Discovering Spatial and Temporal Links among RDF Data

Panayiotis Smeros and Manolis Koubarakis

HELLENIC REPUBLIC
National and Kapodistrian University of Athens

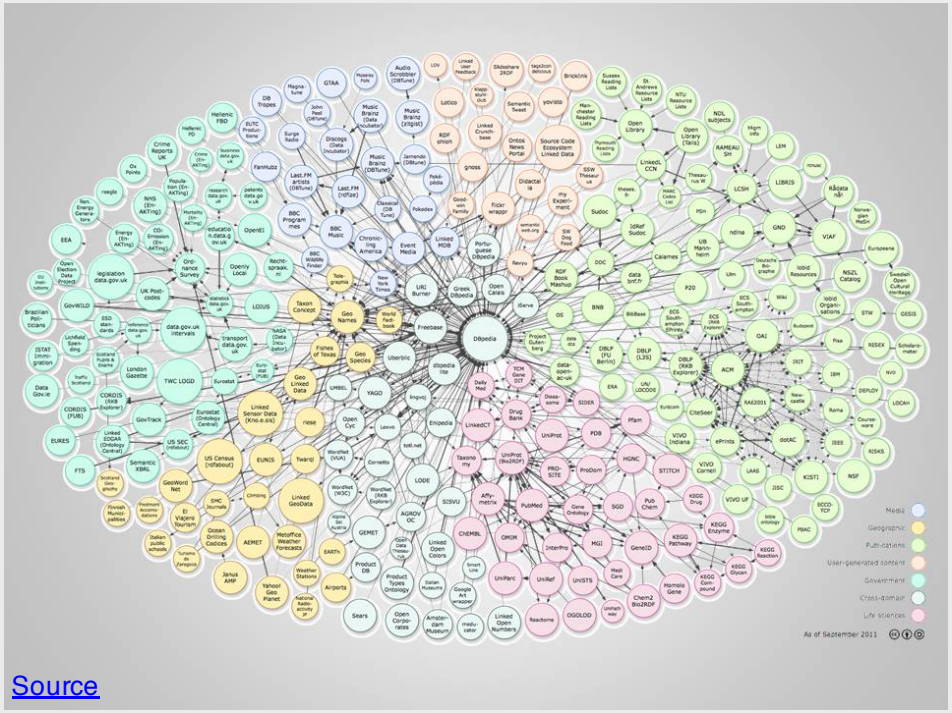
WWW2016 Workshop: Linked Data on the Web (LDOW2016)
April 12, 2016 - Montréal, Canada

Outline

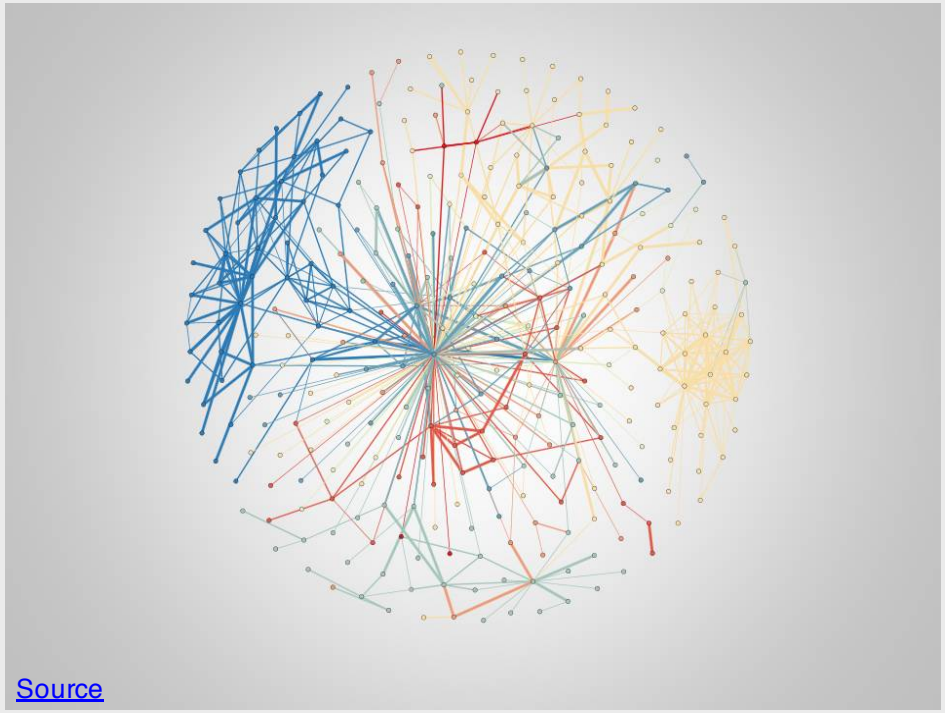
- **Introduction**
- Background
- Developed Methods
- Implementation
- Experimental Evaluation
- Conclusions

