Geospatial RDF stores

Theofilos Ioannidis
@tioannid1
tioannid@di.uoa.gr
tioannid@yahoo.com

Outline

- Part A - Geospatial RDF stores (25 min)
- Part B – Benchmarking geospatial RDF stores (10 min)
- Part C – Q&A (5 min)

Part A - Geospatial RDF stores

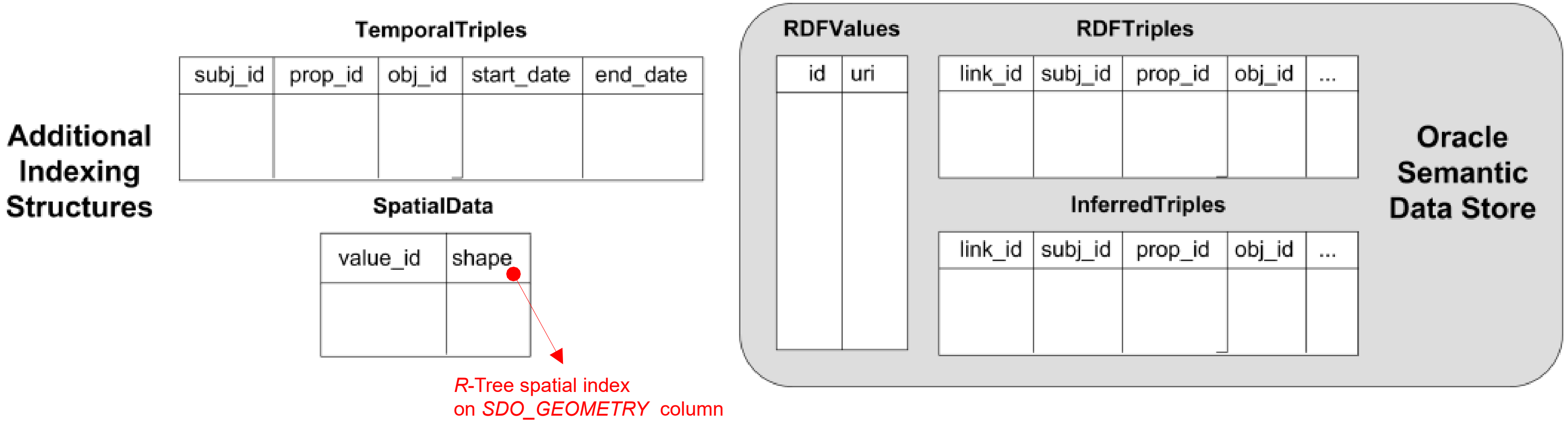
Part A.1 - Early Experimental Systems

Perry's Ph.D dissertation

The system of Brodt et al.

Perry PhD dissertation

- Implementation on top of Oracle 10g by Wright State University
- Support for SPARQL-ST
- GeoRSS GML serialization of geometries
- Spatial and temporal variables
- Spatial and temporal filters (RCC8, Allen)
- R-tree spatial index



Additional Indexing Structures

TemporalTriples

subj_id	prop_id	obj_id	start_date	end_date

SpatialData

value_id	shape

R-Tree spatial index on *SDO_GEOMETRY* column

Oracle Semantic Data Store

RDFValues

id	uri

RDFTriples

link_id	subj_id	prop_id	obj_id	...

InferredTriples

link_id	subj_id	prop_id	obj_id	...

Perry, Matthew Steven. "A framework to support spatial, temporal and thematic analytics over semantic web data." (2008)

Perry PhD dissertation - SPARQL-ST Implementation

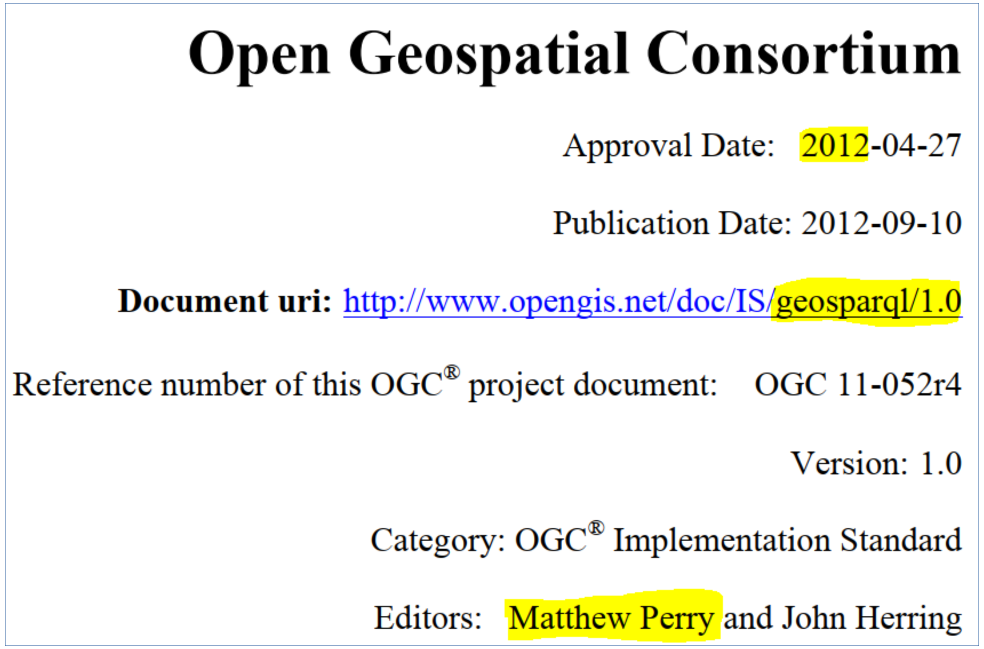
- Uses an Oracle database feature, called **Table Function**: a PL/SQL function that can be invoked inside the FROM clause of a SELECT statement. They return collections, which can be transformed into a dataset of rows and columns with the TABLE clause:

```
SELECT COLUMN_VALUE my_string FROM TABLE (random_strings (count_in => 5))
```

- Perry provides a single SQL table function, **sparql_st**, that inputs a valid SPARQL-ST query and returns a table of the resulting variable mappings: `sparql_st(queryStr VARCHAR) return AnyDataSet;`

```
SELECT * FROM TABLE (sparql_st (
          ‘SELECT ?x, ?z, ?y, %l
          WHERE {
          ?x <assigned to> ?z .
          ?x <participates in> ?y .
          ?y <occurred at> %l .
          SPATIAL FILTER(overlapbdyintersect(%l, GEOM(
          POLYGON((-122.84501 42.240328, -122.8075 42.240328, -122.8075 42.3764,
          -122.84501 42.3764, -122.84501 42.240328)))
          }’
));
```

Perry PhD dissertation – GeoSPARQL v1.0 standard



Open Geospatial Consortium

Approval Date: 2012-04-27

Publication Date: 2012-09-10

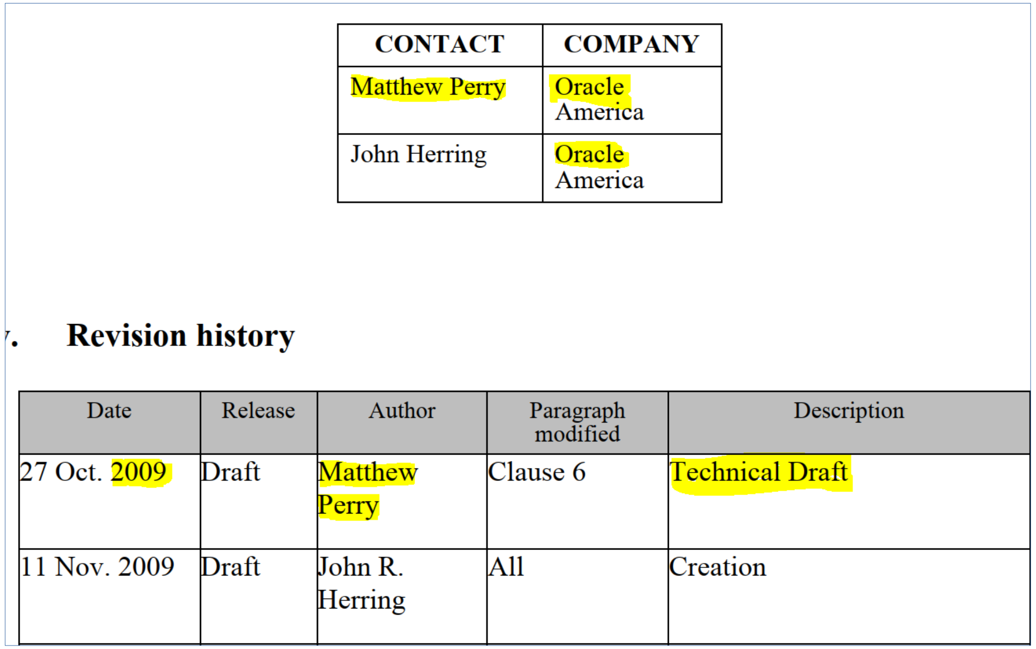
Document uri: http://www.opengis.net/doc/IS/geosparql/1.0

