



Bringing Location Analysis to the Semantic Web with the OGC GeoSPARQL Standard

Matthew Perry

Xavier Lopez

ORACLE[®]

Agenda



- About the GeoSPARQL SWG
- Use Cases & Requirements
- From SPARQL to GeoSPARQL
- GeoSPARQL Technical Details
- Implementation Considerations
- Live Demo

Group Members



- Open Geospatial Consortium standards working group
 - 13 voting members, 36 observers
 - Editors: Matthew Perry and John Herring
 - Chairs: John Herring and Dave Kolas
- Submitting Organizations

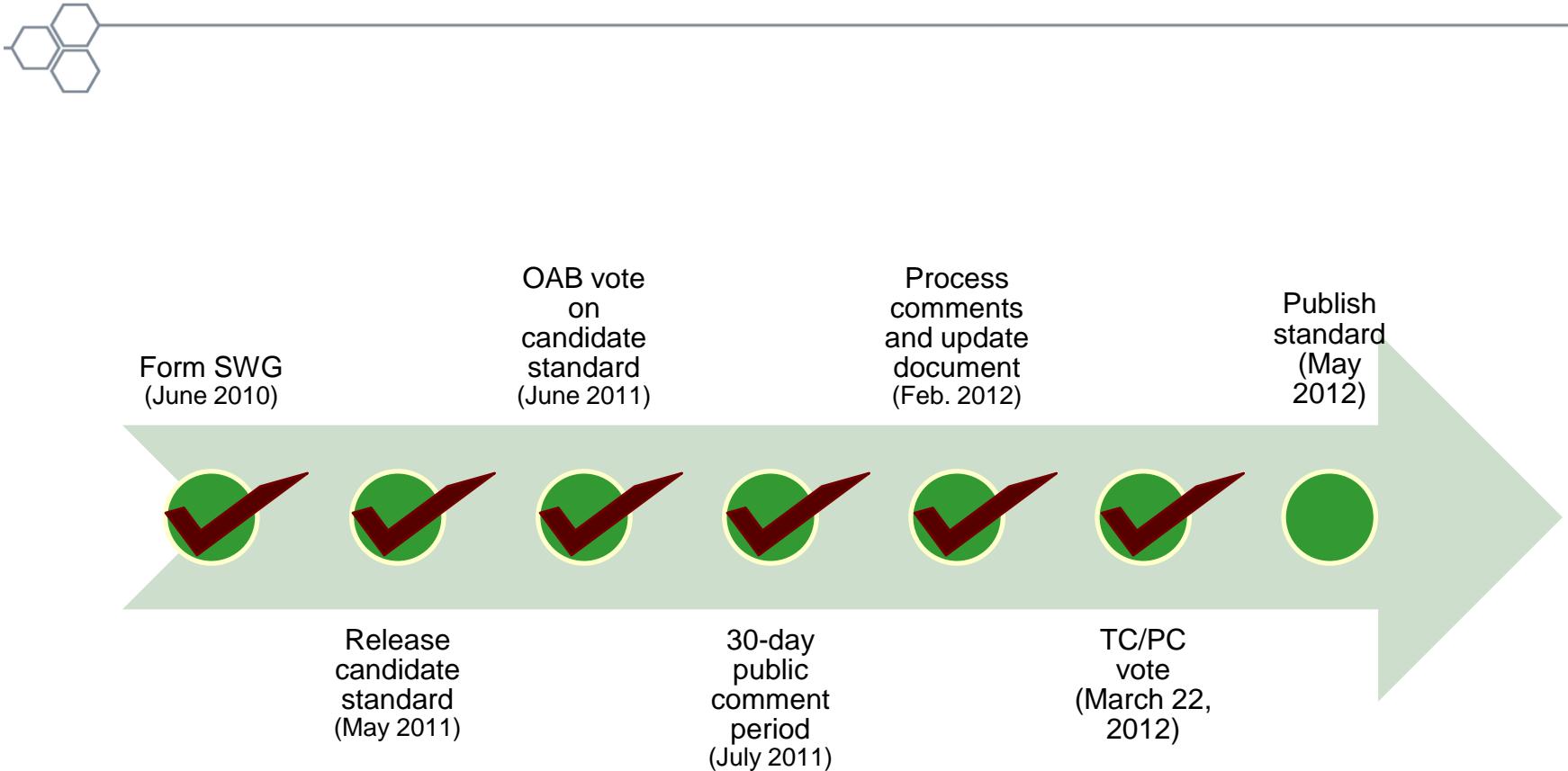


Defence Geospatial Information Working Group



Traverse Technologies, Inc.

Standardization Process



Linked Geo Data



- Many LOD datasets have geospatial components



- Barriers to integration
 - Vendor-specific geometry support
 - Different vocabularies
 - W3C Basic Geo, GML XMLLiteral, Vendor-specific
 - Different spatial reference systems
 - WGS84 Lat-Long, British National Grid

What **DBPedia** Historic Buildings are within walking distance?

What **OpenStreetMap** Dog Parks are inside **Ordnance Survey** Southampton Administrative District?



Gazetteers and Linked Open Data Services

- Provide common terms (place names) to link across existing spatial data resources
- Enable consolidated view across the map layers
- Reconcile differences in data semantics so that they can all “talk” and interoperate
- Resolving semantic discrepancies across databases gazetteers and applications
- Integrate full breath of enterprise content continuum (structured, spatial, email, documents, web services)

Semantic GIS



- GIS applications with semantically complex thematic aspects
 - Logical reasoning to classify features
 - land cover type, suitable farm land, etc.
 - Complex Geometries
 - Polygons and Multi-Polygons with 1000's of points
 - Complex Spatial Operations
 - Union, Intersection, Buffers, etc.

Find parcels with an **area** of at least 3 sq. miles that **touch** a local feeder road and are **inside** an area of suitable farm land.

Towards Qualitative Spatial Reasoning



- Don't always have geometry data
 - Textual descriptions
 - Next to Hilton hotel
 - Inside Union Square
 - Incomplete geometry data
 - Only have geometries for some features
 - Hybrid quantitative and qualitative spatial reasoning
- GeoSPARQL takes some steps in this direction
 - Vocabulary for asserting topological relations
 - Same query specification for qualitative and quantitative systems

Requirements for GeoSPARQL



- Provide a common target for implementers & users
 - Representation and query
- Work within SPARQL's extensibility framework
- Simple enough for general users
 - Keep the common case simple (WGS 84 point data)
- Capable enough for GIS professionals
 - Multiple SRSs, complex geometries, complex operators
- Don't re-invent the wheel!