Spatial and Temporal Link Discovery

Establish **semantic relations** (links) between entities



Source



Source

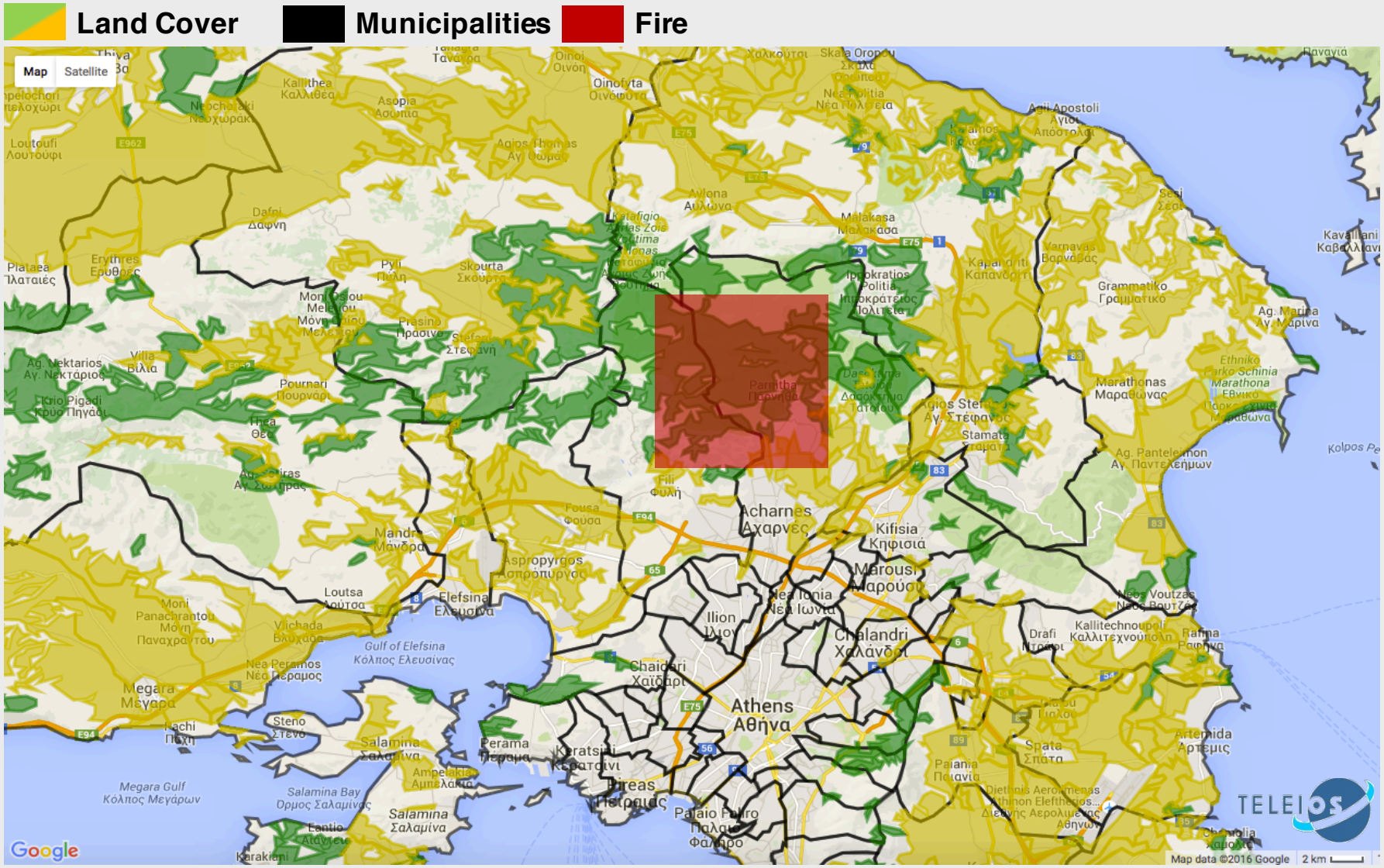
Enrich the information of datasets with Geospatial and Temporal characteristics

From Locations to Complex Geometries

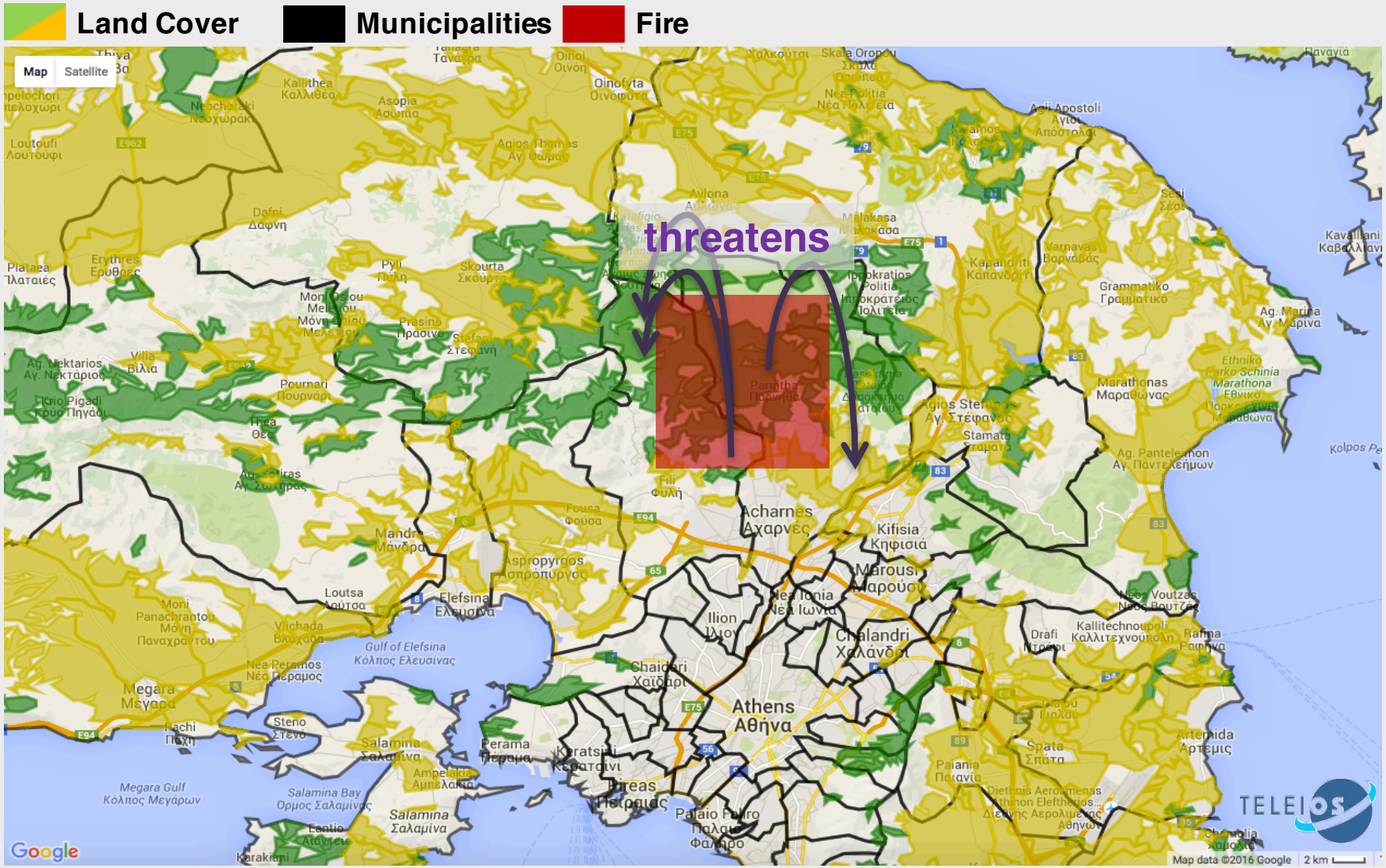
- Geonames, OpenStreetMap, etc. are dominated by **location** (point) information
- GeoSPARQL Standard
- Datasets with **rich geospatial** and **temporal** information
  - Corine Land Cover (http://datahub.io/dataset/corine-land-cover)
  - Urban Atlas (http://datahub.io/dataset/urban-atlas)
  - Products from Satellite Images (http://datahub.io/dataset/sentinel2)
- State-of-the-art works focus on **distance based** (similarity) relations

More spatial and temporal relations can be discovered!

