
Querying Linked Geospatial Data with Incomplete Information

Charalampos Nikolaou
charnik@di.uoa.gr

Manolis Koubarakis
koubarak@di.uoa.gr



Department of Informatics & Telecommunications
National and Kapodistrian, University of Athens



Outline

- Linked geospatial data (motivation)
- Querying complete geospatial information (exact geometries)
- Querying qualitative geospatial information
- Querying incomplete geospatial information
- The RDFⁱ framework
- Future work

Motivation

Linked geospatial data

1. GeoNames
<http://www.geonames.org/ontology/>
2. LinkedGeoData (OpenStreetMap)
<http://linkedgeodata.org/>
3. Administrative geography of Great Britain (Ordnance Survey)
<http://data.ordnancesurvey.co.uk/.html>
4. Greek Administrative Geography
<http://linkedopendata.gr/>
5. Corine Land Cover of Greece
<http://linkedopendata.gr/>
6. Global Administrative Areas (GADM)
<http://www.gadm.org/>
7. DBpedia
<http://dbpedia.org/>

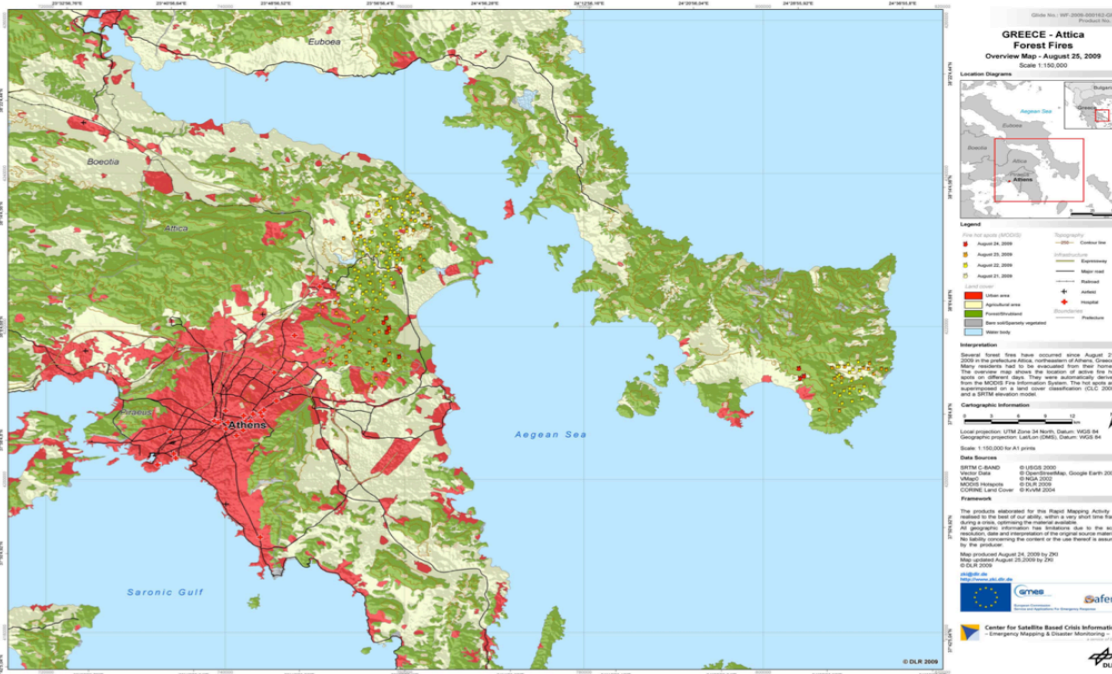
Motivation

over 9.5 million geometries
(points, linestrings, polygons)

Motivation (cont'd)

Exploitation

- Earth observation
 - National Observatory of Athens (NOA)
 - Fire monitoring and burnt scar mapping
 - Risk assessment



Motivation (cont'd)



