GeoGraph :

Normalized Categorization of Multisource Geolocated Data with a Structured Thesaurus. Display on a map of the Neo4j DB content with QGIS

Véronique Gendner

26th Jan 2022

Projet Choucas

Marie-Dominique Van Damme, LASTIG, Univ Gustave Eiffel, IGN-ENSG
Ana-Maria Olteanu-Raimond, LASTIG, Univ Gustave Eiffel, IGN-ENSG

IGN ENSG Géomatique

MEIG

UNIVERSITÉ DE PAU ET DES PAYS DE L'ADOUR

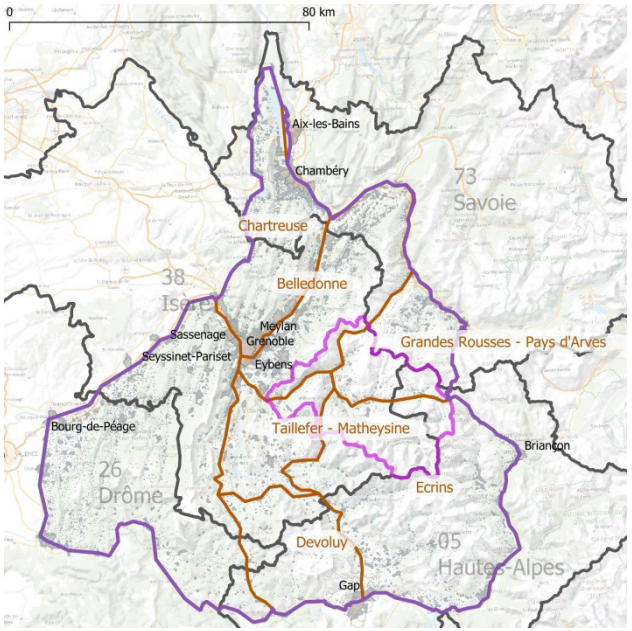
LIG

GENDARMERIE PELVOUX HAUTE MONTAGNE

AGENCE NATIONALE DE LA RECHERCHE ANR

Université Gustave Eiffel

LaSTIG



Study zone *of the Choucas project, in the French Alpes*

Agenda

- Information Design *full stack*
- The Use Case
- The Landmark Objects Ontology (LOO)
- Categorization in Label Property Graph DB
- Modeling by categorization and iteration
- Data import
- LOO instantiation with imported data
- Use of the graph structure
- Display on a map with QGIS script

Information Design

full stack

database

data analysis
/ modeling

application

website

information
architecture

personal
information system

structure

human internal

representation

human external

process

human
language

linguistics

psychoanalysis

formal
language

formal linguistics

programming languages

algorithmic

visual representation

signage

communication

UX / UI

Information Design

applications

programming
script writing

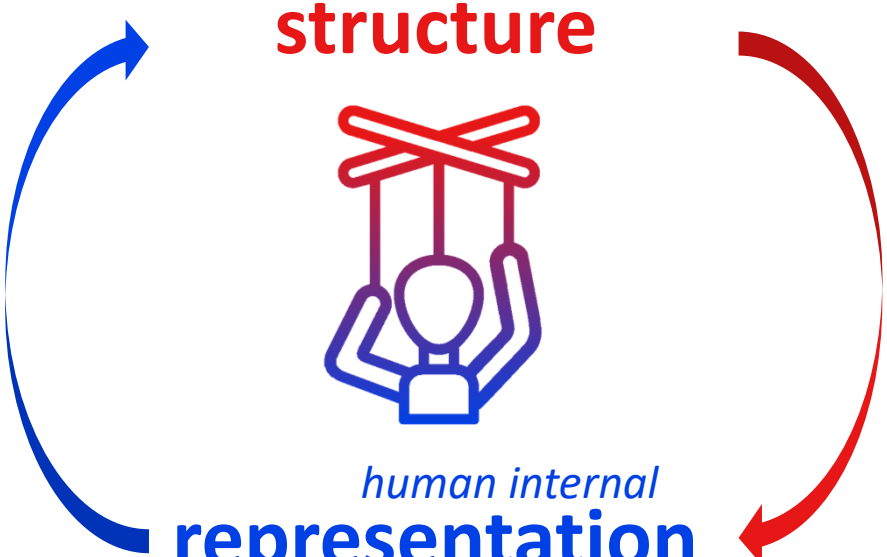
formal linguistics
programming languages
algorithmic

data analysis

web application
creation

UX, UI

flows



database
structuring

time
management

personal
information
system

flows

personal
documentation

text

document
structuring

video

new digital formats

Use case : Choucas project

- Helping mountain rescue team localise victims upon emergency calls

- Examples of localization clues

I took the **trail** in direction of Oursière **waterfall**
j'ai pris le sentier en direction de la cascade de l'Oursière

municipality
I left from[Bourg d'Oisans,] on foot, on the **route** to a **ski resort**
je suis parti de Bourg d'Oisans, à pied, sur chemin, en direction d'une station

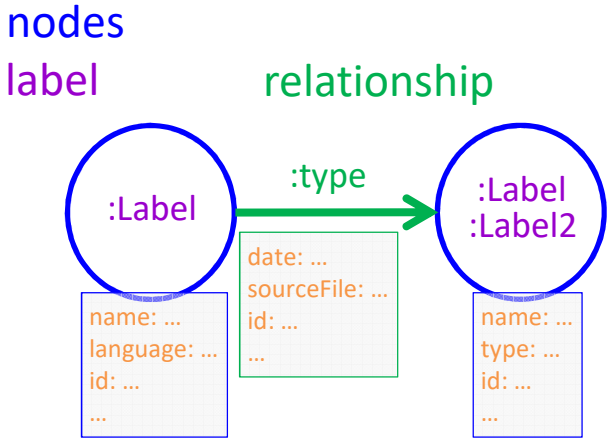
I can see part of a **water body**
je vois une partie de plan d'eau,

I'm under a 3-strand **power line**
je suis sous une ligne électrique 3 brins

-> categories imported data according to Landmark Objects Ontology (LOO)

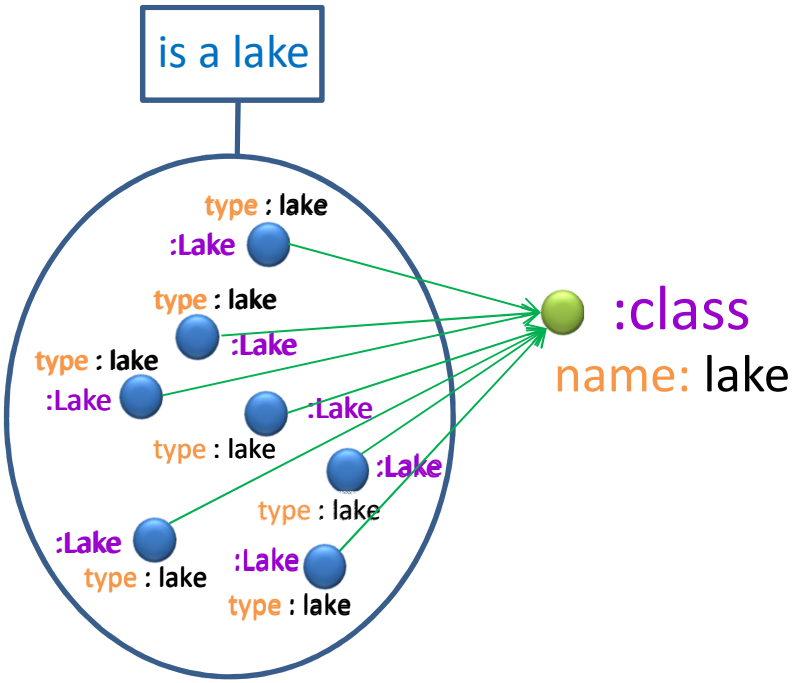
Label Property Graph data base

LPG DB elements



property

name : value



property
type: lake

label
:Lake

relationship *to a node with* Label :class

identical property values,
a set-theoretic representation by label,
links to class,
= formally equivalent ways of categorizing

