and with the solution found for it, this problem will be solved too.

GDPR

The General Data Protection Regulation [19], came into effect the April of 2016 and into force the May of 2018 is the law that protects the data of natural persons in regard of it's processing. This law does affect how the application has been developed and definitely its capabilities, since it uses full names of persons as identification in the nodes of the network and it gathers information about them in the DOGC and the tables in the OGD.

Although the GDPR clearly states in the article 18 :
"This Regulation does not apply to the processing of personal data by a natural person in the course of a purely personal or household activity and thus with no connection to a professional or commercial activity "

is applicable and would left this work out of the scope of the legislation, there are other articles that would represent the violation of this law, since this activities refer to manual and simple processing eg. a personal telephone agenda or a photograph album, this is not the case of the author.

First of all, the author, by the definition is a "controller" and a "processor", legal entities introduced by this text, the first being the entity that determines the propose and means of the processing of this personal data, and the second being the one that does the processing. The author and only developer, being both of them makes it's work a subject of application of this law.

Having this clear, although the data used by the tool is public and the information would be generated (and imported) by the end user if this tool was to be published, the author, during the phase of the development could become the sole actor of an infringement of the GDPR.

To avoid the illegality some measures have taken, that in addition will solve the other legal issues explained in the previous section, “Spanish OGD Laws ”.

1- In the extraction of entities from the Catalan OGD, when the CSV lines are run over to extract edges and nodes, the names won’t be extracted. Instead of using the name of the entity a number is generated so any identification data is never stored in the database and by thus vanishing any doubt with the compliance with the European law.

The final visualization will only show that in the tables of the OGD there is a person related to the parent entity and the user was to be interested in this node, it could follow the method of traceability explained in the documentation of the software.

This design option will influence the corpus that will be generated and used in the extraction of relations from DOGC, that is composed by all the names of the entities that appear in the database. Because of this modification, no node representing a person is going to be chosen as candidate in the linking of the new relations from the journal.

DISCUSSION AND CONCLUSIONS

One of the fist conclusions that had been written in this list for a long time is that during the development of this project has been the frustration that has arisen during the development because although having the relative perception of working with new tools during the interaction with the Catalan OGD sometimes the more basic protocols or appropriate ways to present data have been poorly implemented. Although some great finds (like the usage of APIs and great metadata in some databases) have been happening, the most resources don't offer great functionality and in some cases, some clear mistakes have been found.

It is palpable that great improvements have been achieved but in some cases, the implementation of changes have been done with a notable lack effort to make this ones usable and practical. Lots of examples can be found, in [20] can be seen some incoherence in the format of the commercial formulas of the private entities. When using names as unique identifiers (most of the times the only way to use a unique identifier) this leads to a generation of multiple entities that in reality represent the same one. In the same fashion, as said, multiple file formats to download the data can be found that could greatly improve the re-usability of the data published in the Catalan portal, but in some cases, this formats are implemented deficiency (as explained with the case of the RDF format) and the extra possibilities are diminished.

In addition to the mistakes and other obstacles in the implementation of the OGD in Catalunya, and elsewhere ,there is an other problem in the usability of this initiatives. This is the amount of data that the user is overflowed with, not only during the access to the raw data (a problem that this work has tried to address) but the problem appears too when the user has to manually process the results of the findings, is the capability to process the amount of data that is outputted as a result by the system. The great numbers of tables that populate the OGD portals and the hundreds of pages published each day in the DOGC interfere with the capacity of the standard citizen to access this information and extract conclusions, and by thus diminishes the practical transparency that is achieved.

As can be seen in the Illustration, some results that are generated with reasonable queries can generate can become completely unpractical for any use. This is due the high degree of some elements. This specific example was generated with the functionality bfsForName for the name “Universitat Pompeu Fabra” and depth 2, parameters that by any means would indicate the output of huge network.

By no means this is a specific problem of this sector, the problem of managing great amounts of data is present in almost all of the information systems.