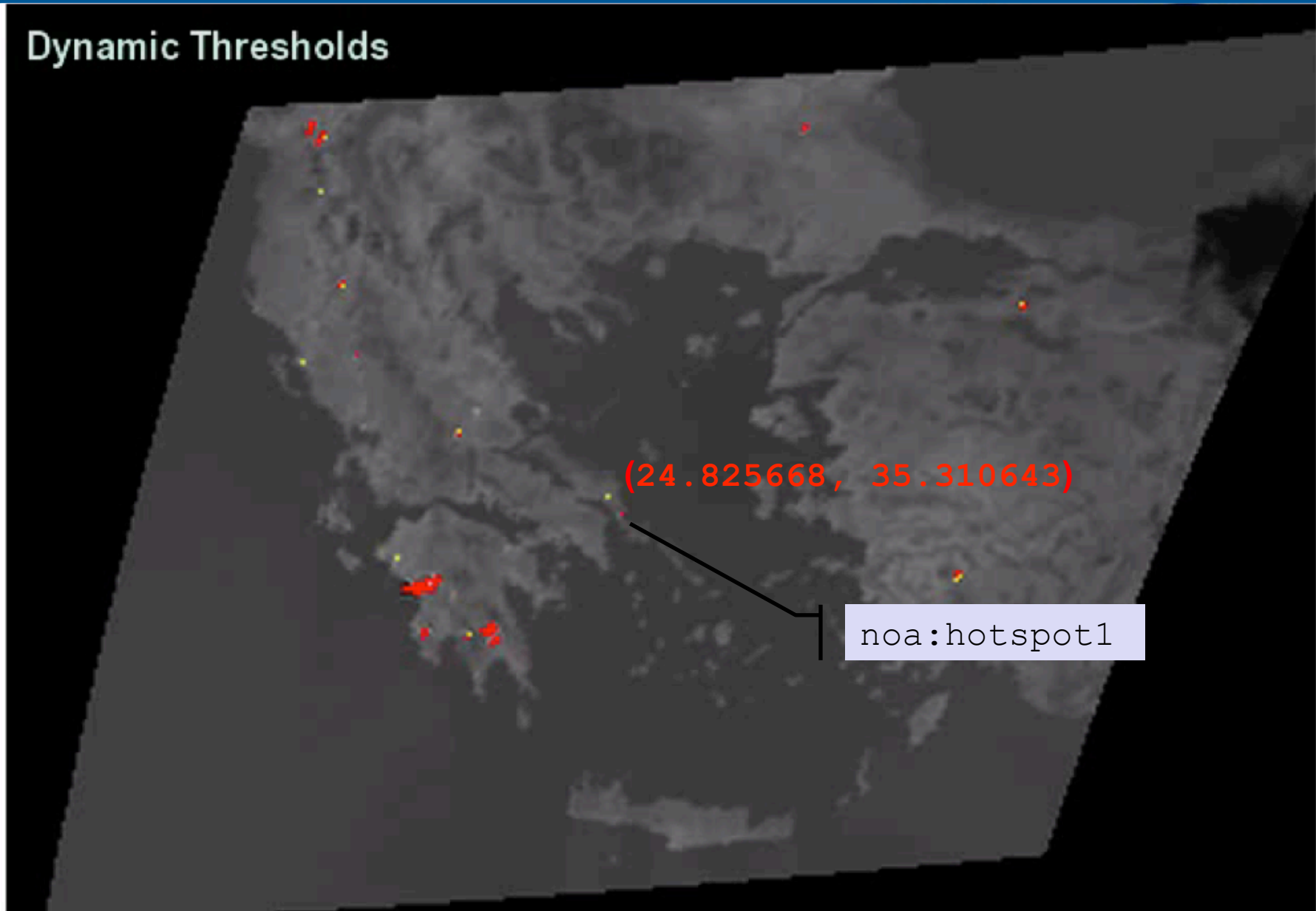
– German Aerospace Center (DLR)

- Management of environmental disasters (oil spills, tsunamis, floods, etc.)
- Land use and regional/urban planning



NOA's representation of hotspots

Dynamic Thresholds



NOA's representation of hotspots (cont'd)

Representation using stRDF

```
noa:hotspot1 rdf:type noa:Hotspot .
```

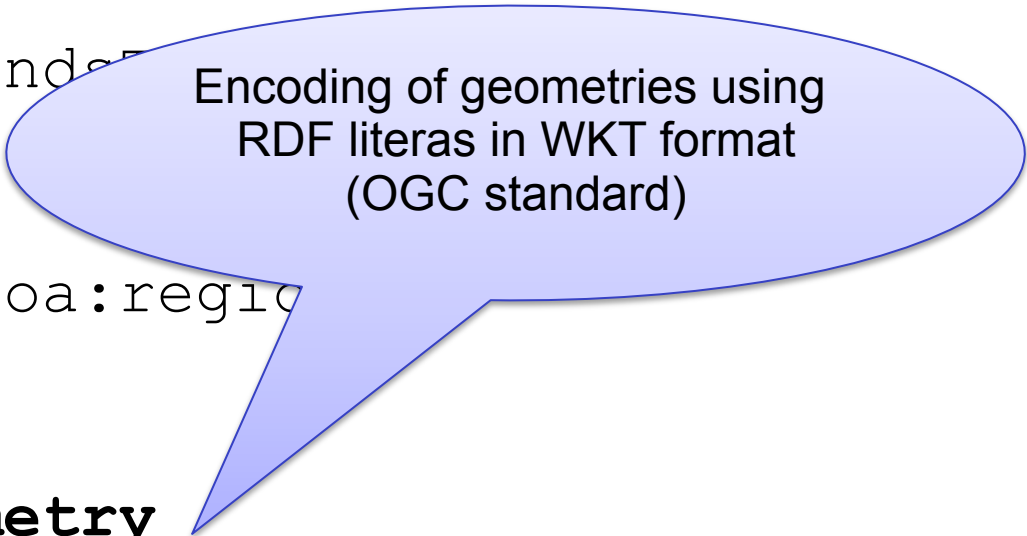
```
noa:fire1 rdf:type noa:Fire .
```

```
noa:hotspot1 noa:corresponds
```

```
noa:fire1 noa:occuredIn noa:region1
```

```
noa:region1 strdf:hasGeometry
```

```
"POINT (24.825668 35.310643) "^^strdf:WKT .
```



