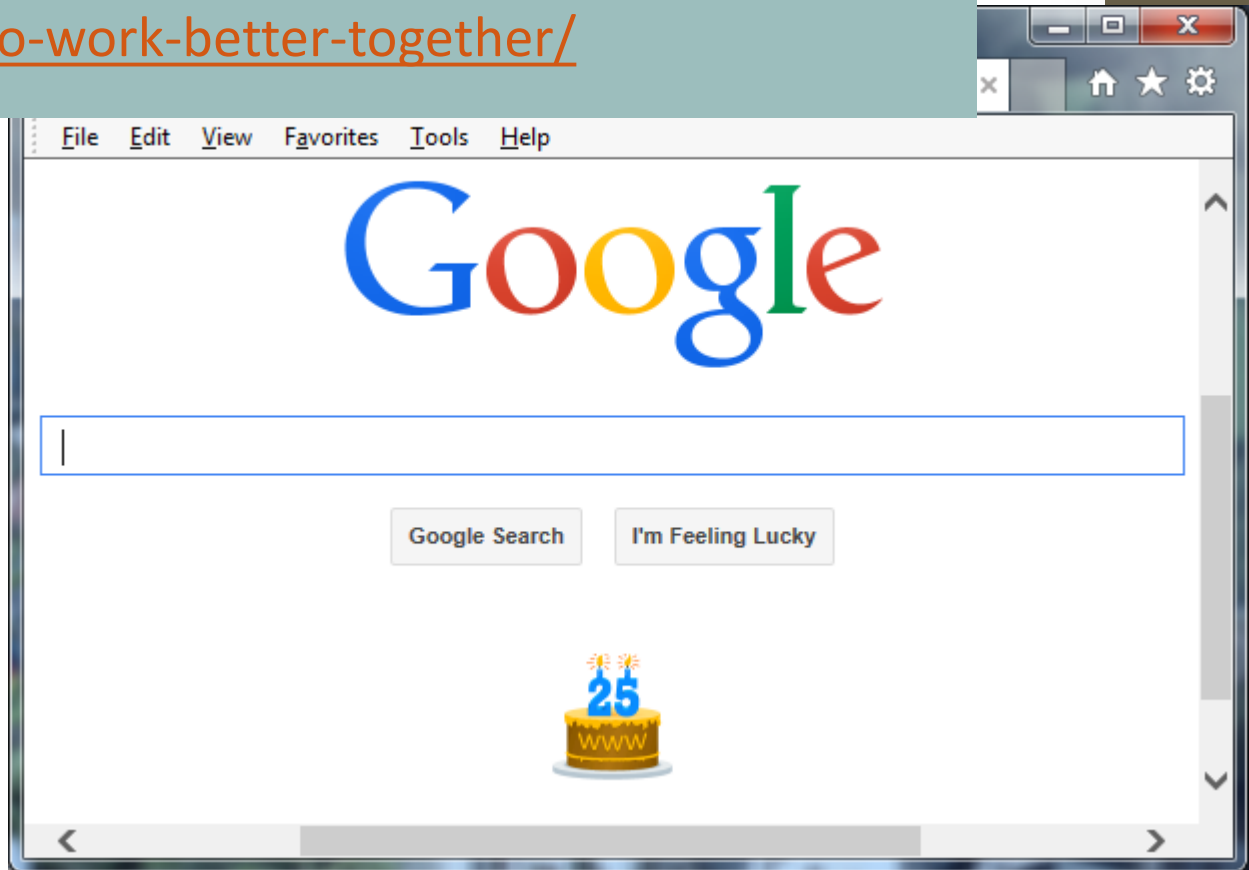
Introduction to SPARQL

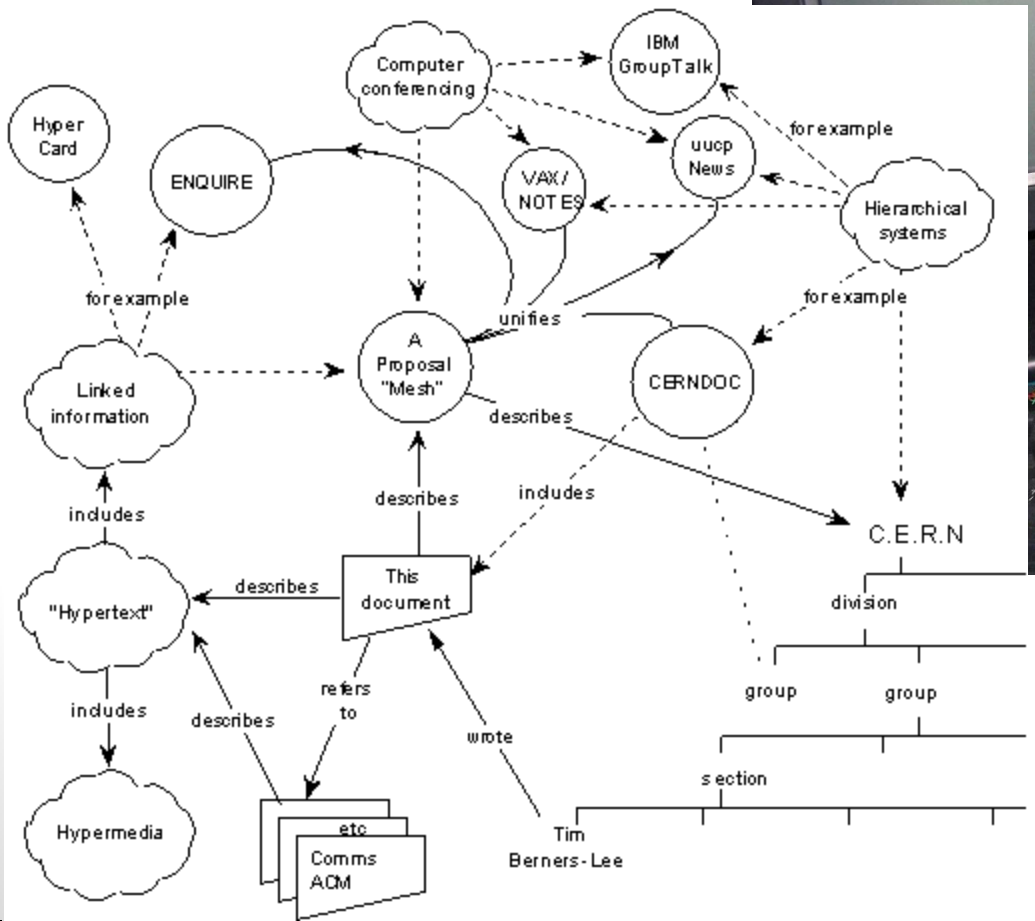
Ernst Henle
ErnstHe@UW.edu
Skype: ernst.predixion

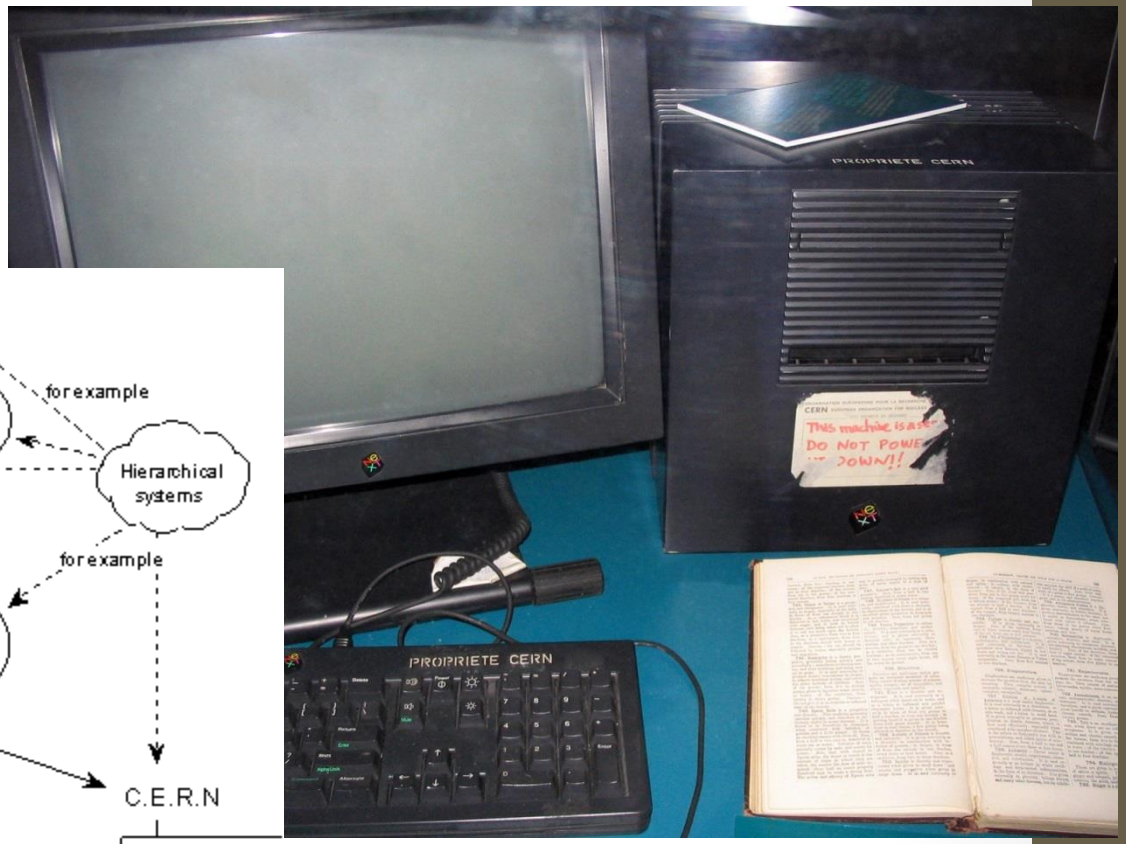
SPARQL

- ***S**PARQL **P**rotocol **a**nd **R**DF **Q**uery **L**anguage*
- "SPARQL will make a huge difference“ http://blog.semantic-web.at/2009/04/22/tim-berners-lee-we-need-data-on-the-web-to-work-better-together/