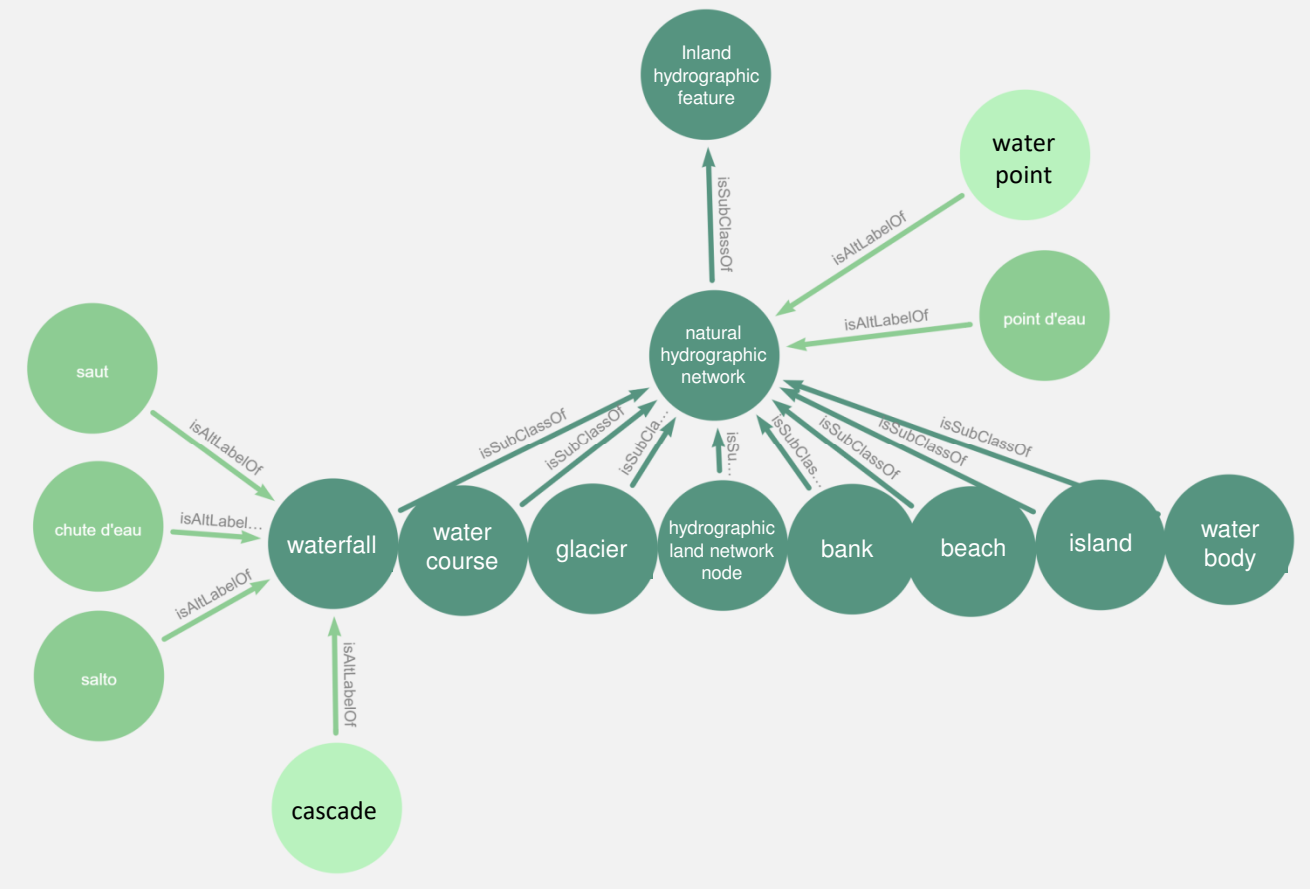
DB modeled by categorization

Different categorization needs :

- by sources (IGN BDTOPO, camp2camp, ENEDIS…)
- extraction date of a data set
- technical classification of nodes (object, geometry, Rtree element)
- geographical object types (landmark, toponym,…)
- according to the Landmark Objects Ontology (LOO)

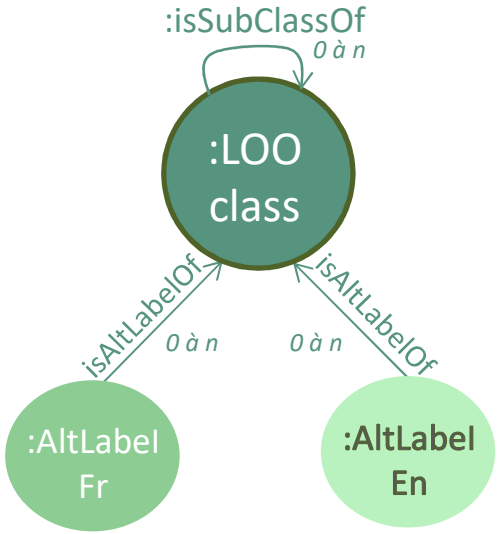
Landmark Objects Ontology (LOO)

excerpt



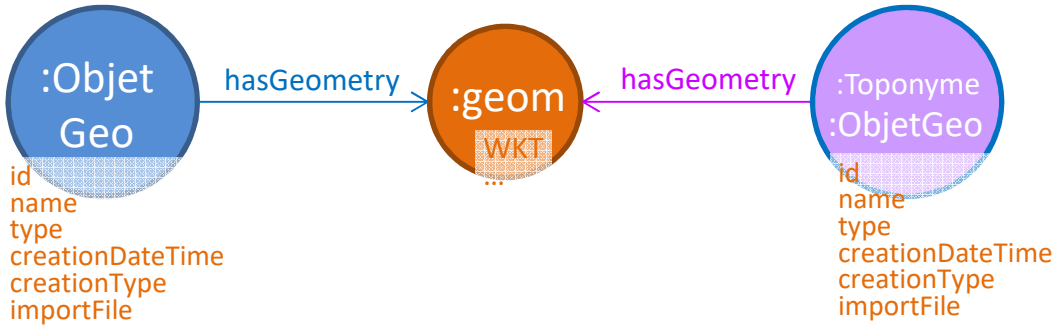
Properties of nodes with label :LOOclass

id
prefLabelFr
prefLabelEn
commentFr
commentEn
isDefinedBy
importFile
creationDateTime
creationType