ISO 19107 – Spatial Schema
ISO 13249 – SQL/MM



Simple Features
Well Known Text (WKT)
GML
KML
GeoJSON



GEOSPARQL TECHNICAL DETAILS



FROM SPARQL TO GEOSPARQL

SPARQL QUERY



RDF Data

```
:res1 rdf:type :House .  
:res1 :baths "2.5"^^xsd:decimal .  
:res1 :bedrooms "3"^^xsd:decimal .  
  
:res2 rdf:type :Condo .  
:res2 :baths "2"^^xsd:decimal .  
:res2 :bedrooms "2"^^xsd:decimal .  
  
:res3 rdf:type :House  
:res3 :baths "1.5"^^xsd:decimal .  
:res3 :bedrooms "3"^^xsd:decimal .
```

SPARQL Query

```
SELECT ?r ?ba ?br  
WHERE { ?r rdf:type :House .  
        ?r :baths ?ba .  
        ?r :bedrooms ?br }
```

Result Bindings

?r	?ba	?br
=====	=====	=====
:res1	"2.5"	"3"
:res3	"1.5"	"3"

SPARQL QUERY



RDF Data

```
:res1 rdf:type :House .  
:res1 :baths "2.5"^^xsd:decimal .  
:res1 :bedrooms "3"^^xsd:decimal .  
  
:res2 rdf:type :Condo .  
:res2 :baths "2"^^xsd:decimal .  
:res2 :bedrooms "2"^^xsd:decimal .  
  
:res3 rdf:type :House  
:res3 :baths "1.5"^^xsd:decimal .  
:res3 :bedrooms "3"^^xsd:decimal .
```

SPARQL Query

```
SELECT ?r ?ba ?br  
WHERE { ?r rdf:type :House .  
        ?r :baths ?ba .  
        ?r :bedrooms ?br  
        FILTER (?ba > 2) }
```

Result Bindings

?r		?ba		?br
=====				
:res1		"2.5"		"3"

Spatial SPARQL QUERY



Spatial RDF Data

```
:res1 rdf:type :House .
:res1 :baths "2.5"^^xsd:decimal .
:res1 :bedrooms "3"^^xsd:decimal .
:res1 ogc:hasGeometry :geom1 .
:geom1 ogc:asWKT "POINT(-122.25 37.46)"^^ogc:wktLiteral .
```

This is what we are standardizing

```
:res3 rdf:type :House
:res3 :baths "1.5"^^xsd:decimal .
:res3 :bedrooms "3"^^xsd:decimal .
:res3 ogc:hasGeometry :geom3 .
:geom3 ogc:asWKT "POINT(-122.24 37.47)"^^ogc:wktLiteral .
```

Vocabulary & Datatypes

Find houses within a search polygon

GeoSPARQL Query

```
SELECT ?r ?ba ?br
WHERE { ?r rdf:type :House .
        ?r :baths ?ba .
        ?r :bedrooms ?br .
        ?r ogc:hasGeometry ?g .
        ?g ogc:asWKT ?wkt
        FILTER(ogcf:sfWithin(?wkt,
                               "POLYGON(...)"^^ogc:wktLiteral)) }
```

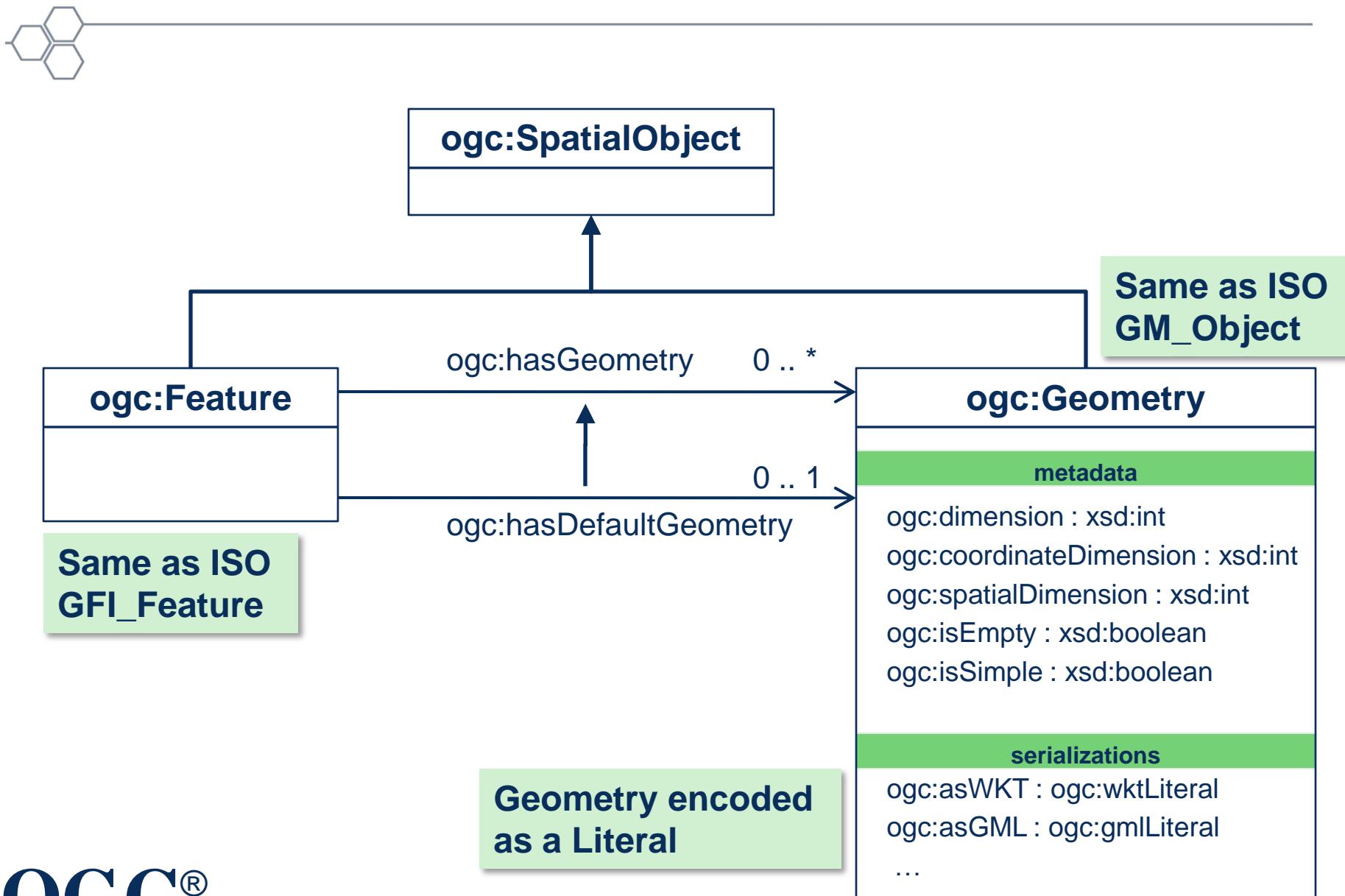
Extension Functions

Components of GeoSPARQL



- Vocabulary for Query Patterns
 - Classes
 - Spatial Object, Feature, Geometry
 - Properties
 - Topological relations
 - Links between features and geometries
 - Datatypes for geometry literals
 - ogc:wktLiteral, ogc:gmlLiteral
- Query Functions
 - Topological relations, distance, buffer, intersection, ...
- Entailment Components
 - RDFS entailment
 - RIF rules to compute topological relations

GeoSPARQL Vocabulary: Basic Classes and Relations



Details of ogc:wktLiteral



All RDFS Literals of type ogc:wktLiteral shall consist of an optional IRI identifying the spatial reference system followed by Simple Features Well Known Text (WKT) describing a geometric value [ISO 19125-1].

"<<http://www.opengis.net/def/crs/OGC/1.3/CRS84>>
POINT (-122.4192 37.7793)"^^ogc:wktLiteral

WGS84 longitude – latitude
is the default CRS

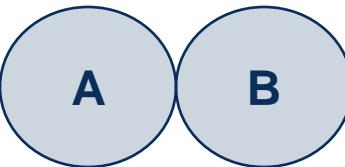
"POINT (-122.4192 37.7793)"^^ogc:wktLiteral

European Petroleum Survey Group (EPSG)
maintains a set of CRS identifiers.

