Using Location in Cloud Applications with Python, Node.js, and More

Siva Ravada
Senior Director of Engineering
Oracle Spatial and Graph Development
Overview

Enabling Spatial Analytics in Cloud applications

- Oracle REST Data Services (ORDS)
- Node.js
- Python
Oracle Spatial and Graph

- Most widely-used Spatial database
- Part of Oracle Database – Secure, high performance, scalable
- Optimized for Exadata
- Leverages existing skills – SQL, Java, and XML
- Easy to program using new scripting and other cloud enabled languages
- Simpler, more scalable applications
Oracle Big Data Spatial and Graph
Bringing location analytics to Hadoop and NoSQL platforms

Data Harmonization using any location attribute (address, postal code, lat/long, placename, etc).

Categorization and filtering based on location and proximity

Preparation, validation and cleansing of Spatial and Raster data

Visualizing and displaying results on a map

Spatial querying and analysis of Hadoop data with SQL
Oracle REST Data Services (ORDS)
Spatial with Oracle Rest Data Services (ORDS)

ORDS Overview

- Java J2EE mid tier application
  - Also supports “Standalone” mode for development
- For input, maps/binds URI to SQL and PL/SQL
- For output, transforms results to JSON and other formats
- Ships with Oracle Database and SQL Developer
Oracle REST Data Services

HTTP(s) API App-Dev with Relational Tables in Oracle Database

https://myhost/ords/hr/customerorders/1001

```json
{  "custno": 1001,  "name": "Scott King",  "address": "500 Main street, Innovation CA",  "orders": [  {  "orderno": 404,  "orderdate": "Feb 27, 2014",  "status": "in process"}  ],  {  "orderno": 303,  "orderdate": "Feb 26, 2014",  "status": "in process"} }
```

ORDS maps standard URI requests to corresponding relational SQL (not schemaless): e.g. SQL SELECT from customers and orders table.
ORDS also transforms the SQL results into the highly popular JavaScript Object Notation (JSON), other formats include HTML, binary and CSV.
Fully committed to supporting any and all standards required by Fusion / SaaS / FMW; we are actively engaged in the ongoing dialog.
Spatial with Oracle Rest Data Services (ORDS)

Use cases

• Create REST endpoints for tables and views with geometries
• Create REST endpoints for Spatial queries and procedures
• Create REST endpoints wrapped as GeoJSON

Approach

• Programming through PL/SQL and Java APIs
• User friendly UI in SQL Developer
Spatial with Oracle Rest Data Services (ORDS)

Demo 1

 Shows how to publish results of a Spatial query as a web service
Node.js
Spatial with Node.js

Node.js overview

• Built for developing server side applications in JavaScript
  – JavaScript instead of Java
• Open-source JavaScript run-time, runs outside of a browser
• Designed for asynchronous, event driven applications
• Built to handle streaming content
• Easily creates REST APIs
Node.js Architecture

https://www.pabbly.com/tutorials/node-js-event-loops
Spatial with Node.js

Use cases

• **Combine** content and analyses from Spatial DB with other geospatial resources and publish REST API
  – For example, publish items from Spatial DB based on containment in a region defined in an external source

• Subscribe to Oracle Database’s Continuous Query Notification (CQN) for Spatial content for near real-time monitoring app.
  – For example, monitoring Tracking Server notifications

• Deploy to Oracle Application Container Cloud Service to easily leverage Spatial in Oracle DBCS
Spatial with Node.js

Approach

• Leverage node-oracledb add-on for Node.js
  – Robust connectivity to Oracle database
  – Handles LOBs
  – Open source, maintained by Oracle

• Leverage github examples

• Since Node.js is Javascript, GeoJSON is a natural. So leverage SDO_UTIL.TO/FROM_GEOJSON()
Demo 2: Shows how to make Spatial calls to DB and web services from Node.js application