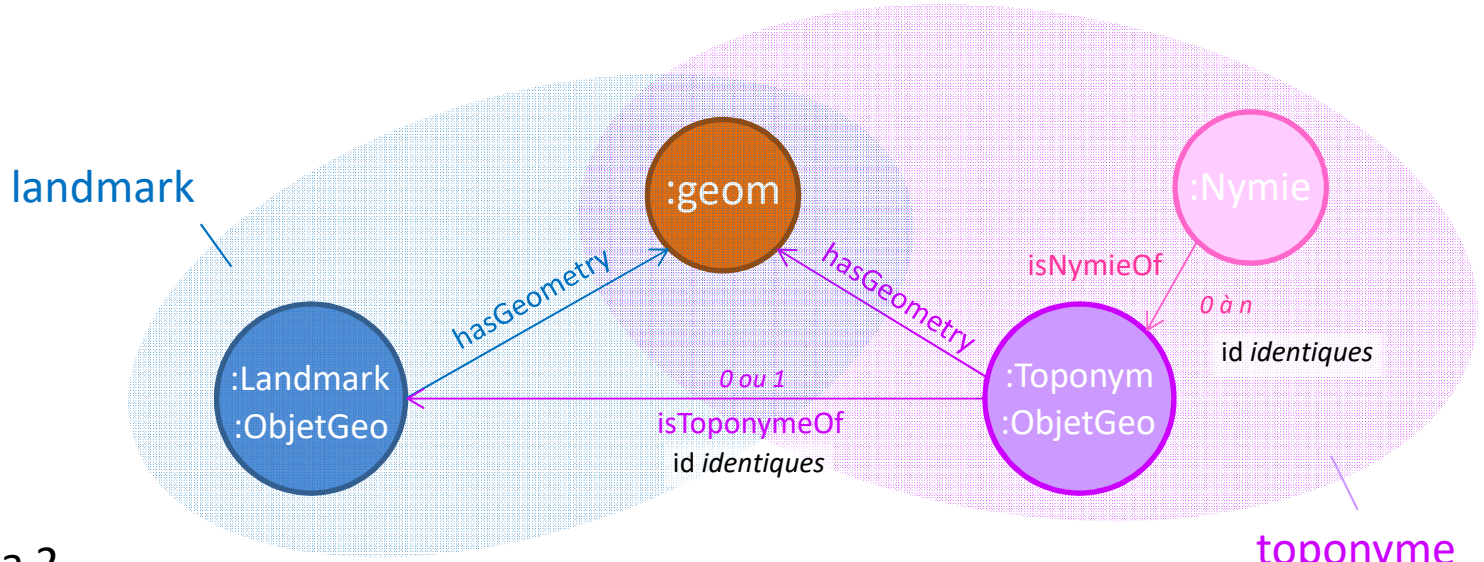


DB modeling by iteration

Schema 1.