Tim Berners-Lee's Web Server



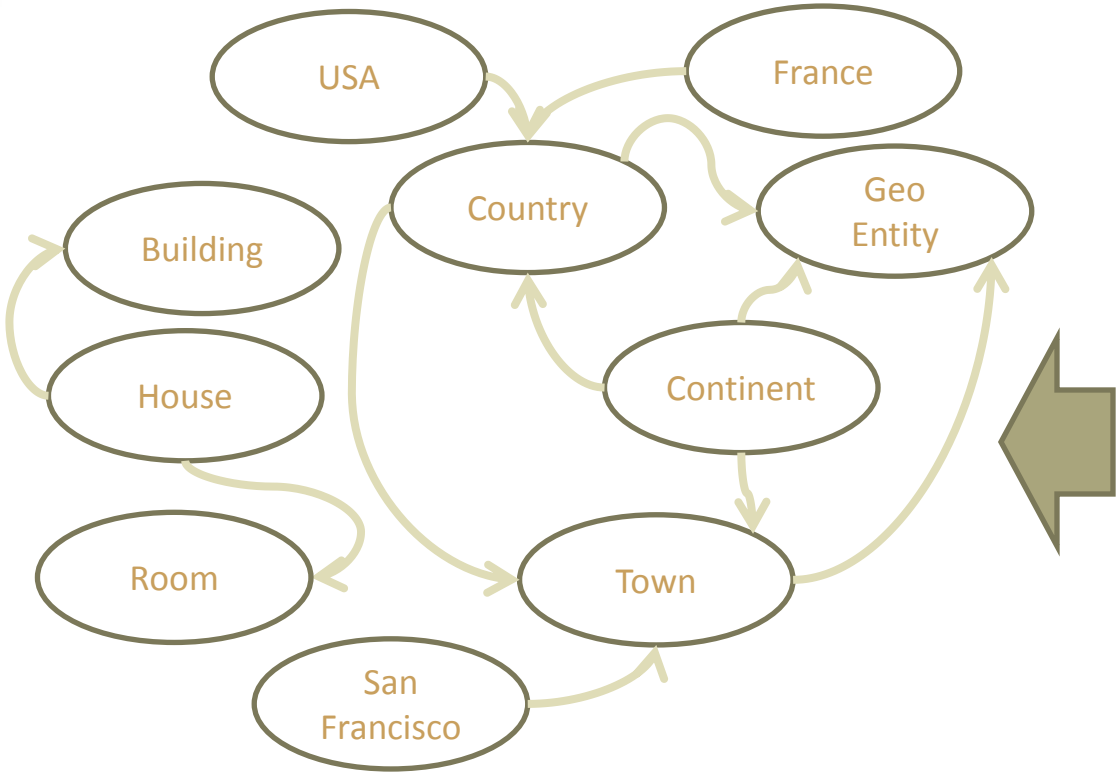


SPARQL: Triplestore

- http://en.wikipedia.org/wiki/Triplestore
- http://en.wikipedia.org/wiki/Sparql

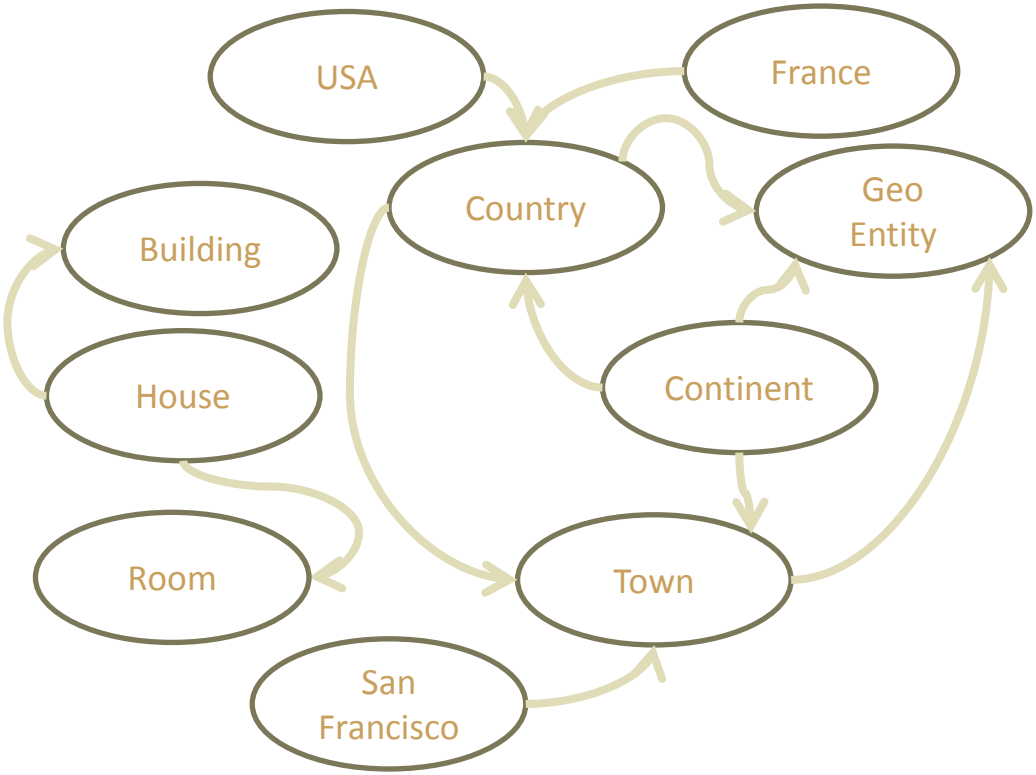
S	P	O
USA	is a	Country
France	is a	Country
Country	is a	Geo Entity
Country	contains	Town
Continent	contains	Country
Continent	is a	Geo Entity
Continent	contains	Town
Town	is a	Geo Entity
San Francisco	is a	Town
House	contains	Room
House	is a	Building

SPARQL: Graph

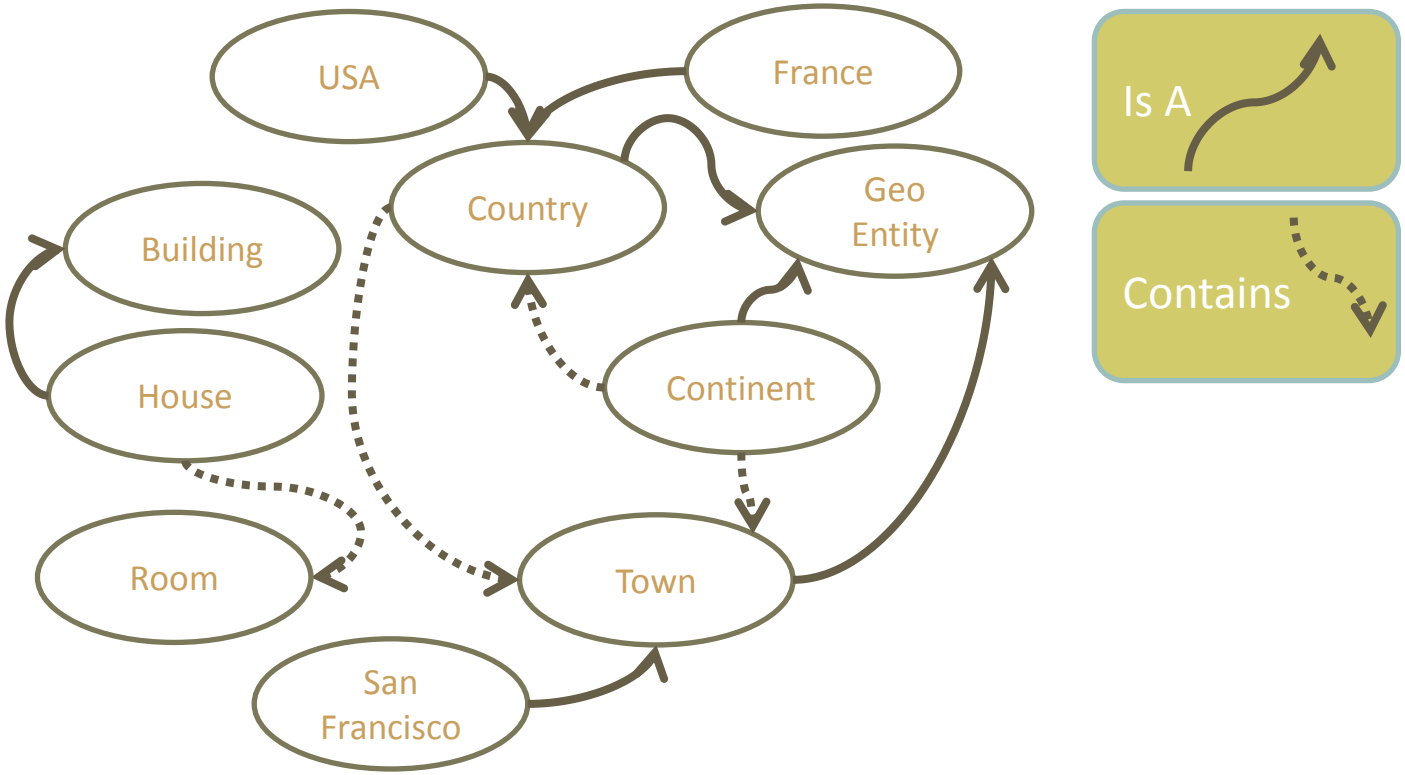


S	P	O
USA	is a	Country
France	is a	Country
Country	is a	Geo Entity
Country	contains	Town
Continent	contains	Country
Continent	is a	Geo Entity
Continent	contains	Town
Town	is a	Geo Entity
San Francisco	is a	Town
House	contains	Room
House	is a	Building

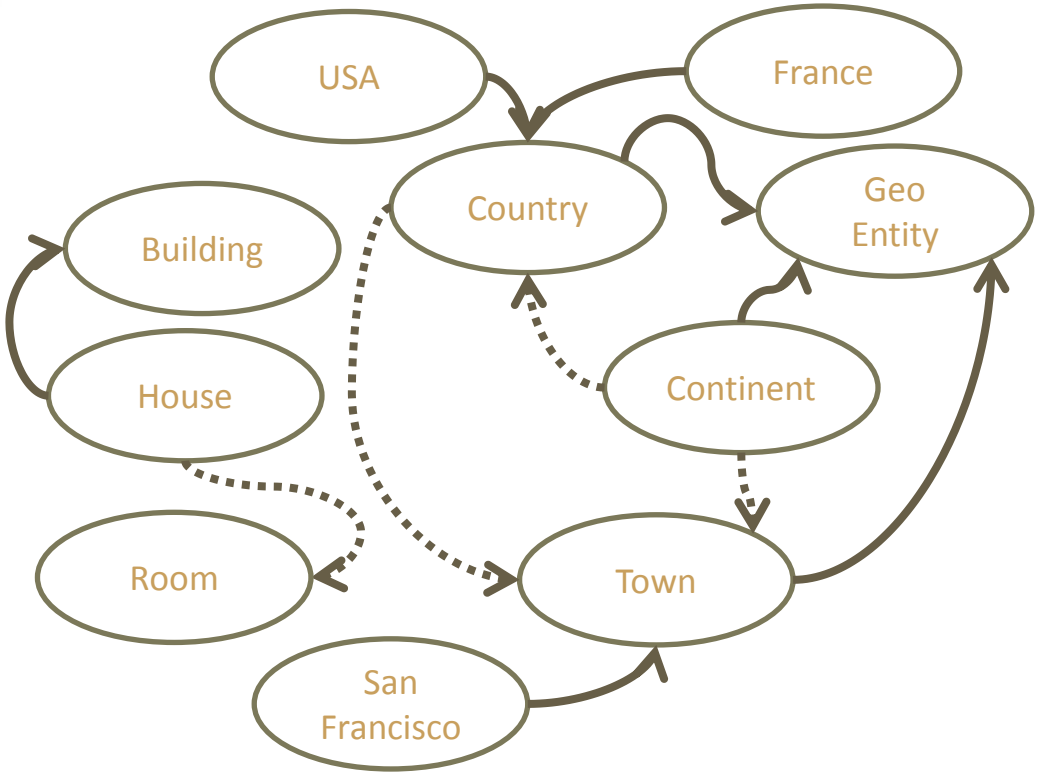
SPARQL: Graph

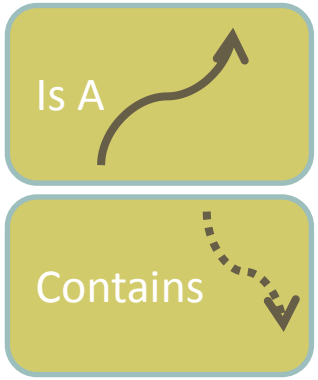


SPARQL: Graph



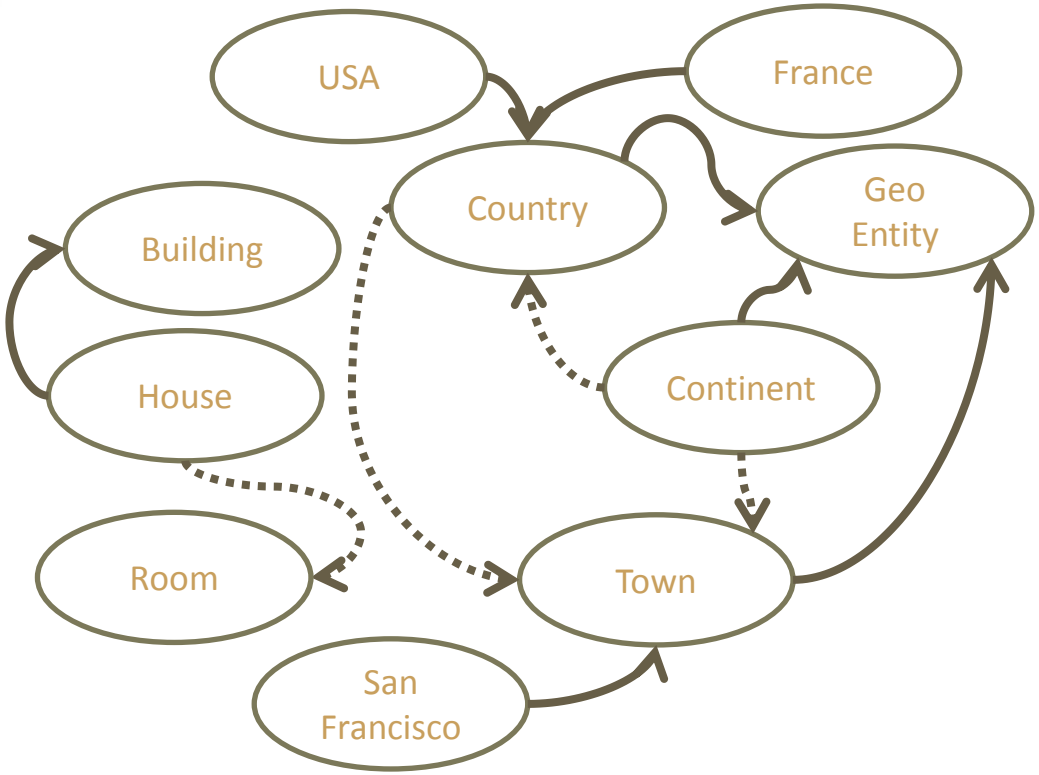
SPARQL: Subjects & Objects

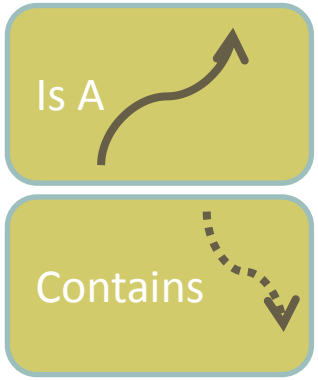




Show node (subject) that contains something
Show node (object) that is contained by something
Show node (subject) that is a Geo Entity