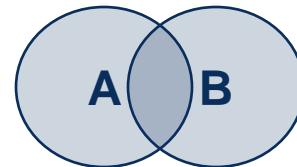
Topological Relations between ogc:SpatialObject



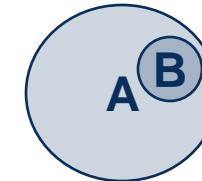
ogc:sfEquals



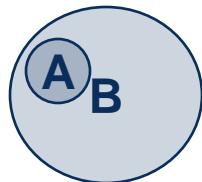
ogc:sfTouches



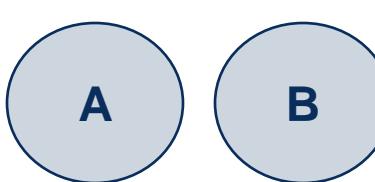
ogc:sfOverlaps



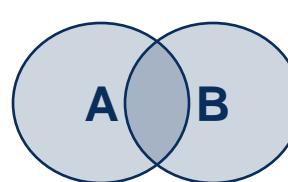
ogc:sfContains



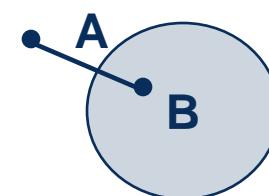
ogc:sfWithin



ogc:sfDisjoint



ogc:sfIntersects



ogc:sfCrosses

- Assumes Simple Features Relation Family
- Also support Egenhofer and RCC8

Example Data



```
:City rdfs:subClassOf ogc:Feature .  
:Park rdfs:subClassOf ogc:Feature .  
:exactGeometry rdfs:subPropertyOf ogc:hasGeometry .
```

Meta Information

```
:SanFrancisco rdf:type :City .  
:UnionSquarePark rdf:type :Park .  
:UnionSquarePark :commissioned "1847-01-01"^^xsd:date .
```

Non-spatial Properties

```
:UnionSquarePark :exactGeometry :geo1 .  
:geo1 ogc:asWKT "Polygon(())"^^ogc:wktLiteral .  
  
:SanFrancisco :exactGeometry :geo2 .  
:geo2 ogc:asWKT "Polygon(())"^^ogc:wktLiteral .  
  
:UnionSquarePark ogc:sfWithin :SanFrancisco .
```

Spatial Properties

Why don't you have `ogc:myFavoriteProperty`?



- GeoSPARQL vocabulary is not comprehensive
 - Just enough to define a reasonable set of query patterns
- There are other efforts for more comprehensive vocabularies
 - ISO / TC 211
 - SOCoP
 - GeoVocamps
- GeoSPARQL vocabulary can easily be extended with other application/domain-specific vocabularies

Why Encode Geometry Data as a Literal?



Advantage: single self-contained unit

Consistent way to select geometry information

Find all water bodies that are within 1 km of Route 3

```
SELECT ?water ?wWKT
WHERE { ?water      rdf:type          :WaterBody .
        ?water      :hasExactGeometry   ?wGeo .
        ?wGeo       ogc:asWKT         ?wWKT .
        :Route_3    :hasExactGeometry   ?r3Geo .
        :r3Geo     ogc:asWKT         ?r3WKT .
        FILTER(ogcf:distance(?r3WKT, ?wWKT,...) <= 1000)
}
```

Consistent way to pass geometry information around

Why don't you support W3C Basic Geo?



- Too simple to meet our requirements
 - Can't use different datums and coordinate systems
 - Limited number of geometry types
- W3C Basic Geo data can easily be converted to wktLiteral

```
PREFIX geo: <http://www.w3.org/2003/01/geo/wgs84_pos#>
PREFIX ogc: <http://www.opengis.net/geosparql#>
SELECT (STRDT(CONCAT("POINT(",?long," ",?lat,")"),  

              ogc:wktLiteral) AS ?wktLit)
WHERE { ?point geo:long ?long .  

        ?point geo:lat ?lat }
```

GeoSPARQL Query Functions



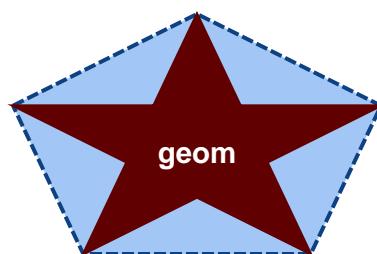
- `ogcf:distance(geom1: ogc:wktLiteral, geom2: ogc:wktLiteral, units: xsd:anyURI) : xsd:double`



- `ogcf:buffer(geom: ogc:wktLiteral, radius: xsd:double, units: xsd:anyURI) : ogc:wktLiteral`



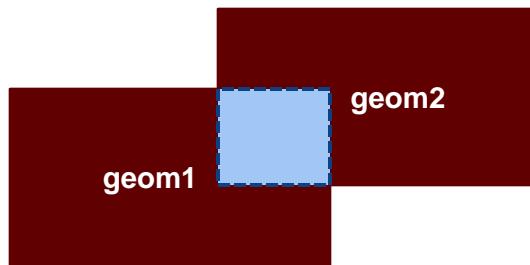
- `ogcf:convexHull(geom: ogc:wktLiteral) : ogc:wktLiteral`



GeoSPARQL Query Functions



- `ogcf:intersection(geom1: ogc:wktLiteral,
geom2: ogc:wktLiteral): ogc:wktLiteral`



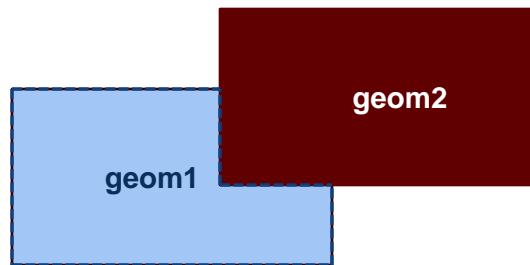
- `ogcf:union(geom1: ogc:wktLiteral,
geom2: ogc:wktLiteral): ogc:wktLiteral`



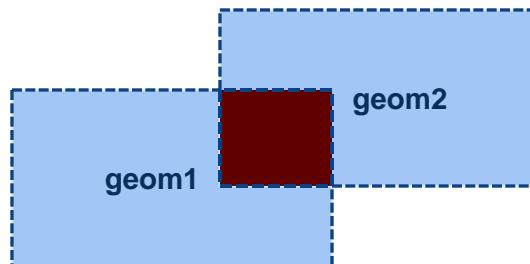
GeoSPARQL Query Functions



- `ogcf:difference(geom1: ogc:wktLiteral,
geom2: ogc:wktLiteral): ogc:wktLiteral`



- `ogcf:symDifference(geom1: ogc:wktLiteral,
geom2: ogc:wktLiteral): ogc:wktLiteral`



GeoSPARQL Query Functions



- `ogcf:envelope(geom: ogc:wktLiteral) : ogc:wktLiteral`



- `ogcf:boundary(geom1: ogc:wktLiteral) : ogc:wktLiteral`



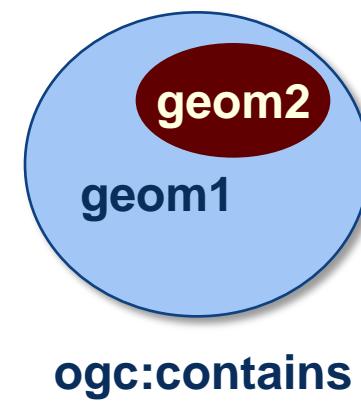
- `ogcf:getSRID(geom: ogc:wktLiteral) : xsd:anyURI`