Encoding of geometries using
RDF literals in WKT format
(OGC standard)

NOA's representation of hotspots (cont'd)

Querying using stSPARQL

Find all fires and hotspots inside Rethymno

```
SELECT ?f ?h
```

```
WHERE {
```

```
  ?h rdf:type noa:Hotspot ;
```

```
    noa:correspondsTo ?f .
```

```
  ?f rdf:type noa:Fire ;
```

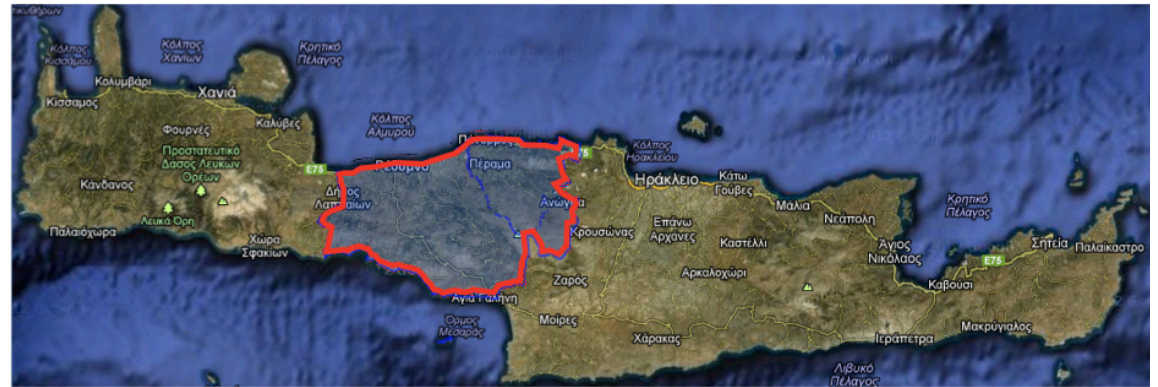
```
    noa:occuredIn ?r .
```

```
  ?r strdf:hasGeometry ?rgeo .
```

```
  gag:Rethymno strdf:hasGeometry ?rethGeo .
```

```
  FILTER (strdf:contains(?rethGeo, ?rgeo))
```