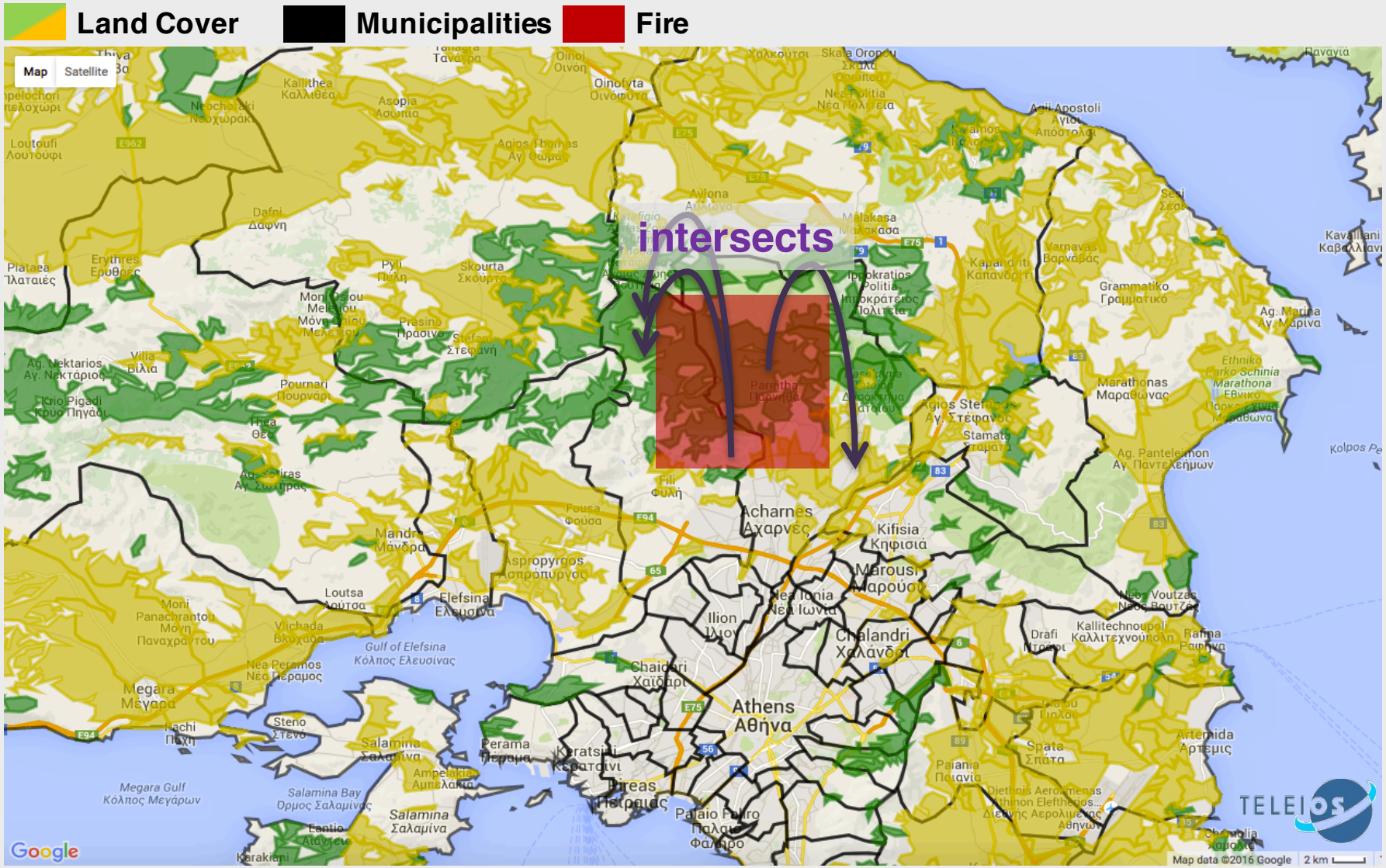
Link Discovery in Fire Monitoring (Example)



Link Discovery in Fire Monitoring (Example)



Link Discovery in Fire Monitoring (Example)



Heterogeneity: Geospatial Datasets

GeoSPARQL

```
_:1 rdf:type geo:Geometry .
_:1 geo:hasGeometry
"<http://www.opengis.net/def/crs/EPSG/0/4326>
POINT(10 20)"^^geo:wktLiteral .
```

StRDF

```
_:1 rdf:type strdf:Geometry .
_:1 strdf:hasGeometry
"<gml:Point crsName="EPSG:2100"><gml:coordinates>10,20
</gml:coordinates></gml:Point>"^^strdf:GML .
```

W3C GEO

```
_:1 rdf:type wgs84Geo:Point .
_:1 wgs84Geo:lat "10"^^xsd:double .
_:1 wgs84Geo:long "20"^^xsd:double .
```

Heterogeneity: Geospatial Datasets

```
_:1 rdf:type geo:Geometry .
_:1 geo:hasGeometry
"<http://www.opengis.net/def/crs/EPSG/0/4326>
POINT(10 20)"^^geo:wktLiteral .
```

GeoSPARQL

```
_:1 rdf:type strdf:Geometry .
_:1 strdf:hasGeometry
"<gml:Point crsName="EPSG:2100"><gml:coordinates>10,20
</gml:coordinates></gml:Point>"^^strdf:GML .
```

stRDF

- ***Different Vocabularies***

Heterogeneity: Geospatial Datasets

```
_:1 rdf:type geo:Geometry .
_:1 geo:hasGeometry
"<http://www.opengis.net/def/crs/EPSG/0/4326>
POINT(10 20)"^^geo:wktLiteral .
```

GeoSPARQL

```
_:1 rdf:type strdf:Geometry .
_:1 strdf:hasGeometry
"<gml:Point crsName="EPSG:2100"><gml:coordinates>10,20
</gml:coordinates></gml:Point>"^^strdf:GML .
```

stRDF

- ***Different Vocabularies***
- ***Different Serializations of Geometries***

Heterogeneity: Geospatial Datasets

```
_:1 rdf:type geo:Geometry .
_:1 geo:hasGeometry
"<http://www.opengis.net/def/crs/EPSG/0/4326>
POINT(10 20)"^^geo:wktLiteral .
```

