ORACLE®

Developing GeoSPARQL Applications with Oracle Spatial and Graph

Matthew Perry
Ana Estrada
Souri Das
Jay Banerjee

First Joint International Workshop on Semantic Sensor Networks and Terra Cognita
SSN-TC 2015
October 11, 2015

ORACLE

ISWC2015
Bethlehem, Pennsylvania

Safe Harbor Statement

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 decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Agenda

1. GeoSPARQL Overview
2. Implementation in Oracle
3. Demonstration with Oracle 12c

Agenda

1. GeoSPARQL Overview
2. Implementation in Oracle
3. Demonstration with Oracle 12c

OGC GeoSPARQL

- GeoSPARQL – A Geographic Query Language for RDF Data
  - OGC Standard (document 11-052r4)
  - Published in June 2012
  - Submitting Organizations

Australian Government Bureau of Meteorology

Bentley

CSIRO

GeoConnections

Defence Geospatial Information Working Group

USGS science for a changing world

interactive instruments

ORACLE

Ordnance Survey

Raytheon

Traverse Technologies, Inc.

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

This is what GeoSPARQL standardizes

```
: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 .

: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

Extension Functions

```
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)) }
```

ORACLE

Details of ogc:WTKLiteral

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
```

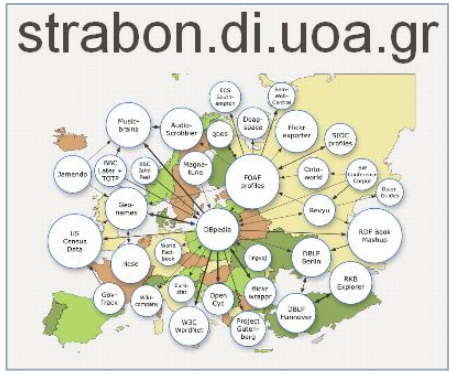
European Petroleum Survey Group (EPSG) maintains a set of CRS identifiers.

GeoSPARQL Spatial Function Library

- Topological Relations
  - `ogcf:relate, ogcf:sfContains, ogcf:sfCrosses, ogcf:sfDisjoint, ogcf:sfEquals, ogcf:sfIntersects, ogcf:sfOverlaps, ogcf:sfTouches, ogcf:sfWithin`
- Distance-based Operations
  - `ogcf:distance, ogcf:buffer`
- Geometry Operations
  - `ogcf:boundary, ogcf:convexHull, ogcf:envelope, ogcf:getSRID,`
- Geometry-Geometry Operations
  - `ogcf:difference, ogcf:intersection, ogcf:symDifference, ogcf:union`

Implementations

Raytheon
Parliament™



ORACLE DATABASE 12c

OPEN SAHARA | uSeekM

Agenda

1. GeoSPARQL Overview
2. Implementation in Oracle
3. Demonstration with Oracle 12c

GeoSPARQL Support in Oracle

- Oracle Spatial and Graph supports the following conformance classes for GeoSPARQL
  - Core
  - Topology Vocabulary Extension (Simple Features)
  - Geometry Extension (WKT, 1.2.0)
  - Geometry Topology Extension (Simple Features, WKT, 1.2.0)
  - RDFS Entailment Extension (Simple Features, WKT, 1.2.0)

ORACLE

OGC wktLiteral Datatype

- Optional leading Spatial Reference System URI followed by OGC WKT geometry string.
  <http://xmlns.oracle.com/rdf/geo/srid/{srid}>
- WGS 84 Longitude, Latitude is the default SRS (assumed if SRS URI is absent)

```
SRS: WGS84 Longitude, Latitude
"POINT(-122.4192 37.7793)"^^ogc:wktLiteral

SRS: NAD27 Longitude, Latitude
"<http://xmlns.oracle.com/rdf/geo/srid/8260>
                 POINT(-122.4181 37.7793)"^^ogc:wktLiteral
```

What Types of Spatial Data are Supported?

- Spatial Reference Systems
  - Built-in support for 1000's of SRS
  - Plus you can define your own
  - Coordinate system transformations applied transparently during indexing and query
- Geometry Types
  - Support OGC Simple Features geometry types
    - Point, Line, Polygon
    - Multi-Point, Multi-Line, Multi-Polygon
    - Geometry Collection
  - Up to 500,000 vertices per Geometry