Reference number of this OGC® project document: OGC 11-052r4

Version: 1.0

Category: OGC® Implementation Standard

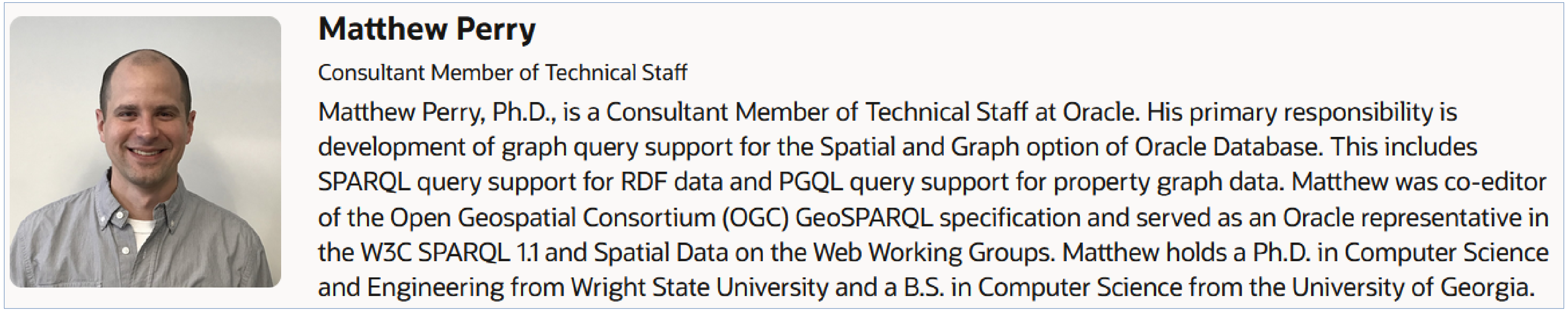
Editors: Matthew Perry and John Herring



CONTACT	COMPANY
Matthew Perry	Oracle America
John Herring	Oracle America

Revision history

Date	Release	Author	Paragraph modified	Description
27 Oct. 2009	Draft	Matthew Perry	Clause 6	Technical Draft
11 Nov. 2009	Draft	John R. Herring	All	Creation



Matthew Perry

Consultant Member of Technical Staff

Matthew Perry, Ph.D., is a Consultant Member of Technical Staff at Oracle. His primary responsibility is development of graph query support for the Spatial and Graph option of Oracle Database. This includes SPARQL query support for RDF data and PGQL query support for property graph data. Matthew was co-editor of the Open Geospatial Consortium (OGC) GeoSPARQL specification and served as an Oracle representative in the W3C SPARQL 1.1 and Spatial Data on the Web Working Groups. Matthew holds a Ph.D. in Computer Science and Engineering from Wright State University and a B.S. in Computer Science from the University of Georgia.

Brodt et al.

- Built on top of RDF-3X by University of Stuttgart
- No GeoSPARQL support
- Geometries represented as typed WKT literals
- WGS84 supported
- OGC-SFA spatial operations as SPARQL filter functions
- R-tree supported (but only used for spatial selections)

Brodt, Andreas, Daniela Nicklas, and Bernhard Mitschang. "Deep integration of spatial query processing into native RDF triple stores." Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems. 2010

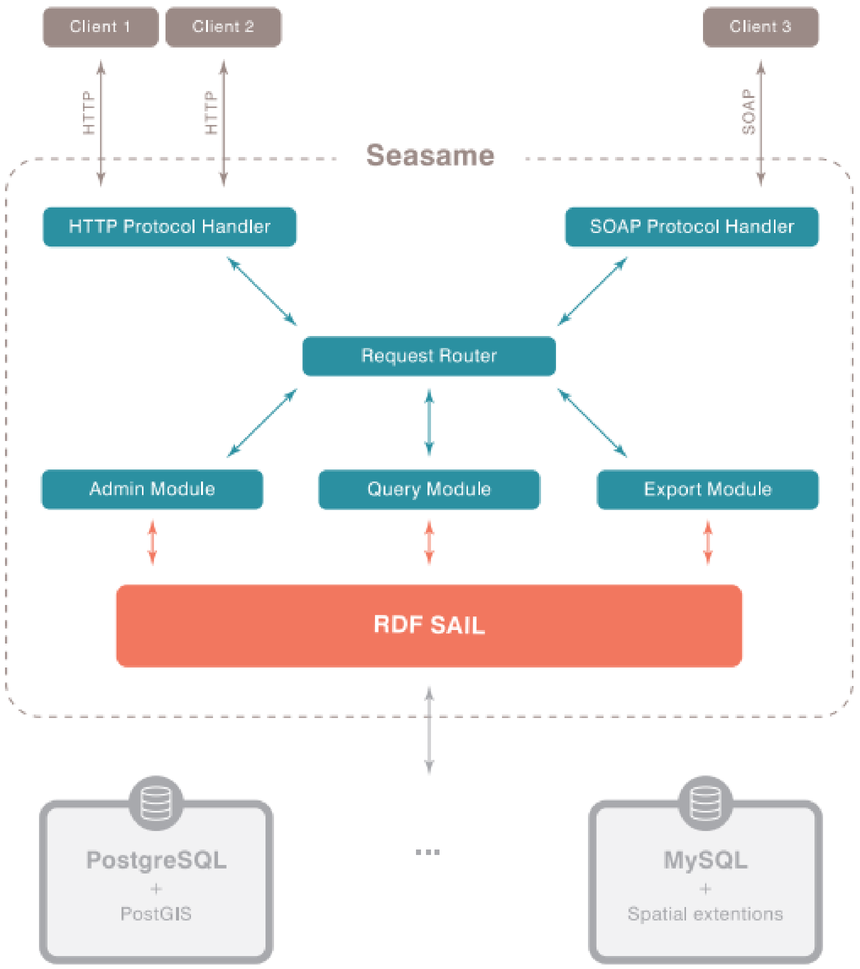
Part A.2 - Outdated Systems

OpenRDF Sesame

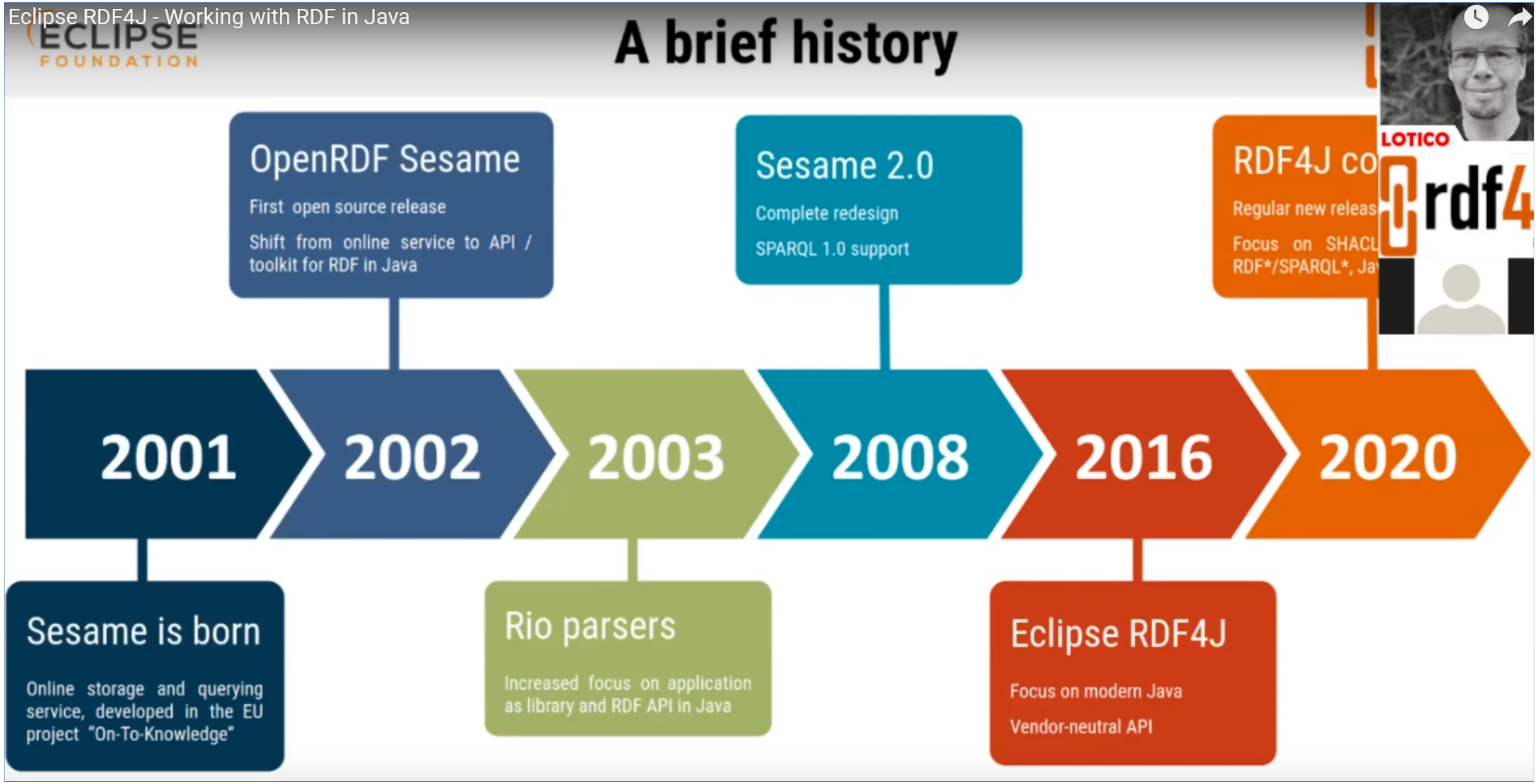
uSeekM

OpenRDF Sesame

- Java framework for processing and handling RDF data by Aduna.
- SAIL (Storage and Inference Layer), stackable architecture
- Spatial extensions: extending RDBMS SAIL with spatial databases
- Major releases available: v2.x and v4.x
- Open source available at: https://sourceforge.net/projects/sesame/