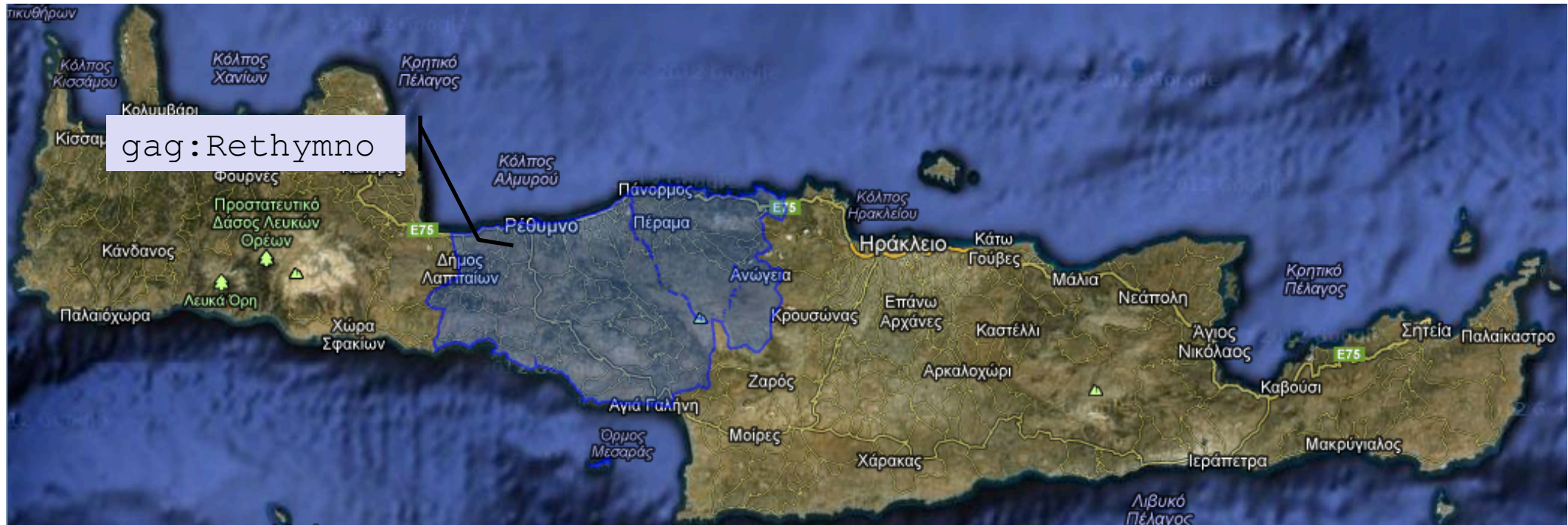
```
}
```



Greek Administrative Geography

Spatial filtering

Extending the previous example with topological information



`gag:Rethymno rdf:type gag:Perfection .`

Extending the previous example with topological information



```
gag:Rethymno rdf:type gag:Perfection .
```

```
gag:Mylopotamos rdf:type gag:Municipality .
```

```
gag:Rethymno geo:sfContains gag:Mylopotamos .
```

```
gag:Mylopotamos geo:sfContains noa:region1 .
```

Topology vocabulary extension of GeoSPARQL

Extending the previous example with topological information (cont'd)

Querying using GeoSPARQL

Find all fires and hotspots inside Rethymno

```
SELECT ?f ?h
```

```
WHERE {
```

```
  ?h rdf:type noa:Hotspot ;
```

```
    noa:correspondsTo ?f .
```

```
  ?f rdf:type noa:Fire ;
```

```
    noa:occuredIn ?r .
```

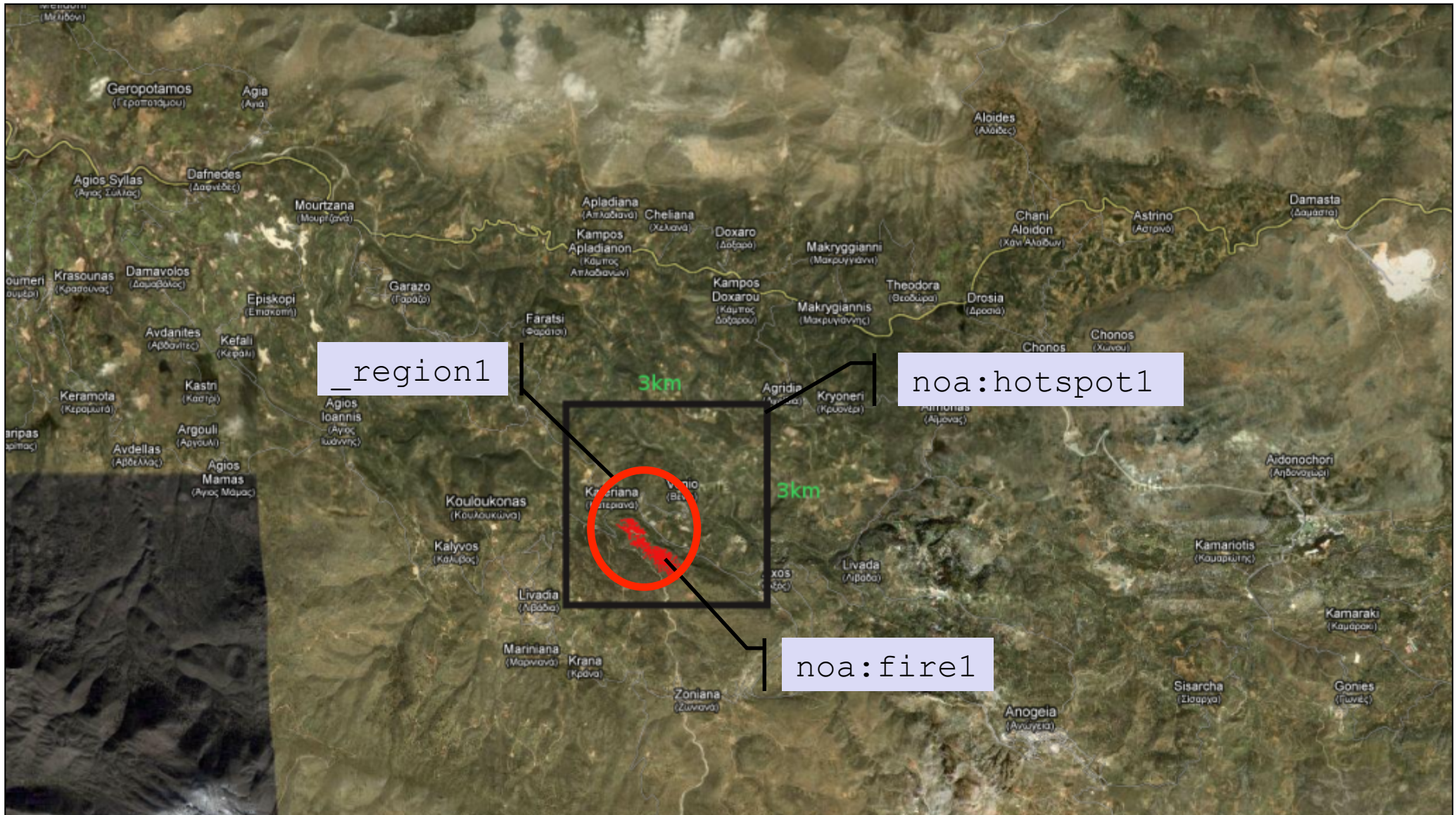
```
  gag:Rethymno geo:sfContains ?r .
```

```
}
```