Furthermore, as will be explained in the following chapter, if this tool where to be enriched with the relations of a realistic number of DOGC journals (for example, the ones from the present legislature), the increase in the density in some visualizations would make the tool unpractical. This can be considered a critical situation if a huge effort is not done in the interface with the tool.

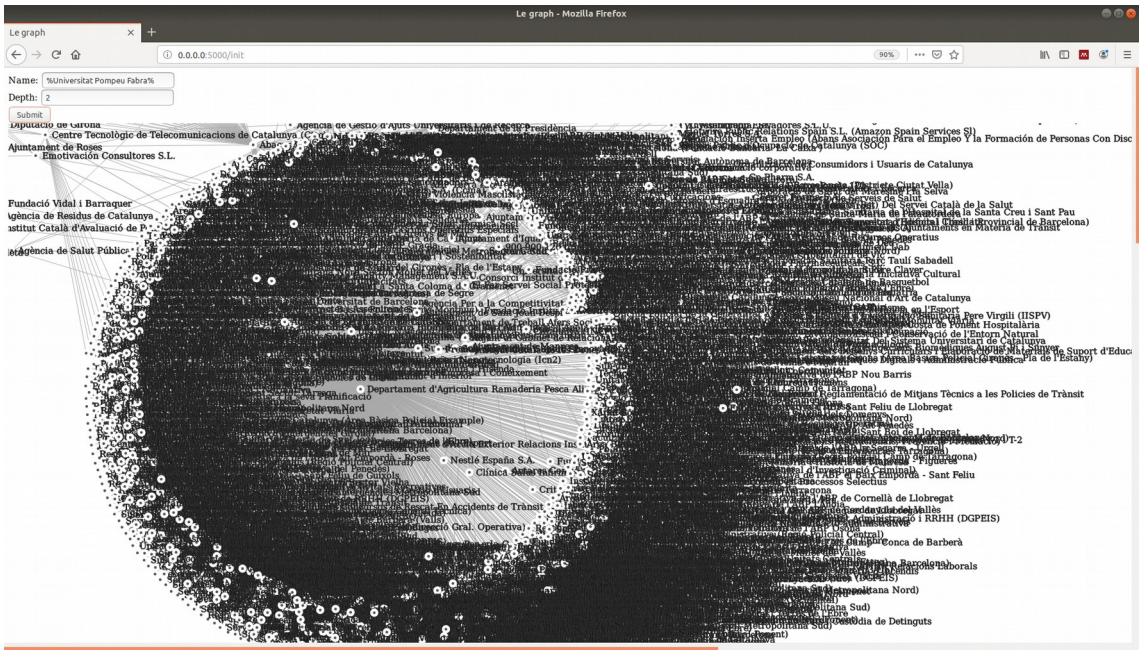


Illustration 32: Example of unpractical result

FUTURE WORK

One of the fist conclusions that had been written in this list for a long time is that during the development of this project has been the frustration that has arisen during the development because although having the relative perception of working with new tools during the interaction with the Catalan OGD sometimes the more basic protocols or appropriate ways to present data have been poorly implemented. Although some great finds (like the usage of APIs and great metadata in some databases) have been happening, the most resources don't offer great functionality and in some cases, some clear mistakes have been found.

It is palpable that great improvements have been achieved but in some cases, the implementation of changes have been done with a notable lack effort to make this ones usable and practical. Lots of examples can be found, in [20] can be seen some incoherence in the format of the commercial formulas of the private entities. When using names as unique identifiers (most of the times the only way to use a unique identifier) this leads to a generation of multiple entities that in reality represent the same one. In the same fashion, as said, multiple file formats to download the data can be found that could greatly improve the re-usability of the data published in the Catalan portal, but in some cases, this formats are implemented deficiency (as explained with the case of the RDF format) and the extra possibilities are diminished.