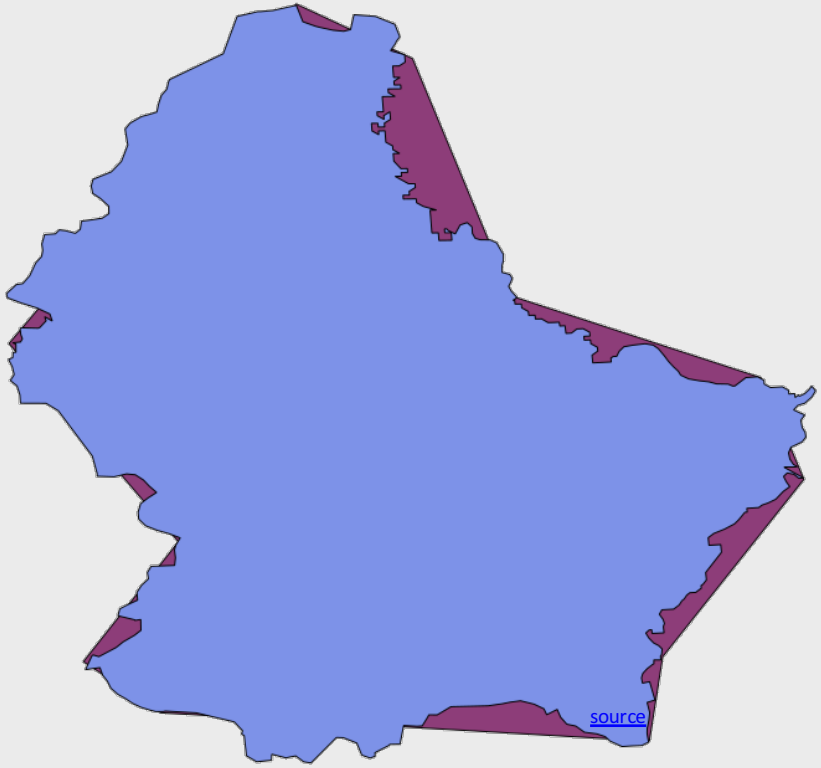
GeoSPARQL

```
_:1 rdf:type strdf:Geometry .
_:1 strdf:hasGeometry
"<gml:Point crsName="EPSG:2100"><gml:coordinates>10,20
</gml:coordinates></gml:Point>"^^strdf:GML .
```

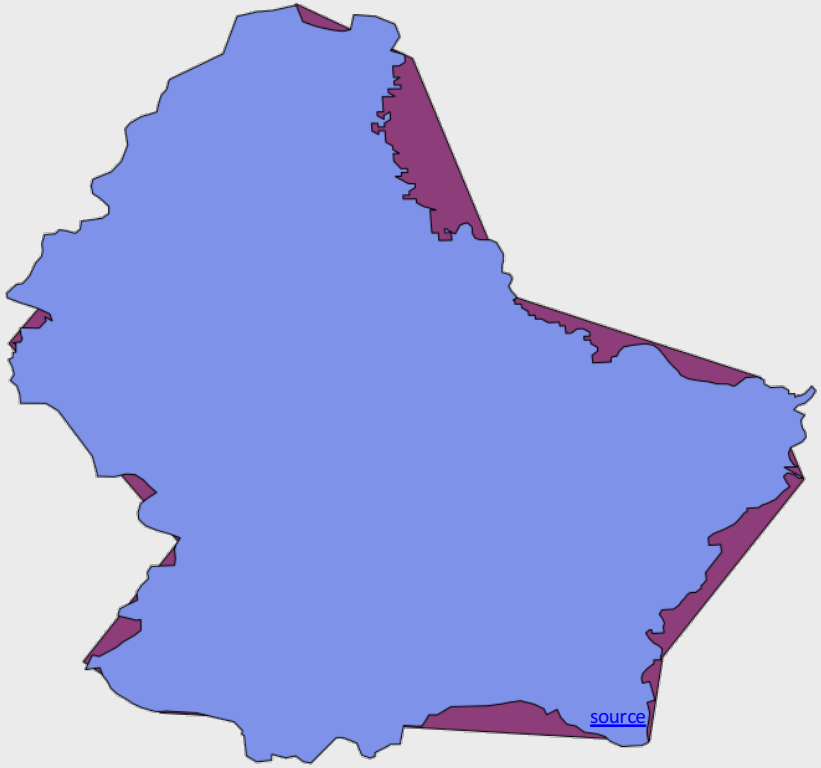
stRDF

- ***Different Vocabularies***
- ***Different Serializations of Geometries***
- ***Geometries expressed in Different Coordinate Reference Systems (CRS)***

Heterogeneity: Geospatial Datasets



Heterogeneity: Geospatial Datasets



- ***Different Sampling Values***
- ***Different Granularity***
- ***Different Rounding Effects***

Heterogeneity: Temporal Datasets

```
_:1 ex:hasBirthday "1989-09-
24T11:05:00+01:00"xsd:dateTime .
```

RDF

```
_:1 ex:hasAffiliation ex:UoA
"[2007-09-01T00:00:00+03:00,
  2015-08-31T00:00:00+04:00)"^^strdf:Period .
```

StRDF

Heterogeneity: Temporal Datasets

```
_:1 ex:hasBirthday "1989-09-
24T11:05:00+01:00"xsd:dateTime .
```

RDF

```
_:1 ex:hasAffiliation ex:UoA
"[2007-09-01T00:00:00+03:00,
  2015-08-31T00:00:00+04:00)"^^strdf:Period .
```

StRDF

- ***Different Vocabularies***

Heterogeneity: Temporal Datasets

```
_:1 ex:hasBirthday "1989-09-
24T11:05:00+01:00"^^xsd:dateTime .
```

RDF

```
_:1 ex:hasAffiliation ex:UoA
"[2007-09-01T00:00:00+03:00,
  2015-08-31T00:00:00+04:00)"^^strdf:Period .
```

StRDF

- ***Different Vocabularies***
- ***Different Time Zones***

Heterogeneity: Temporal Datasets

```
_:1 ex:hasBirthday "1989-09-
24T11:05:00+01:00"xsd:dateTime .
```

RDF

```
_:1 ex:hasAffiliation ex:UoA
"[2007-09-01T00:00:00+03:00,
  2015-08-31T00:00:00+04:00)"^^strdf:Period
```

StRDF

- ***Different Vocabularies***
- ***Different Time Zones***
- ***Time Instants and Periods***

Outline

- Introduction
- **Background**
- Developed Methods
- Implementation
- Experimental Evaluation
- Conclusions

Link Discovery (Definition)

Let S and T be two sets of entities and R the set of relations that can be discovered between entities. For a relation $r \in R$, w.l.o.g., we define a distance function d_r and a distance threshold θ_{d_r} as follows:

$$d_r : \mathrm{S} \times \mathrm{T} \rightarrow [0,1] \ , \ \theta_{d_r} \in [0,1]$$

We define the set of discovered links for relation r (DL_r) as follows:

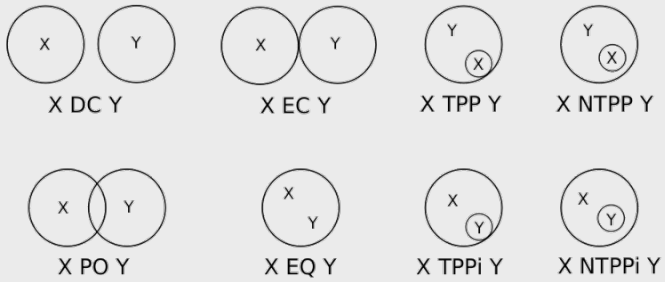
$$DL_r = \{(\mathrm{s}, \mathrm{r}, \mathrm{t}) \mid s \in S \wedge t \in T \wedge d_r(s,t) \leq \theta_{d_r}\}$$