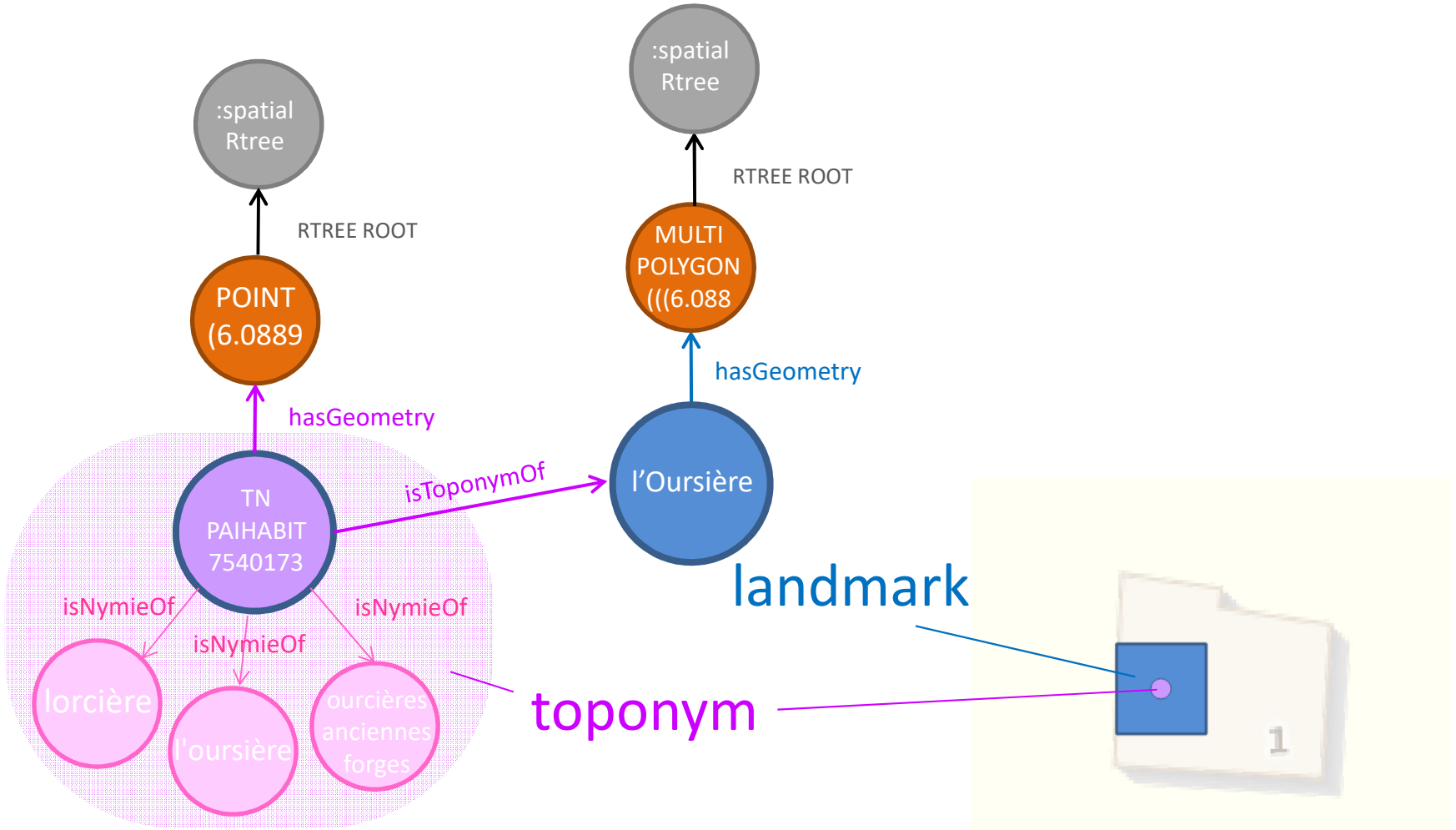


1. bis

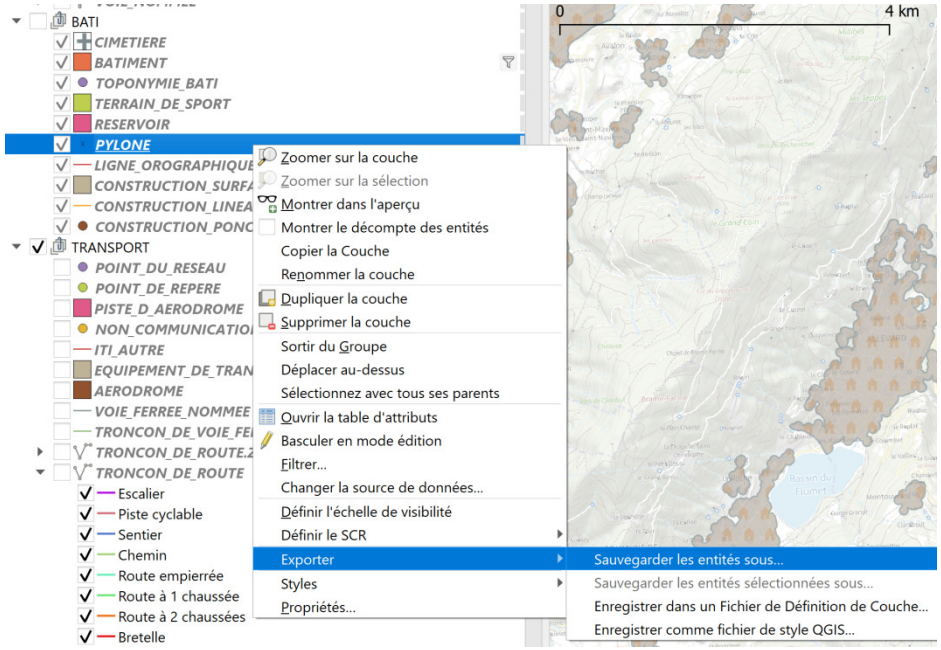


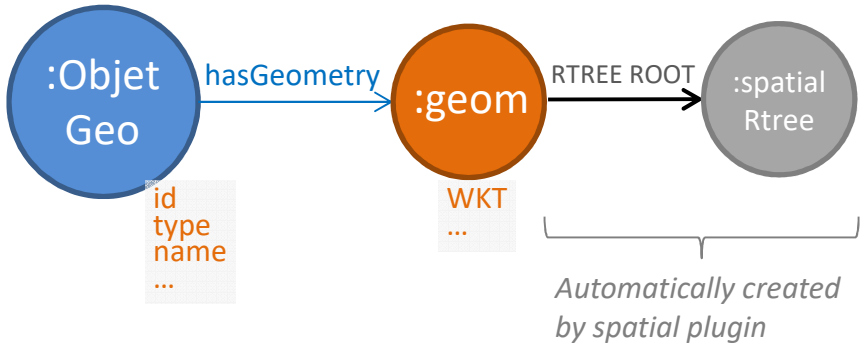
Schema 2.

Landmark and toponym instance

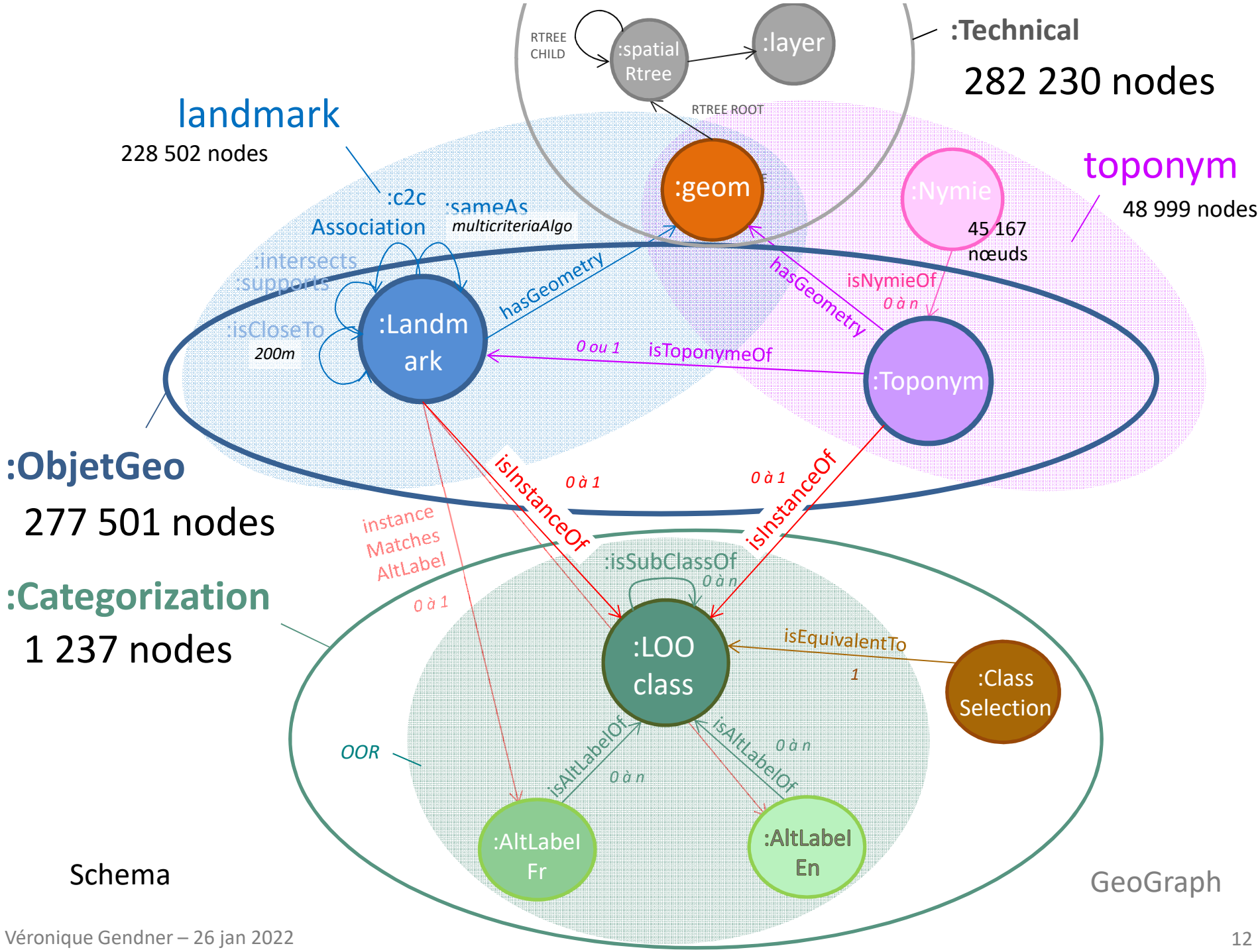


Sources data import






```
:param importFile => BDTOPO.2021.pylone.WKT.csv;
call apoc.load.csv($importFile) YIELD map
call spatial.addWKT('points',map.WKT) yield node
set node:geom:Tech,
node.creationDateTime=localdatetime({timezone:'Europe/Paris'})
, node.creationType=import
with map,node
create (:ObjetGeo:ObjetRepère:Pylone:sourceBDTOPO:v2021
{
   name: replace(map.ID,'000000',' ')
   ,creationDateTime:localdatetime({ timezone: 'Europe/Paris' })
   ,creationType:import
 ,importFile:$importFile
   ,id:map.ID
   //,etat:map.ETAT
   ,srcDateCrea:map.DATE_CREAT
   ,srcDateMAJ:map.DATE_MAJ
   ,date_conf:map.DATE_CONF
   ,srcSource:map.SOURCE
   //,id_source:map.ID_SOURCE
   ,prec_plani:map.PREC_PLANI
   ,prec_alti:map.PREC_ALTI
   ,hauteur:map.HAUTEUR
})-[:hasGeometry]->(node);
```