SPARQL: Subjects & Objects

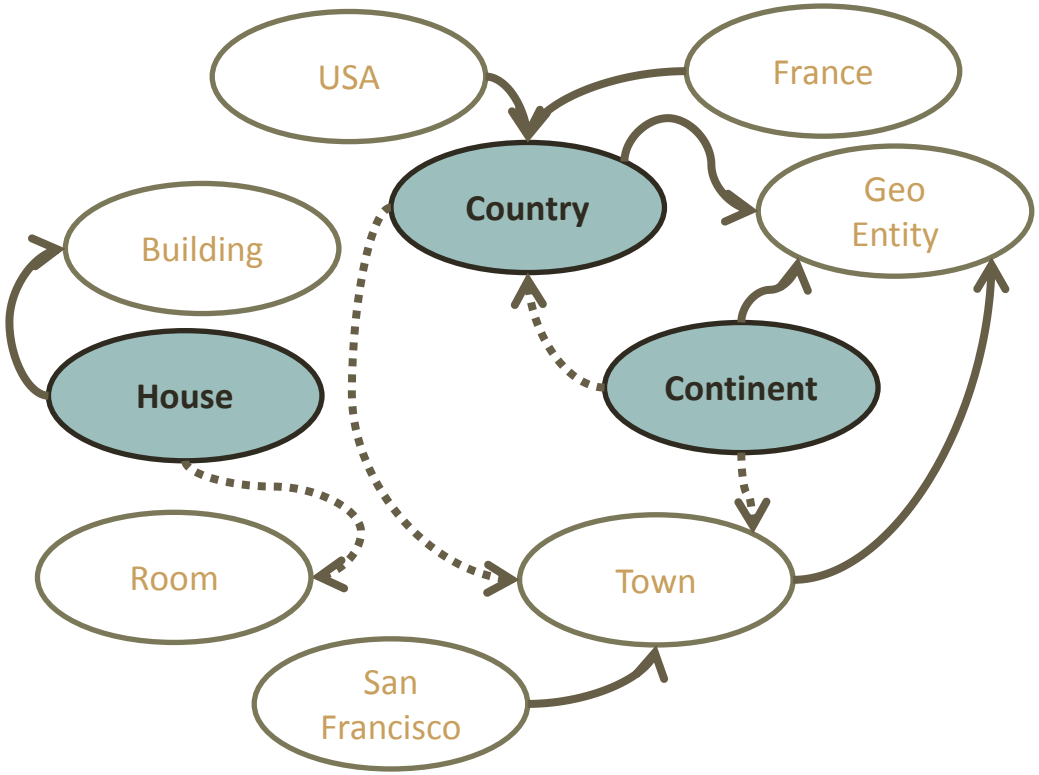


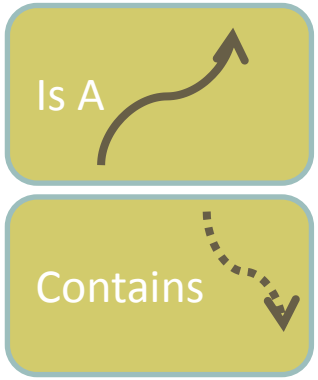



```
SELECT <Columns to be Presented>
FROM <Triplestore>
WHERE { <condition>}
```

- Show node (subject) that contains something
- Show node (object) that is contained by something
- Show node (subject) that is a Geo Entity

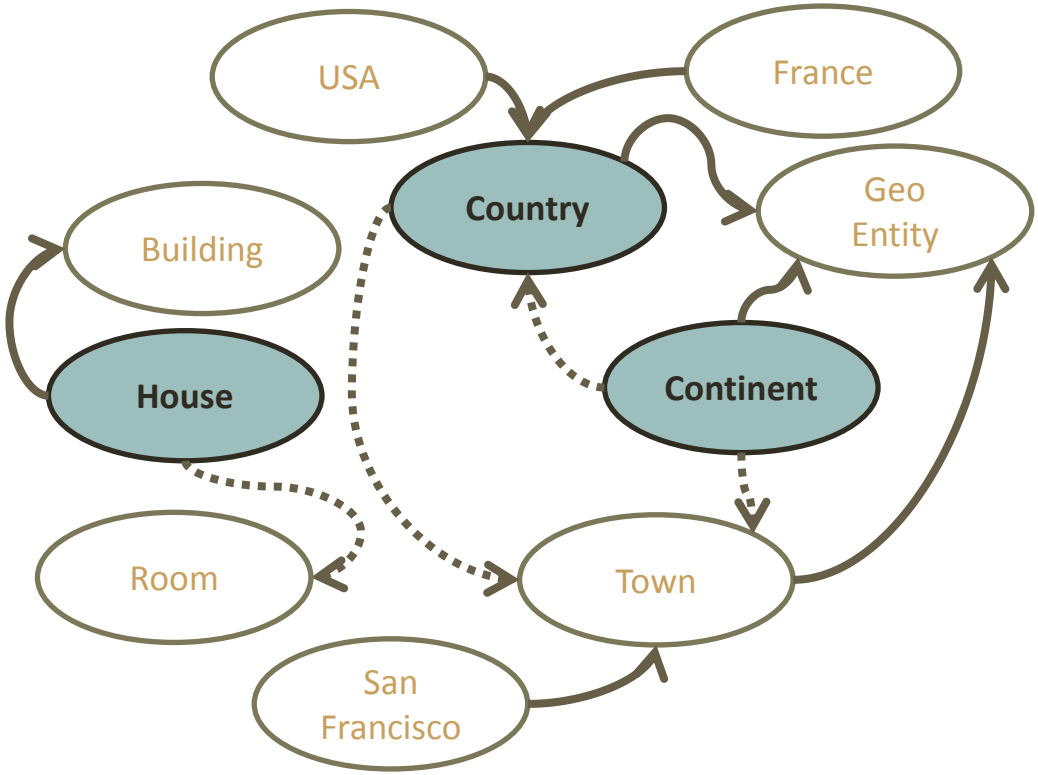
SPARQL: Subjects & Objects

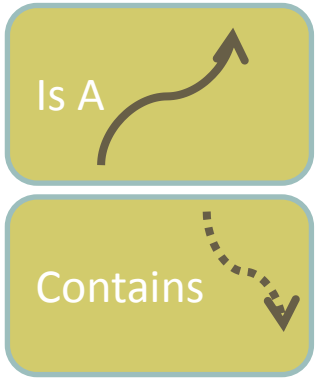




Show node (subject) that contains something
Show node (object) that is contained by something
Show node (subject) that is a Geo Entity

SPARQL: Subjects & Objects

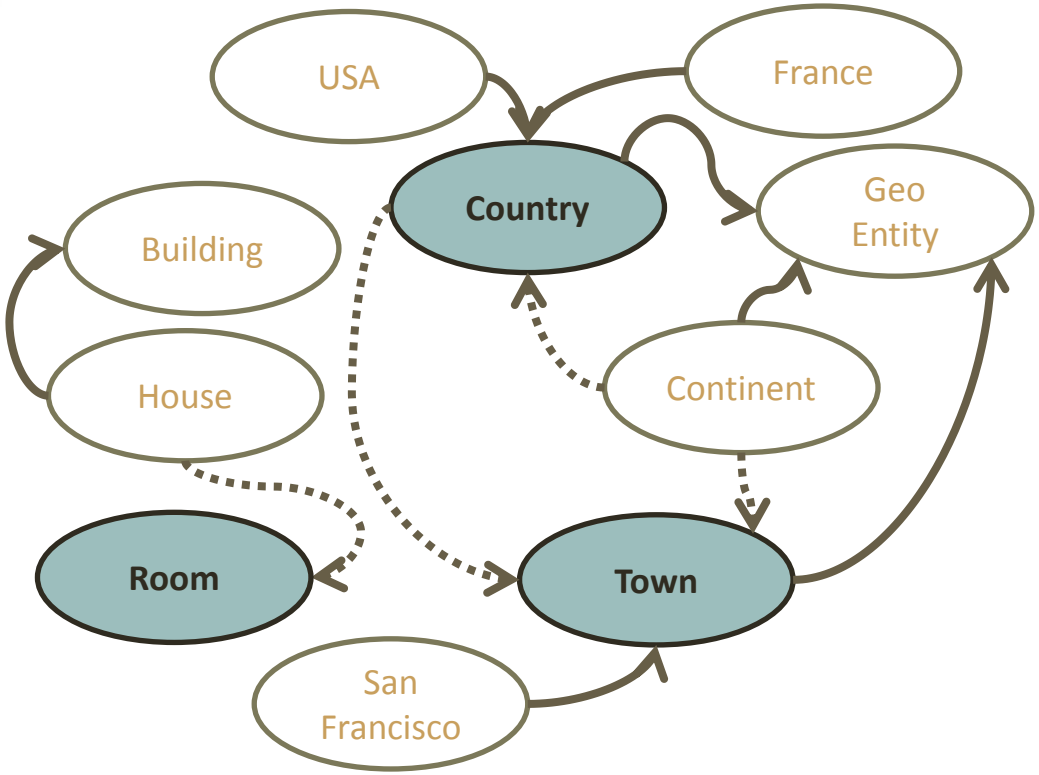


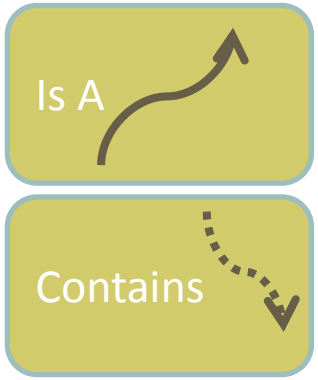



```
SELECT ?s
FROM <myTripleStore>
WHERE {
?s contains ?o
}
```

Show node (subject) that contains something
Show node (object) that is contained by something
Show node (subject) that is a Geo Entity

SPARQL: Subjects & Objects



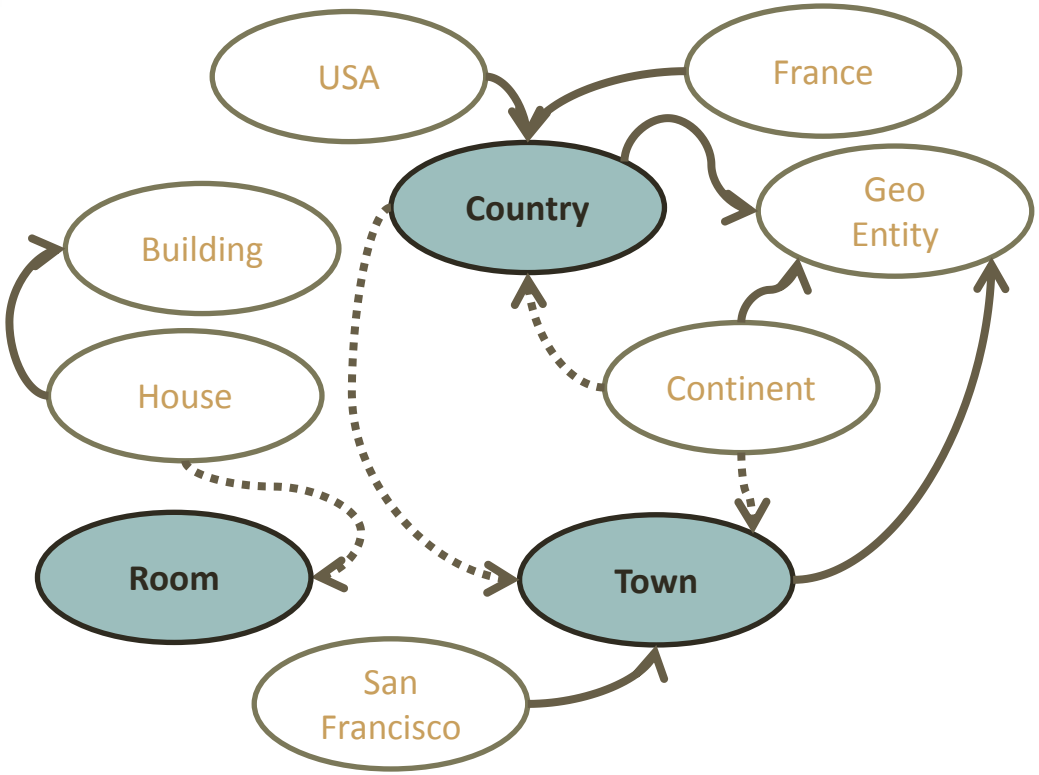


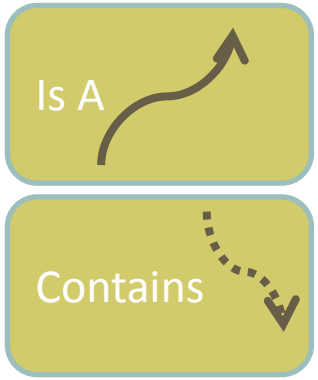
Show node (subject) that contains something

Show node (object) that is contained by something

Show node (subject) that is a Geo Entity

SPARQL: Subjects & Objects





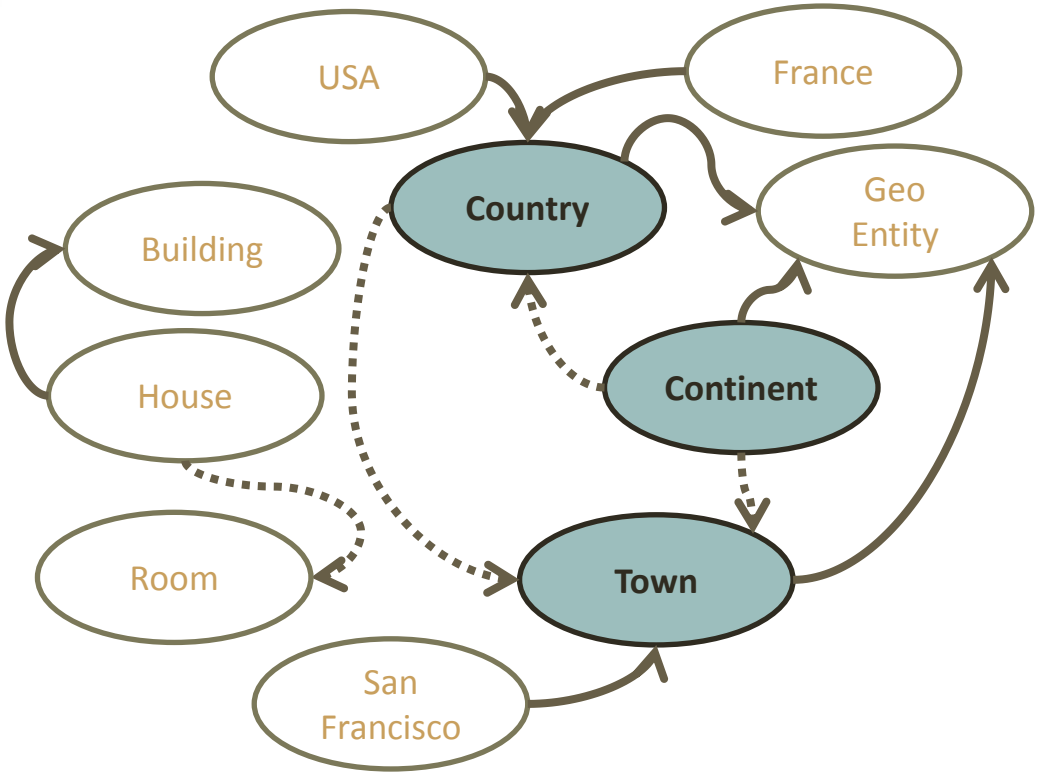

```
SELECT ?o
FROM <myTripleStore>
WHERE {
?s contains ?o}
```