GeoSPARQL Topological Query Functions



- `ogcf:relate(geom1: ogc:wktLiteral,
geom2: ogc:wktLiteral,
patternMatrix: xsd:string) : xsd:boolean`

		geom2		
		Interior	Boundary	Exterior
geom1	Interior	T	T	T
	Boundary	F	F	T
	Exterior	F	F	T



patternMatrix: TTTFFFTFF

GeoSPARQL Topological Query Functions



- **ogcf:sfEquals**(geom1: `ogc:wktLiteral`,
geom2: `ogc:wktLiteral`): `xsd:boolean`
- **ogcf:sfDisjoint**(geom1: `ogc:wktLiteral`,
geom2: `ogc:wktLiteral`): `xsd:boolean`
- **ogcf:sfIntersects**(geom1: `ogc:wktLiteral`,
geom2: `ogc:wktLiteral`): `xsd:boolean`
- **ogcf:sfTouches**(geom1: `ogc:wktLiteral`,
geom2: `ogc:wktLiteral`): `xsd:boolean`
- **ogcf:sfCrosses**(geom1: `ogc:wktLiteral`,
geom2: `ogc:wktLiteral`): `xsd:boolean`
- **ogcf:sfWithin**(geom1: `ogc:wktLiteral`,
geom2: `ogc:wktLiteral`): `xsd:boolean`
- **ogcf:sfContains**(geom1: `ogc:wktLiteral`,
geom2: `ogc:wktLiteral`): `xsd:boolean`
- **ogcf:sfOverlaps**(geom1: `ogc:wktLiteral`,
geom2: `ogc:wktLiteral`): `xsd:boolean`

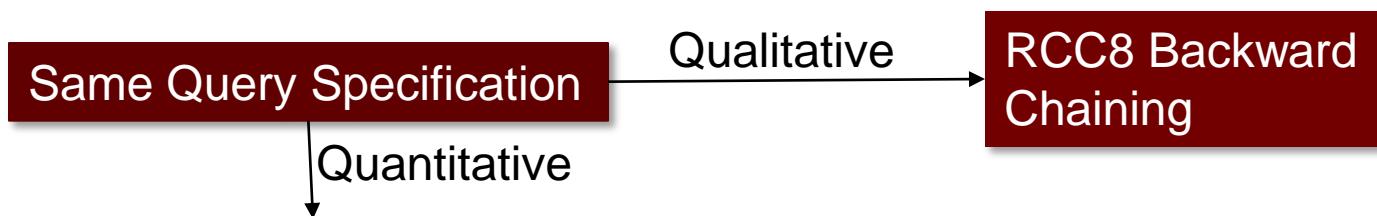
Assumes Simple Features
Relation Family

GeoSPARQL Query Rewrite Extension



Find all water bodies within New Hampshire

```
SELECT ?water
WHERE { ?water rdf:type :WaterBody .
         ?water ogc:rcc8Within :NH }
```



```
SELECT ?water
WHERE { ?water rdf:type :WaterBody .
         ?water ogc:hasDefaultGeometry ?wGeo .
         ?wGeo ogc:asWKT ?wWKT .
         :NH ogc:hasDefaultGeometry ?nGeo .
         ?nGeo ogc:asWKT ?nWKT .
         FILTER(ogc:rcc8Within(?wWKT, ?nWKT)) }
```

Query
Rewrite

Specified with
a RIF rule



IMPLEMENTATION CONSIDERATIONS

Implementing Spatial Operations



- These are standard OGC operators that have been around for some time
- Lots of infrastructure available
 - Open Source

GEOS

Geometry
Engine
Open
Source



pysal

Python Spatial Analysis Library



PostGIS

JTS Topology Suite

- Commercial



ORACLE®
DATABASE

OGC®

Other Considerations



- Have to handle geometries from multiple Spatial Reference Systems simultaneously
 - Normalize to common SRS on-the-fly during computation
 - Pre-normalize ahead of time
- Spatial Indexing very important for performance
 - Normalize to common SRS during indexing

Summary



- GeoSPARQL Defines:
 - Basic vocabulary, Query functions, Entailment component
- Based on existing OGC/ISO standards
 - WKT, GML, Simple Features, ISO 19107
- Uses SPARQL's built-in extensibility framework
- Modular specification
 - Allows flexibility in implementations
 - Easy to extend
- Accommodates qualitative and quantitative systems
 - Same query specification for qualitative (core + topology vocabulary) and quantitative (all components, incl. query rewrite)



GEOSPARQL DEMO WITH ORACLE DATABASE SEMANTIC TECHNOLOGIES

THE FOLLOWING IS INTENDED TO OUTLINE OUR GENERAL PRODUCT DIRECTION. IT IS INTENDED FOR INFORMATION PURPOSES ONLY, AND MAY NOT BE INCORPORATED INTO ANY CONTRACT. IT IS NOT A COMMITMENT TO DELIVER ANY MATERIAL, CODE, OR FUNCTIONALITY, AND SHOULD NOT BE RELIED UPON IN MAKING PURCHASING DECISION. THE DEVELOPMENT, RELEASE, AND TIMING OF ANY FEATURES OR FUNCTIONALITY DESCRIBED FOR ORACLE'S PRODUCTS REMAINS AT THE SOLE DISCRETION OF ORACLE.

Demo Setup



US Census Polygon Data

California Cities, Counties, School Districts

- 6 k triples
- 2 k (Multi)Polygons (**NAD83 Long Lat**)
- Avg 206 points per polygon
- Max 7707 points per polygon

ORACLE®
DATABASE

GeoSPARQL
Prototype *

1

Load RDF from Linked GeoData

- 1) Load into staging table
- 2) Replace `virttrdf:Geometry` with `ogc:wktLiteral`
- 3) Bulk Load into RDF Store

ORACLE®

* Equivalent functionality is available in Oracle Database 11g Release 2 using proprietary datatypes and extension functions.

Demo Setup



US Census Polygon Data

California Cities, Counties, School Districts

- 6 k triples
- 2 k (Multi)Polygons (**NAD83 Long Lat**)
- Avg 206 points per polygon
- Max 7707 points per polygon

Linked GeoData

Relevant Nodes Dataset

- 70 m triples
- 5.9 m points (**WGS84 Long Lat**)



GeoSPARQL
Prototype *

2

Generate and Load RDF from
Census Data

- 1) Load .shp file into Oracle Spatial
- 2) Convert to RDF using SDO_GEOOMETRY.GET_WKT()
- 3) Bulk load into RDF Store



* Equivalent functionality is available in Oracle Database 11g Release 2 using proprietary datatypes and extension functions.

Demo Setup



Linked GeoData

Relevant Nodes Dataset

- 70 m triples
- 5.9 m points (**WGS84 Long Lat**)

ORACLE®
DATABASE

GeoSPARQL
Prototype *

US Census Polygon Data

California Cities, Counties, School Districts

- 6 k triples
- 2 k (Multi)Polygons (**NAD83 Long Lat**)
- Avg 206 points per polygon
- Max 7707 points per polygon

3

Create spatial index with native
SDO_GEOOMETRY object type
(normalize to common **SRS**)

```
exec sem_apis.add_datatype_index(  
    'http://www.opengis.net/geosparql#wktLiteral',  
    options=>'TOLERANCE=1 SRID=8307  
    DIMENSIONS=( (LONGITUDE,-180,180) (LATITUDE,-90,90))',  
    parallel=>4);
```