Topology vocabulary
extension of GeoSPARQL

NOA's representation of hotspots (revisited)

Incomplete information



NOA's representation of hotspots (revisited)

Representation using RDFⁱ

```
noa:hotspot1 rdf:type noa:Hotspot .
```

```
noa:fire1 rdf:type noa:Fire .
```

```
noa:hotspot1 noa:correspondsTo noa:fire1 .
```

```
noa:fire1 noa:occuredIn _region1 .
```

e-literal

RCC-8

```
_region1 geo:sfWithin "POLYGON((24.81 35.32, 24.84 35.33,  
24.84 35.30, 24.81 35.30, 24.81 35.32));<http://  
spatialreference.org/ref/epsg/4121/>"^^strdf:geometry
```

Qualitative **spatial** constraint

Visualization of a certainty query



Certainty queries

- Find all fires that have **certainly** occurred inside the rectangle defined in WKT as `POLYGON((24.79 35.34, 24.85 35.34, 24.85 35.29, 24.79 35.29, 24.79 35.34))`

New (modal) operator

```
CERTAIN SELECT ?F
```

```
WHERE {
```

```
  ?F rdf:type noa:Fire ;
```

```
  noa:occuredIn ?R .
```

```
  FILTER(geof:sfWithin(?R, "POLYGON((24.79 35.34, 24.85  
35.34, 24.85 35.29, 24.79 35.29, 24.79 35.34))") )
```

```
}
```

New topological operator

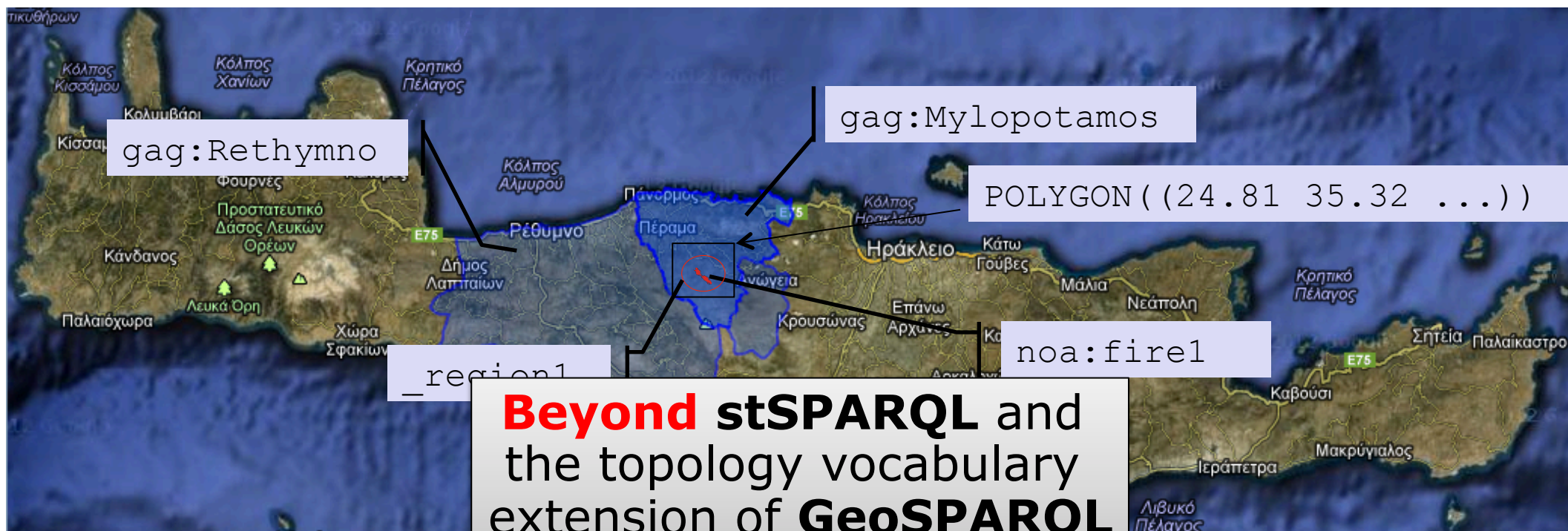
Extending the previous example with topological information



```

gag:Rethymno rdf:type gag:Perfecture .
gag:Mylopotamos rdf:type gag:Municipality .
gag:Mylopotamos geo:sfWithin gag:Rethymno .
_region1 geo:sfWithin "POLYGON((24.81 35.32...))"^^strdf:WKT
  