ORACLE

Spatial Function Library

Standard OGC functions

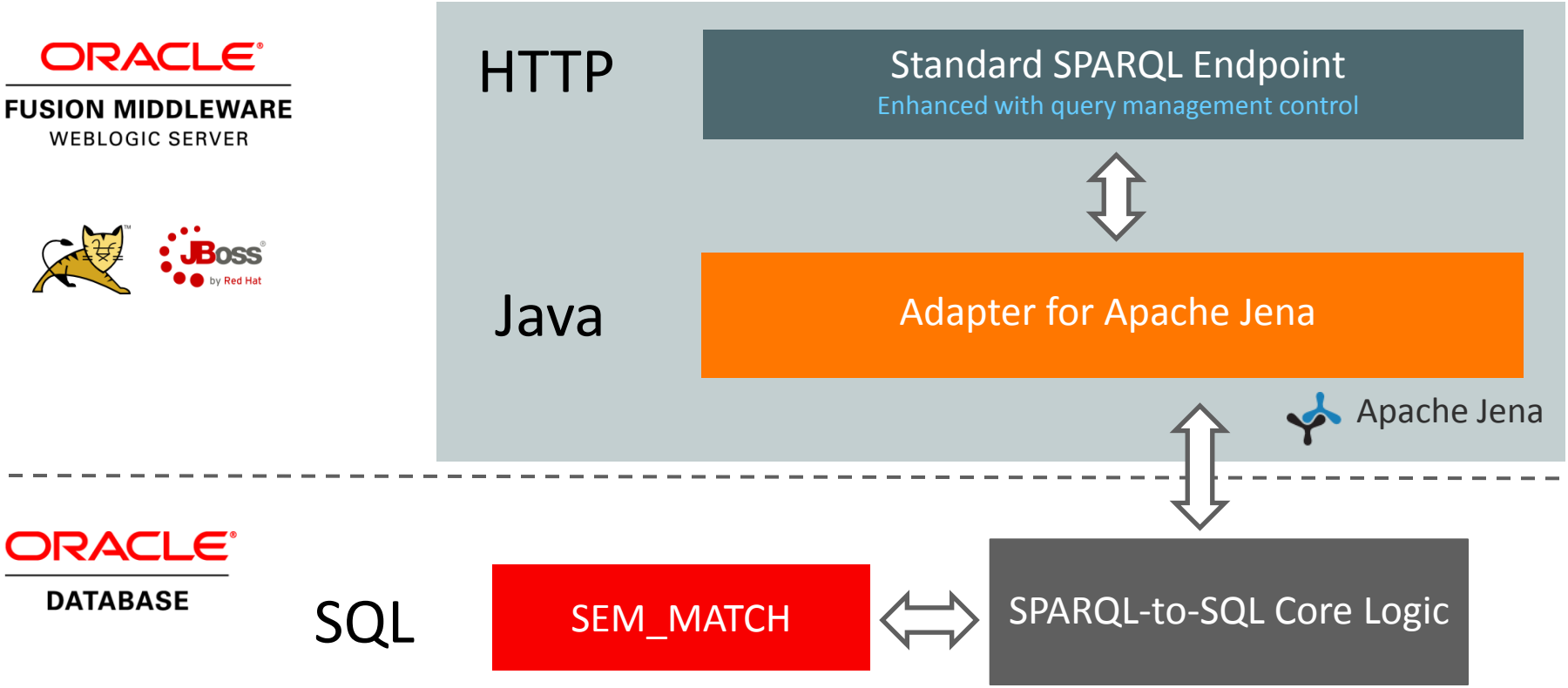
- Topological Relations
  - `ogcf:relate, ogcf:sfContains, ogcf:sfCrosses, ogcf:sfDisjoint, ogcf:sfEquals, ogcf:sfIntersects, ogcf:sfOverlaps, ogcf:sfTouches, ogcf:sfWithin`
- Distance-based Operations
  - `ogcf:distance, ogcf:buffer`
- Geometry Operations
  - `ogcf:boundary, ogcf:convexHull, ogcf:envelope, ogcf:getSRID,`
- Geometry-Geometry Operations
  - `ogcf:difference, ogcf:intersection, ogcf:symDifference, ogcf:union`

Spatial Function Library

Oracle Extensions

- Topological Relations
  - `orageo:relate`
- Distance-based Operations
  - `orageo:distance, orageo:withinDistance, orageo:buffer, orageo:nearestNeighbor`
- Geometry Operations
  - `orageo:area, orageo:length`
  - `orageo:centroid, orageo:mbr, orageo:convexHull`
- Geometry-Geometry Operations
  - `orageo:intersection, orageo:union, orageo:difference, orageo:xor`

SPARQL and SPARQL in SQL Query Architecture



Data Storage Tables

Unique key: PCSGM

RDF_LINK$

S (subj id)	P (pred id)	C (c. obj id)	G (graph id)	M (model id)
100	300	200		1

Partition P1

S (subj id)	P (pred id)	C (c. obj id)	G (graph id)	M (model id)

Partition P2

RDF_VALUE$

pkey

Unique key: Vpfx-Vtyp-Vsfx-Lit-Lang

val. id	vname_prefix	vname_suffix	value_type	Literal type	Lang type
100	http://xyz.com/	geom1	UR		
200	POINT(...)		TL	ogc:wktLiteral	
300	http://opengis..	asWKT	UR		