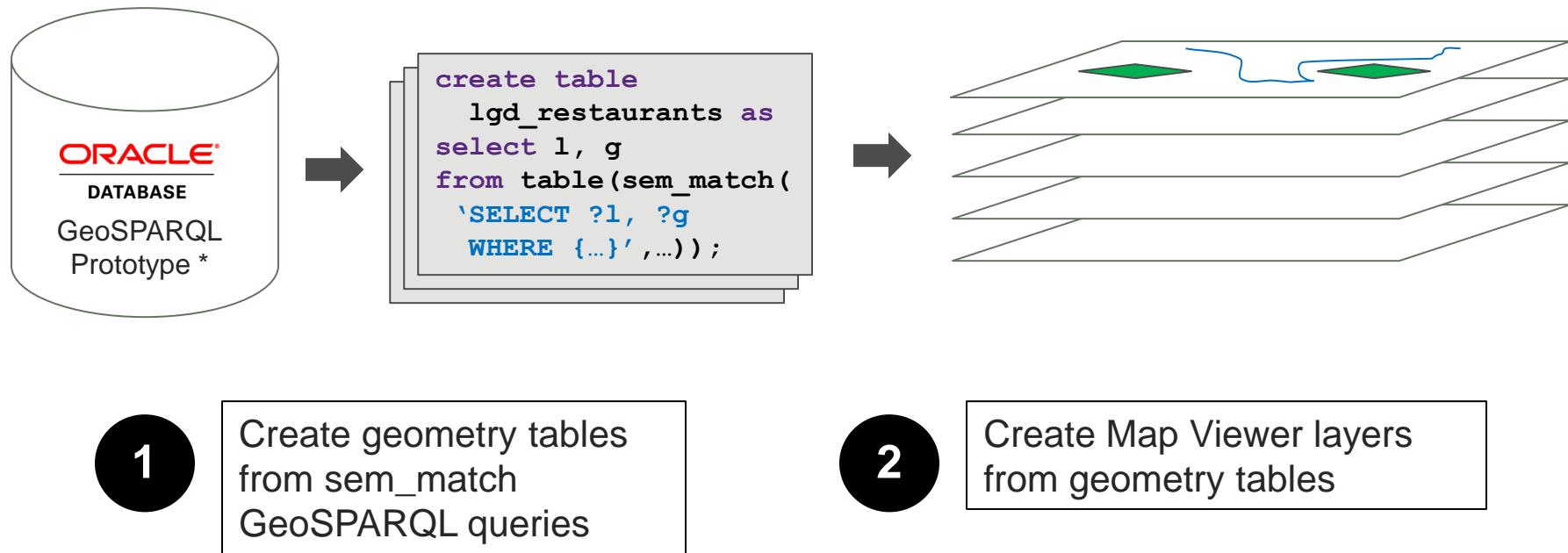
ORACLE®

* Equivalent functionality is available in Oracle Database 11g Release 2 using proprietary datatypes and extension functions.

Map Creation

ORACLE®
FUSION MIDDLEWARE

Map Viewer



ORACLE®

* Equivalent functionality is available in Oracle Database 11g Release 2 using proprietary datatypes and extension functions.



Thanks to all members of
the GeoSPARQL SWG !

Questions?

Linked GeoData Queries - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://localhost:8888/mapviewer/fsmc/tutorial/samples/linkedGeoDataQueries.html

Linked GeoData Queries

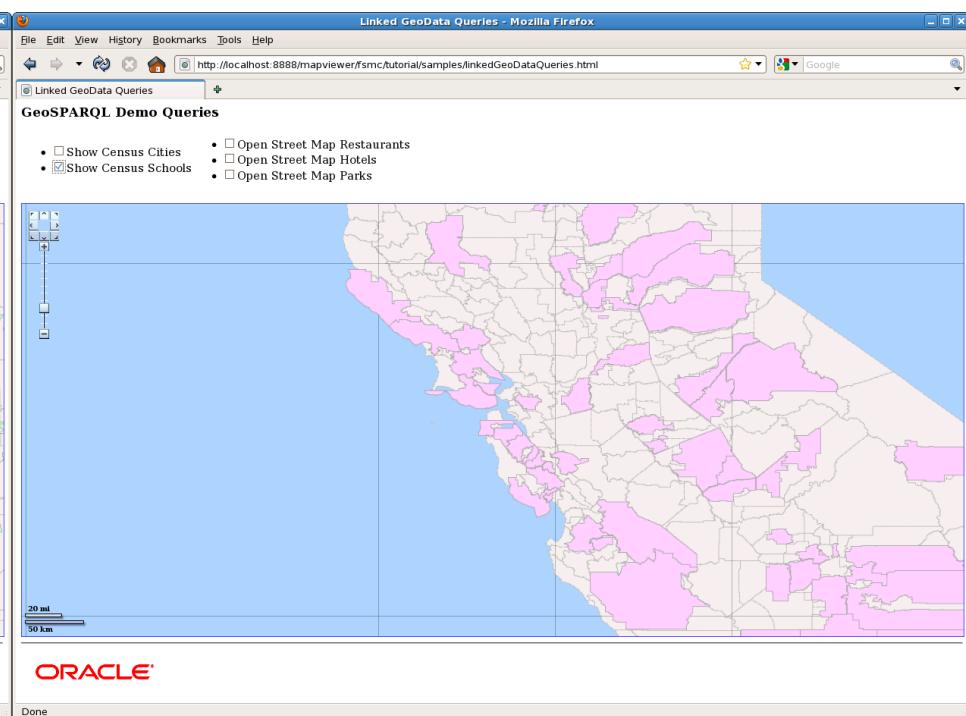
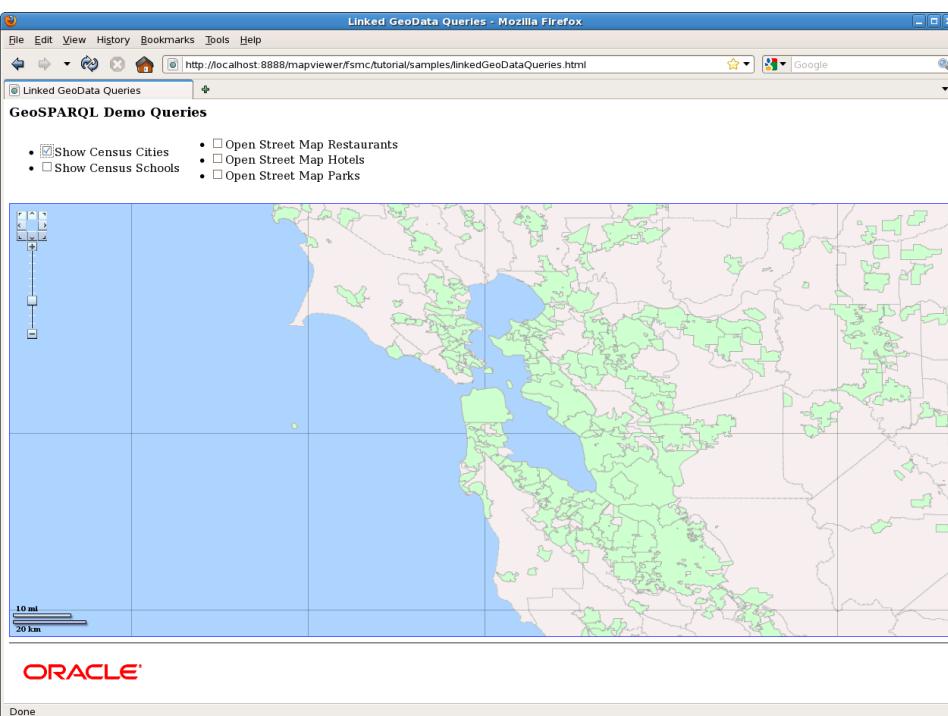
GeoSPARQL Demo Queries

- Show Census Cities
- Show Census Schools
- Open Street Map Restaurants
- Open Street Map Hotels
- Open Street Map Parks