State-of-the-art Spatial Relations

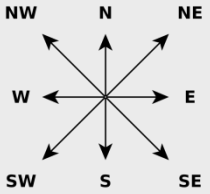
- Dimensionally Extended 9-Intersection Model
- Egenhofer's Model
- OGC Simple Features Model

Intersects, Overlaps, Equals, Touches, Disjoint, Contains, Crosses, Covers, CoveredBy and Within

- Region Connection Calculus
  - e.g., RCC8



- Cardinal Direction Calculus



State-of-the-art Temporal Relations

- Allen's Interval Calculus

Relation	Illustration
X before Y Y after X	X _____ Y _____
X meets Y Y isMetBy X	X _____ Y _____
X overlaps Y Y isOverlappedBy X	X _____ Y _____
X starts Y Y isStartedBy X	X _____ Y _____
X during Y Y contains X	X ___ Y _____
X finishes Y Y isFinishedBy X	X _____ Y _____
X equals Y	X _____ Y _____

Outline

- Introduction
- Background
- **Developed Methods**
- Implementation
- Experimental Evaluation
- Conclusions

Introduced Relations

- Spatial (R_s), Temporal (R_t), Spatiotemporal (R_{st}) relations
- Subsets of Boolean relations (R_B)

$$R_s, R_t, R_{st} \subset R_B \subset R$$

- R_B constitutes a special subset of R. The distance function d_r and the distance threshold θ_{d_r} for a relation $r \in R_B$ are defined as follows:

$$d_r(\mathrm{s,t}) = \begin{cases} 0 & if\ \ r\ holds \\ 1 & elsewhere \end{cases}, \quad \theta_{d_r} = 0$$

Introduced Transformations (1/2)

- **Vocabulary** Transformation
  - converts the vocabulary of geometry literals into GeoSPARQL
- **Serialization** Transformation
  - converts the serialization of geometries into WKT
- **CRS** Transformation
  - converts the CRS of geometries into the World Geodetic System (WGS 84)
- **Validation** Transformation
  - converts not valid geometries (e.g., self-intersecting polygons) to valid ones
- **Simplification** Transformation
  - simplifies geometries according to a given distance tolerance

Introduced Transformations (2/2)

- **Envelope** Transformation
  - computes the envelope (minimum bounding rectangle) of geometries
- **Area** Transformation
  - computes the area of geometries in square metres
- **Points-To-Centroid** Transformation
  - computes the centroid of a cluster of points
- **Time-Zone** Transformation
  - converts the time zone of time elements to Coordinated Universal Time (UTC)
- **Period** Transformation
  - converts time instants to periods with the same starting and ending point

Techniques for Checking the Relations

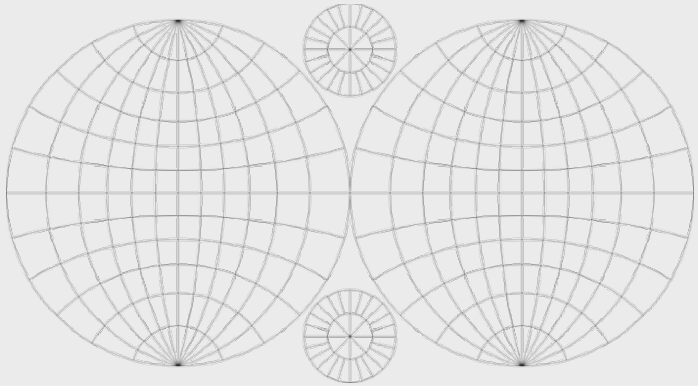
- **Cartesian Product** Technique (Naive)
  - Exhaustive checks between the pairs of the entities of datasets
  - Complete
  - Complexity: O(|S||T|) checks

- **Blocking** Technique
  - Decreases the number of checks
  - Divides the entities into blocks
  - Complexity: O(|S||T|) checks (worst case), O(|L|) checks (best case)

* |S|, |T|: number of entities in datasets S and T; |L|: number of links between datasets S and T

Blocking Technique (algorithm)

1. **Divide** the surface of the earth into **curved rectangles** / the time into **intervals** (blocks)



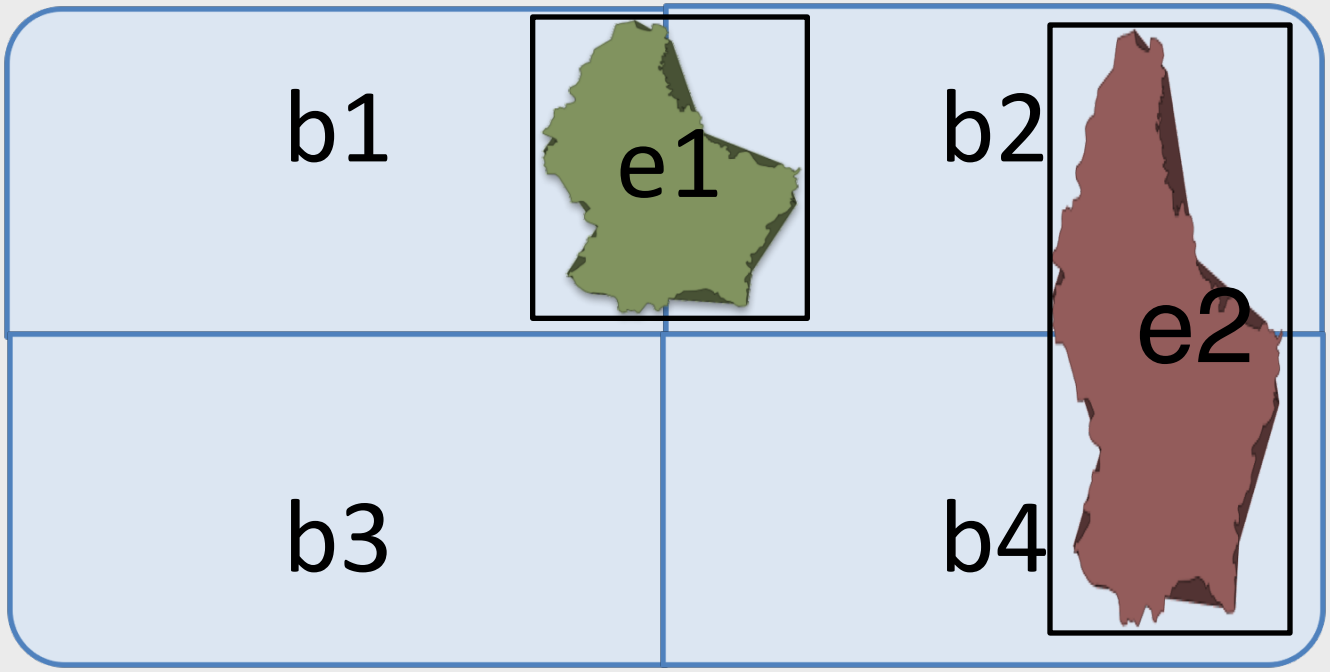
2. **Adjust** the size of the blocks with a **blocking factor** (sbf or tbf)