Spatial Index on ogc:wktLiteral Datatype

- Prepare for spatial querying by creating a spatial index for the ogc:wktLiteral datatype

```
SQL> exec sem_apis.add_datatype_index(
       'http://www.opengis.net/ont/geosparql#wktLiteral',
       options=>'TOLERANCE=0.1 SRID=8307
         DIMENSIONS=((LONGITUDE,-180,180)(LATITUDE,-90,90))');
```

Spatial Indexing

RDF_VALUE$ columns

Target SRS

Helper function to generate SDO_GEOMETRY objects

```
FUNCTION getV$GeometryVal (
  value_type          IN VARCHAR2
, vname_prefix        IN VARCHAR2
, vname_suffix        IN VARCHAR2
, literal_type        IN VARCHAR2
, language_type       IN VARCHAR2
, srid                IN NUMBER
) RETURN        MDSYS.SDO_GEOMETRY
```

RDF_VALUE$

val. id	vname_prefix	vname_suffix	value_type	Literal type	Lang type
100	http://xyz.com/	geom1	UR		
200	POINT(…)		TL	ogc:wktLiteral	
300	http://opengis..	asWKT	UR		

Spatial Indexing

ogcf:sfOverlaps(?geom, "POLYGON((…))")

1) Primary Filter → Function-based R-Tree Index (BBox)

2) Secondary Filter → getV$GeometryVal(value_type, vname_prefix, …)

RDF_VALUE$

val. id	vname_prefix	vname_suffix	value_type	Literal type	Lang type
100	http://xyz.com/	geom1	UR		
200	POINT(…)		TL	ogc:wktLiteral	
300	http://opengis..	asWKT	UR		

Spatial Query Execution

```
SELECT ?s ?l ?wkt
WHERE {
  ?s rdf:type lgd:Monument .
  ?s rdfs:label ?l .
  ?s geovocab:geometry ?geom .
  ?geom ogc:asWKT ?wkt .
  FILTER(
    ogcf:sfWithin(
      ?wkt,
      "POLYGON((-71.44 42.50,
                -71.42 42.40,
                -71.08 42.39,
                -71.03 42.56,
                -71.44 42.50))"^^ogc:wktLiteral))}
```

```
-----------------------------------------------------------------------
| Id  | Operation                                   | Name              |
-----------------------------------------------------------------------
|   0 | SELECT STATEMENT                            |                   |
|   1 |  NESTED LOOPS                               |                   |
|   2 |   NESTED LOOPS                              |                   |
|   3 |    VIEW                                     |                   |
|   4 |     NESTED LOOPS                            |                   |
|   5 |      NESTED LOOPS                           |                   |
|   6 |       NESTED LOOPS                          |                   |
|   7 |        NESTED LOOPS                         |                   |
|   8 |         TABLE ACCESS BY INDEX ROWID         | RDF_VALUE$        |
|*  9 |          DOMAIN INDEX (SEL: 0.100000 %)     | RDF_V$GEO_IDX     |
|  10 |         PARTITION LIST SINGLE               |                   |
|* 11 |          INDEX RANGE SCAN                   | RDF_LNK_CPSGM_IDX |
|  12 |        PARTITION LIST SINGLE                |                   |
|* 13 |         INDEX RANGE SCAN                    | RDF_LNK_CPSGM_IDX |
|  14 |       PARTITION LIST SINGLE                 |                   |
|* 15 |        INDEX RANGE SCAN                     | RDF_LNK_CPSGM_IDX |
|  16 |      PARTITION LIST SINGLE                  |                   |
|* 17 |       INDEX RANGE SCAN                      | RDF_LNK_PSCGM_IDX |
|  18 |    TABLE ACCESS BY INDEX ROWID              | RDF_VALUE$        |
|* 19 |     INDEX UNIQUE SCAN                       | C_PK_VID          |
|  20 |   TABLE ACCESS BY INDEX ROWID               | RDF_VALUE$        |
|* 21 |    INDEX UNIQUE SCAN                        | C_PK_VID          |
-----------------------------------------------------------------------
```

Agenda

1. GeoSPARQL Overview
2. Implementation in Oracle
3. Demonstration with Oracle 12c

Demonstration with Oracle 12c

ORACLE DATABASE 12c

ORACLE FUSION MIDDLEWARE

Oracle Big Data Lite VM
+
Spatial & Graph Support
+
Map Viewer

HistoricThing.node
4.1 Million Triples
268K Points (WGS 84)

Create semantic model → Bulk load into model → Create spatial index → Execute GeoSPARQL queries → Create Map Viewer layer

Integrated Cloud

Applications & Platform Services

ORACLE

ORACLE®