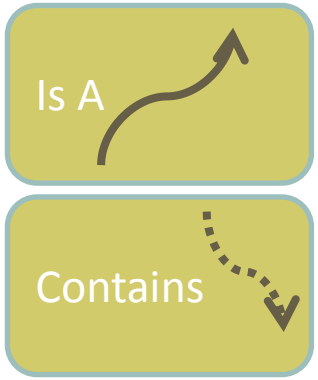
Show node (subject) that contains something

Show node (object) that is contained by something

Show node (subject) that is a Geo Entity

SPARQL: Subjects & Objects



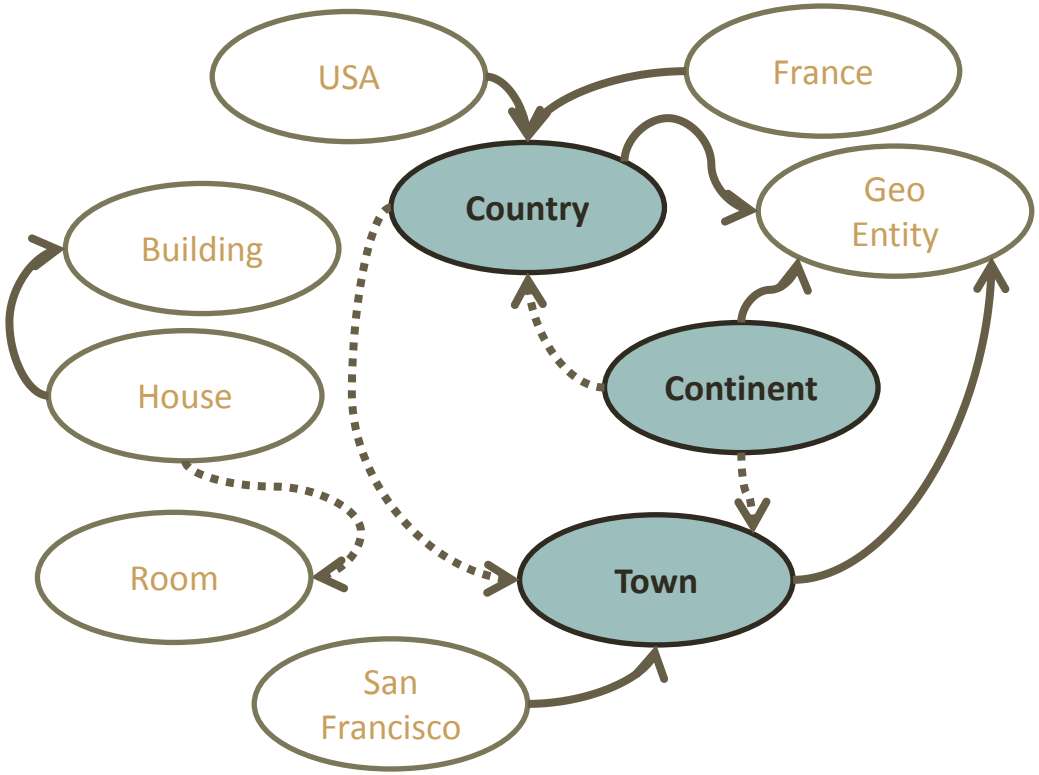


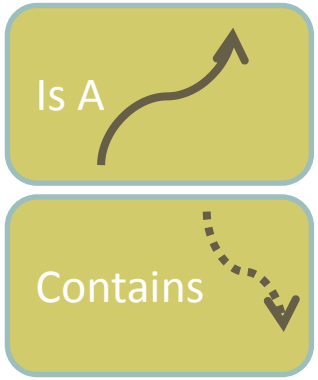
Show node (subject) that contains something

Show node (object) that is contained by something

Show node (subject) that is a Geo Entity

SPARQL: Subjects & Objects





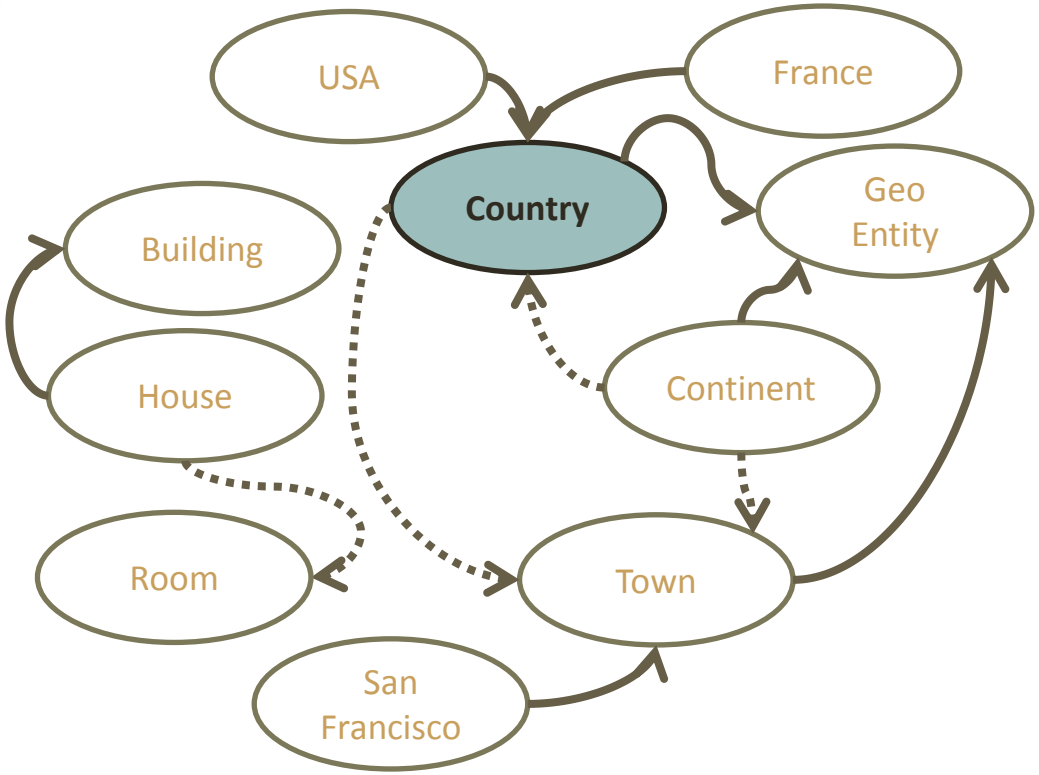

```
SELECT ?s
FROM <myTripleStore>
WHERE {
?s is a Geo Entity}
```

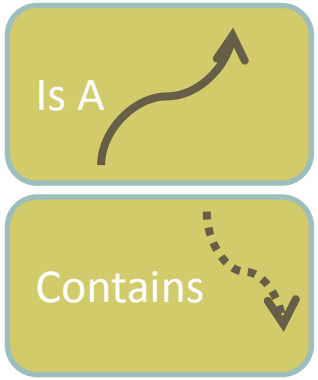
Show node (subject) that contains something

Show node (object) that is contained by something

Show node (subject) that is a Geo Entity

SPARQL: Subjects & Objects



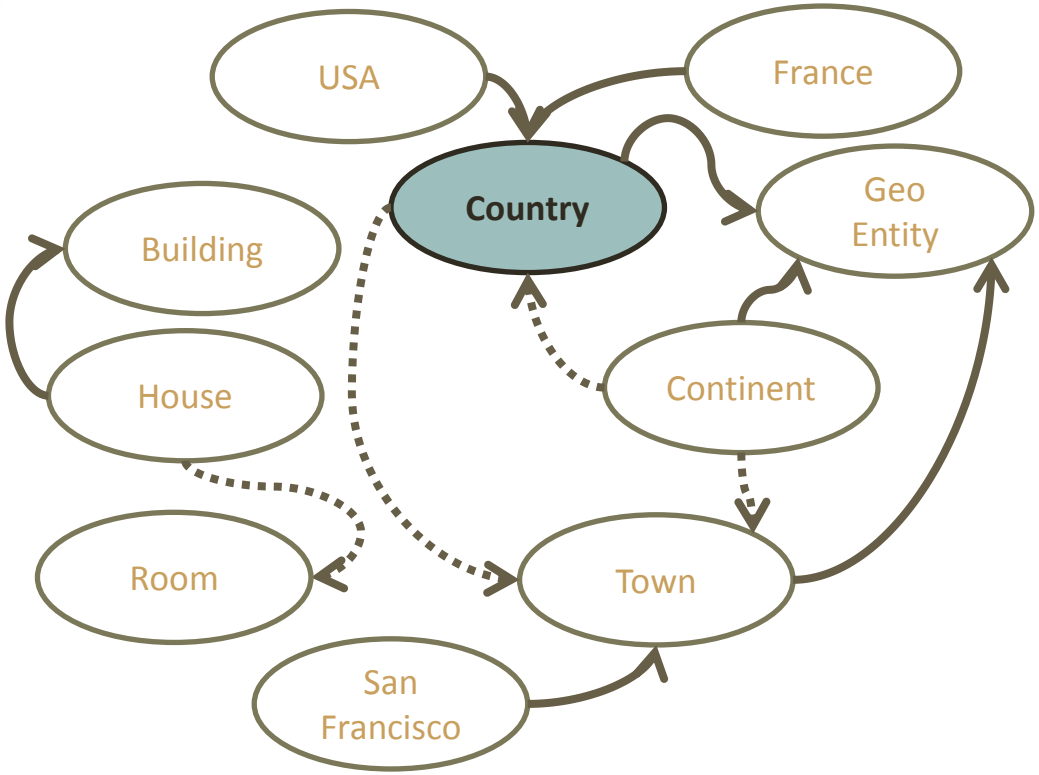


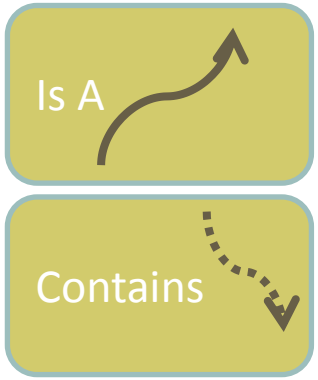
Show node (subject) that contains something

Show node (object) that is contained by something

Show node (subject) that is a Geo Entity

SPARQL: Subjects & Objects





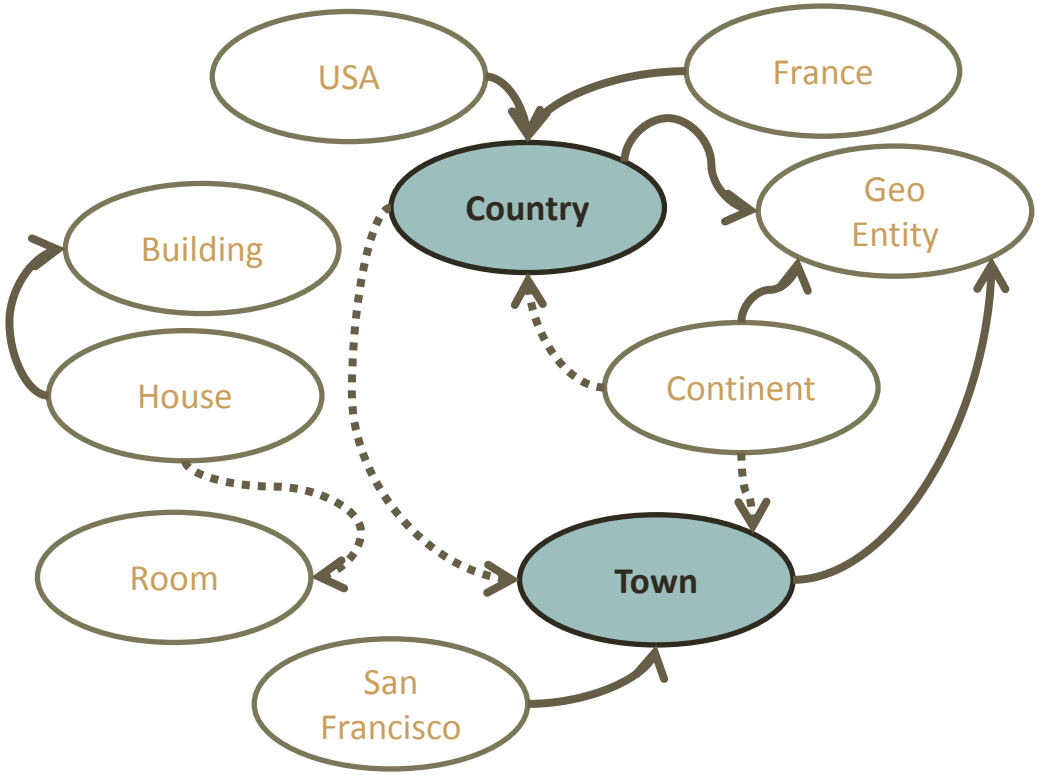

```
SELECT ?s
FROM <myTripleStore>
WHERE {
?s Contains ?o .
?x Contains ?s}
```