OpenRDF Sesame → RDF4J



Eclipse RDF4J - Working with RDF in Java

A brief history

2001 2002 2003 2008 2016 2020

Sesame is born

Online storage and querying service, developed in the EU project "On-To-Knowledge"

OpenRDF Sesame

First open source release

Shift from online service to API / toolkit for RDF in Java

Rio parsers

Increased focus on application as library and RDF API in Java

Sesame 2.0

Complete redesign

SPARQL 1.0 support

Eclipse RDF4J

Focus on modern Java

Vendor-neutral API

RDF4J co

Regular new releas

Focus on SHACL RDF*/SPARQL*, Ja

Eclipse RDF4J - Working with RDF in Java, https://www.youtube.com/watch?v=OOgXDUf8Cps&t=52s

uSeekM

OPEN SAHARA

- Spatial plugin for Sesame by OpenSahara. Supports:
- GeoSPARQL support
    - Core
    - Topology Vocabulary Extension (Simple Features, Egenhofer, RCC8)
    - Geometry Extension (WKT)
    - Geometry Topology Extension (Simple Features, Egenhofer, RCC8, WKT)
    - RDFS Entailment Extension (Simple Features, Egenhofer, RCC8, WKT)
- Only WGS84 CRS
- Available spatial indexes: R-Tree-over-GiST, Quadtree, Geohash (PostGIS or ElasticSearch as providers)
- Open source (Apache v2.0). Currently, no available official source code location.
- https://web.archive.org/web/20140415085418/https://dev.opensahara.com/projects/useekm

Part A.3 - Systems with Limited Geospatial/GeoSPARQL Functionality

AllegroGraph
OpenLink Virtuoso
Blazegraph DB
MarkLogic Server

Allegrograph

FRANZ INC.

- Quad store developed by Franz Inc. Supports:
- No GeoSPARQL
- Point geometries (N-dimensional Geospatial)
- Non standard CRSs
- Only a few spatial operations supported (Buffer, Bounding Box, Distance)
- Includes temporal functionalities: datetimes, time points, and time intervals and relations (properties) between these
- Is horizontally scalable through federation and sharding (FedShard)
- Latest v7.3.0 (May 2022)
- Closed source. Free edition available at https://allegrograph.com/downloads/

OpenLink Virtuoso

- Developed by OpenLink. Virtuoso Open Source (VOS)
- 3 Java APIs: Jena (v2.x, v3.0.x, v4.3.x), Sesame (v1, v2, v4), RDF4J v2.x.
- No GeoSPARQL (stable/7 branch), GeoSPARQL* (develop/7 branch)
- Points only
- Serialized as typed literals (datatype `virtrdf:Geometry`)
- Spatial operations (subset of SQL/MM)
- Multiple CRS
- WKT for geometry types
- R-tree spatial index
- Latest v7.2.8 (Oct 2022)
- Latest from develop/7 branch v7.2.6-rc1 (Oct 2018)
- VOS available at http://vos.openlinksw.com/owiki/wiki/VOS

Blazegraph DB

- Previously known as Bigdata. Developed by SYSTAP, LLC. Supports:
- Written in Java, uses Sesame, no GeoSPARQL yet
- Dedicated, typed literals
  - http://www.bigdata.com/rdf/geospatial/literals/v1#lat-lon
  - http://www.bigdata.com/rdf/geospatial/literals/v1#lat-lon-time
- Rectangle and distance queries over geospatial coordinates, possibly combined with range scans over the temporal coordinate
- Geospatial queries are done through a custom SERVICE extension http://www.bigdata.com/rdf/geospatial#search
- Generic z-order index implementation for multi-dimensional range scans
- Latest v2.1.6 RC (Feb 2020)
- Available at https://blazegraph.com/

MarkLogic Server

- NoSQL database that supports multi-model data representation
- JSON/XML documents, RDF triples and relational data
- Data access languages: JavaScript/XQuery, SPARQL and SQL
- Triple query languages: SPARQL, XQuery or JavaScript
- No GeoSPARQL
- Geospatial serialization vocabularies: GML, KML, GeoRSS and GeoJSON
- Geospatial types : Point, Box, Circle, Polygon, Complex Polygon (WKT), Linestring (WKT)
- Supports Cartesian, WGS84 coordinate systems and ETRS89 (European specific)
- Spatial indexes: (i) Point index, fast, in memory, point-only, (ii) Region path index, Geohash or R-tree, allows DE-9IM operations
- Latest v10.0-9.5 (Sep 2022)
- Available (free Developer License) at https://www.marklogic.com/

Part A.4 - Systems with Partial GeoSPARQL Support

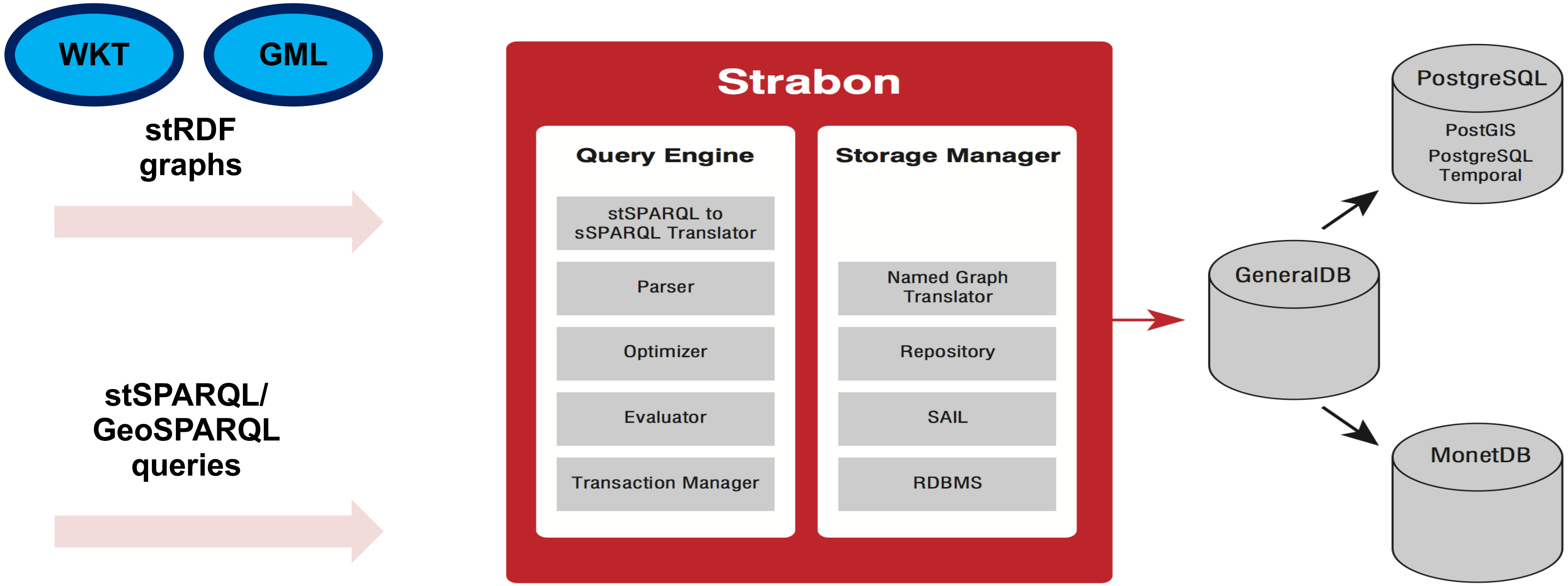
Stardog
Strabon
Eclipse RDF4J

Stardog

- Developed by Stardog. Supports:
- partial GeoSPARQL support (after 2018)
  - WKT serialization
    - points and rectangles, by default
    - All shapes, by installing JTS (Java Topology Suite https://locationtech.github.io/jts/)
  - Functions geof:relate, geof:distance, geof:within, geof:nearby and geof:area
- Spatial Indexing
  - spatial indexing, based on Lucene Spatial, geohash prefix tree (precision 11 for sub-meter accuracy)
  - approximate matching controlled with a precision parameter
  - precision specified upon database creation
- Supports WGS84 Geo Positioning RDF vocabulary (World Geodetic System 1984) https://www.w3.org/2003/01/geo/wgs84_pos
- Provides support for Jena and Sesame APIs
- Latest v8.2.0 (Dec 2022)
- Closed source. Available at: https://www.stardog.com

Strabon

Find more at: http://www.strabon.di.uoa.gr/



[1] Kyzirakos, Kostis, Manos Karpathiotakis, and Manolis Koubarakis. "Strabon: a semantic geospatial DBMS." International Semantic Web Conference. Springer, Berlin, Heidelberg, 2012
[2] Kyzirakos, Kostis, et al. “The Spatiotemporal RDF Store Strabon.” SSTD 2013 Proceedings of the 13th International Symposium on Advances in Spatial and Temporal Databases - Volume 8098, 2013, pp. 496–500.