3. **Insert** the entities into the corresponding blocks

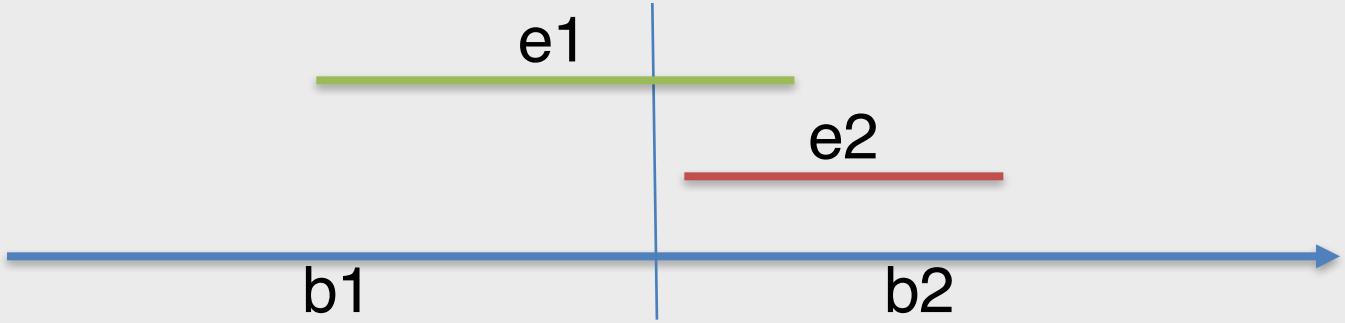
4. **Check** for the actual relation within each block

5. **Aggregate** the links from all the blocks to construct DL_r

Blocking Technique (algorithm)



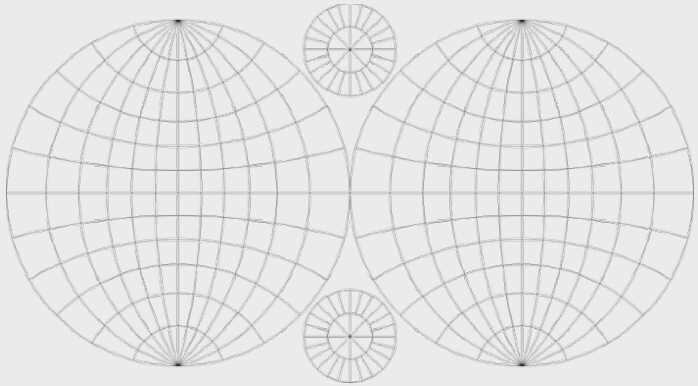
e1: b1, b2
e2: b2, b4



e1: b1, b2
e2: b2

Blocking Technique (algorithm)

1. **Divide** the surface of the earth into **curved rectangles** / the time into **intervals** (blocks)



2. **Adjust** the size of the blocks with a **blocking factor** (sbf or tbf)

3. **Insert** the entities into the corresponding blocks

4. **Check** for the actual relation within each block

5. **Aggregate** the links from all the blocks to construct DL_r

Blocking Technique (accuracy)

- **Sound** and **complete**

- $Precision = \frac{TDL}{TDL+FDL} = \frac{TDL}{TDL} = 100\%$
- $Recall = \frac{TDL}{TDL+FNDL} = \frac{TDL}{TDL} = 100\%$

TDL: True Discovered Links

FDL: False Discovered Links

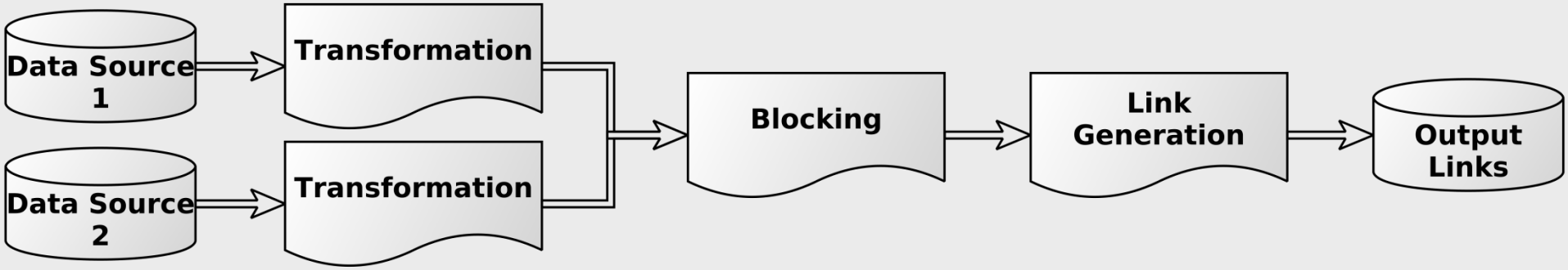
FNDL: False Not Discovered Links

Guaranteed **100% accurate links**

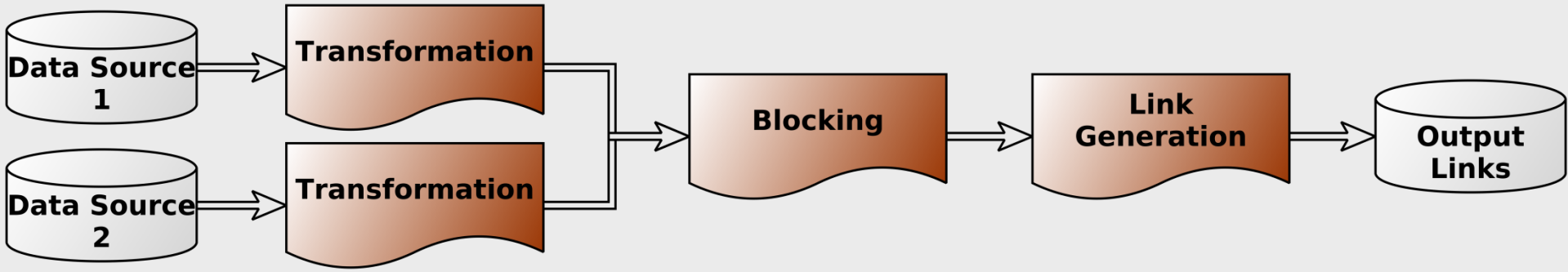
Outline

- Introduction
- Background
- Developed Methods
- **Implementation**
- Experimental Evaluation
- Conclusions

Extensions to the Silk Framework



Extensions to the Silk Framework



- Implemented as **Plugins**
- **Transparent** to all the applications of Silk (Single Machine, MapReduce and Workbench)
- Included in the the **default** Silk distribution (from release 2.6.1 and above)
- https://github.com/silk-framework/silk