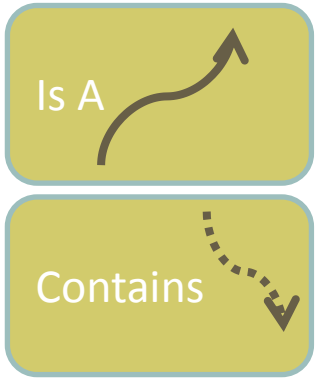
Show node (subject) that contains something

Show node (object) that is contained by something

Show node (subject) that is a Geo Entity

SPARQL: Subjects & Objects



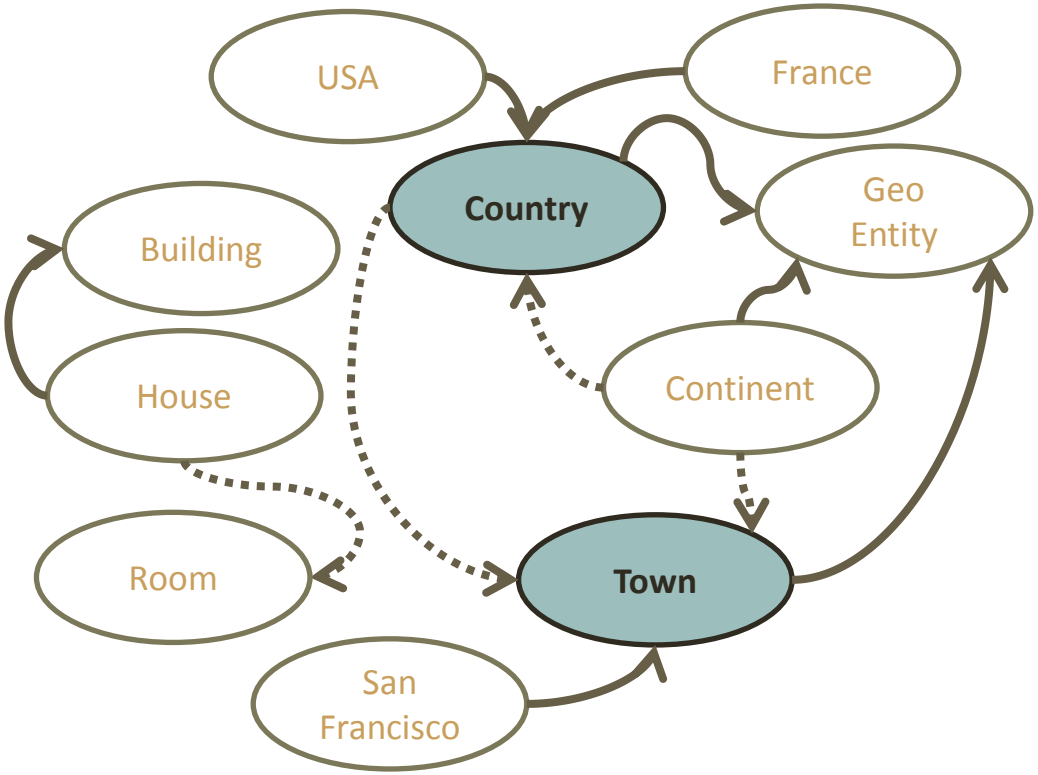


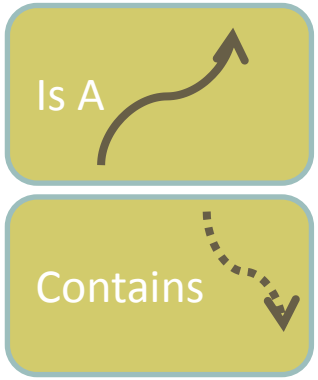
Show node (subject) that contains something

Show node (object) that is contained by something

Show node (subject) that is a Geo Entity

SPARQL: Subjects & Objects





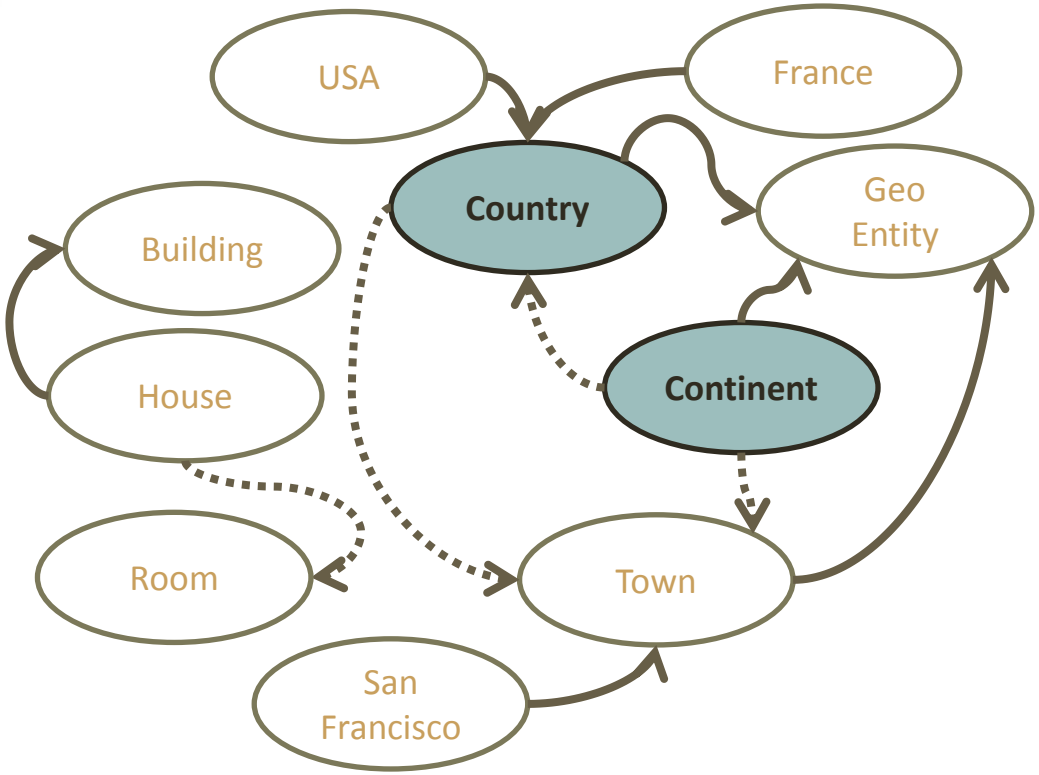

```
SELECT ?o
FROM <myTripleStore>
WHERE {
?x Contains ?o .
?o is a Geo Entity}
```

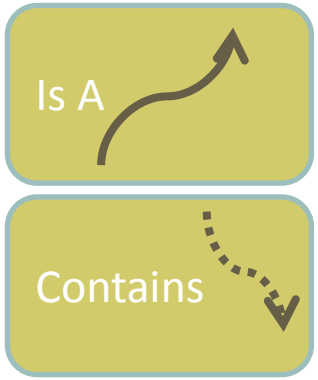
Show node (subject) that contains something

Show node (object) that is contained by something

Show node (subject) that is a Geo Entity

SPARQL: Subjects & Objects



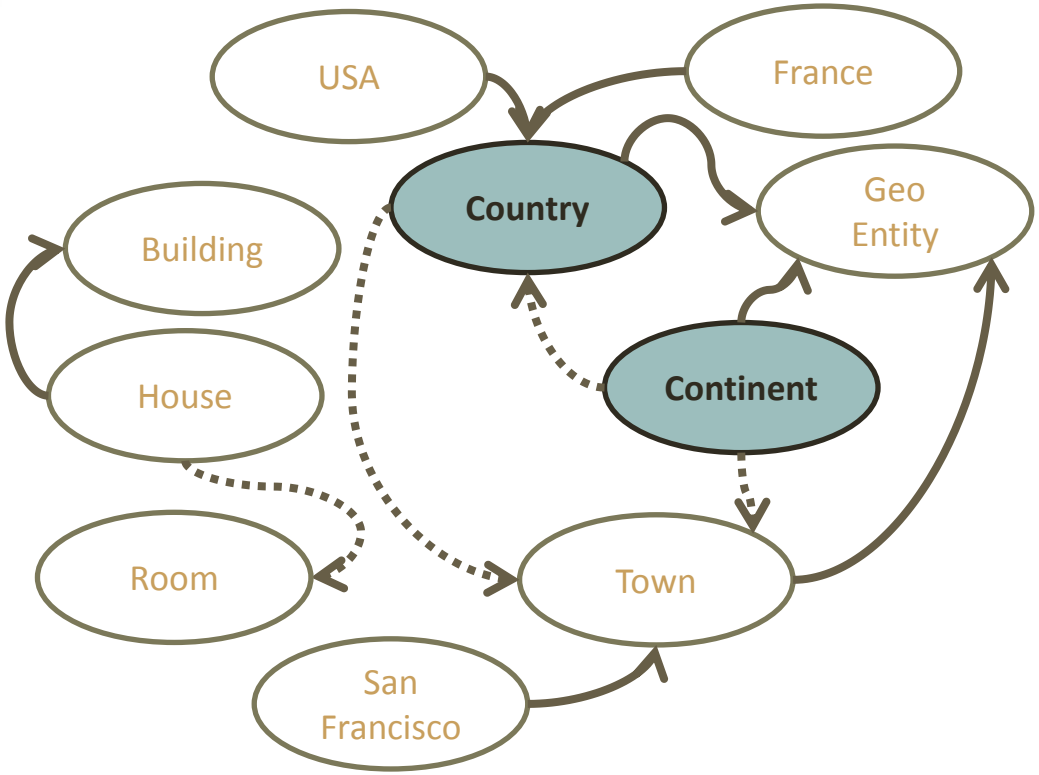


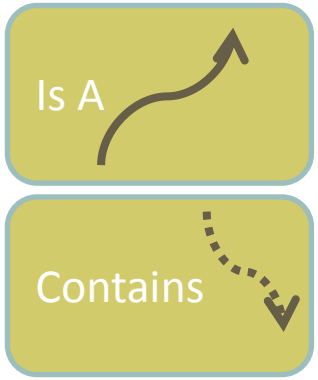
Show node (subject) that contains something

Show node (object) that is contained by something

Show node (subject) that is a Geo Entity

SPARQL: Subjects & Objects





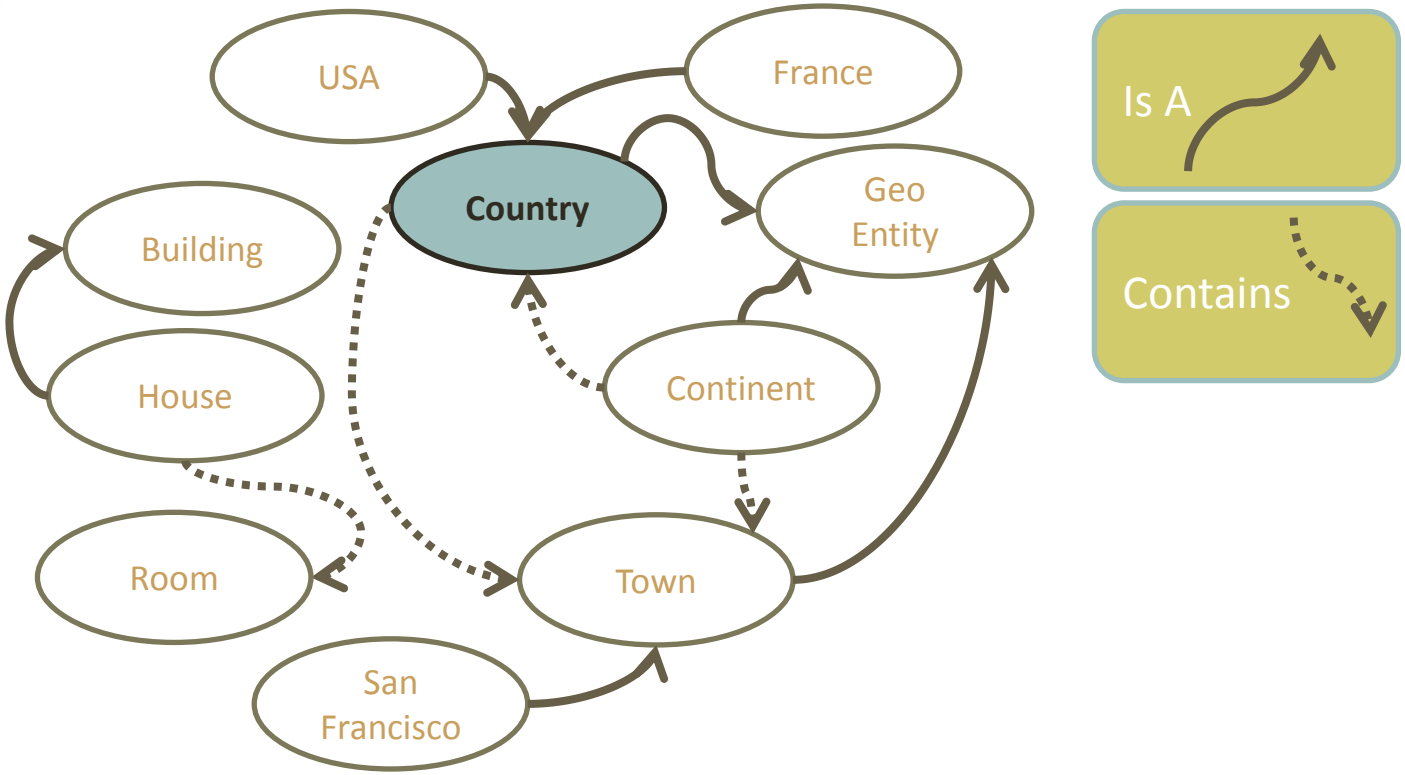

```
SELECT ?o
FROM <myTripleStore>
WHERE {
?s Contains ?o .
?s is a Geo Entity}
```