Strabon - Geospatial features

Support for:

- stRDF and stSPARQL
- GeoSPARQL support
  - Core
  - Geometry Extension (WKT, GML)
  - Geometry Topology Extension (Simple Features, Egenhofer, RCC8, WKT, GML)
- Multiple Coordinate Reference Systems (CRS)
- Builds on Sesame RDBMS Sail
- Geospatial relational database as back-end (PostGIS, MonetDB)
- R-tree over GiST index (PostGIS)
- Latest v3.2.9. Available at http://www.strabon.di.uoa.gr/

Eclipse RDF4J

- Java framework for processing and handling RDF data (former OpenRDF Sesame by Aduna).
- Based on Spatial4J and JTS libraries for geospatial reasoning.
- SAIL (Storage and Inference Layer), stackable architecture
- Memory store, NativeStore, Lucene SAIL and its derivates (the SolrSail and the ElasticSearchSail), (Sesame's RDBMS Sail removed), LMDB Store fast embeddable key-value based on the Symas Lightning Memory-Mapped Database (https://www.symas.com/lmdb)
- GeoSPARQL :
  - By default supported on any type of store (memory, native, etc)
  - The Lucene SAIL and its derivates (the SolrSail and the ElasticSearchSail) have built-in optimizations for geospatial querying.
  - Default spatially indexed property is geo:asWKT
  - LuceneSail configuration allows for customization of spatially indexed properties
    - Core
    - Geometry (partial) (WKT)
    - Geometry Topology Extension (Simple Feature, Egenhofer, RCC8, WKT)
    - Only WKT serializations
    - RDFS entailment
- Latest v4.2.0 (Sep 2022)
- Open source available at: https://rdf4j.org

Part A.4 - Systems with Extensive GeoSPARQL Support

Parliament

Oracle RDF Spatial and Graph

AnzoGraph DB

GraphDB

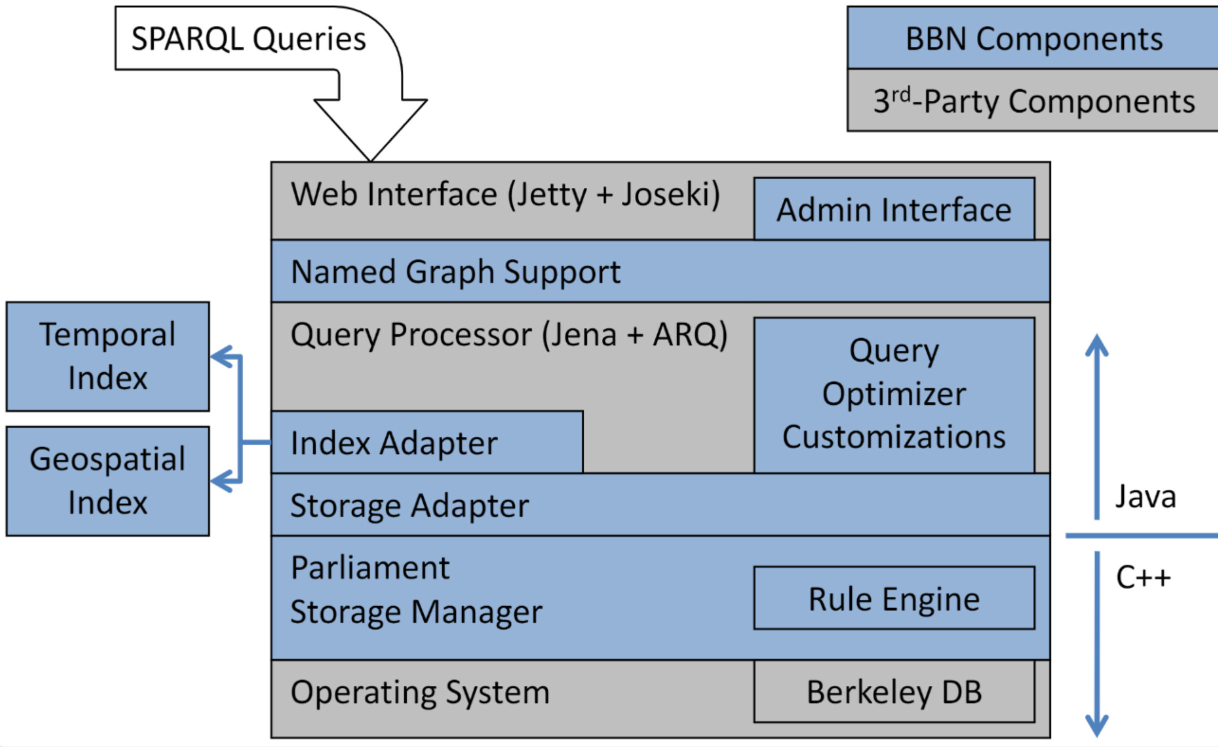
Apache Jena GeoSPARQL

Parliament

- Developed (2009) by Raytheon BBN Technologies (Dave Kolas).
- First GeoSPARQL implementation. Supports:
  - Core
  - Topology vocabulary
  - Geometry
  - Geometry Topology
  - Multiple CRSs
  - Both WKT and GML serializations
  - RDF entailment required for (geo:asWKT, geo:asGML)
- It uses BerkeleyDB as backend and Jena+ARQ as SPARQL processor
- Standard R-tree index

Latest v2.8.1 (June 2022) features: separation between Parliament's software and data files, Java 8 and 11 support, offered as Docker image

- Open source available (since 2018) at : https://github.com/SemWebCentral/parliament



Oracle Spatial and RDF Graph

- Developed by Oracle
- GeoSPARQL support
  - Core
  - Topology Vocabulary Extension (Simple Features)
  - Geometry Extension (WKT 1.2.0, GML 3.1.1)
  - Geometry Topology Extension (Simple Features, WKT 1.2.0, GML 3.1.1)
  - RDFS Entailment Extension (Simple Features, WKT 1.2.0, GML 3.1.1)
- CRS support
- R-Tree as spatial index for up to 4 dimensions or composite B-tree index on point data for non spatial join operations
- Virtual RDF graphs (as of Oracle Spatial and Graph 12c R2)
- Latest version 21c (April 2022)

AnzoGraph DB

CAMBRIDGE SEMANTICS

- Massively parallel processing native graph database built for data harmonization and analytics
- Scales from single server to multiple servers in a cluster and cloud environments
- It holds the record for the fastest execution of the LUBM 1 Trillion Triple Benchmark (Google Cloud Platform, October 2016)
- GeoSPARQL support
  - Core
  - Topology Vocabulary Extension (Simple Features, Egenhofer, RCC8)
  - Geometry Extension (WKT 1.x, GML 2.x)
  - Geometry Topology Extension (Simple Features, Egenhofer, RCC8, WKT 1.x, GML 2.x)
  - RDFS Entailment Extension (Simple Features, Egenhofer, RCC8, WKT 1.x, GML 2.x)
- CRS support
- Geospatial aggregate functions provided through user-defined aggregate extensions
- Latest v2.5.10
- Closed source. Available at: https://cambridgesemantics.com/

GraphDB Free

ontotext

- Former OWLIM. Based on RDF4J framework, developed by Ontotext.
- GeoSPARQL
  - Core
  - Topology Vocabulary Extension (Simple Features, Egenhofer, RCC8)
  - Geometry Extension (WKT 1.x, GML 3.x)
  - Geometry Topology Extension (Simple Features, Egenhofer, RCC8, WKT 1.x, GML 2.x)
  - RDFS Entailment Extension (Simple Features, Egenhofer, RCC8, WKT 1.x, GML 2.x)
  - WKT serialization
- Multiple CRSs (v9.x onward)
- GeoSPARQL support through the GeoSPARQL plugin
  - Indexing algorithms, based on Lucene Spatial, quad or geohash prefix tree
  - Approximate matching controlled with a precision parameter
  - Reconfiguration of algorithm and precision allowed through SPARQL update statements
  - Rebuilding GeoSPARQL index on demand
- Offers useful GeoSPARQL extensions based on the USeekMSail, i.e., ext:isValid(geometry), ext:area(geometry)
- Supports WGS84 Geo Positioning RDF vocabulary (World Geodetic System 1984) https://www.w3.org/2003/01/geo/wgs84_pos
- Features 2 very fast bulk importers (LoadRDF, Preload)
- Latest v10.0.2 (Aug 2022)
- Closed source. Available at: https://graphdb.ontotext.com/

Apache Jena GeoSPARQL

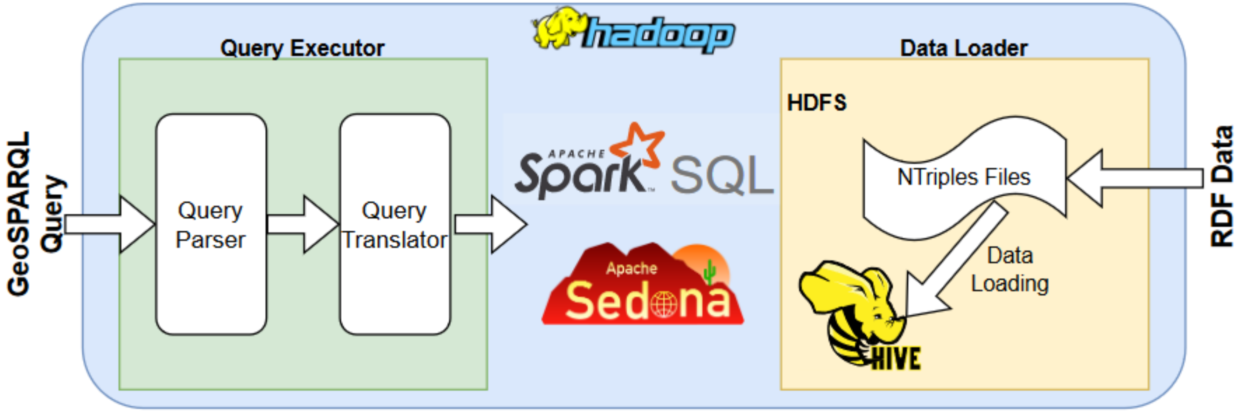
- Pure Java implementation which does not require any setup or configuration of any third party relational databases and geospatial extensions
- Full GeoSPARQL support
  - Core
  - Topology Vocabulary Extension (Simple Feature, Egenhofer and RCC8)
  - Geometry Extension (WKT, GML 2.0)
  - Geometry Topology Extension (Simple Feature, Egenhofer, RCC8, WKT, GML 2.0)
  - RDFS Entailment Extension (Simple Feature, Egenhofer, RCC8, WKT, GML 2.0)
  - Query Rewrite (Simple Feature, Egenhofer, RCC8, WKT, GML 2.0)
- All indexing and caching is performed during query execution
- 3 indexes (Geometry Literal, Geometry Transform, Query Rewrite)
- Also supports WGS84 Geo Positioning RDF vocabulary (World Geodetic System 1984)
- Latest v3.16.0 (Jul 2020) as a maven artifact org.apache.jena:jena-geosparql:3.16.0
- Open source available at: https://github.com/galbiston/geosparql-jena