1. Web service request to get a fire perimeter

2. Response in JSON

3. Query DB to find the RR affected by the fire perimeter

4. Response in JSON

Node.js application: node-spatial-rest-api.js

USGS Service for Fire perimeters
Demo 3: Shows how to build a simple tracking and notification application

- Define countries as regions of interest to track moving objects
- Generate random moving objects on the globe
- Generate notifications when a moving object moves into a country
Location tracking Server

• Track objects against a set of polygons and generate notifications when the moving objects enter the regions of interest

• Track objects within a set of polygons and generate notifications when the moving objects leaves the regions of interest
Continuous Query Notification (CQN)

• CQN lets an application register queries with the database
  – object change notification (the default)
  – query result change notification

• If a query is registered for query result change notification (QRCN), the database notifies the application whenever a transaction changes the result of the query and commits

• CQN provides a push mechanism for generating notifications for changes in the DB
Demo 3: Software components

CQN and Location Tracking Server

1- server.js app registers notifications table with CQN
2- tracking server queues messages
3- dequeued messages are inserted to table
4- CQN pushes notifications
5- server.js processes notifications and takes actions
Python
Spatial with Python

Python overview

• Very popular general-purpose open-source language
• Interpreted, high level, easy to get started
• Widely adopted for data science and general data analysis
• Tons of available well-maintained and widely adopted and libraries
Spatial with Python

Use cases

• Pull content from Spatial to Python to perform specialized geostatistical analysis

• Invoke Spatial from Python for large scale spatial processing such as spatial join, dynamic segmentation etc and retrieve results for further work

• Combine content from Spatial with other geospatial sources in Python such as ShapeFiles, GeoJSON files and services
Spatial with Python Approach

- Leverage `cx_Oracle` module
  - Robust connectivity to Oracle database
  - Open source, maintained by Oracle
  - SDO_GEOMETRY bindings, LOB handling
  - see github examples incl. Spatial
- Leverage geospatial library ecosystem
  - GeoPandas, Shapely, PySAL, etc
- Use notebook (i.e. Jupyter) to analyze and visualize
Jupyter Notebook

• Jupyter Notebook is an open-source web application that allows you to create and share code and documents

• Benefits of this environment
  – you can document your code
  – run code, look at the outcome
  – visualize data and see the results without leaving the environment
  – A handy tool for performing end to end data science workflows – data cleaning, statistical modeling, building and training machine learning models, visualizing data
Demo 4: Shows how to use shapefile data to query against Spatial database and visualize result

• Regions of interest provided in a shapefile
• Pick one region of interest from the file and use it to query against the spatial database
• Visualize results on a map and as a bar graph
• All these tasks can be done from one environment
We have "regions of interest" as a local ShapeFile. We will retrieve auto crash data from Oracle for locations within the one of the regions for analysis in Python.

```python
%matplotlib inline

# geospatial data operations and basic plotting
from shapely.wkb import loads
import geopandas as gpd
import matplotlib.pyplot as plt

# robust Oracle database access
import cx_Oracle

Load ShapeFile to GeoDataFrame

```gpd.read_file('/opt/data/TX_ESC_Regions/TX_ESC_Regions.shp')
regions_gdf.set_index('REGION', inplace=True)

Display the region of interest (Region 13)

```f, ax = plt.subplots(1, figsize=(8, 7))
regions_gdf.plot(ax=ax, linewidth=5, edgecolor='black')
```
The Spatial & Graph SIG User Community

We are a vibrant community of customers and partners that connects and exchanges knowledge online, and at conferences and events.

Join us online
tinyurl.com/oraclespatialcommunity

LinkedIn  Google+  IOUG  IOUG SIG

@oraspatsig

oraspatialsig@gmail.com
Call for Speakers now open!
Submit an abstract to share your use case or technical session

Analytics and Data Summit
All Analytics. All Data. No Nonsense.
March 12 – 14, 2019

Formerly called the BIWA Summit with the Spatial and Graph Summit
Same great technical content...new name!
www.AnalyticsandDataSummit.org
Questions?