100 mi
200 km

ORACLE®

Done



GeoSPARQL Demo Queries

- Show Census Cities
 - Show Census Schools
 - Open Street Map Restaurants
 - Open Street Map Hotels
 - Open Street Map Parks



ORACLE®

Done

Q1: Open Street Map restaurants near Hilton Hotel

```
SELECT ?x ?l ?g
WHERE
{ ?x rdf:type lgd:Restaurant .
  ?x rdfs:label ?l .
  ?x geo:asWKT ?g
  FILTER (geof:distance(?g,
    "POINT(-122.41 37.7858)"^^geo:wktLiteral,
    uom:KM) <= 0.5)
}
```

File Edit View Navigate Run Versioning Tools Help

Con... x Q1sql x Q2.sql x Q3.sql x Q4.sql x Q5.sql x Q6.sql x

SQL Worksheet History rdfuser

```
-- Q1: Open Street Map Restaurants within 1/2 KM of Hilton Hotel
select x, l, g
from table(sem_match(
'PREFIX geo: <http://www.opengis.net/geosparql#>
PREFIX geof: <http://www.opengis.net/def/geosparql/function/>
PREFIX lgd: <http://linkedgeodata.org/ontology/>
PREFIX uom: <http://xmlns.oracle.com/rdf/geo/uom/>
SELECT ?x ?l ?g
WHERE
{ ?x rdf:type lgd:Restaurant .
?x rdfs:label ?l .
?x geo:asWKT ?g
FILTER (geof:distance(?g, "POINT(-122.41 37.7858)"^^geo:wktLiteral, uom:KM) <= 0.5 )',
sem_models('lg_nodes'),null,null,null,null,
' ));
```

Query Result x

SQL | All Rows Fetched: 65 in 0.393 seconds

	L	G
1	http://linkedgeodata.org/triplify/node1229745497	Fleur de Lys POINT(-122.413 37.7885)
2	http://linkedgeodata.org/triplify/node725100831	Biscuits & Blues POINT(-122.41 37.7873)
3	http://linkedgeodata.org/triplify/node725100726	Max's POINT(-122.41 37.7873)
4	http://linkedgeodata.org/triplify/node621818393	Postrio POINT(-122.411 37.7879)
5	http://linkedgeodata.org/triplify/node808930739	Tad's Steaks POINT(-122.408 37.786)
6	http://linkedgeodata.org/triplify/node383507544	Pinecrest Diner POINT(-122.41 37.787)
7	http://linkedgeodata.org/triplify/node420432891	Denny's POINT(-122.405 37.7842)
8	http://linkedgeodata.org/triplify/node317081652	Johns Grill Live Jazz POINT(-122.407 37.7855)
9	http://linkedgeodata.org/triplify/node808884874	The Old Siam POINT(-122.41 37.7852)
10	http://linkedgeodata.org/triplify/node725082462	Taqueria El Sol POINT(-122.413 37.7867)
11	http://linkedgeodata.org/triplify/node818060774	Supremo Pizza POINT(-122.413 37.7857)
12	http://linkedgeodata.org/triplify/node818060761	Pakwan POINT(-122.413 37.7857)
13	http://linkedgeodata.org/triplify/node808884850	Hana Zen POINT(-122.409 37.7856)
14	http://linkedgeodata.org/triplify/node381943359	Caffe Bella Venezia POINT(-122.414 37.7877)
15	http://linkedgeodata.org/triplify/node808930748	Lori's Diner POINT(-122.408 37.7861)
16	http://linkedgeodata.org/triplify/node927572195	Shalimar POINT(-122.413 37.7863)
17	http://linkedgeodata.org/triplify/node621818430	Honey Honey Cafe & Crepery POINT(-122.412 37.7878)
18	http://linkedgeodata.org/triplify/node725082459	Millenium POINT(-122.413 37.7869)
19	http://linkedgeodata.org/triplify/node725100771	Lefty O'Doul's POINT(-122.408 37.7872)

Q2: Open Street Map hotels within 1 KM of Hilton Hotel

```
SELECT ?x ?l ?g
WHERE
{ ?x rdf:type lgd:TourismHotel .
  ?x rdfs:label ?l .
  ?x geo:asWKT ?g
  FILTER (geof:distance(?g,
    "POINT(-122.41 37.7858)"^^geo:wktLiteral,
    uom:KM) <= 1)
}
```

File Edit View Navigate Run Versioning Tools Help

Con... x Q1.sql x Q2.sql x Q3.sql x Q4.sql x Q5.sql x Q6.sql x

SQL Worksheet History

Connections rdfuser

```
-- Q2: Open Street Map Hotels within 1 KM of Hilton Hotel
select x, l, g
from table(sem_match(
PREFIX geo: <http://www.opengis.net/geosparql#>
PREFIX geof: <http://www.opengis.net/def/geosparql/function/>
PREFIX lgd: <http://linkedgeodata.org/ontology/>
PREFIX uom: <http://xmlns.oracle.com/rdf/geo/uom/>
SELECT ?x ?l ?g
WHERE
{ ?x rdf:type lgd:TourismHotel .
?x rdfs:label ?l .
?x geo:asWKT ?g
FILTER (geof:distance(?g, "POINT(-122.41 37.7858)"^^geo:wktLiteral, uom:KM) <= 1 )},
sem_models('lg_nodes'),null,null,null,null,
' ));
```

Query Result x

SQL | All Rows Fetched: 32 in 0.059 seconds

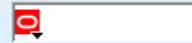
	L	G
1	http://linkedgeodata.org/triplify/node769361407	Hotel Abri
2	http://linkedgeodata.org/triplify/node621818388	Prescott Hotel
3	http://linkedgeodata.org/triplify/node973620751	Herbert Hotel
4	http://linkedgeodata.org/triplify/node621793545	Hotel Beresford Arms
5	http://linkedgeodata.org/triplify/node529780287	Coast Hotel
6	http://linkedgeodata.org/triplify/node275550581	The Mosser
7	http://linkedgeodata.org/triplify/node725096450	Hotel Adagio
8	http://linkedgeodata.org/triplify/node768247786	W Hotel
9	http://linkedgeodata.org/triplify/node416346666	Campton Place Hotel
10	http://linkedgeodata.org/triplify/node412571265	Grand Hyatt San Francisco
11	http://linkedgeodata.org/triplify/node841414817	Hotel Triton
12	http://linkedgeodata.org/triplify/node432822084	Hotel Embassy
13	http://linkedgeodata.org/triplify/node1232832490	The Opal Hotel
14	http://linkedgeodata.org/triplify/node393126126	Hotel Union
15	http://linkedgeodata.org/triplify/node470872778	Pickwick hotel
16	http://linkedgeodata.org/triplify/node416342090	Hotel Whitcomb
17	http://linkedgeodata.org/triplify/node621818350	The Donatello
18	http://linkedgeodata.org/triplify/node621851564	Marine's Memorial Club & Hotel
19	http://linkedgeodata.org/triplify/node313215409	The Westin Saint Francis

Q3: 20 closest Open Street Map hotels to Hilton Hotel

```
SELECT ?x ?l ?g
WHERE
{ ?x rdf:type lgd:TourismHotel .
?x rdfs:label ?l .
?x geo:asWKT ?g
FILTER (geof:distance(?g,
"POINT(-122.41 37.7858)"^^geo:wktLiteral,
uom:KM) <= 1)
}
ORDER BY ASC(geof:distance(?g,
"POINT(-122.41 37.7858)"^^geo:wktLiteral,
uom:KM))
LIMIT 21
```