Part A.5 – Distributed RDF Stores with GeoSPARQL Support

Strabo2

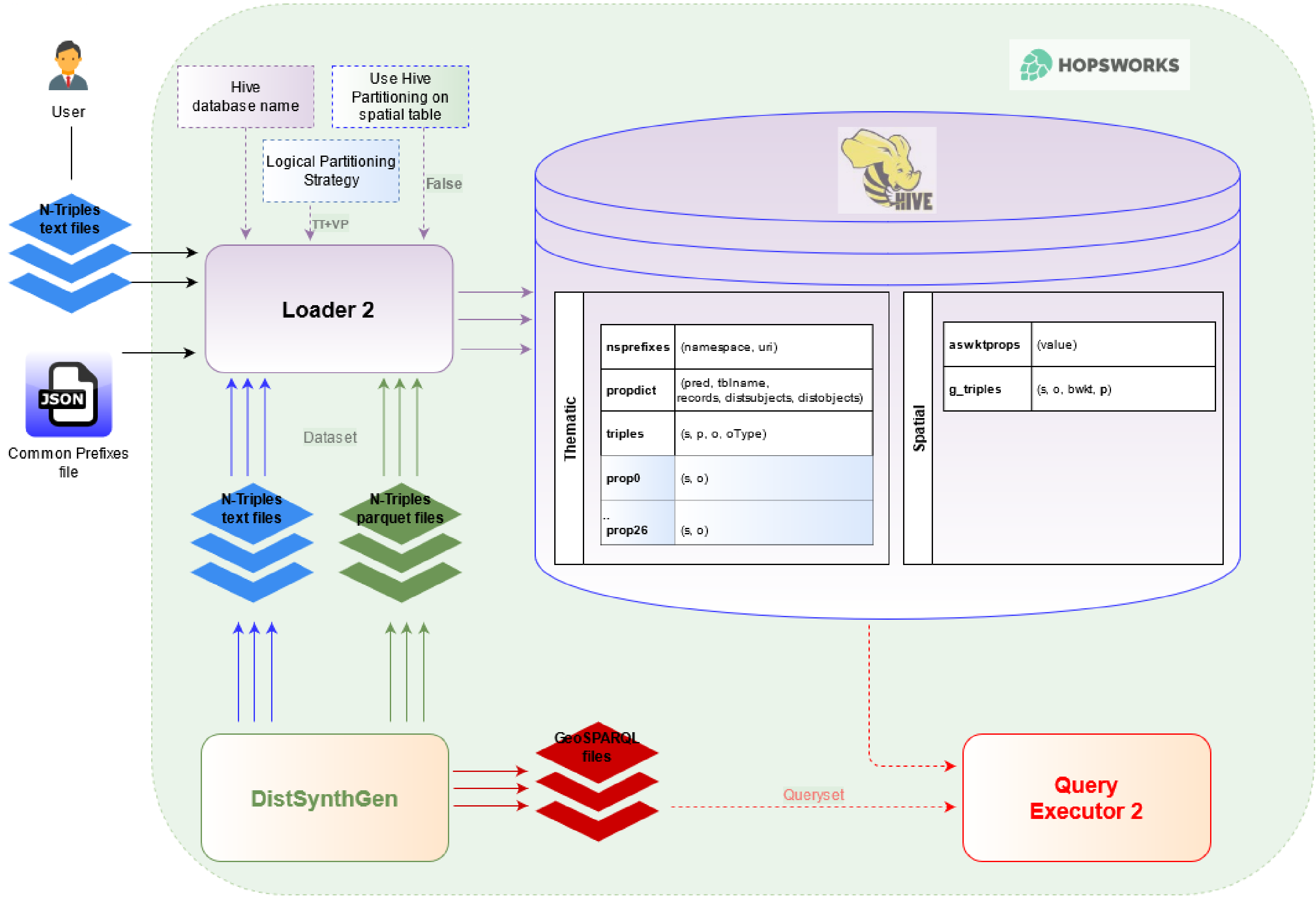
Strabo 2

- Apache Spark Java implementation for Hadoop clusters
- Extends Ontop-spatial for GeoSPARQL to Spark-SQL translation
- Geospatial capabilities provided by Apache Sedona (former GeoSpark)
- Apache Hive as persistence layer with Parquet storage format
- Vertical Partitioning used as the logical partitioning strategy
- GeoSPARQL support: Simple Feature family, WKT serialization
- Open source available at: https://github.com/db-ee/Strabo-2



ISWC 2022: Main Track 6b: Querying, 26th October 2022
Strabo 2: Distributed Management of Massive Geospatial RDF Datasets
Dimitris Bilidas, Theofilos Ioannidis, Nikos Mamoulis and Manolis Koubarakis

Strabo 2 – Operation Flow



nsprefixes	(namespace, uri)
propdict	(pred, tblname, records, distsubjects, distobjects)
triples	(s, p, o, oType)
prop0	(s, o)
.. prop26	(s, o)

aswktprops	(value)
g_triples	(s, o, bwkt, p)

Part B - Geographica 2 benchmark

Useful links

Source code, Datasets, Query sets and most of the results are available on the following 2 web sites:

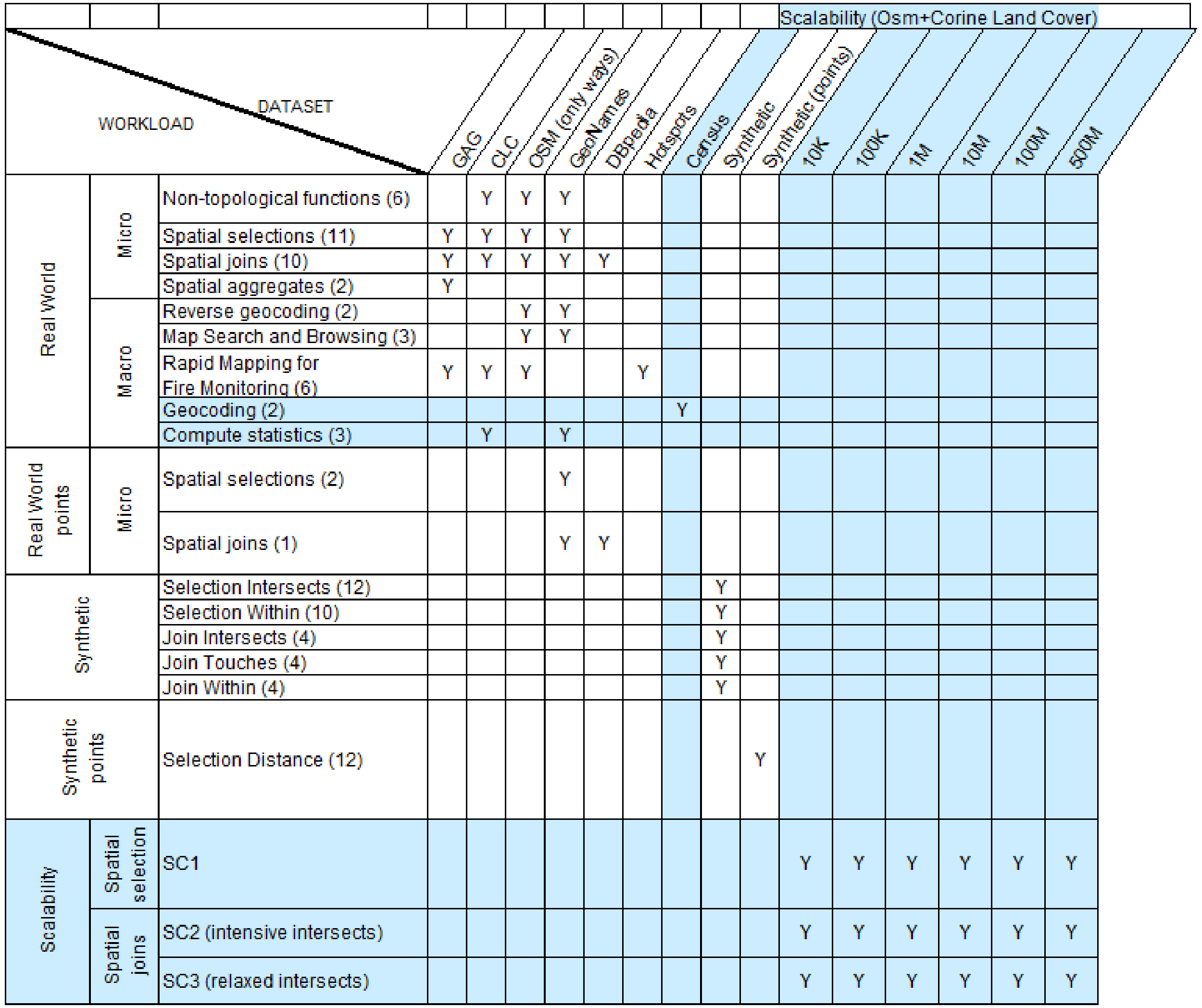
- http://geographica.di.uoa.gr/
- http://geographica2.di.uoa.gr/