Outline

- Introduction
- Background
- Developed Methods
- Implementation
- **Experimental Evaluation**
- Conclusions

Real-world Scenario (Fire Monitoring)

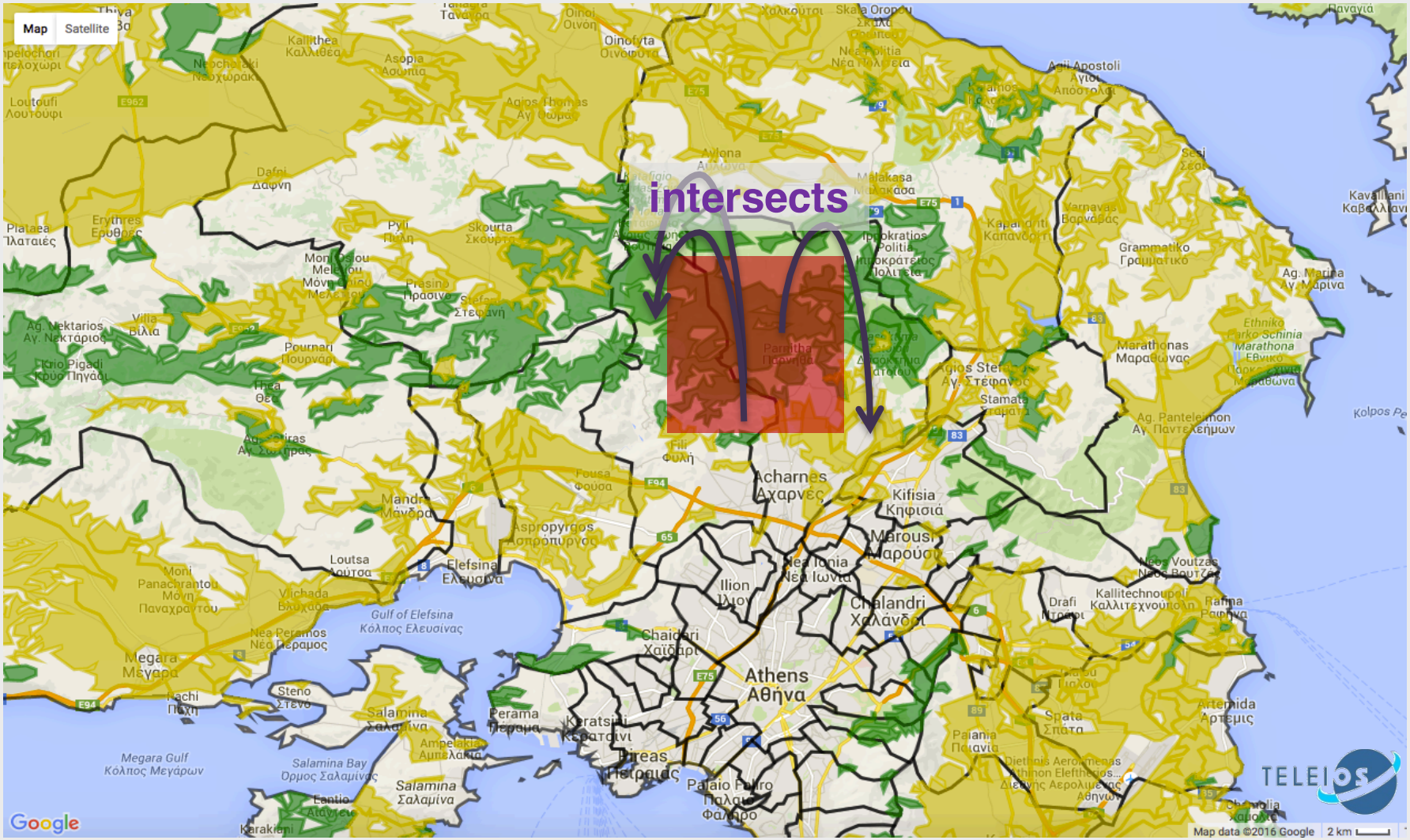
- Which fires (hotspots) threaten forests?
- Which municipalities are threatened by fires?

Dataset	#Entities	Geometries		Time Elements	
		Type	#Points	Type	#Instants
Municipalities from Greek Administrative Geography (GAG)	325	Polygons	979,929	Periods	650
Forests from CORINE Land Cover of Greece (CLCG)	4,868	Polygons	8,004,058	Periods	9,736
Hotspots of Greece (HG)	37,048	Polygons	148,192	Instants	37,048

- Using **Silk**: Discover the relation **intersects** between **HG-GAG** and **HG-CLCG**

Real-world Scenario (Fire Monitoring)

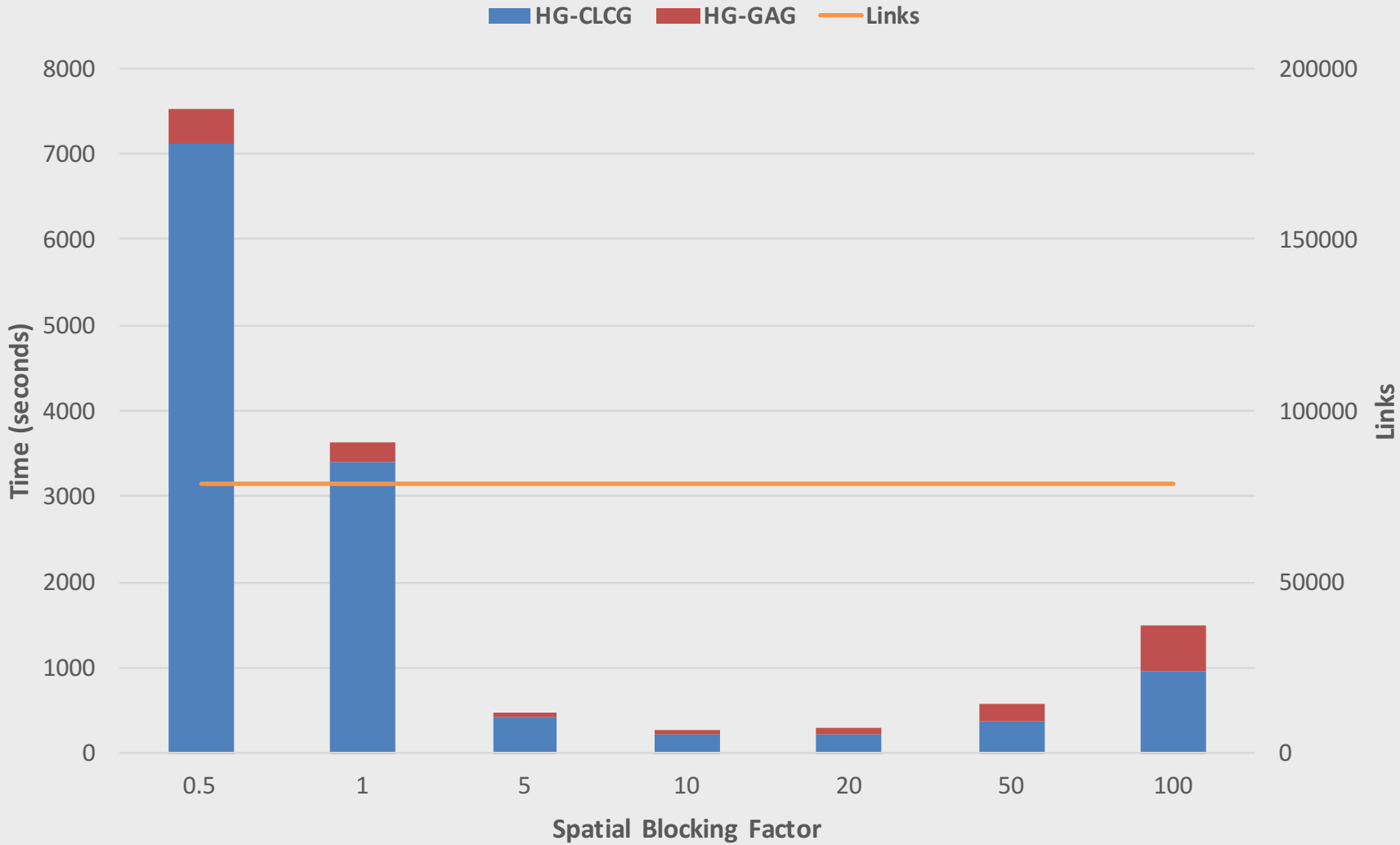
Land Cover (CLCG) Municipalities (GAG) Fire (HG)