Show node (subject) that contains something

Show node (object) that is contained by something

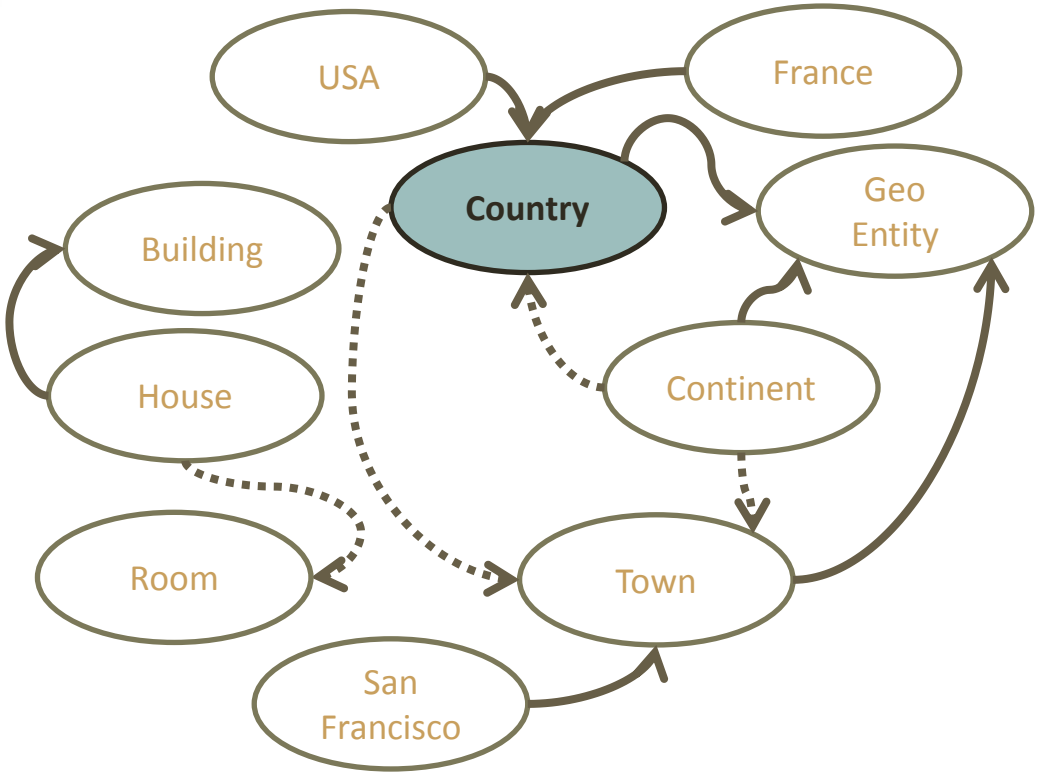
Show node (subject) that is a Geo Entity

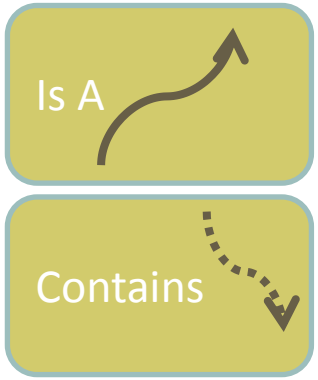
SPARQL: Subjects & Objects



Show node (subject) that contains something
Show node (object) that is contained by something
Show node (subject) that is a Geo Entity

SPARQL: Subjects & Objects






```
SELECT ?o
FROM <myTripleStore>
WHERE {
?s Contains ?o .
?x Contains ?s .
?s is a Geo Entity}
```

Show node (subject) that contains something

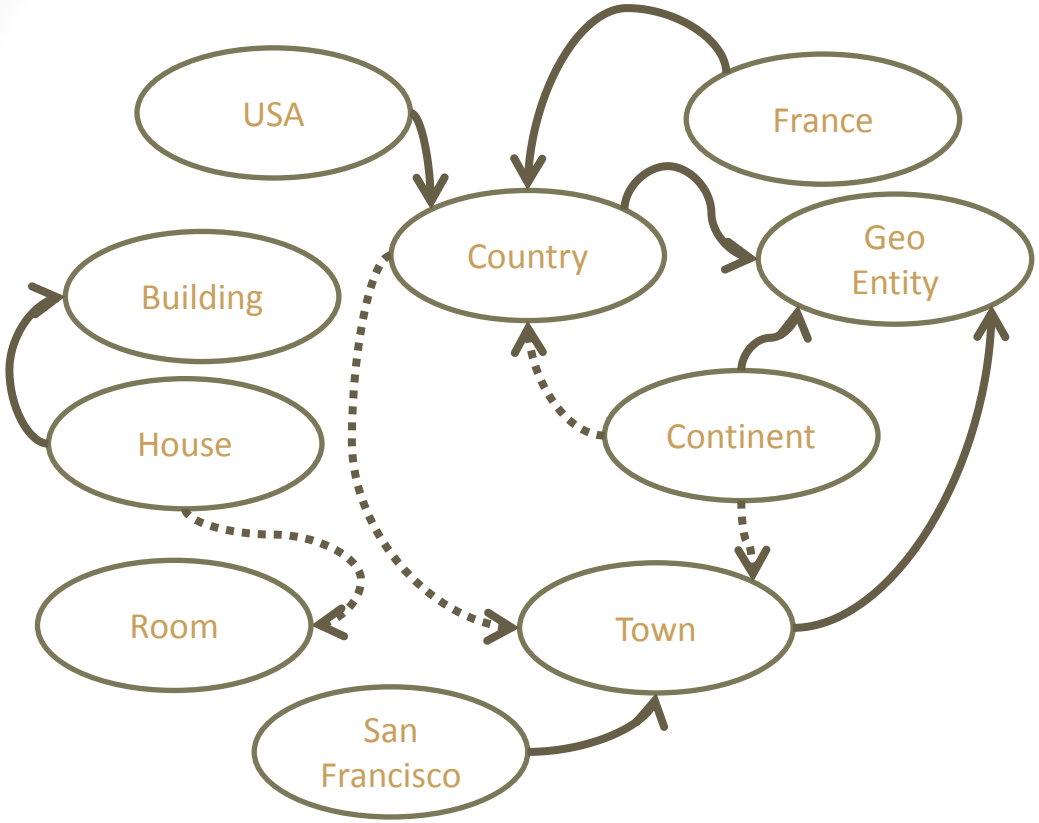
Show node (object) that is contained by something

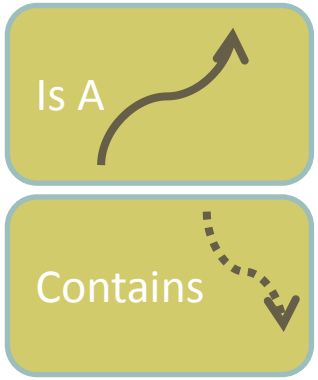
Show node (subject) that is a Geo Entity