Geographica 2

- **Purpose**: evaluate RDF stores supporting GeoSPARQL and stSPARQL
- **Workloads**:
  - Real world (micro and macro benchmarks)
  - Synthetic
  - Scalability: OSM+Corine Land Cover Reference Dataset 1 (RD-1)
- **Datasets**: GAG, CLC, LGD (OSM), GeoNames, DBpedia, Hotspots, Synthetic-512, Synthetic-Points-1024, Census, Scalability (OSM+CLC RD-1: 10K, 100K, 1M, 10M, 100M, 500M triples)
- **Scenarios**: Micro, Macro (Reverse geocoding, Map search & browsing, Rapid mapping for fire monitoring, Geocoding, Computation of Statistics), Scalability
- **Systems**: uSeekM, Parliament, Strabon, System X (Par, Ser), GraphDB, RDF4J (NativeSail, LuceneSail+NativeSail)
- **Limited functionality**: VOS (Virtuoso OpenLink Server), System Y in addition to the 6 (8 variants) other systems against points only versions of Real World and Synthetic datasets

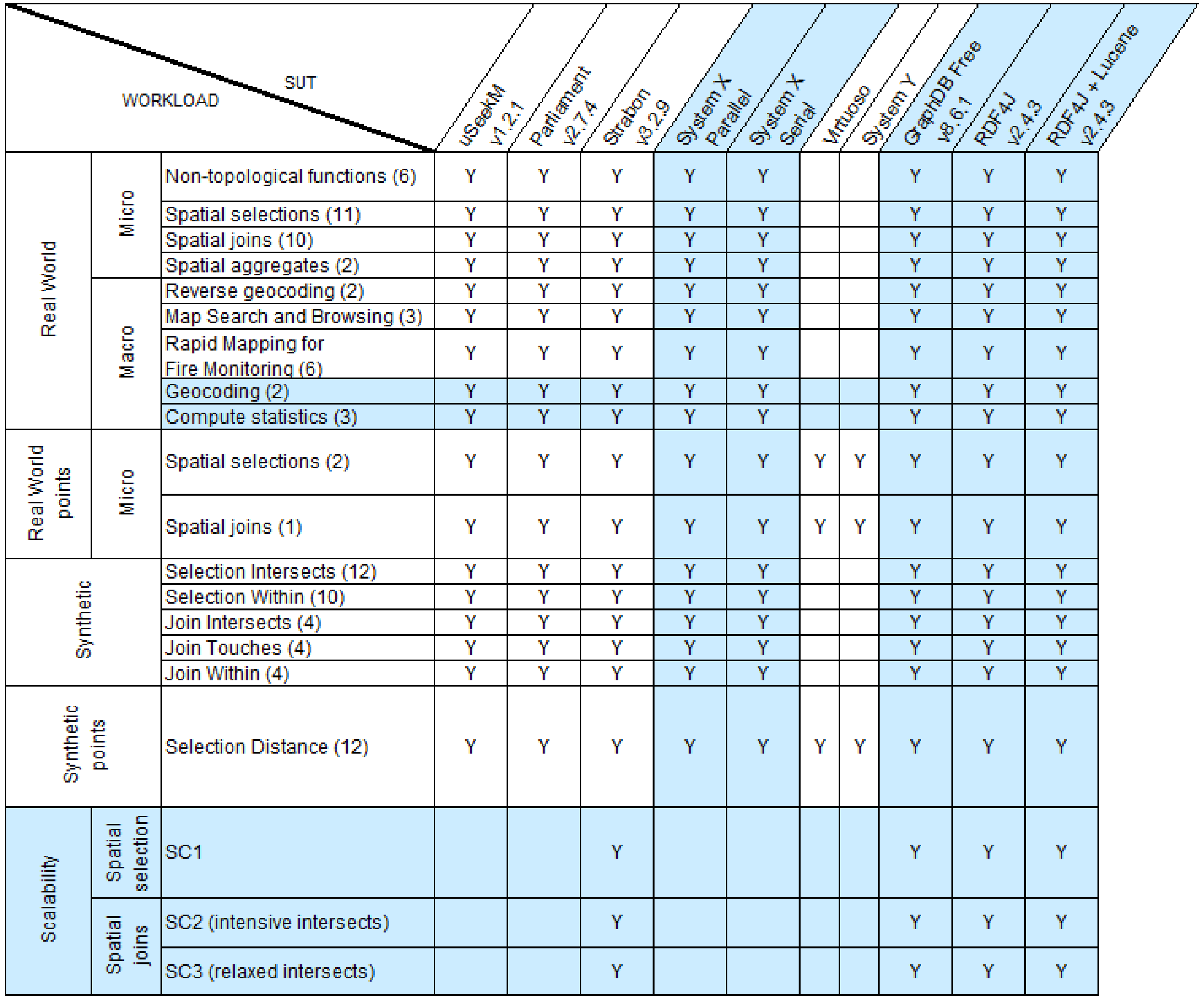
Ioannidis Theofilos, Garbis George, Kyzirakos Kostis, Bereta Konstantina, & Koubarakis Manolis (2021). “Evaluating geospatial RDF stores using the benchmark Geographica 2”. Journal on Data Semantics, 10(3), 189-228.

Geographica 2 – Workload/Dataset matrix



												Scalability (Osm+Corine Land Cover)					
		WORKLOAD DATASET	GAG	CLC	OSM (only ways)	GeoNames	DBpedia	Hotspots	Census	Synthetic	Synthetic (points)	10K	100K	1M	10M	100M	500M
Real World	Micro	Non-topological functions (6)		Y	Y	Y											
		Spatial selections (11)	Y	Y	Y	Y											
		Spatial joins (10)	Y	Y	Y	Y	Y										
		Spatial aggregates (2)	Y														
	Macro	Reverse geocoding (2)			Y	Y											
		Map Search and Browsing (3)			Y	Y											
		Rapid Mapping for Fire Monitoring (6)	Y	Y	Y			Y									
		Geocoding (2)							Y								
		Compute statistics (3)		Y		Y											
Real World points	Micro	Spatial selections (2)				Y											
		Spatial joins (1)				Y	Y										
Synthetic		Selection Intersects (12)								Y							
		Selection Within (10)								Y							
		Join Intersects (4)								Y							
		Join Touches (4)								Y							
		Join Within (4)								Y							
Synthetic points		Selection Distance (12)									Y						
Scalability	Spatial selection	SC1										Y	Y	Y	Y	Y	Y
	Spatial joins	SC2 (intensive intersects)										Y	Y	Y	Y	Y	Y
		SC3 (relaxed intersects)										Y	Y	Y	Y	Y	Y