Q1.sql x | Q2.sql x | Q3.sql x | Q4.sql x | Q5.sql x | Q6.sql x



SQL Worksheet History

```
Run Statement (Ctrl+Enter)
SELECT x, l, g
FROM table(sem_match(
  PREFIX geo: <http://www.opengis.net/geosparql#>
  PREFIX geof: <http://www.opengis.net/def/geosparql/function/>
  PREFIX lgd: <http://linkedgeodata.org/ontology/>
  PREFIX uom: <http://xmlns.oracle.com/rdf/geo/uom/>
  SELECT ?x ?l ?g
  WHERE
    { ?x rdf:type lgd:TourismHotel .
      ?x rdfs:label ?l .
      ?x geo:asWKT ?g
      FILTER (geof:distance(?g, "POINT(-122.41 37.7858)"^^geo:wktLiteral, uom:KM) <= 1) }
    ORDER BY ASC(geof:distance(?g, "POINT(-122.41 37.7858)"^^geo:wktLiteral, uom:KM))
    LIMIT 21',
    sem_models('lg_nodes'),null,null,null,null,
    ')
  ORDER BY sem$rownum;
```

Query Result x

SQL | All Rows Fetched: 21 in 0.139 seconds

x	l	g
1 http://linkedgeodata.org/triplify/node647024126	Hilton San Francisco Union Square	POINT(-122.41 37.7858)
2 http://linkedgeodata.org/triplify/node648270469	Hotel Nikko San Francisco	POINT(-122.409 37.786)
3 http://linkedgeodata.org/triplify/node924624361	Hotel Union Square	POINT(-122.408 37.7858)
4 http://linkedgeodata.org/triplify/node973620751	Herbert Hotel	POINT(-122.408 37.786)
5 http://linkedgeodata.org/triplify/node394952757	Diva	POINT(-122.411 37.7872)
6 http://linkedgeodata.org/triplify/node769361407	Hotel Abri	POINT(-122.408 37.7851)
7 http://linkedgeodata.org/triplify/node313215409	The Westin Saint Francis	POINT(-122.409 37.7877)
8 http://linkedgeodata.org/triplify/node621818350	The Donatello	POINT(-122.41 37.7879)
9 http://linkedgeodata.org/triplify/node621818379	Post Hotel	POINT(-122.411 37.7878)
10 http://linkedgeodata.org/triplify/node621818388	Prescott Hotel	POINT(-122.411 37.7879)
11 http://linkedgeodata.org/triplify/node529780287	Coast Hotel	POINT(-122.413 37.7859)
12 http://linkedgeodata.org/triplify/node621818364	JW Marriott	POINT(-122.41 37.7882)
13 http://linkedgeodata.org/triplify/node621793550	Worth Hotel	POINT(-122.412 37.7877)
14 http://linkedgeodata.org/triplify/node621793523	Emperor Norton Inn	POINT(-122.412 37.7877)
15 http://linkedgeodata.org/triplify/node944758375	The Andrews Hotel	POINT(-122.412 37.7879)

Q4: Open Street Map restaurants inside a query window

```
SELECT ?x ?l ?g
WHERE
{ ?x rdf:type lgd:Restaurant .
  ?x rdfs:label ?l .
  ?x geo:asWKT ?g
  FILTER (geof:sfWithin(?g,
    "POLYGON((-122.25 37.46, -123.27 37.46,
              -123.27 37.60, -122.25 37.60,
              -122.25 37.46))"^^geo:wktLiteral))
}
```

Oracle SQL Developer : /home/oracle/Data/LinkedGeoData/Q4.sql

File Edit View Navigate Run Versioning Tools Help

Con... x Q1.sql x Q2.sql x Q3.sql x Q4.sql x Q5.sql x Q6.sql x

SQL Worksheet History rdfuser

```
-- Q4: Restaurants inside a query window
select x, l, g
from table(sem_match(
PREFIX geo: <http://www.opengis.net/geosparql#>
PREFIX geof: <http://www.opengis.net/def/geosparql/function/>
PREFIX lgd: <http://linkedgeodata.org/ontology/>
PREFIX uom: <http://xmlns.oracle.com/rdf/geo/uom/>
SELECT ?x ?l ?g
WHERE
{ ?x rdf:type lgd:Restaurant .
?x rdfs:label ?l .
?x geo:asWKT ?g
FILTER (geof:sfWithin(?g,
"POLYGON((-122.25 37.46, -123.27 37.46, -123.27 37.60, -122.25 37.60, -122.25 37.46))"^^geo:wktLiteral)) }',
sem_models('lg_nodes'),null,null,null,null,
' ));
```

Query Result x

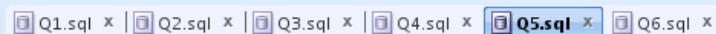
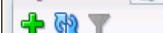
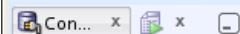
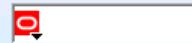
SQL | All Rows Fetched: 67 in 0.735 seconds

	L	G	
1	http://linkedgeodata.org/triplify/node1116384569	Sonoma Valley Bagel	POINT(-122.328 37.534)
2	http://linkedgeodata.org/triplify/node451572042	Nectar Wine Lounge	POINT(-122.346 37.5791)
3	http://linkedgeodata.org/triplify/node598480405	Ketch Joanne	POINT(-122.482 37.5036)
4	http://linkedgeodata.org/triplify/node853379258	Cafe Gibraltar	POINT(-122.476 37.5042)
5	http://linkedgeodata.org/triplify/node527664092	BostonMarket, Jamba Juice, Noahs, Starbucks	POINT(-122.274 37.5607)
6	http://linkedgeodata.org/triplify/node598480402	Barbara's Fish Trap	POINT(-122.485 37.5035)
7	http://linkedgeodata.org/triplify/node452312108	Baskin-Robbins	POINT(-122.434 37.4692)
8	http://linkedgeodata.org/triplify/node451546603	Trapeze	POINT(-122.346 37.5789)
9	http://linkedgeodata.org/triplify/node299404847	Joe's	POINT(-122.333 37.58)
10	http://linkedgeodata.org/triplify/node443809744	Seventh St. Montara	POINT(-122.516 37.5425)
11	http://linkedgeodata.org/triplify/node1151499486	Coyote's Mexican Cafe	POINT(-122.294 37.5117)
12	http://linkedgeodata.org/triplify/node1199004281	Ramano's Macaroni Grill	POINT(-122.298 37.5359)
13	http://linkedgeodata.org/triplify/node334405696	Ramen Club	POINT(-122.352 37.5824)
14	http://linkedgeodata.org/triplify/node267482502	Steelhead Brewery	POINT(-122.346 37.58)
15	http://linkedgeodata.org/triplify/node276581563	Panda Express	POINT(-122.349 37.5769)

Line 2 Column 1 Insert Unix/Mac: LF Editing

Q5: What US Census areas contain SFO airport

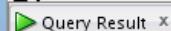
```
SELECT ?x ?l ?t
WHERE
{ ?x rdf:type ?t .
  ?x dc:title ?l .
  ?x geo:asWKT ?g
  FILTER (geof:sfContains(?g,
    "POINT(-122.2230 37.3708)"^^geo:wktLiteral))
}
```



SQL Worksheet History

rdfuser▼

```
-- Q5: What US Census areas contain SFO airport
select x, l, t
from table(sem_match(
PREFIX geo: <http://www.opengis.net/geosparql#>
PREFIX geof: <http://www.opengis.net/def/geosparql/function/>
PREFIX uom: <http://xmlns.oracle.com/rdf/geo/uom/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
SELECT ?x ?l ?t
WHERE
{ ?x rdf:type ?t .
?x dc:title ?l .
?x geo:asWKT ?g
FILTER (geof:sfContains(?g,
"POINT(-122.2230 37.3708)"^^geo:wktLiteral)) }',
sem_models('vm1'),null,null,null,null,
' ALLOW_DUP=T '));
```