:Technical
282 230 nodes
RTREE CHILD
:spatial Rtree
:layer
RTREE ROOT
:geom
landmark
228 502 nodes
:c2c Association
:sameAs
multicriteriaAlgo
:intersects
:supports
:isCloseTo
200m
:Landm ark
hasGeometry
hasGeometry
toponym
48 999 nodes
:Nymie
45 167 nœuds
isNymieOf
0 à n
0 ou 1
isToponymeOf
:Toponym
:ObjetGeo
277 501 nodes
isInstanceOf
0 à 1
0 à 1
isInstanceOf
instance Matches AltLabel
0 à 1
:Categorization
1 237 nodes
:isSubClassOf
0 à n
:LOO class
isEquivalentTo
1
:Class Selection
OOR
isAltLabelOf
0 à n
0 à n
isAltLabelOf
:AltLabel Fr
:AltLabel En
Schema
GeoGraph

Labels : object type, version, source,…

Toponyme

IGN BDTOPO layers
hydro
bati
…

Landmark

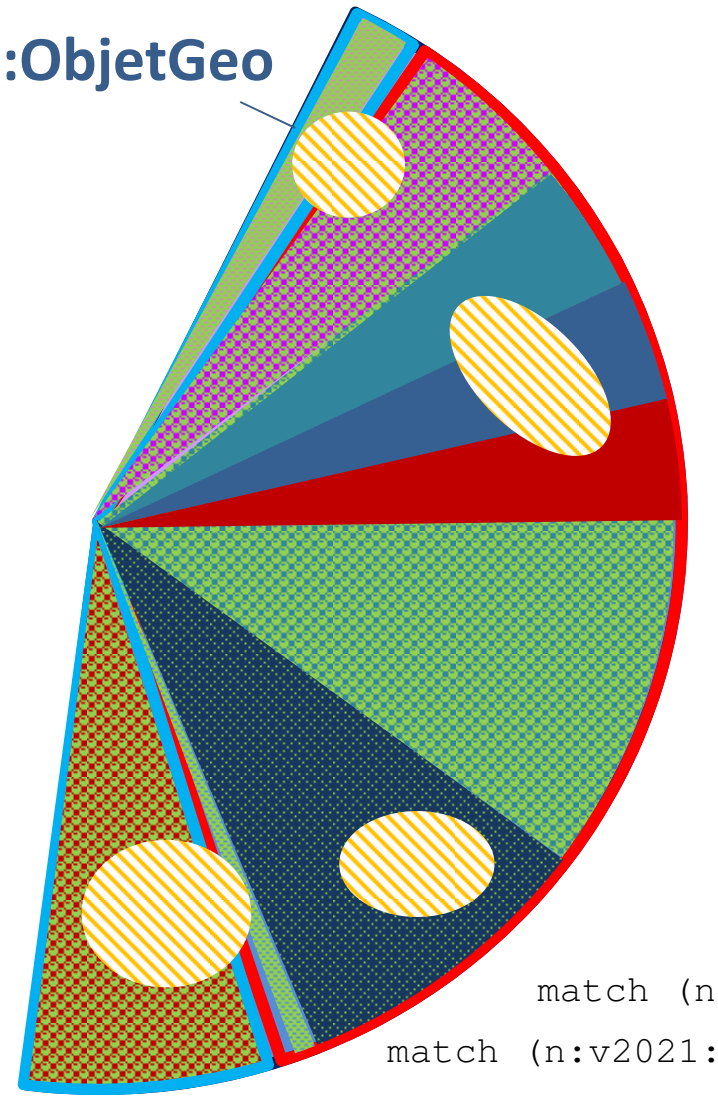
IGN BDTOPO layers

water course

building

electric line

…



Version 2018

Version 2021

sourceBDTOPO

sourceENEDIS

sourceC2c

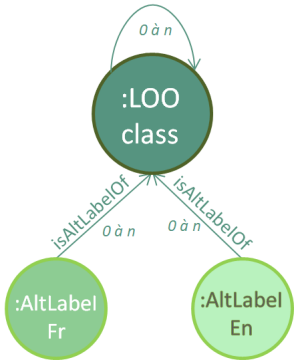
InZoneFilRouge

= restricted study area

```
match (n:v2018) detach delete n
match (n:v2021:sourceC2c:InZoneFilRouge)
```

Landmark Objects Ontology (LOO) import

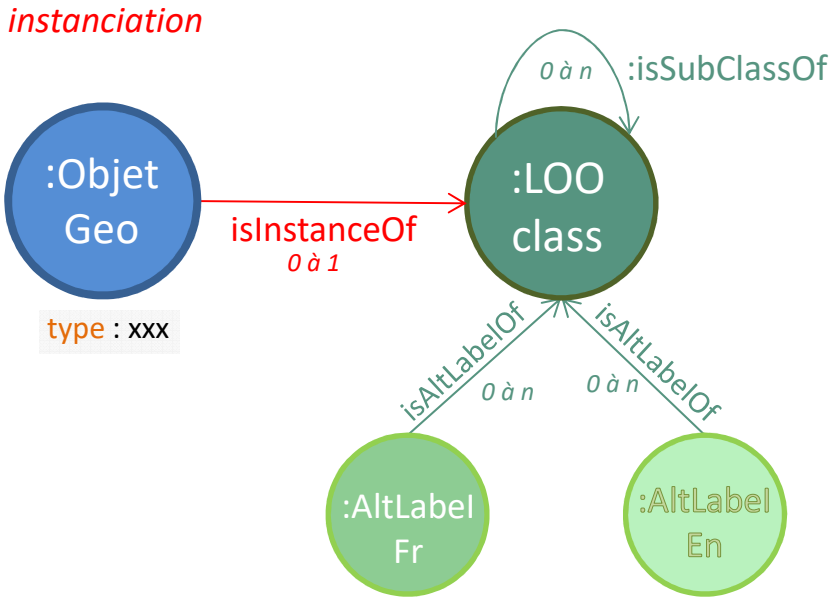
Properties of nodes with label :LOOclass



id
prefLabelFr
prefLabelEn
commentFr
commentEn
isDefinedBy
importFile
creationDateTime
creationType