SPARQL: Predicates

- Search for predicates

SPARQL: Predicates

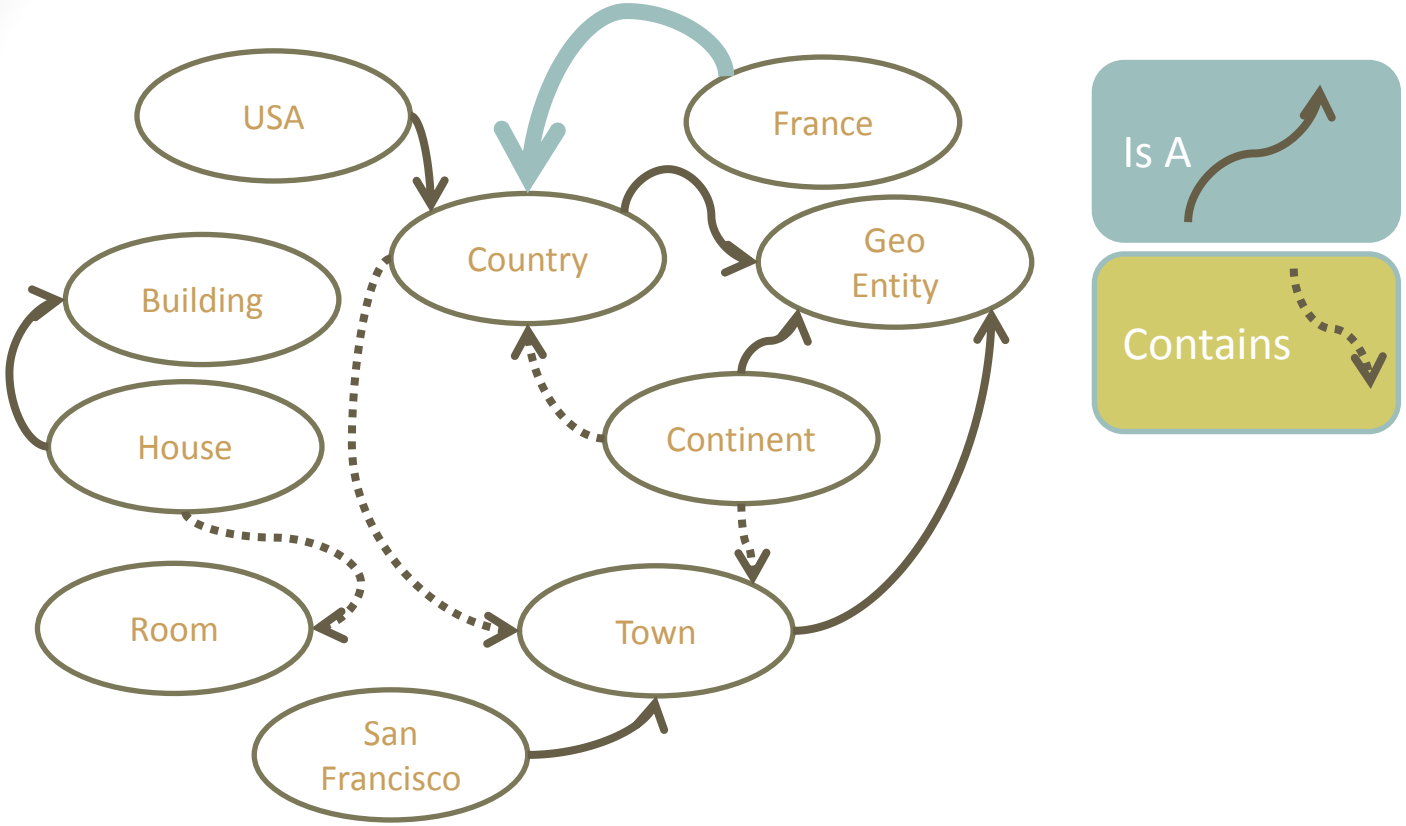




Show relationships (Predicate) from France to Anything

Show relationships from Country to Town

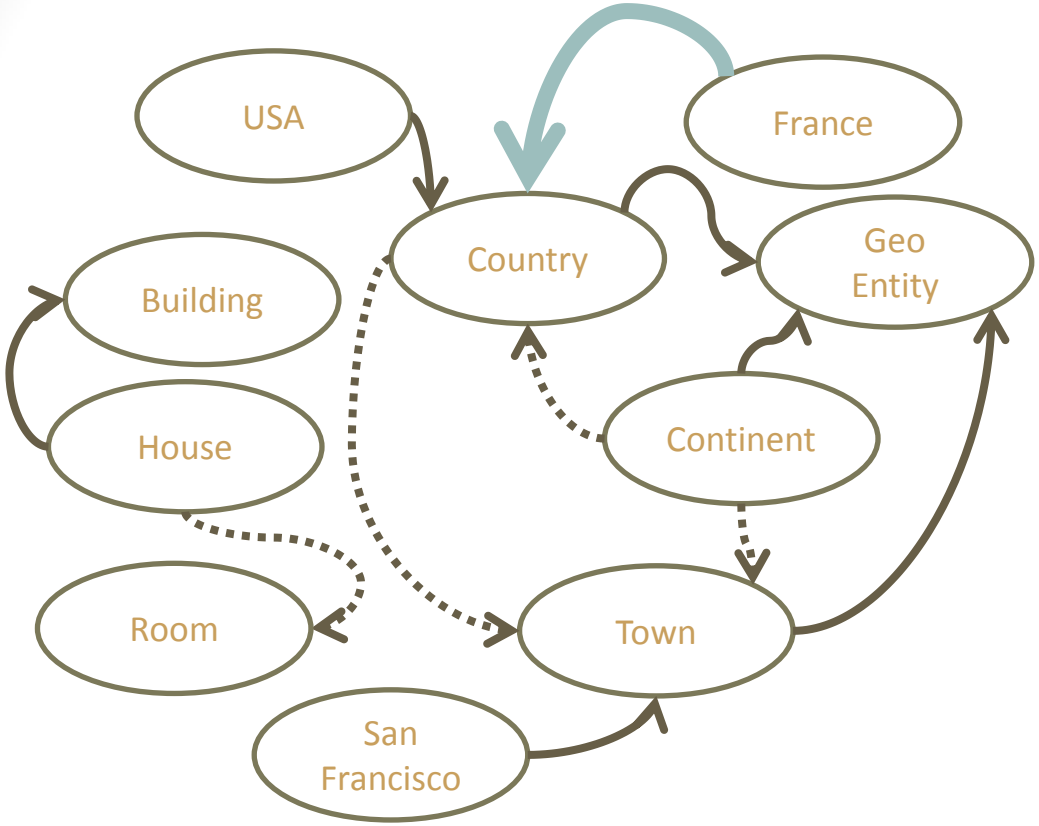
SPARQL: Predicates

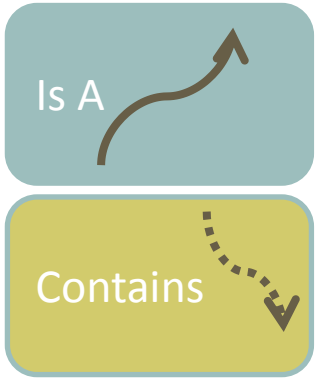


Show relationships (Predicate) from France to Anything

Show relationships from Country to Town

SPARQL: Predicates



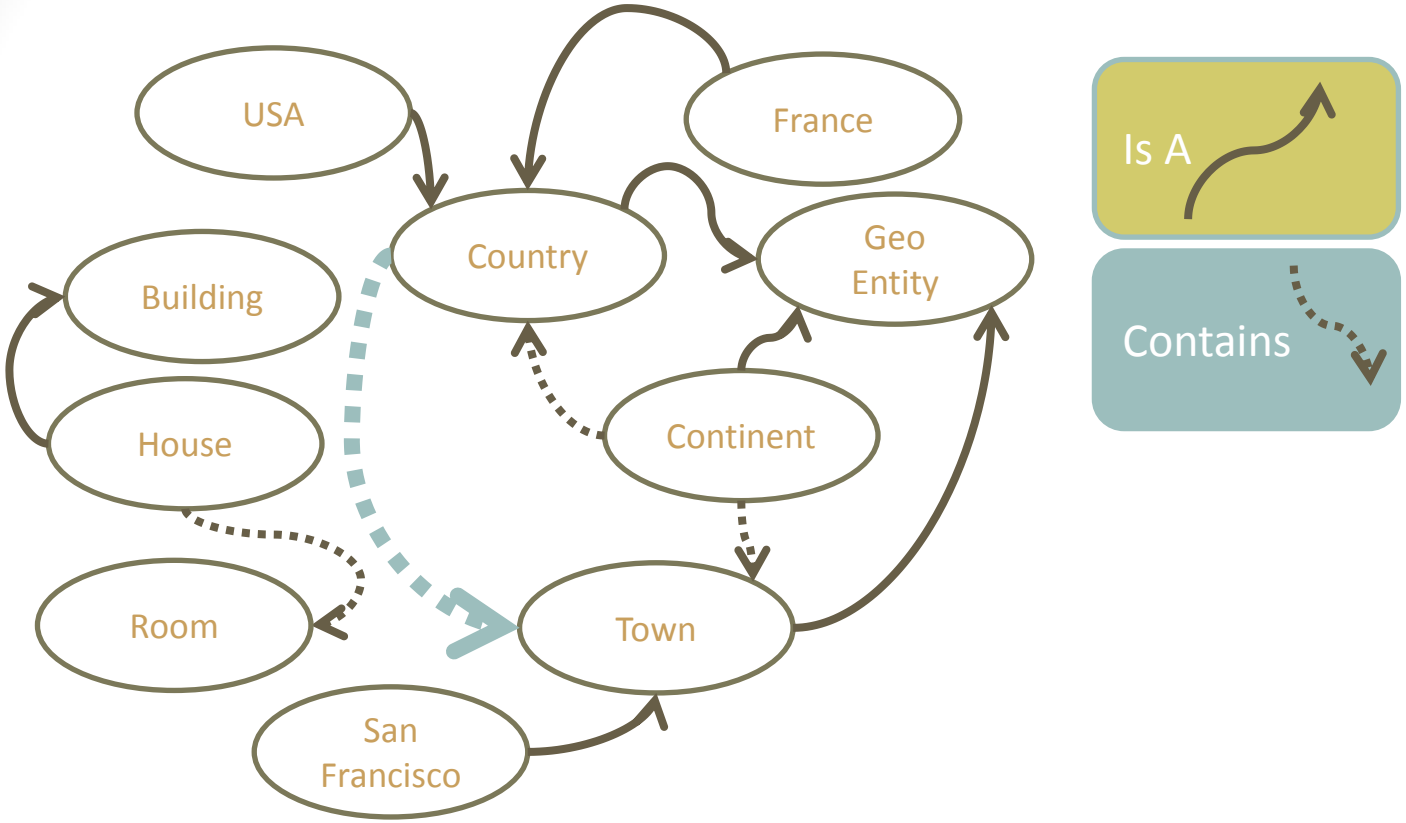



```
SELECT ?p
FROM <myTripleStore>
WHERE {
France ?p ?o}
```

Show relationships (Predicate) from France to Anything

Show relationships from Country to Town

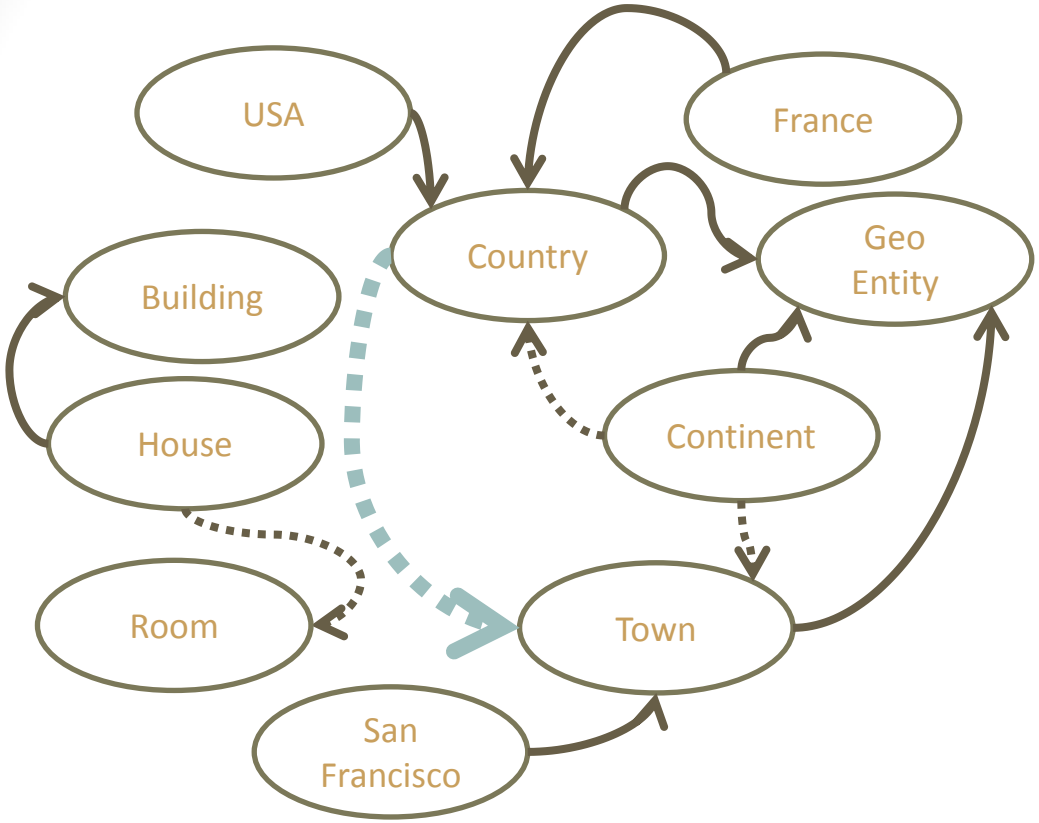
SPARQL: Predicates

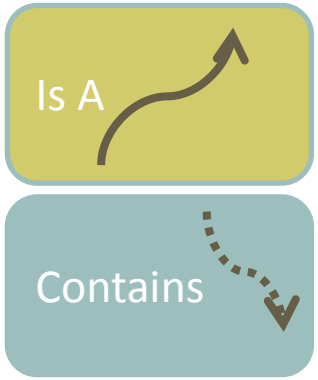


Show relationships (Predicate) from France to Anything

Show relationships from Country to Town

SPARQL: Predicates



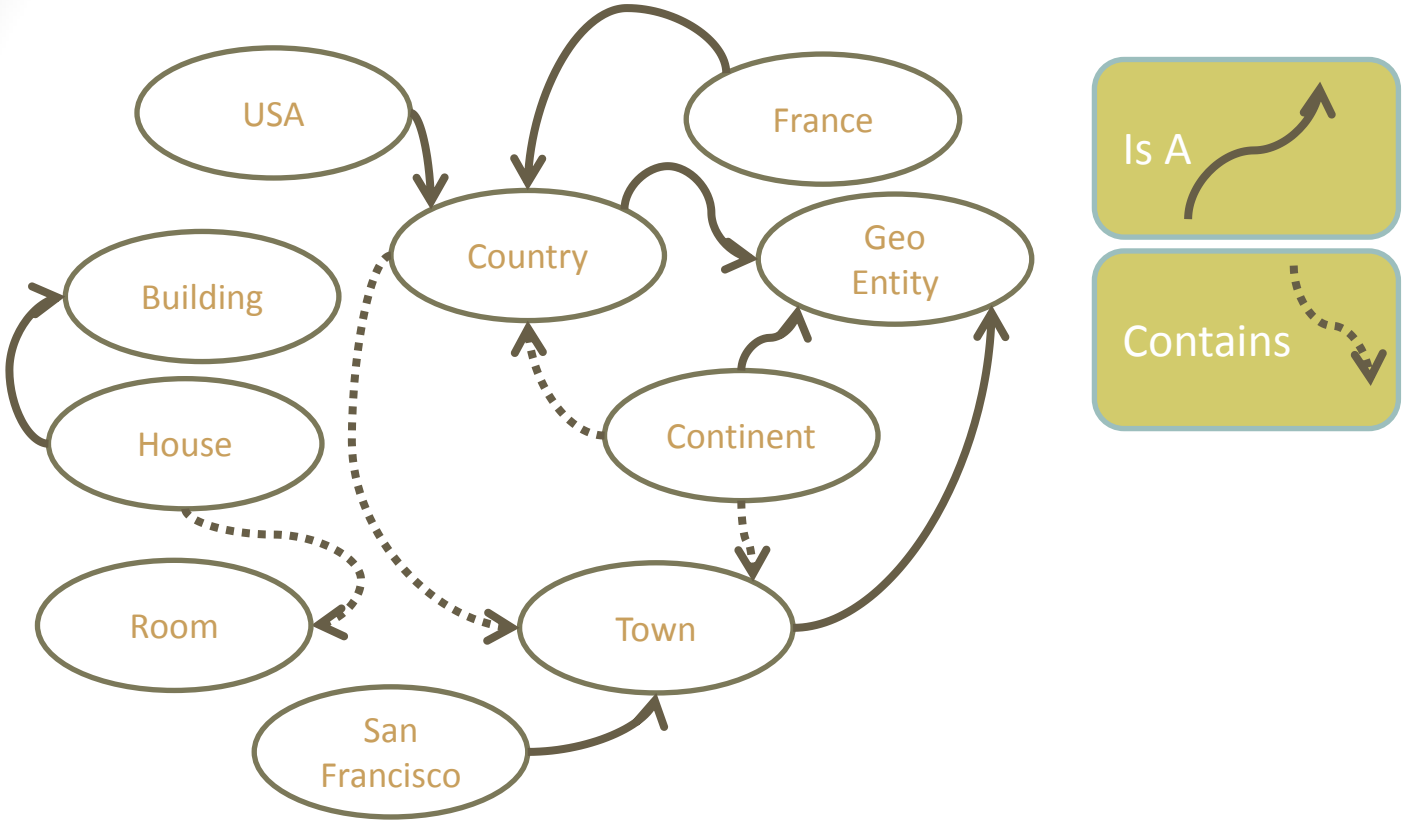



```
SELECT ?p
FROM <myTripleStore>
WHERE {
Country ?p Town}
```