```

Extending the previous example with topological information



Beyond stSPARQL and the topology vocabulary extension of GeoSPARQL

```
gag:Rethymno rdf:type gag:Perfecture .
```

```
gag:Mylopotamos rdf:type gag:Municipality .
```

```
gag:Mylopotamos geo:sfWithin gag:Rethymno .
```

```
_region1 geo:sfWithin "POLYGON((24.81 35.32...))"^^strdf:WKT
```

Certainty queries

Find all fires that have **certainly** occurred inside Rethymno

```
CERTAIN SELECT ?F
WHERE {
    ?F rdf:type noa:Fire ;
        noa:occuredIn ?R .
    FILTER(geof:sfWithin(?R, gag:Rethymno))
}
```

Computing the answer

```
gag:Mylopotamos geo:sfWithin gag:Rethymno .  
gag:Mylopatamos strdf:hasGeometry "POLYGON((<M>))"^^strdf:WKT .  
gag:Rethymno strdf:hasGeometry "POLYGON((<R>))"^^strdf:WKT .  
_region1 geo:sfWithin "POLYGON((<H>))"^^strdf:WKT
```

D
A
T
A
B
A
S
E

Geometry for
Mylopotamos

Geometry for
Rethymno

Geometry for
Hotspot

Computing the answer

```
gag:Mylopotamos geo:sfWithin gag:Rethymno .  
gag:Mylopatamos strdf:hasGeometry "POLYGON((<M>))"^^strdf:WKT .  
gag:Rethymno strdf:hasGeometry "POLYGON((<R>))"^^strdf:WKT .  
_region1 geo:sfWithin "POLYGON((<H>))"^^strdf:WKT
```

DATA
BASE

Geometry for
Mylopotamos

Geometry for
Rethymno

Geometry for
Hotspot

```
strdf:Inside ("POLYGON ( (<H> ) ) ", "POLYGON ( (<M> ) ) ")
```

```
"POLYGON ( (<H> ) ) " geo:sfWithin "POLYGON ( (<M> ) ) "
```

Vocabulary
translation

Computing the answer

```
gag:Mylopotamos geo:sfWithin gag:Rethymno .  
gag:Mylopatamos strdf:hasGeometry "POLYGON((<M>))"^^strdf:WKT .  
gag:Rethymno strdf:hasGeometry "POLYGON((<R>))"^^strdf:WKT .  
_region1 geo:sfWithin "POLYGON((<H>))"^^strdf:WKT
```

DATA
BASE

Geometry for
Mylopotamos

Geometry for
Rethymno

Geometry for
Hotspot

```
strdf:Inside ("POLYGON ( (<H> ) ) ", "POLYGON ( (<M> ) ) ")  
"POLYGON ( (<H> ) ) " geo:sfWithin "POLYGON ( (<M> ) ) "
```

Vocabulary
translation



```
_region1 geo:sfWithin "POLYGON ( (<H> ) ) "  
_region1 geo:sfWithin gag:Mylopotamos
```

Qualitative
spatial reasoning

Computing the answer

```
gag:Mylopotamos geo:sfWithin gag:Rethymno .  
gag:Mylopatamos strdf:hasGeometry "POLYGON((<M>))"^^strdf:WKT .  
gag:Rethymno strdf:hasGeometry "POLYGON((<R>))"^^strdf:WKT .  
_region1 geo:sfWithin "POLYGON((<H>))"^^strdf:WKT
```

DATA
BASE

Geometry for
Mylopotamos

Geometry for
Rethymno

Geometry for
Hotspot

```
strdf:Inside ("POLYGON((<H>))", "POLYGON((<M>))")  
"POLYGON((<H>))" geo:sfWithin "POLYGON((<M>))"
```

Vocabulary
translation



```
_region1 geo:sfWithin "POLYGON((<H>))"  
_region1 geo:sfWithin gag:Mylopotamos
```

Qualitative
spatial reasoning



```
gag:Mylopotamos geo:sfWithin gag:Rethymno  
_region1 geo:sfWithin gag:Rethymno
```

Combined
algorithm

The Framework RDFⁱ

- Extension of RDF with **incomplete information**
- New kind of literals (**e-literals**) for each datatype
 - Property values that *exist* but are *unknown* or *partially known*
- **Partial knowledge**: captured by constraints
(appropriate constraint language L)
- RDF graphs **extended** to RDFⁱ databases: pair **(G, φ)**
 - G : RDF graph with e-literals
 - φ : quantifier-free formula of L