In addition to the mistakes and other obstacles in the implementation of the OGD in Catalunya, and elsewhere ,there is an other problem in the usability of this initiatives. This is the amount of data that the user is overflowed with, not only during the access to the raw data (a problem that this work has tried to address) but the problem appears too when the user has to manually process the results of the findings, is the capability to process the amount of data that is outputted as a result by the system. The great numbers of tables that populate the OGD portals and the hundreds of pages published each day in the DOGC interfere with the capacity of the standard citizen to access this information and extract conclusions, and by thus diminishes the practical transparency that is achieved.

As can be seen in the Illustration, some results that are generated by reasonable queries can generate networks completely unpractical for any use. This is due the high degree of some elements. This specific example was generated with the functionality bfsForName for the name “Universitat Pompeu Fabra” and depth 2, parameters that by any means would indicate the output of huge network.

By no means this is a specific problem of this sector, the problem of managing great amounts of data is present in almost all of the information systems.

Furthermore, as will be explained in the following chapter, if this tool where to be enriched with the relations of a realistic number of DOGC journals (for example, the ones from the present legislature), the increase in the density in some visualizations would make the tool unpractical. This can be considered a critical situation if a huge effort is not done in the interface with the tool.

Bibliography

1: A.-L. Barabási, R. Albert, H. Jeong, Scale-free characteristics of random networks: the topology of the world wide web , 2000
2: S. H. Yook, H. Jeong, A.-L. Barabási, Modeling the internet's large-scale topology , 2002
3: Stephen P. Borgatti, Ajay Mehra, Daniel J. Brass, Giuseppe Labianca, Network Analysis in the Social Sciences, 2009
4: H. Gallego, D. Laniado, A. Kaltenbrunner, V. Gómez, P. Aragón, Statistical analysis of the social network and discussion threads in slashdot, 2008
5: Gallego H, Laniado D, Kaltenbrunner A, Gómez V, Aragón P, Lost in re-election: a tale of two Spanish online campaigns, 2017 Sep 13-15
6: Generalitat de Catalunya, Sistema de Monitoratge del RIS3CAT, , http://unics.cloud/ris3/mcat/#/
7: Fatemeh Ahmadi Zeleti, Adegboyega Ojo, Edward Curry, Exploring the economic value of open government data, 2016
8: Peter Conradieac ,Sunil Choenniab, On the barriers for local government releasing open data, 2014
9: Kawaljeet Kapoor, Vishanth Weerakkody, Uthayasankar Sivarajah, Open Data Platforms and Their Usability: Proposing aFramework for Evaluating Citizen Intentions, 2015
10: , Pressupostos Municipals, , http://pressupostosmunicipals.transparenciacatalunya.cat
11: Kalpana Shankar, Scientific data archiving: the state of the art in information, data, and metadata management, 2003
12: W3 Consortium, Linked Data, , https://www.w3.org/standards/semanticweb/data
13: , RDF Primer, 2004, https://www.w3.org/TR/2004/REC-rdf-primer-20040210/
14: Generalitat de Catalunya, Contractació de Catalunya,
15: Marcio Victorino, Edgard Costa Oliveira, Maristela Terto de Holanda, George Ghinea Brunel, Edison Ishikawa Brasília, Sammohan Chhetri Brunel , Proposal of a Brazilian Database Government Open Linked Data: DBgoldbr, 2017
16: Eleni Galiotou, Pavlina Fragkou, Applying Linked Data Technologies to Greek Open Government Data: A Case Study, 2013
17: Entitat Autònoma del Diari Oficial i de Publicacions, Criteris d'elaboracióde documents per al DOGC, , https://dogc.gencat.cat/web/.content/continguts/serveis/dogc_en_linia/documents/arxius/criteris_documents_dogc_v2.pdf
18: The SciPy community, numpy.array, , https://docs.scipy.org/doc/numpy/reference/generated/numpy.array.html
19: , REGULATION (EU) 2016/679 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL, 2016
20: Generalitat de Catalunya, Contractació de Catalunya,