Show relationships (Predicate) from France to Anything

Show relationships from Country to Town

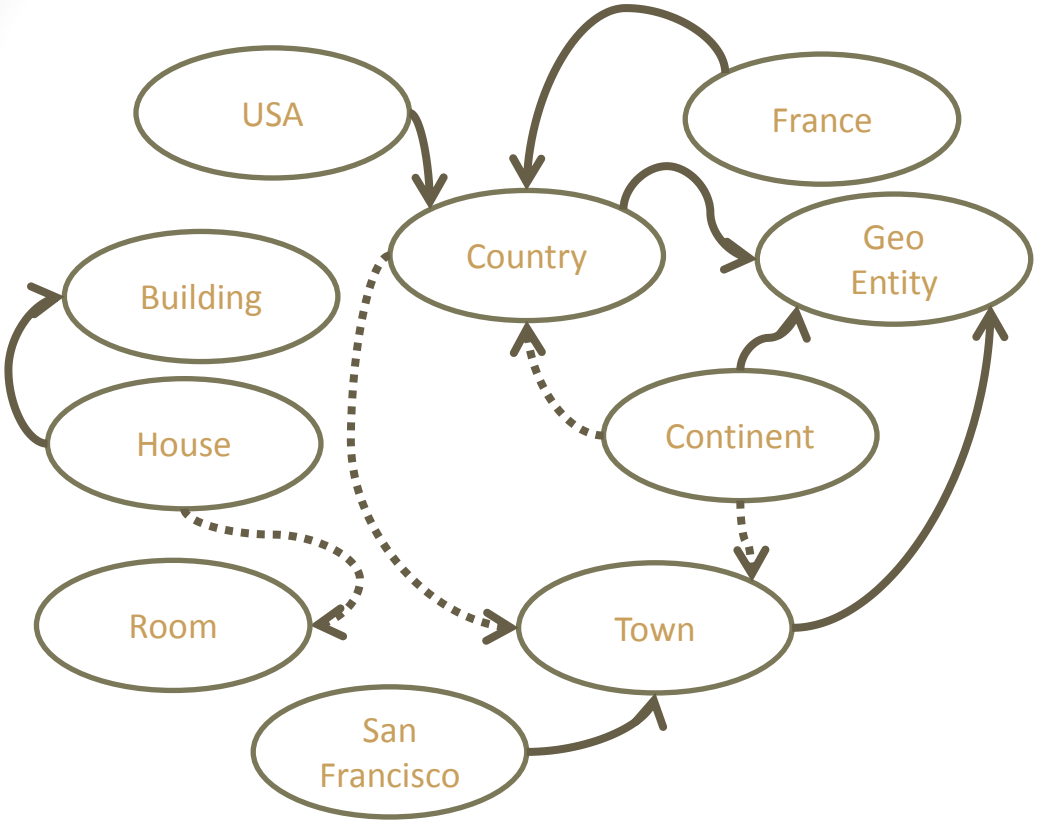
SPARQL: Predicates

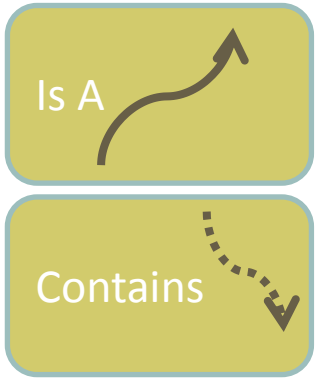


Show relationships (Predicate) from France to Anything

Show relationships from Country to Town

SPARQL: Predicates



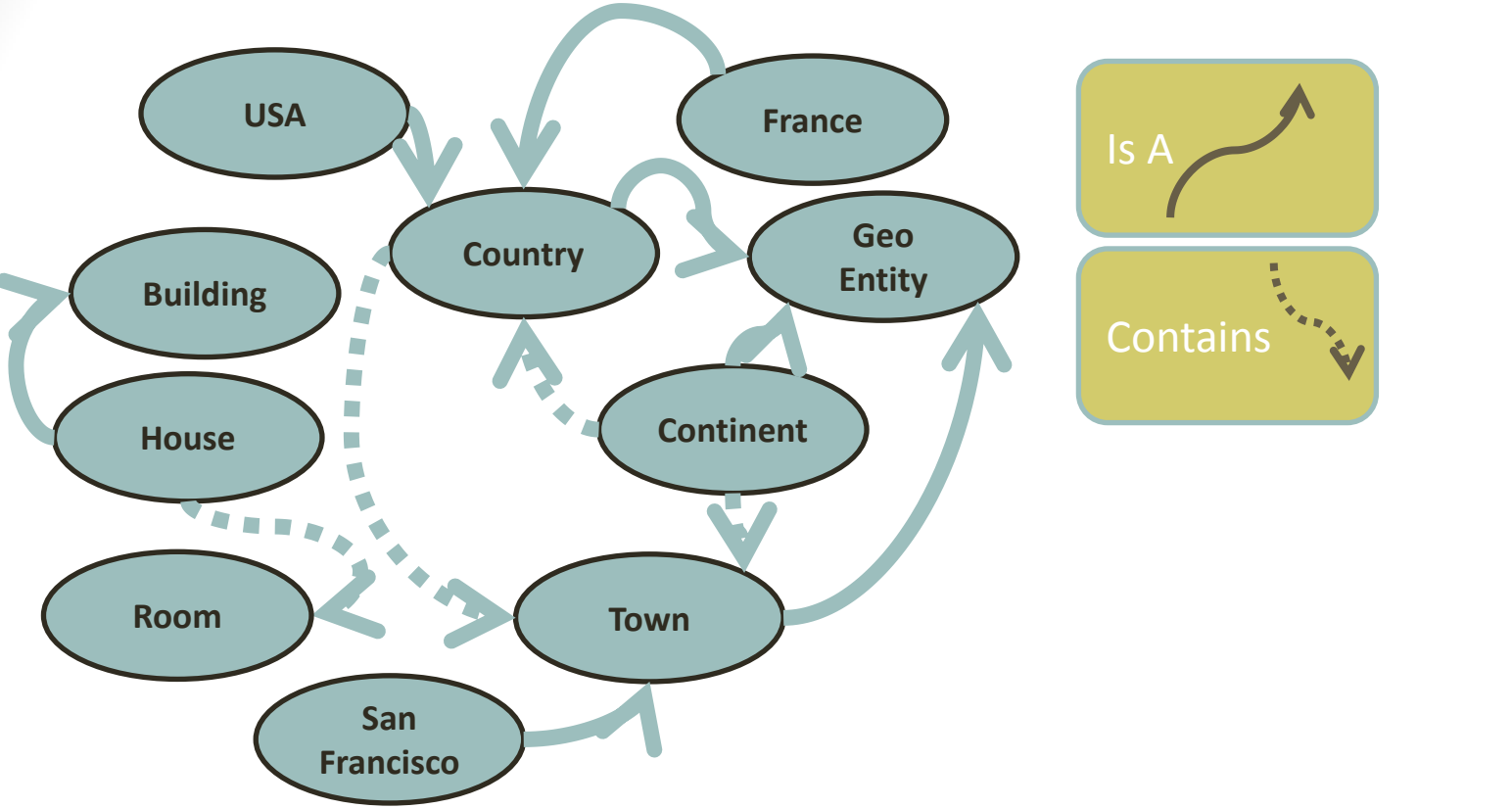



```
SELECT ?p
FROM <myTripleStore>
WHERE {
France ?p ?o .
Country ?p Town}
```

Show relationships (Predicate) from France to Anything

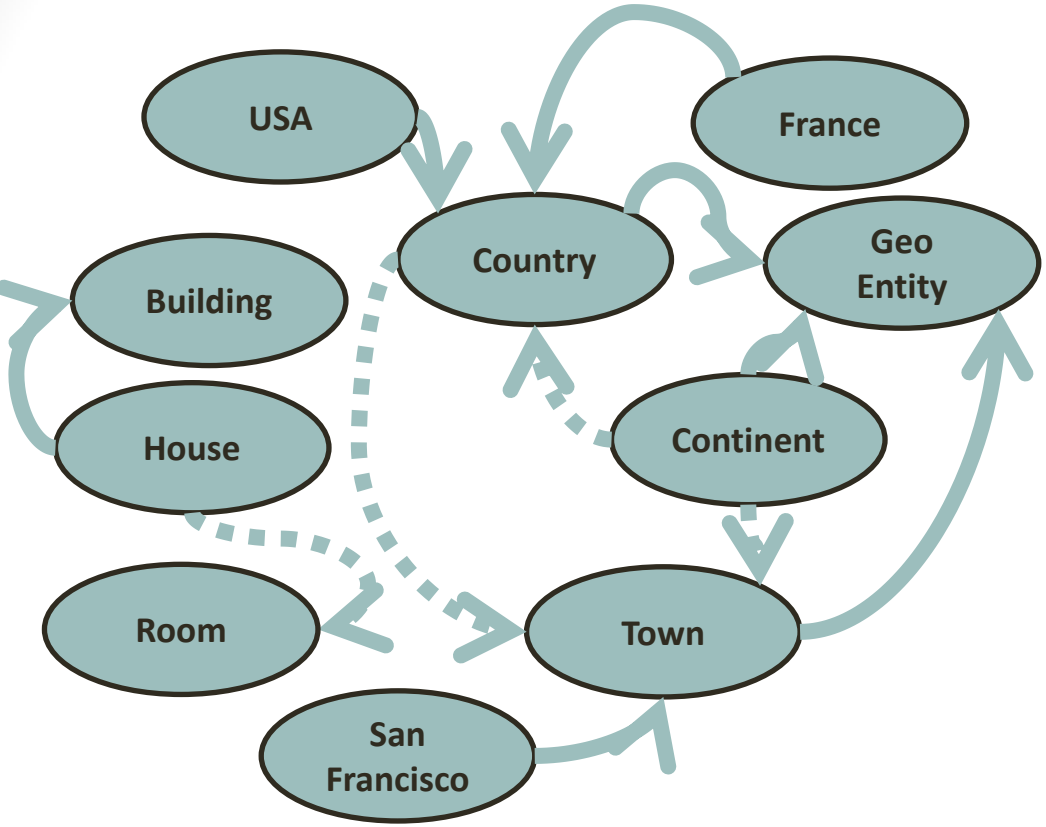
Show relationships from Country to Town

SPARQL: Predicates



Show any node and relationship

SPARQL: Predicates




```
SELECT ?s ?p ?o
FROM <myTripleStore>
WHERE {
?s ?p ?o}
```

Show any node and relationship

SPARQL Examples (0)

- http://dbpedia.org/sparql
- http://lod.openlinksw.com/sparql/

```
#Total number of triples:
SELECT (COUNT(*) AS ?NumberOfTriples)
FROM <http://dbpedia.org>
WHERE
{?s ?p ?o}
```

```
PREFIX property:
<http://dbpedia.org/property/>
SELECT ?s ?club (max(?o) as ?maxGoals)
FROM <http://dbpedia.org>
WHERE
{
?s property:totalgoals ?o .
?s property:currentclub ?club
filter(?o > 10 && ?o < 1000)
}
ORDER BY DESC(?maxGoals) limit 10
```

SPARQL Examples (1)

- http://dbpedia.org/sparql
- http://lod.openlinksw.com/sparql/

```
PREFIX property:
<http://dbpedia.org/property/>
SELECT DISTINCT ?p
FROM <http://dbpedia.org>
WHERE
{
?s property:years  ?o.
?s ?p ?o2
} limit 10
```

```
PREFIX property:
<http://dbpedia.org/property/>
SELECT DISTINCT ?p
FROM <http://dbpedia.org>
WHERE
{
?s property:years  ?o .
?s ?p ?o
} limit 10
```

SPARQL Examples (2)

- http://dbpedia.org/sparql
- http://lod.openlinksw.com/sparql/

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT DISTINCT ?name ?email
WHERE {
?person a foaf:Person .
?person foaf:name ?name .
?person foaf:mbox ?email . }
```

SPARQL Exercise 1

1. You have the following triple store <http://ImaginaryTripleStore.com>:
A X B
C X B
B X D
B Y E
F Y B
F X D
F Y E
E X D
G X E

2. You have the following SPARQL statement:
SELECT ?x
FROM
<http://ImaginaryTripleStore.com>
WHERE
{F ?p ?o .
?o ?p ?x}

3. Solution

3.1 List all items that fulfil "F ?p ?o". Place them below the where clause.

```
F  ?p  ?o  .  ?o  ?p   ?x
F   Y   B
F   Y   D
F   Y   E
```

3.2 List all items that fulfil "?o ?p ?x" where ?o is B, D, or E and ?p is Y. Place them below the where clause.

```
F  ?p  ?o  .  ?o  ?p   ?x
F   Y   B  .  B   Y    E
F   Y   D
F   Y   E
```

3.3 List all items that were placed under "?x"

SPARQL Exercise 2

1. You have the following triple store <http://ImaginaryTripleStore.com>:

A X B
C X B
B X D
B Y E
F Y B
F X D
F Y E
E X D
G X E

2. You have the following SPARQL statement:

```
SELECT ?s
FROM
<http://ImaginaryTripleStore.com>
WHERE
{?s Y ?o .
?o X D}
```

3. Solution

3.1 Look at the 2nd triple from the where clause first. Find all triplets that use X as a predicate where the object is D (fulfils "?o X D"). Place them below the where clause:

```
?s  Y  ?o  .  ?o  X  D
              B   X  D
              F   X  D
              E   X  D
```

3.2 The subject of the above triplets are: B, F, and E. Now, look at the first triplet of the where clause. Look for all triplets that have an object that is B, F, or E and that have a predicate Y (fulfils "?s Y ?o"). Place those triplets below the where clause:

```
?s  Y  ?o  .  ?o  X  D
F   Y   B     B   X  D
              F   X  D
F   Y   E     E   X  D
B   Y   E     E   X  D
```

3.3 List all items that were placed under "?s"

Some Links

- http://dbpedia.org/sparql
- http://lod.openlinksw.com/sparql/
- http://librdf.org/query

- Online tutorial: http://www.cambridgesemantics.com/semantic-university/sparql-by-example
-
- http://sparql.org/sparql.html
- http://demo.openlinksw.com/sparql

Introduction to SPARQL