RDFⁱ Semantics

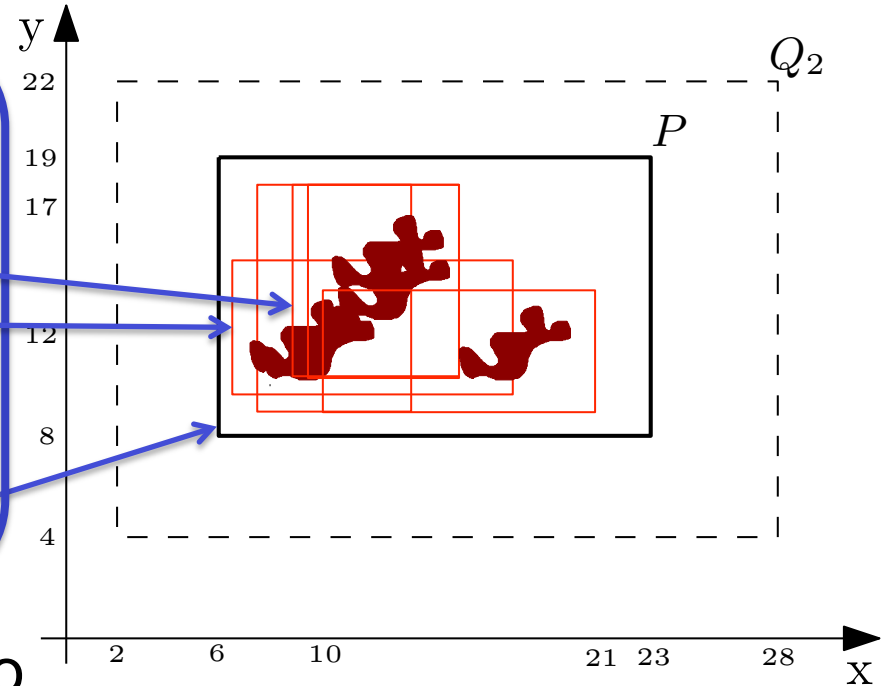
hotspot1 type Hotspot .

fire1 type Fire .

hotspot1 correspondsTo fire1 .

fire1 occurredIn _region1 .

_region1 geo:sfWithin "x ≥ 6 ∧ x ≤ 23 ∧ y ≥ 8 ∧ y ≤ 19"



corresponds to

$\{ G_1, G_2, G_3, G_4, \dots \}$

hotspot1 type Hotspot .

fire1 type Fire .

hotspot1 correspondsTo fire1 .

fire1 occurredIn "x ≥ 8 ∧ x ≤ 14 ∧ y ≥ 10 ∧ y ≤ 18" .

set of RDF graphs
(possible worlds)

Certain answers

```
CERTAIN SELECT ?F
```

```
WHERE {
```

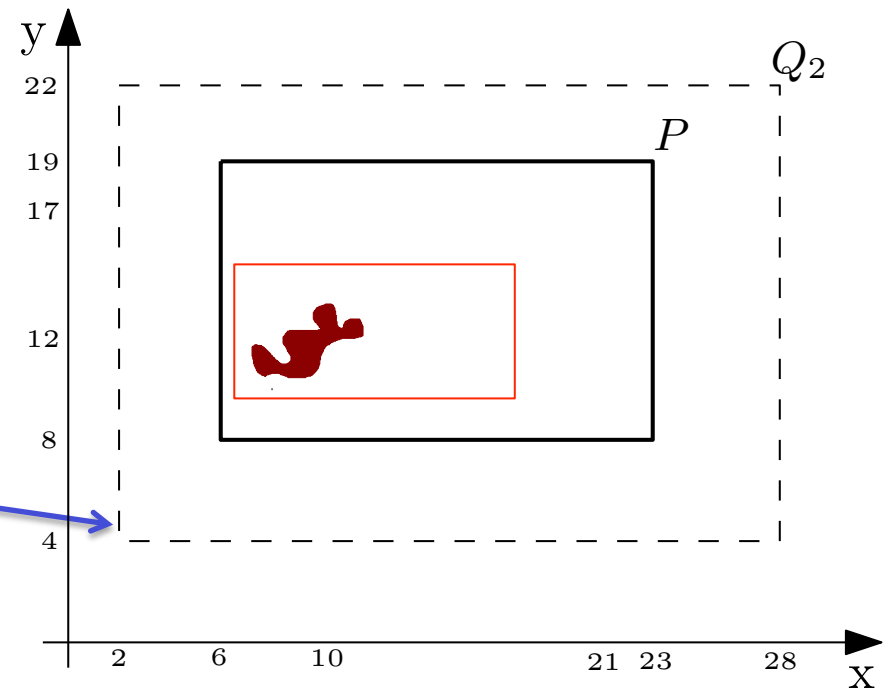
```
  ?F rdf:type noa:Fire ;
```

```
  noa:occuredIn ?R .
```

```
  FILTER(geof:sfWithin(?R,
```

```
    "x ≥ 2 ∧ x ≤ 28 ∧ y ≥ 4 ∧ y ≤ 22")
```

```
  }
```