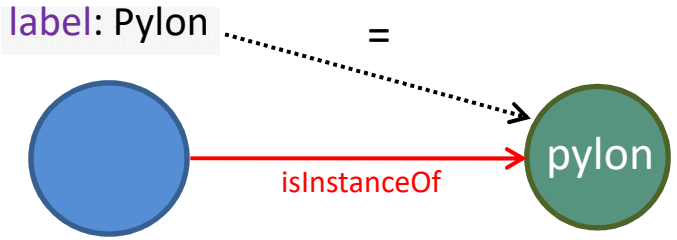
```
:param importFile => OOR.AM.210505.json;
call apoc.load.json($importFile) YIELD value as item
with item
where item.`@type`[0] = http://www.w3.org/2002/07/owl#Class
create (classOOR:ClassOOR:`Catégorisation` {
    id: item.`@id`,
    creationDateTime: localdatetime({ timezone: 'Europe/Paris' }),
    creationType: import,
    prefLabelFr : [prefLabel in item.`http://www.w3.org/2000/01/rdf-schema#prefLabel`
where prefLabel.`@language` = 'fr' | prefLabel.`@value`][0],
    prefLabelEn : [prefLabel in item.`http://www.w3.org/2000/01/rdf-schema#prefLabel`
where prefLabel.`@language` = 'en' | prefLabel.`@value`][0],
    commentFr : [comment in item.`http://www.w3.org/2000/01/rdf-schema#comment` where
comment.`@language` = 'fr' | comment.`@value`],
    commentEn : [comment in item.`http://www.w3.org/2000/01/rdf-schema#comment` where
comment.`@language` = 'en' | comment.`@value`],
    importFile: $importFile
})
WITH item.`http://www.w3.org/2000/01/rdf-schema#altLabel` as altLabels,classOOR
UNWIND altLabels  as altLabel
    MERGE (n:AltLabel:`Catégorisation` {name:altLabel.`@value`})
                    ON CREATE SET n.lang=coalesce(altLabel.`@language`,
'',n.creationDateTime=localdatetime({ timezone: 'Europe/Paris'
}),n.creationType=import,n.importFile=$importFile
    MERGE (n)-[:isAltLabelOf {importFile:$importFile,creationDateTime:localdatetime({
timezone: 'Europe/Paris' }),creationType:import}]->(classOOR);
```

Landmark Objects Ontology : modeling in DB and instantiation

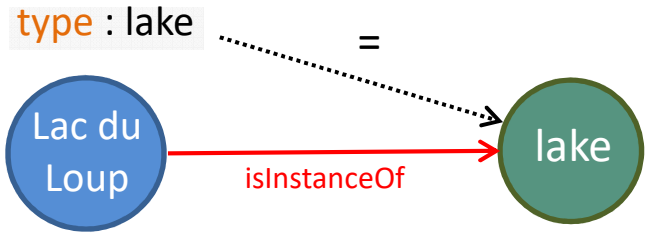


Schema

- Rule on source layer = label

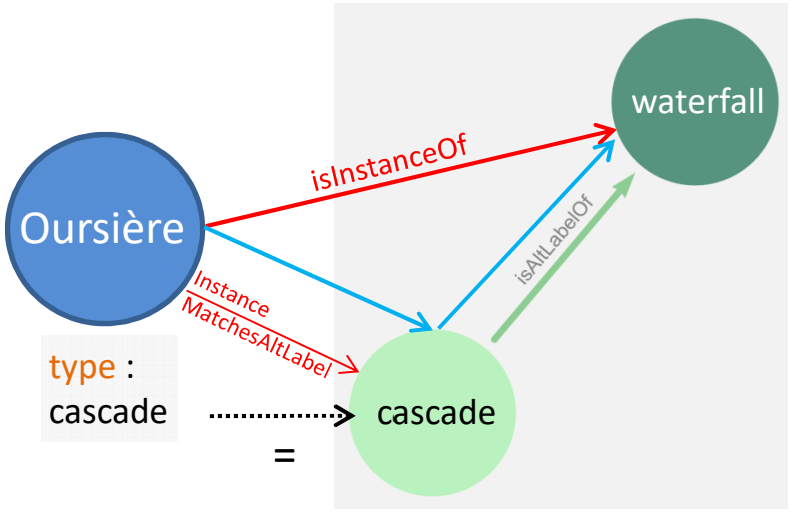


- Rule on source categorization info = type

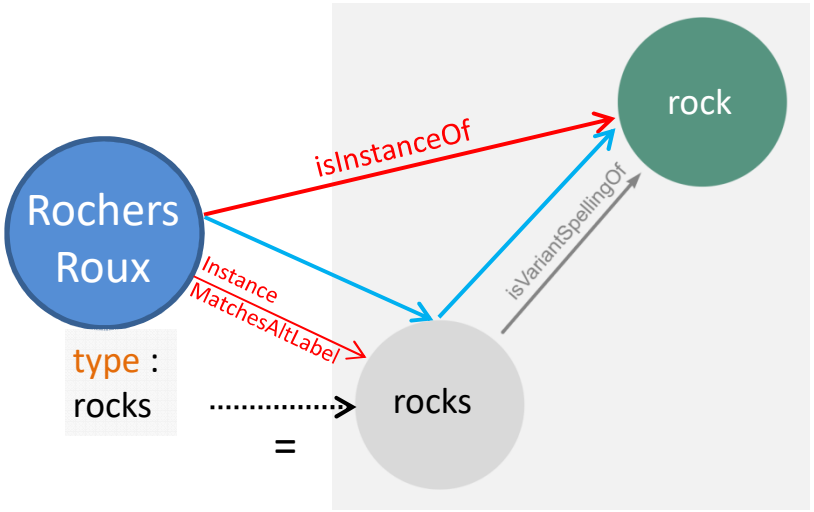


Examples

Use of relation transitivity to normalize categorization of different sources

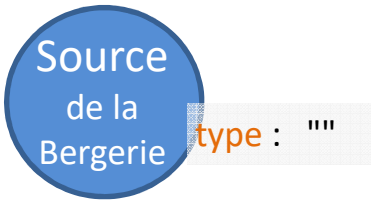


transitivity



The same principle is used for variant spellings and some plurals forms that where found in the sources

LOO subdomain extraction for more specific rules



- Use of First word in names ? -> over generation

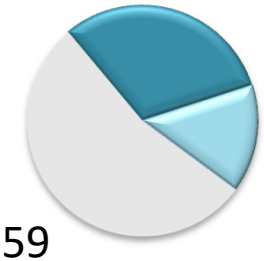
Ex « cascade de la pisse » that is a car park

- Applying rule on first word only on a subdomain


```
match (cHydro:ClassOOR)-[:isSubClassOf*]->(:ClassOOR {prefLabelFr:"hydrographie"})
with collect(distinct cHydro.prefLabelFr) as hydrographie
match (cInfra:ClassOOR)-[:isSubClassOf*]->(:ClassOOR {prefLabelFr:"infrastructure eau"})
with hydrographie,collect(distinct cInfra.prefLabelFr) as infraEau
match (o:ObjetGeo) where (o:DetailHydro or o:CoursEau or o:SurfaceHydro or o:Hydro)
and not exists((o)-[:isInstanceOf]-(:ClassOOR))
with o, toLower(split(o.name, )[0]) as firstWord where firstWord in hydrographie or
firstWord in infraEau
match (c:ClassOOR) where  apoc.text.compareCleaned(c.prefLabelFr, firstWord )
         create (o)-[:isInstanceOf {rule:nameFirstWord-
ClassOORprefLabelFr,creationDateTime:localdatetime({ timezone: 'Europe/Paris'
}),creationType:cypherCreate}]->(c) ;
```