SQL | All Rows Fetched: 3 in 0.104 seconds

X	L	T
1 http://www.census.gov/geo/0600000US0608192870	San Mateo	http://www.census.gov/geo/CCD
2 http://www.census.gov/geo/0636390	Sequoia Union High School District	http://www.census.gov/00
3 http://www.census.gov/geo/1600000US0658380	Portola Valley	http://www.census.gov/geo/town

Q6: Pairs of US Census cities and Open Street Map parks

```
SELECT ?t ?l
WHERE
{ ?sf rdf:type ucs:city .
?sf dc:title ?t .
?sf geo:asWKT ?sgeo .
?x rdf:type lgd:Park .
?x rdfs:label ?l .
?x geo:asWKT ?g
FILTER (
  geof:sfWithin(?g,
    "POLYGON((-124.40959 32.534156, -114.13443 32.534156,
              -114.13443 42.009518, -124.40959 42.009518,
              -124.40959 32.534156))"^^geo:wktLiteral)
  &&
  geof:sfContains(?sgeo, ?g) )
} LIMIT 100
```

File Edit View Navigate Run Versioning Tools Help

Con... x Q1.sql x Q2.sql x Q3.sql x Q4.sql x Q5.sql x Q6.sql x

SQL Worksheet History rdfuser

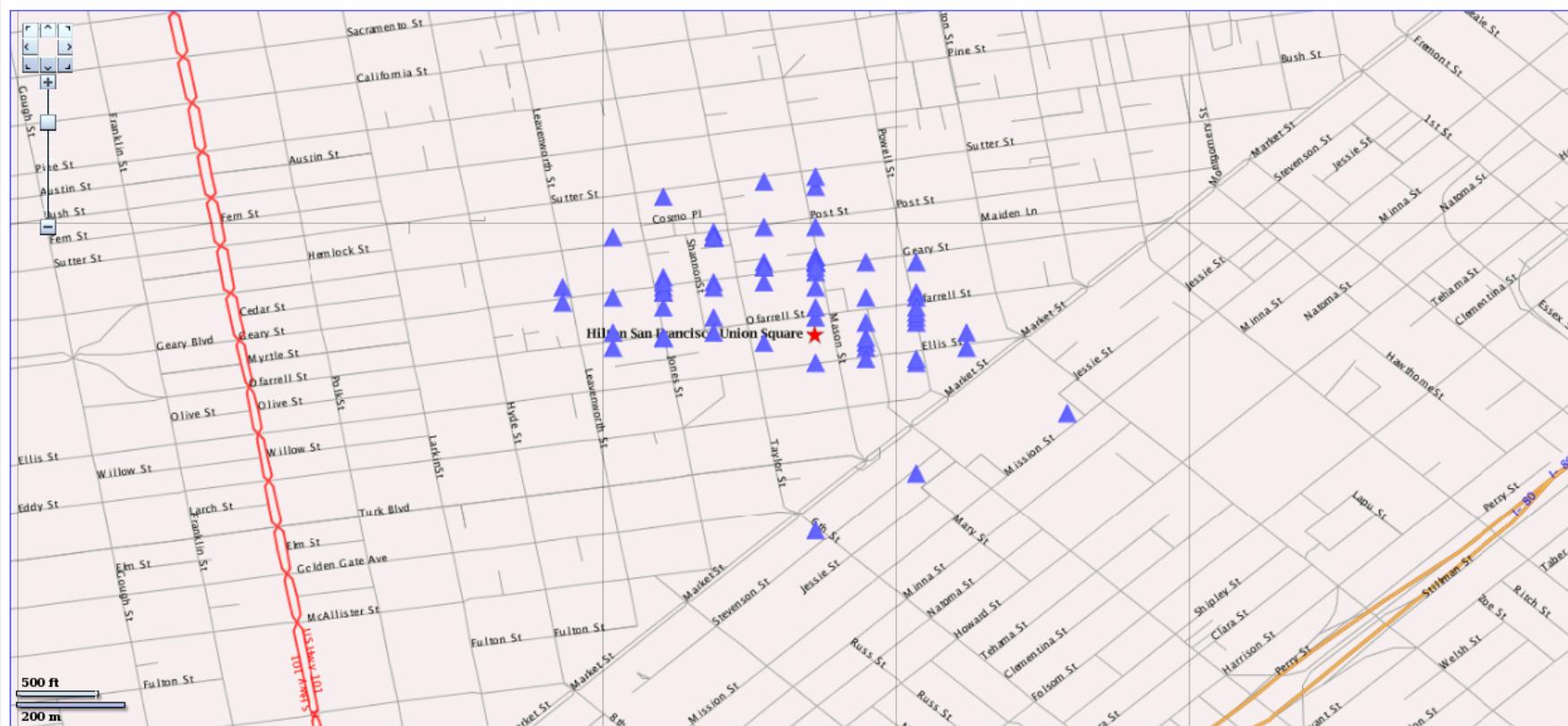
```
-- Q6: Pairs of US Census Cities and Open Street Map Parks
set define off;
select t, 1
from table(sem_match(
'PREFIX geo: <http://www.opengis.net/geosparql#>
PREFIX geof: <http://www.opengis.net/def/geosparql/function/>
PREFIX uom: <http://xmlns.oracle.com/rdf/geo/uom/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX lgd: <http://linkedgeodata.org/ontology/>
PREFIX ucs: <http://www.census.gov/geo/>
SELECT ?t ?1
WHERE
{ ?st rdf:type ucs:city .
?sf dc:title ?t .
?sf geo:asWKT ?sgeo .
?x rdf:type lgd:Park .
?x rdfs:label ?1 .
?x geo:asWKT ?g
FILTER (
geo:sfWithin(?g,
"POLYGON((-124.40959 32.534156, -114.13443 32.534156, -114.13443 42.009518, -124.40959 42.009518, -124.40959 32.534156))"^^geo:wktLiteral)
&&
geo:sfContains(?sgeo, ?g) )
} LIMIT 100',
sem_models('vm1'),null,null,null,null,
' ALLOW_DUP=T HINTO={ LEADING(?g t5 t3 ?sgeo t2 to t1 t4) } '));
```

Statement Output x Query Result x

SQL | All Rows Fetched: 100 in 5.037 seconds

T	L
1 San Francisco	Yerba Buena Park (historical)
2 San Francisco	Vicente Park
3 San Francisco	Lake Merced Sports Center
4 San Francisco	Koshland Park
5 San Francisco	Mountain Lake Park
6 San Francisco	Golden Gate Park Polo Field and Stadium
7 San Francisco	USS San Francisco Memorial
8 San Francisco	San Francisco Maritime National Historical Park
9 San Francisco	Kezar Stadium
10 San Francisco	Queen Wilhelmina Tulip Garden
11 San Francisco	Ina Coolbrith Park

- Show Census Cities
 - Show Census Schools
 - Open Street Map Restaurants
 - Open Street Map Hotels
 - Open Street Map Parks



ORACLE®



Linked GeoData Queries

GeoSPARQL Demo Queries

- Show Census Cities
- Open Street Map Hotels
- Show Census Schools
- Open Street Map Parks



ORACLE®

GeoSPARQL Demo Queries

- Show Census Cities
- Open Street Map Restaurants
- Show Census Schools
- Open Street Map Parks

**ORACLE**