Geographica 2 – Workload/System matrix

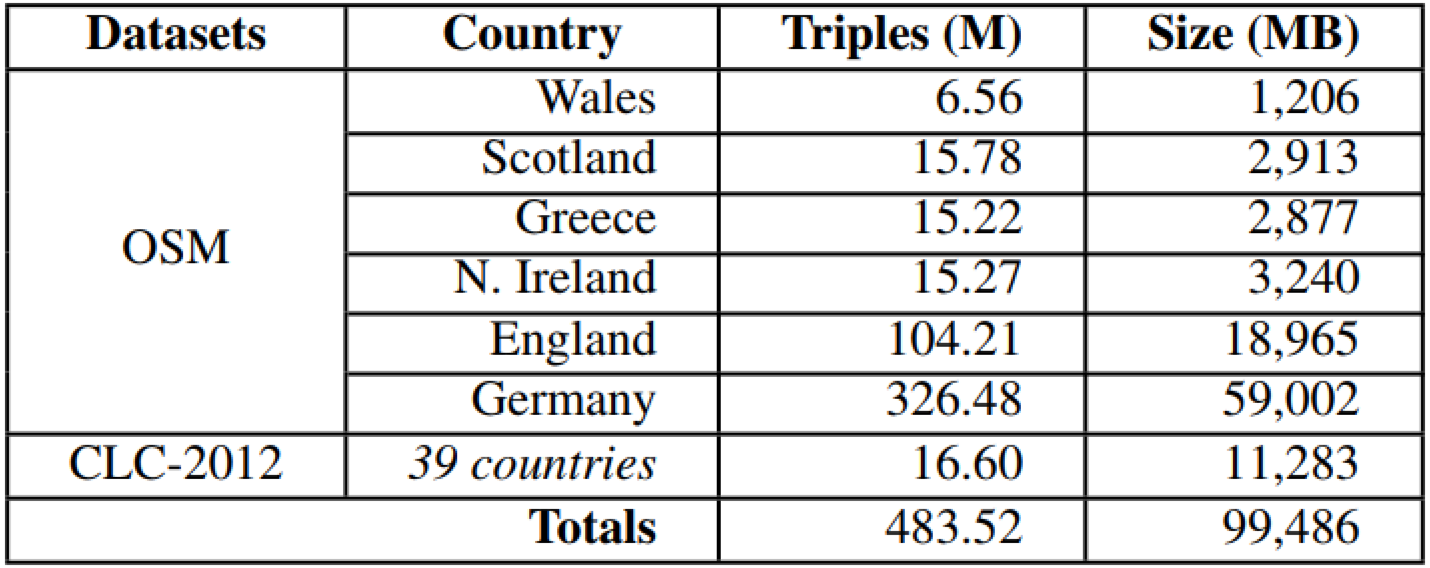


		SUT WORKLOAD	uSeekM v1.2.1	Parliament v2.7.4	Strabon v3.2.9	System X Parallel	System X Serial	Virtuoso	System Y	GraphDB Free v8.6.1	RDF4J v2.4.3	RDF4J + Lucene v2.4.3
Real World	Micro	Non-topological functions (6)	Y	Y	Y	Y	Y			Y	Y	Y
		Spatial selections (11)	Y	Y	Y	Y	Y			Y	Y	Y
		Spatial joins (10)	Y	Y	Y	Y	Y			Y	Y	Y
		Spatial aggregates (2)	Y	Y	Y	Y	Y			Y	Y	Y
	Macro	Reverse geocoding (2)	Y	Y	Y	Y	Y			Y	Y	Y
		Map Search and Browsing (3)	Y	Y	Y	Y	Y			Y	Y	Y
		Rapid Mapping for Fire Monitoring (6)	Y	Y	Y	Y	Y			Y	Y	Y
		Geocoding (2)	Y	Y	Y	Y	Y			Y	Y	Y
		Compute statistics (3)	Y	Y	Y	Y	Y			Y	Y	Y
Real World points	Micro	Spatial selections (2)	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		Spatial joins (1)	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Synthetic		Selection Intersects (12)	Y	Y	Y	Y	Y			Y	Y	Y
		Selection Within (10)	Y	Y	Y	Y	Y			Y	Y	Y
		Join Intersects (4)	Y	Y	Y	Y	Y			Y	Y	Y
		Join Touches (4)	Y	Y	Y	Y	Y			Y	Y	Y
		Join Within (4)	Y	Y	Y	Y	Y			Y	Y	Y
Synthetic points		Selection Distance (12)	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Scalability	Spatial selection	SC1			Y					Y	Y	Y
	Spatial joins	SC2 (intensive intersects)			Y					Y	Y	Y
		SC3 (relaxed intersects)			Y					Y	Y	Y

Geographica 2 - Scalability Workload

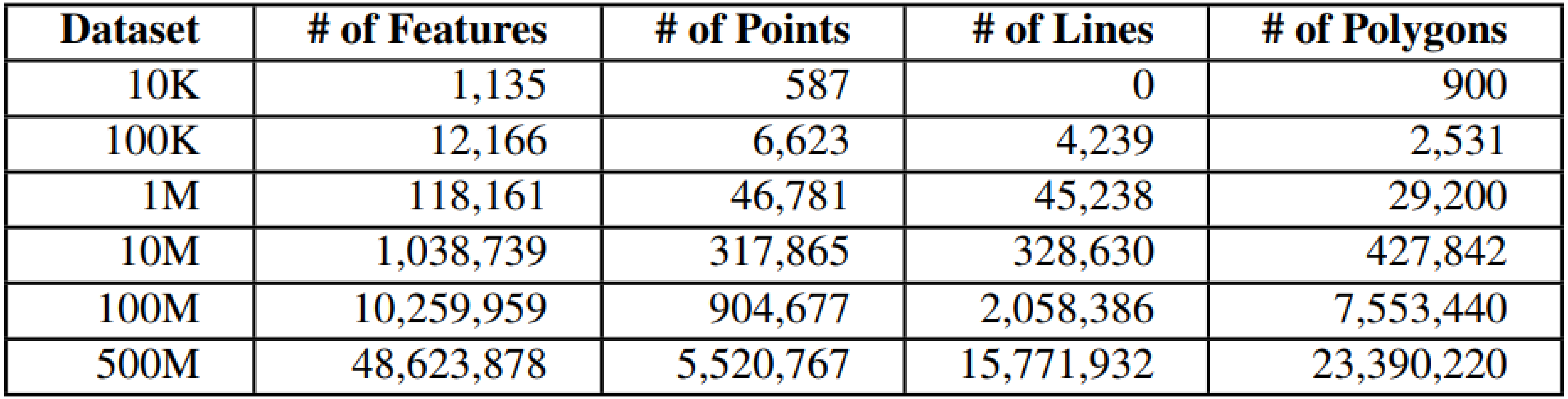
- **Purpose**: the scalability experiment aims at discovering the limits of the systems under test as the number of triples in the dataset increase
- **Method**: Each selected system is tested against six increasingly bigger, proper subsets of a big real world reference geospatial dataset. For each system-dataset combination we measure:
  - ✔ the repository size on disk,
  - ✔ the bulk loading time taking into consideration the limitations of loading methods of each system and
  - ✔ the response time in three queries which represent a spatial selection, a heavy spatial join with higher spatial selectivity and a lighter spatial join with lower spatial selectivity
- **Systems**: Strabon, GraphDB, RDF4J

Geographica 2 - Scalability Reference Dataset characteristics



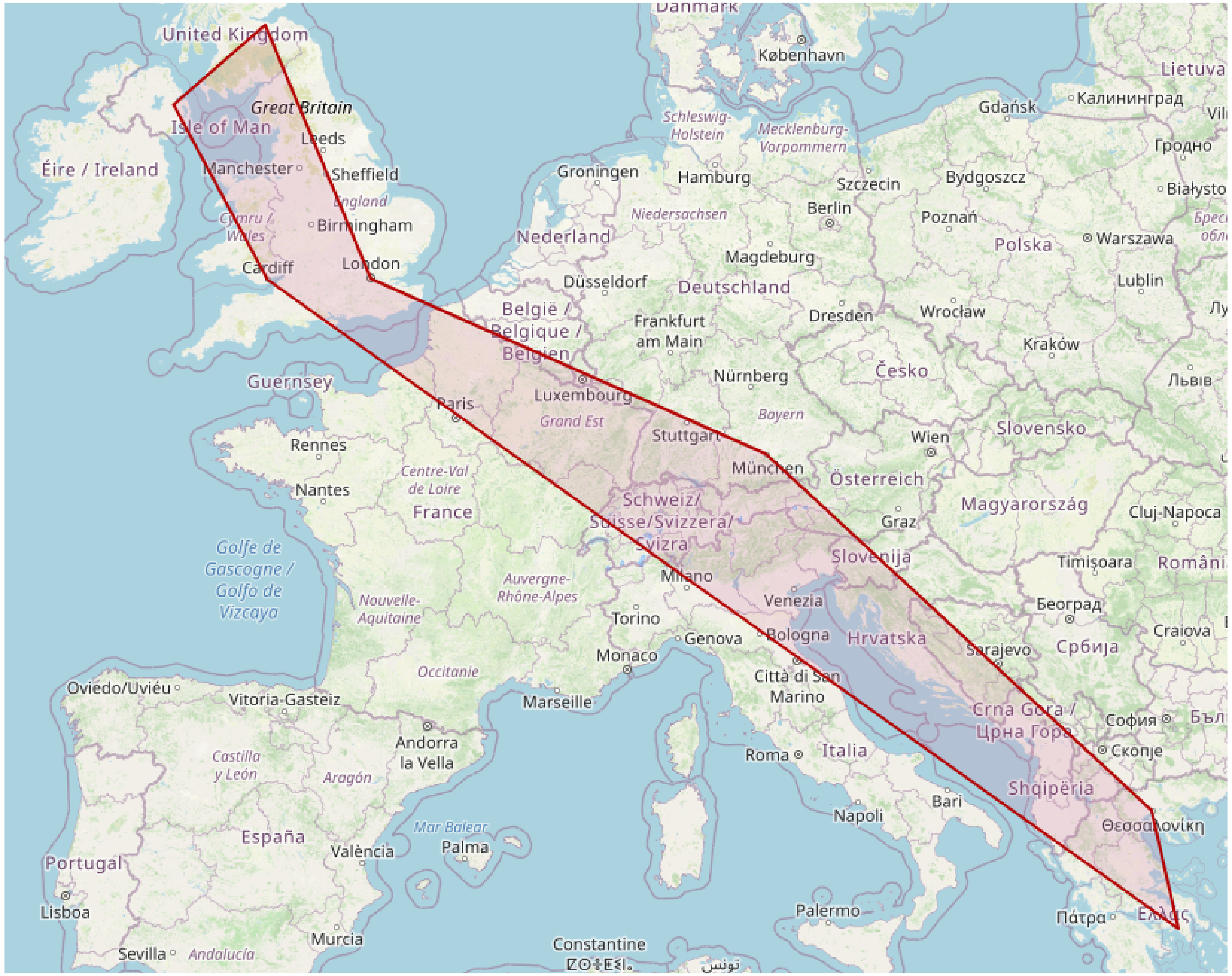
Datasets	Country	Triples (M)	Size (MB)
OSM	Wales	6.56	1,206
	Scotland	15.78	2,913
	Greece	15.22	2,877
	N. Ireland	15.27	3,240
	England	104.21	18,965
	Germany	326.48	59,002
CLC-2012	*39 countries*	16.60	11,283
	Totals	483.52	99,486

Geographica 2 - Scalability Datasets basic characteristics



Dataset	# of Features	# of Points	# of Lines	# of Polygons
10K	1,135	587	0	900
100K	12,166	6,623	4,239	2,531
1M	118,161	46,781	45,238	29,200
10M	1,038,739	317,865	328,630	427,842
100M	10,259,959	904,677	2,058,386	7,553,440
500M	48,623,878	5,520,767	15,771,932	23,390,220

Geographica 2 - Scalability Query SC1 (Spatial Selection)



Geographica 2 - Scalability Query SC1 (Spatial Selection)