LOO instantiation results

- **482 LOO classes**



150 directly instantiated

+73 instantiated by cumulating with children's instantiation

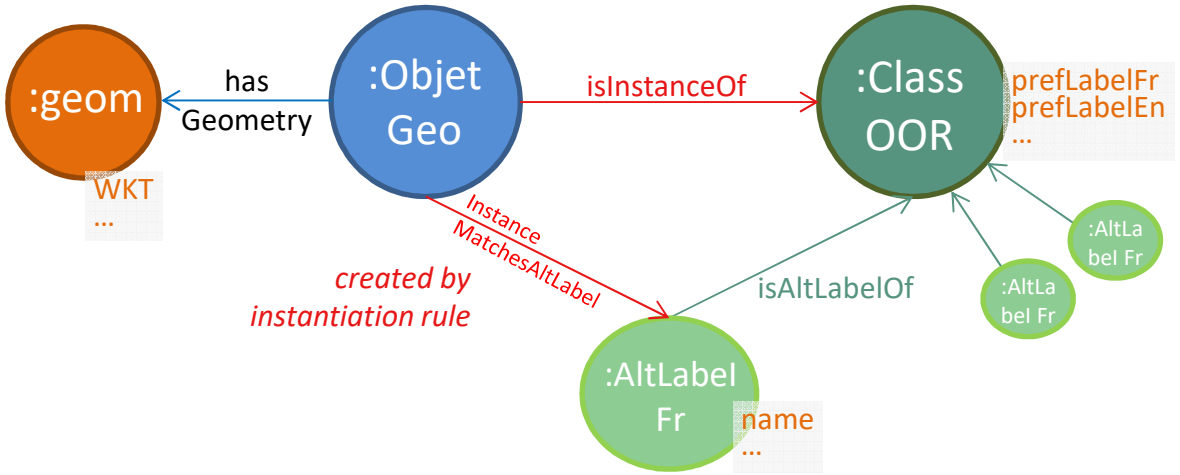
259

- **277 501** `ObjetGeo`

249 681 objects attached to a class

- Evaluation
  - What reference ?
  - Looking at the data (manually, via confusion matrix), to filter out errors and iteratively adjust rules
- Instantiation brought out inconsistencies in the LOO

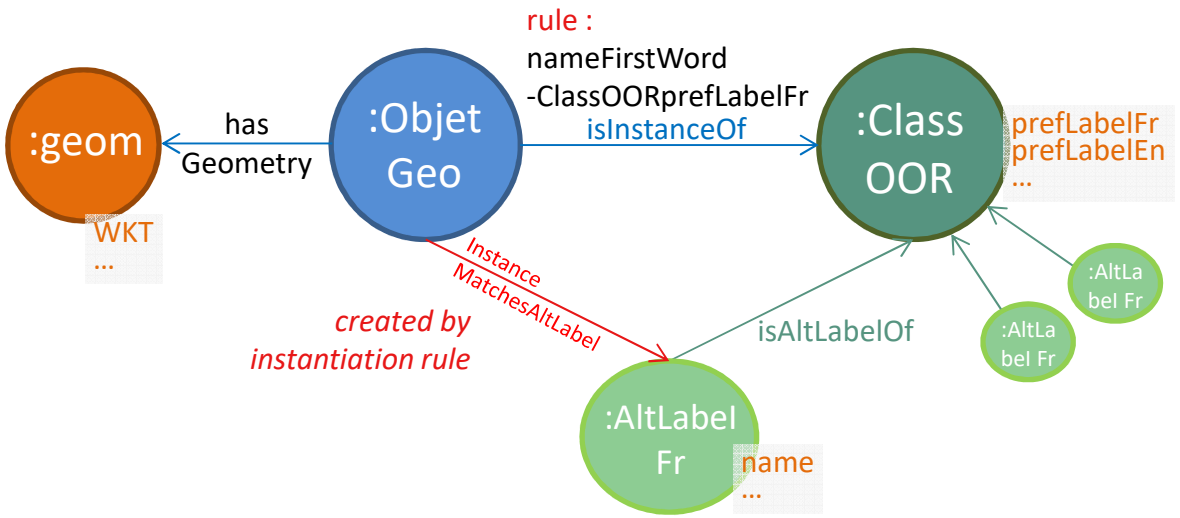
Queries to analyze instantiation process (1)




```
match (c:ClassOOR)-[:isInstanceOf]-(o:ObjetGeo)-(g:geom)
 optional match (o)-[:instanceMatchesAltLabelFr]-(a:AltLabelFr)
return distinct c.prefLabelFr,count(o),
collect(DISTINCT split(g.WKT, )[0]) as geomTypes,
 collect(DISTINCT a.name) as UsedAltLabelsFr
```

c.prefLabelFr	count(o)	geomTypes	UsedAltLabelsFr
abri de montagne	397	[POINT, MULTIPOLYGON]	[abri]
aire d'autoroute	33	[POINT]	[aire de repos, aire de service]
antenne	840	[POINT]	[]
aérogare	1	[MULTIPOLYGON]	[]
barrage	580	[MULTIPOLYGON, MULTILINESTRING, POINT]	[retenue-barrage]

Queries to analyze instantiation process (2)