$$\text{Cert}(q) = q(G_1) \cap q(G_2) \cap q(G_3) \cap q(G_4) \cap \dots$$

Certain answers

```
CERTAIN SELECT ?F
```

```
WHERE {
```

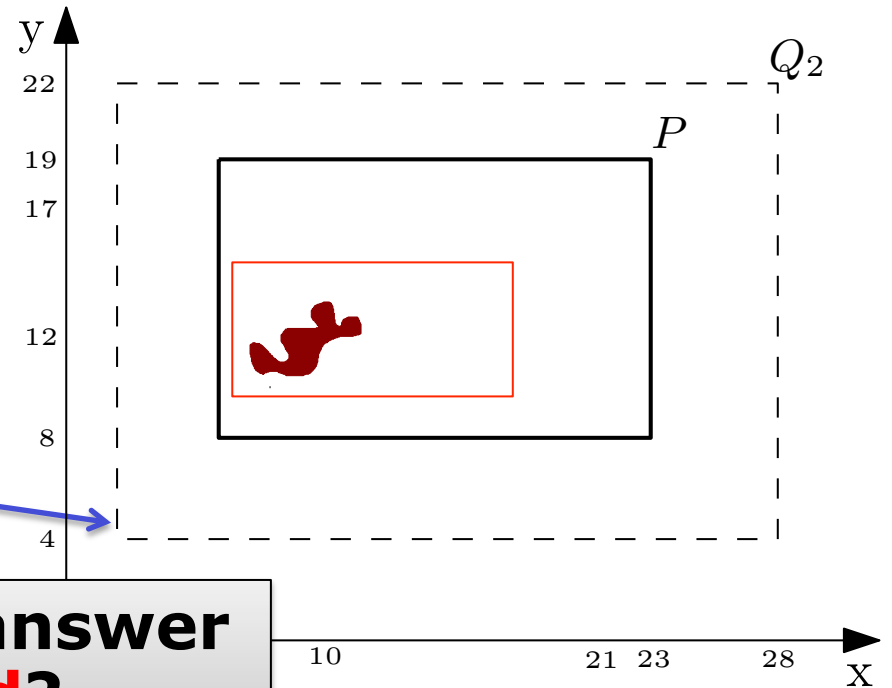
```
  ?F rdf:type noa:Fire ;
```

```
  noa:occuredIn ?R .
```

```
  FILTER (geof:sfWithin (?R,
```

```
    "x ≥ 2 ∧ x ≤ 28 ∧ y ≥ 4 ∧ y ≤ 22" )
```

```
}
```



How the **certain answer**
is **computed**?

$$\text{Cert}(q) = q(G_1) \cap q(G_2) \cap q(G_3) \cap q(G_4) \cap \dots$$

The Framework RDFⁱ (cont'd)

- Formal semantics for RDFⁱ and SPARQL query evaluation
- **Representation Systems:**
 - CONSTRUCT with AUF graph patterns
 - CONSTRUCT with well-designed graph patterns
- **Certain Answer:** semantics, algorithms, computational complexity when L is a language of spatial topological constraints
- Implementation in the system **Strabon** has started with L being PCL (topological constraints between variables and polygon constants)

Charalampos Nikolaou and Manolis Koubarakis
Incomplete Information in RDF
arXiv:1209.3756v2 [cs.DB] 18 Sep 2012
<http://arxiv.org/pdf/1209.3756v2.pdf>

Future work

- **How** do we implement **querying with topological relations** in RDF stores for stSPARQL/GeoSPARQL?
 - **How** do we implement **certainty queries** for RDFⁱ?
 - DL reasoners with RCC-8 support **offer topological reasoning already** (implementing a path-consistency algorithm)
 - RacerPro [Möller et al.], [Wessel-Möller, JAPLL'09]
 - PelletSpatial [Stocker-Sirin, OWLED'09]
 - RDFⁱ goes **beyond**
 - Reason about **qualitative and quantitative** geospatial information
 - Can be used in other application domains (e.g., temporal)
-

Thank you for your attention!

Questions?

References

[Weiming Liu et al.]

Weiming Liu, Sheng-sheng Wang, Sanjiang Li, Dayou Liu: *Solving Qualitative Constraints Involving Landmarks*. CP 2011:523-537

[Wessel-Möller, JAPLL'09]

Michael Wessel, Ralf Möller: *Flexible software architectures for ontology-based information systems*. J. Applied Logic (JAPLL) 7(1): 75-99 (2009)

[Stocker-Sirin, OWLED'09]

Markus Stocker, Evren Sirin: *PelletSpatial: A Hybrid RCC-8 and RDF/OWL Reasoning and Query Engine*. OWLED 2009
