GeoSPARQL - A geographic query language for RDF data

A proposal for an OGC Draft Candidate Standard

Xavier Lopez, Oracle
Objectives

• Spatially-enable SPARQL “endpoints”
  • Enterprise knowledge bases
  • Linked Open Data
    • GeoNames
    • data.gov
    • data.gov.uk
    • Online mapping services
• Build on existing standards
  • W3C Semantic Web (RDF, OWL, SPARQL)
  • OGC (Simple Features, Spatial Relations)
• Leverage $B investments in geospatial data
• Provide foundation for spatial “reasoning”
GeoSPARQL WG Submitters

- Oracle
- US Geological Survey
- UK Ordnance Survey
- BBN Technologies
- Orbis Technologies
- Interactive Instruments

- We Invite others to join as Charter members!
Overview

- **Objective:** Define minimal RDF schema for geospatial data
  - Based on General Feature Model
  - Provides a standardized vocabulary for representing linked geodata
  - Provides a standardized vocabulary for writing SPARQL queries against geospatial RDF data
- **GeoSPARQL** reuses common geometry serialization formats
  - Provides FILTER functions that understand GML, KML, WKT strings encoded as RDF Literals.
- Relatively straightforward to implement on top of a spatial database, GIS, file system
- Other possibilities: OWL-based spatial reasoning
Handling Spatial Data in RDF: Objectives

- To develop best practices for managing spatial data in RDF.
- To define structured vocabulary and semantics for geographic features (metadata) and relationships.
  - E.g.: `ogc:dimension` property on a Spatial Object can capture the dimension of the object.
- To manage geographic data as RDF terms using standard serialization formats.
  - E.g.: GML captured as text with appropriate RDF literal type.
- To add the ability to answer queries involving geographic features and relationships.
  - E.g.: The `ogc:touchs` relationship can link two Spatial Objects in a SPARQL triple pattern.
**Goal: GeoSPARQL**

Find all land parcels with some type of commercial zoning that touch some arterial street

```
SELECT ?parcel ?hwy
WHERE { ?parcel rdf:type :Commercial .
    ?parcel rdf:type ogc:GeometryObject .
    ?hwy rdf:type :Arterial_Street .
    ?hwy rdf:type ogc:GeometryObject .
    ?parcel ogc:touches ?hwy }
```

Find all land parcels with some type of residential zoning that are within 10 KM of a boating lake

```
SELECT ?parcel ?feature
WHERE { ?parcel rdf:type :Residential .
    ?parcel rdf:type ogc:GeometryObject .
    ?feature ogc:hasGML ?fGML .
    FILTER (ogc:within_distance(?pGML, ?fGML, 10, "km"))
```
Spatial Ontology – Requirements*

- Language should be able to represent
  - Spatial concepts
    - Point, Line, Polygon, …
  - Spatial and non-spatial properties of geographic features
    - geometry and population of a Census block group
  - Metadata for each spatial object
    - dimension, SRID, …
  - Relationships between spatial objects
    - binary: touches, contains, … n-ary: within_distance
  - Specialization/generalization concept hierarchies
    - Point is a specialization of Geometry
  - Simple composition hierarchies – concepts made of sets of other concepts
    - Waterfront_Property is the set of all Land_Parcells that touch some Water_Body

Ontology for Spatial Modeling

- An **agreement** on the **vocabulary** used to represent spatial concepts
- An **agreement** on the encoding of (some) **spatial semantics** using OWL/RDFS vocabulary

**Building Blocks:**

- OGC Simple Features Specification
- WKT
- KML
- GML
Geometry Class Hierarchy

Properties for OGC:GeometryObject

Listed properties all have rdfs:domain ogc:GeometryObject

- **Datatype Properties**
  - ogc:dimension :range xsd:integer
  - ogc:srid :range xsd:integer
  - ogc:isEmpty :range xsd:boolean
  - ogc:isSimple :range xsd:boolean
  - ogc:envelope :range ogc:GMLType
  - ogc:boundary :range ogc:GMLType
  - ogc:hasGML :range ogc:GMLType

- **Object Properties**
  - ogc:equals :range ogc:GeometryObject
  - ogc:disjoint :range ogc:GeometryObject
  - ogc:intersects :range ogc:GeometryObject
  - ogc:touches :range ogc:GeometryObject
  - ogc:crosses :range ogc:GeometryObject
  - ogc:within :range ogc:GeometryObject
  - ogc:contains :range ogc:GeometryObject
  - ogc:overlaps :range ogc:GeometryObject

Detailed geometry information encoded as RDF XML Literal (GML in this case)

Properties taken from Simple Features Specification
An Example Query

Design Decision: Encoding Spatial Data as XML Literal (GML)

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 ?wGML
  ?water ogc:hasGML ?wGML .
  :Route_3 ogc:hasGML ?r3GML .
  FILTER(ogc:within_distance(?r3GML, ?wGML, 1, "km"^^xsd:string)) }
```

Consistent way to pass geometry information around
Expressing Spatial Queries with SPARQL

- Types of spatial properties, operations and relationships
  - Descriptive datatype properties (e.g., dimension)
  - Binary relations (e.g., touches, intersects, contains)
  - Parameterized relations (e.g., within distance)
  - Operations that produce new objects (e.g., buffer, union, intersect)
- SPARQL features to use (rely on standard SPARQL syntax)
  - Triple patterns
  - Extensible FILTER functions
- Issues
  - What should be in a FILTER clause and what should be in a graph pattern?
  - How do we test relationships with transient spatial objects?
  - What should the arguments be to Spatial FILTER functions?
Next Steps

• Draft Candidate OGC Specification
  • Spatial Query for SPARQL
  • Spatial Ontology
• Inaugural GeoSPARQL WG Meeting
  • Silver Springs, Thursday, June 17, 2010
  • Finalize WG charter, WG members
• Open Review
  • Late 2010
• Interoperability Tests (2010-11)
  • USGS, UK OS, others?
• Future OGC Working Groups?
  • Catalog Services
  • Feature Services