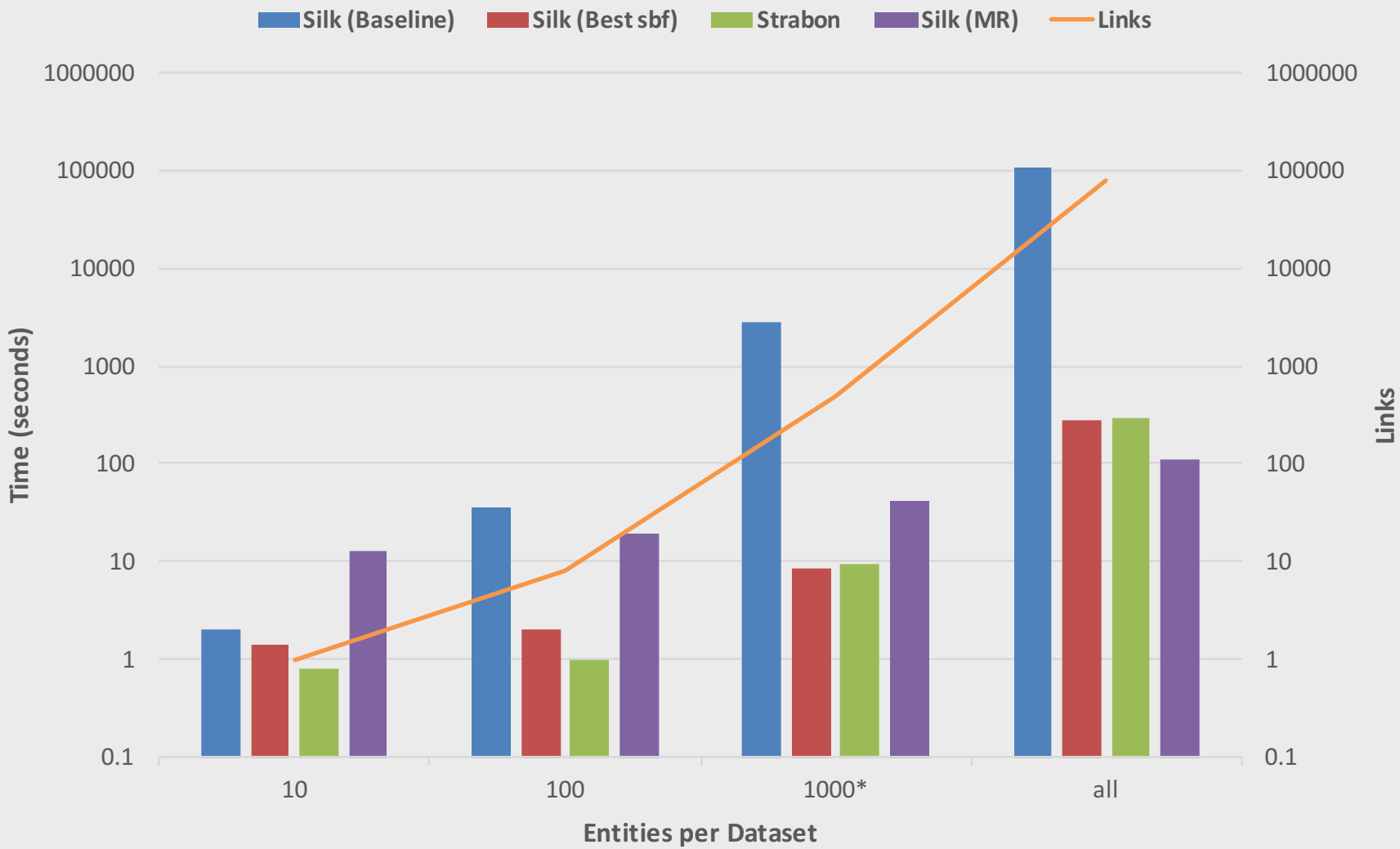
Environment of Experiments

- **Single machine environment**
  - 2 Intel Xeon E5620 processors, 12MB L3 cache, 2.4 GHz, 32 GB RAM, RAID-5. 4 disks, 32 MB cache, 7200 rpm
- **Distributed environment**
  - cluster provided by the European Public Cloud Provider Interoute (1 Master Node + 20 Slave Nodes: 2 CPUs, 4GB RAM, 10GB disk)
- More details: **http://silk.di.uoa.gr**

Experiment 1: Adjusting the Spatial Blocking Factor (sbf)



Experiment 2: Adjusting the number of Entities per Dataset



Outline

- Introduction
- Background
- Developed Methods
- Implementation
- Experimental Evaluation
- **Conclusions**

Conclusions & Future Work

- Methods for Spatial and Temporal Link Discovery
- Implementation on the Silk framework
- Employed efficiently in Real-World Applications

- Support more relation models/calculi
- Make the algorithm parameter free
  - Estimate the optimal value for the bfs
  - Pose preprocessing queries
- Use approximate blocking techniques

Thanks for your attention!
Questions?