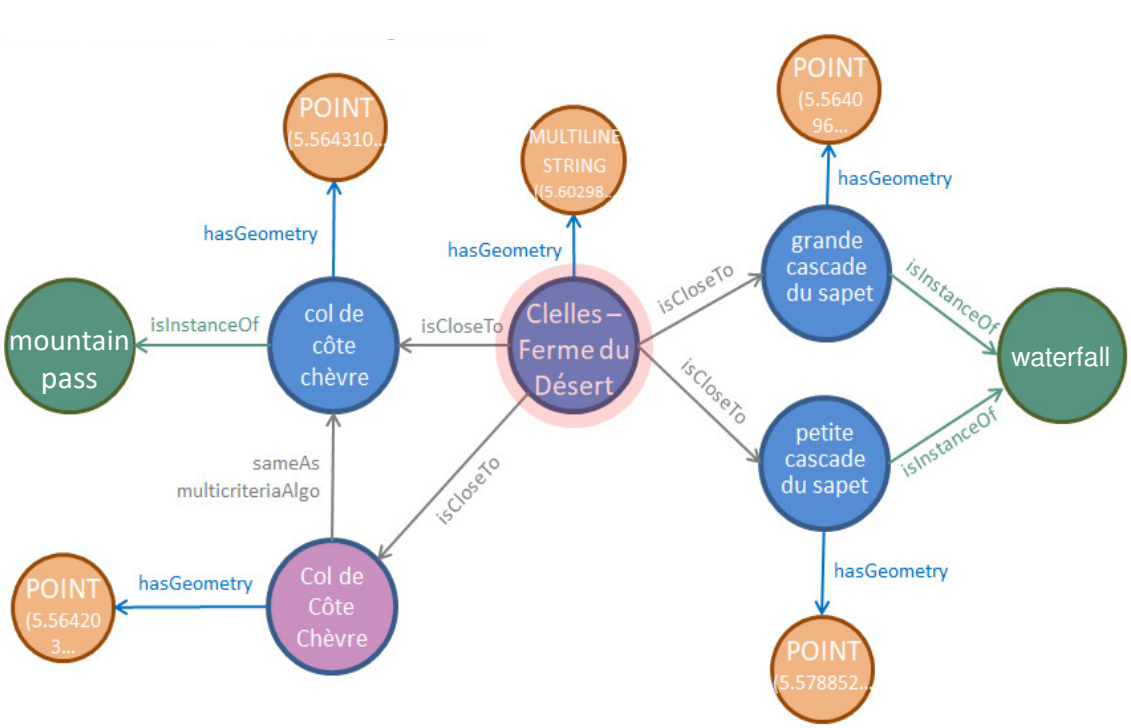


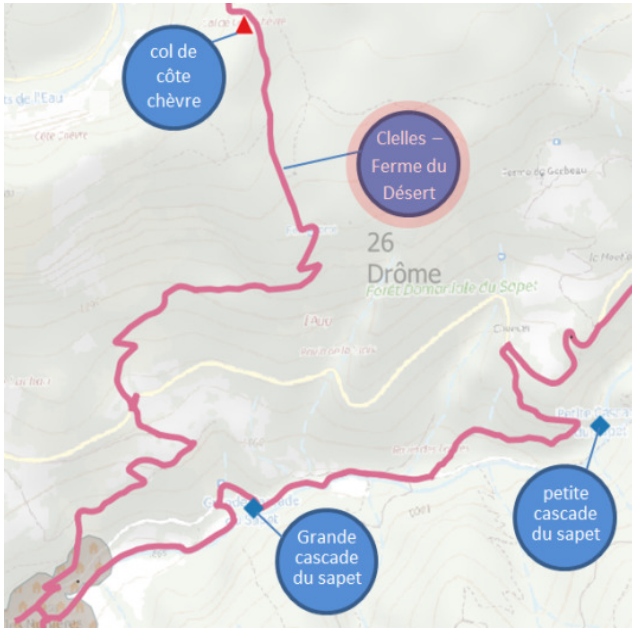

```
match ()-[r:isInstanceOf]->()
return distinct r.rule, count(r) as count order by count desc
```

r.rule	count
type-ClassOORprefLabelFr	94 934
label	82 714
type-ClassOORaltLabelFr	39 238
label-type_voie	17 990
type-ClassOORVariantSpellingFr	9 024
nameFirstWord-ClassOORprefLabelFr	3 493
type-ClassOORprefLabelEn	1 667
type-ClassOORaltLabelEn	576
nameSecondWord-ClassOORprefLabelFr	23
nameFirstWord-ClassOORaltLabelFr	22

Graph pattern query

« I'm on a route to Sapet waterfall and I passed a mountain pass »





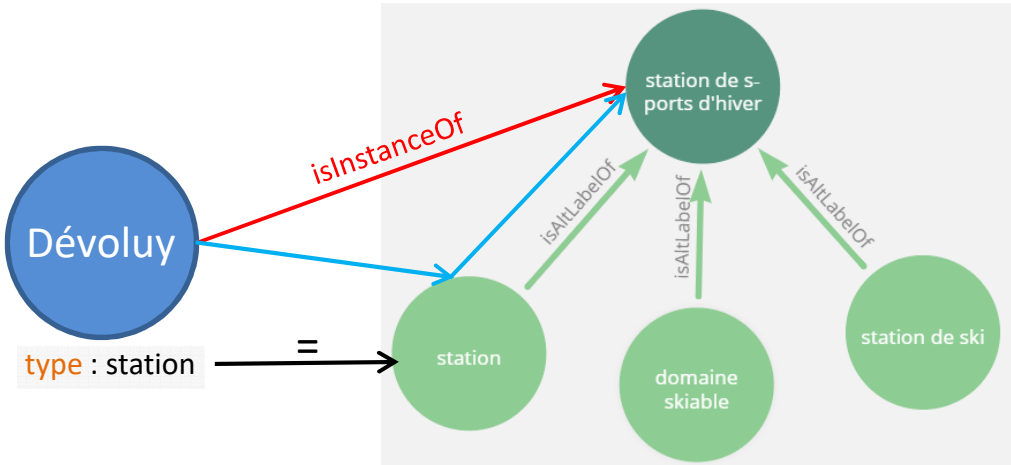
Using the graph structure of the DB (1)

- Relation between landmarks
  - initial idea, still to further implement with more relationships
- the schema can easily be adjusted
  & not one table projection privileged by construction :
  you can start importing with a modelling hypothesis based on one of the main source and adjust as you import more data
  (= modelling by iteration)
- Data analysis
  - Extraction of any desired table to analyse data from different angles
  - Search by pattern
  - Granularity (whole table or a few objects) of selection is easily adaptable

	count
match (o:InZoneFilRouge) return count(o);	36 382
match (o:ObjetGeo) where o.name =~ (?i).*vianney.* return count(o);	15
match(c:ClassOOR{prefLabelFr:"église"})--(o:ObjetGeo) where o.name=~(?i).*vianney.* return o.name, labels(o)	2

Using the graph structure of the DB (2)

- In LOO instantiation rules (= connecting data to normalized categories)
  - Relations transitivity used to normalize categorization through different alternative labels



  - Context specific rules : applied only on a subdomain

Graph : a better structure than tables to *explore* data (1)

- IGN BDTOPO = over 50 layers (= table) grouped by theme

(French national mapping agency)

- Objects of interest

(identified in the ontology)

escalier	ligne électrique
fontaine	nappe d'eau
forêt	parking
forêt de conifères	pic
forêt de feuillus	piste de ski
glacier	plaine
gorge	point d'eau
grotte	pont
hameau	prairie
hydrographie	pylône de remontée mécanique
infrastructure de déplacement terrestre	pylône électrique
lac	refuge
lande ligneuse	relief

- Graph DB integration
  1. All interesting layers in the graph DB
  2. Normalized categorization

→ easy exploration

Queries to explore the data

- In which layers do we have lakes ?

```
match (c:ClassOOR {prefLabelFr:"lac"})--(o:sourceBDTOPO:v2021)
return distinct [lab in labels(o) where lab in $couchesBDTOPO|lab][0] as coucheBDTOPO,
count(o)
```

Hydro 279 Plan Eau 282 Surface Hydro 408

- How many objects in the religious building subdomain ?

```
match (cHydro:ClassOOR)-[:isSubClassOf*]->(:ClassOOR {prefLabelFr: "bâtiment religieux"})
with collect(distinct cHydro.prefLabelFr) as hydrographie
match (o:ObjetGeo)--(c:ClassOOR) where c.prefLabelFr in hydrographie
return count(o)
```

1 805

- Which types of the camp2camp source have not been instantiated ?

```
match (a) where a:AltLabelEn or a:AltLabelFr with collect(a.name) as AltLab
match (c:ClassOOR) with AltLab, collect(c.prefLabelFr) as PrefLabFr,collect(c.prefLabelEn) as PrefLabEn
match (o:sourceC2c) where not o.type in PrefLabEn and not o.type in PrefLabFr and
not o.type in AltLab return distinct o.type,count(o) as count order by count desc
```

o.type	count
access	651
pass	619
local product	106
gite	63
paragliding takeoff	39
…	

-  Choucas Project http://choucas.ign.fr/
- Landmark Objects Ontology
  http://choucas.ign.fr/doc/ontologies/index-fr.html
- The GeoGraph Data Base
  - uses Neo4j spatial plugin (Craig Taverner)
    https://neo4j-contrib.github.io/spatial/
  - is used by GASPAR (Matthieu Viry, Univ. Grenoble)
    a user interface that helps geolocalize victims
- More about GeoGraph Data Base :

http://www.e-tissage.net/GeoGraph

  @v2belleville  @e_tissage

 https://www.linkedin.com/in/veroniquegendner

Thank you for your attention !

Data display on a map with QGIS script : demo