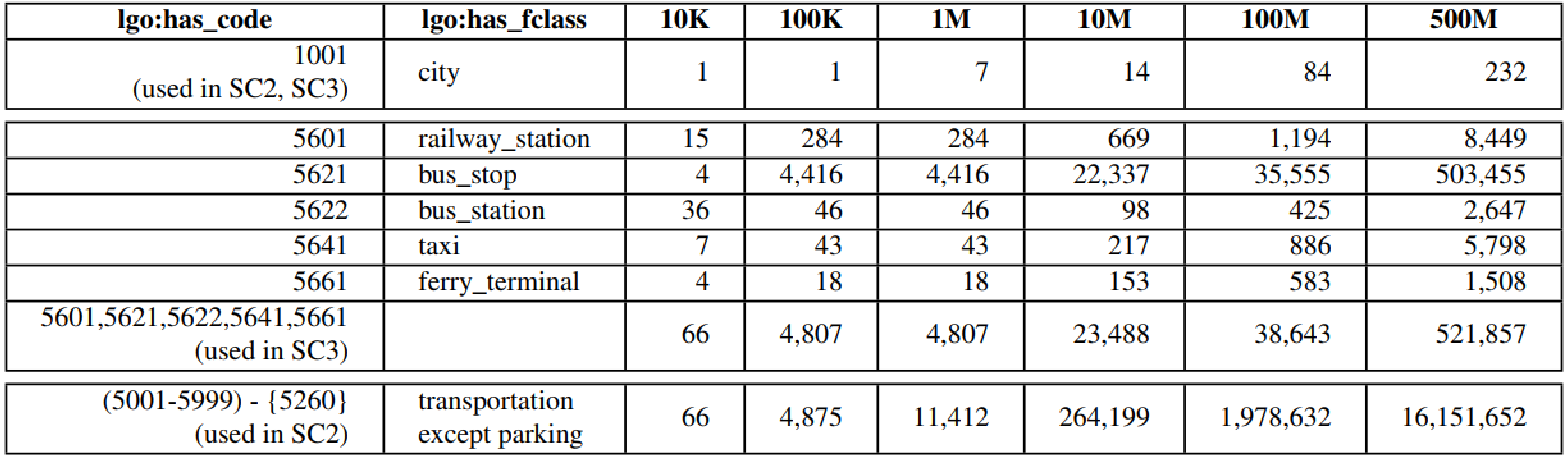
SC1_Geometries_Intersects_GivenPolygon: Find all geometries that intersect with the given polygon

```
PREFIX geof: <http://www.opengis.net/def/function/geosparql/>
PREFIX geo: <http://www.opengis.net/ont/geosparql#>
PREFIX lgd: <http://data.linkedeodata.eu/ontology#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

SELECT ?s1 ?o1
WHERE {
 ?s1 geo:asWKT ?o1 .
 FILTER(geof:sfIntersects(?o1, "POLYGON((23.708496093749996 37.95719224376526,22.906494140625
40.659805938378526,11.524658203125002 48.16425348854739,-0.1181030273437499 51.49506473014367,-
3.2189941406250004 55.92766341247031,-5.940856933593749 54.59116279530599,-3.1668090820312504
51.47967237816337,23.708496093749996
37.95719224376526))"^^<http://www.opengis.net/ont/geosparql#wktLiteral>)).
}
```

Geographica 2 - Scalability Value distribution of `lgo:has_code` property

`lgo:has_code ⇒ <http://data.linkedeodata.eu/ontology#has_code>`



lgo:has_code	lgo:has_fclass	10K	100K	1M	10M	100M	500M
1001 (used in SC2, SC3)	city	1	1	7	14	84	232
5601	railway_station	15	284	284	669	1,194	8,449
5621	bus_stop	4	4,416	4,416	22,337	35,555	503,455
5622	bus_station	36	46	46	98	425	2,647
5641	taxi	7	43	43	217	886	5,798
5661	ferry_terminal	4	18	18	153	583	1,508
5601,5621,5622,5641,5661 (used in SC3)		66	4,807	4,807	23,488	38,643	521,857
(5001-5999) - {5260} (used in SC2)	transportation except parking	66	4,875	11,412	264,199	1,978,632	16,151,652

Geographica 2 - Scalability Query SC2 (Spatial Join)

SC2_Intensive_Geometries_Intersect_Geometries: Find all transportation-related features (except parking) within cities

```
PREFIX geof: <http://www.opengis.net/def/function/geosparql/>
PREFIX geo: <http://www.opengis.net/ont/geosparql#>
PREFIX lgo: <http://data.linkedeodata.eu/ontology#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

SELECT ?s1 ?s2
WHERE {
 ?s1 geo:hasGeometry [ geo:asWKT ?o1 ] ;
    lgo:has_code "1001"^^xsd:integer .
 ?s2 geo:hasGeometry [ geo:asWKT ?o2 ] ;
    lgo:has_code ?code2 .
 FILTER(?code2>5000 && ?code2<6000 && ?code2 != 5260) .
 FILTER(geof:sfIntersects(?o1, ?o2)).
}
```

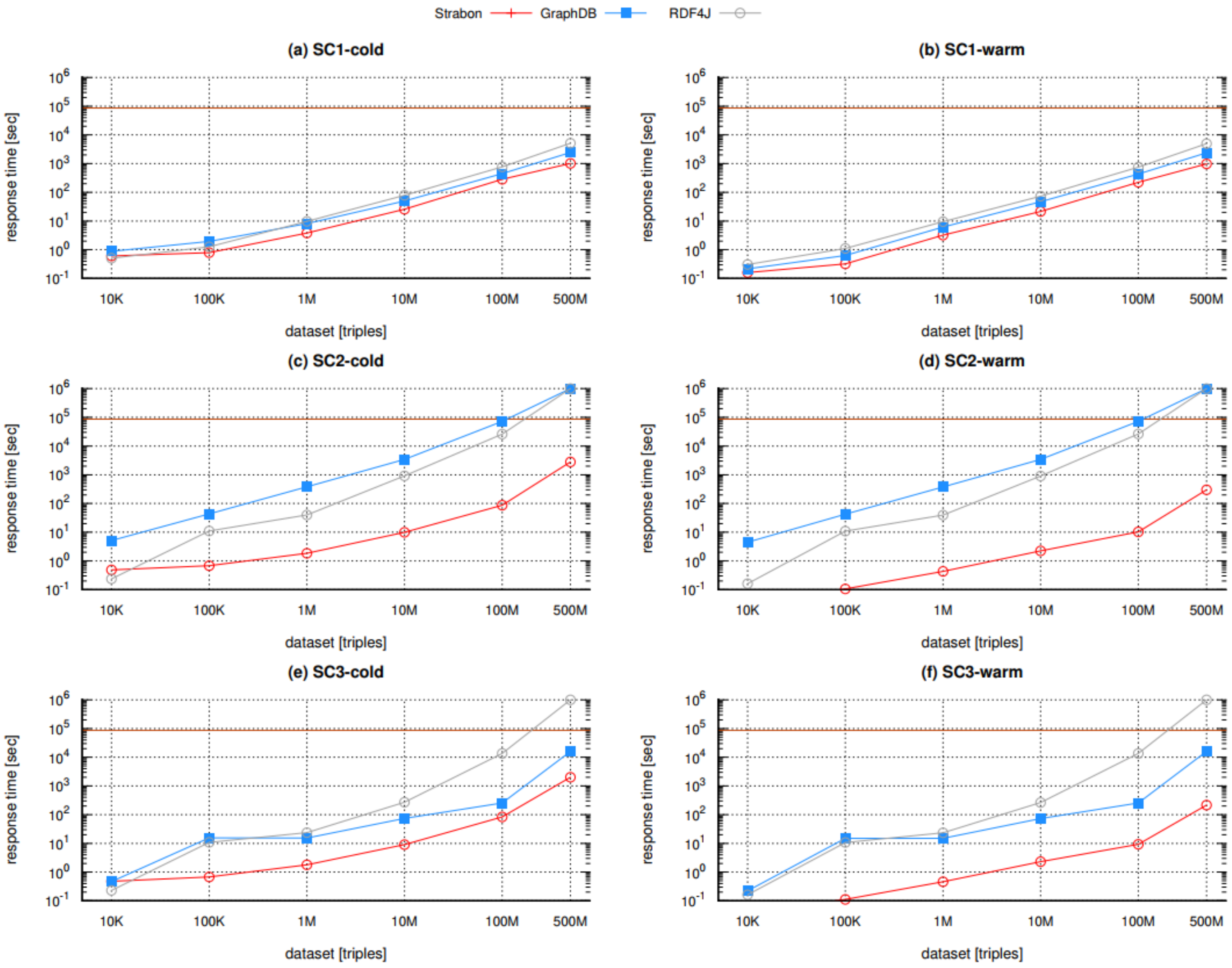
Geographica 2 - Scalability Query SC3 (Spatial Join)

SC3_Relaxed_Geometries_Intersect_Geometries: Find all bus stops, bus stations, railway stations, taxis and ferry terminals within cities

```
PREFIX geof: <http://www.opengis.net/def/function/geosparql/>
PREFIX geo: <http://www.opengis.net/ont/geosparql#>
PREFIX lgo: <http://data.linkedeodata.eu/ontology#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

SELECT ?s1 ?s2
WHERE {
 ?s1 geo:hasGeometry [ geo:asWKT ?o1 ] ;
    lgo:has_code "1001"^^xsd:integer .
 ?s2 geo:hasGeometry [ geo:asWKT ?o2 ] ;
    lgo:has_code ?code2 .
 FILTER(?code2 IN (5622, 5601, 5641, 5621, 5661)) .
 FILTER(geof:sfIntersects(?o1, ?o2)).
}
```

Geographica 2 - Scalability results



Part C – Q & A