Oracle Spatial 101 for APEX Developers

Karin Patenge

Product Manager | Oracle Spatial & Graph Technologies

APEX World 2024 (EN) | Amersfoort/NL | March 21, 2024

NL ORACLE USER GROUP

„Everything is related to everything else. But near things are more related than distant things.”

Tobler's first law of geography

Agenda for today

- What is Oracle Spatial?
- How do spatial data look like?
- How are spatial data stored in the Oracle Database?
- How can you access and use spatial data?
- Your choices for geocoding address data
- ADB Geocoder SQL API with APEX
- Custom Background Maps in APEX
- Q&A
- More information

What is Oracle Spatial?

Oracle Spatial is free

- ✓ Every edition
- ✓ Every Oracle DB service
- ✓ Wherever it is deployed

Oracle Spatial Technologies

*OGC = Open Geospatial Consortium

Oracle Database

- In-depth support for different data models and types, including
  - Geospatial data
    - Vector data (aka Simple features, geometries)
    - Network data
    - Geo-referenced raster imagery
    - 3D point cloud data
    - Topology data
    - Streaming point data
  - Spatial processing
  - Spatial analysis and mining
- Included in
  - All editions of the Oracle database
  - All Database Cloud Services
- Available through Oracle Cloud Infrastructure accounts including Always Free

Deployable Services

- Map visualization
- Geocoding
- Routing
- Network models
- Publishing (OGC* Web Services)

Interfaces

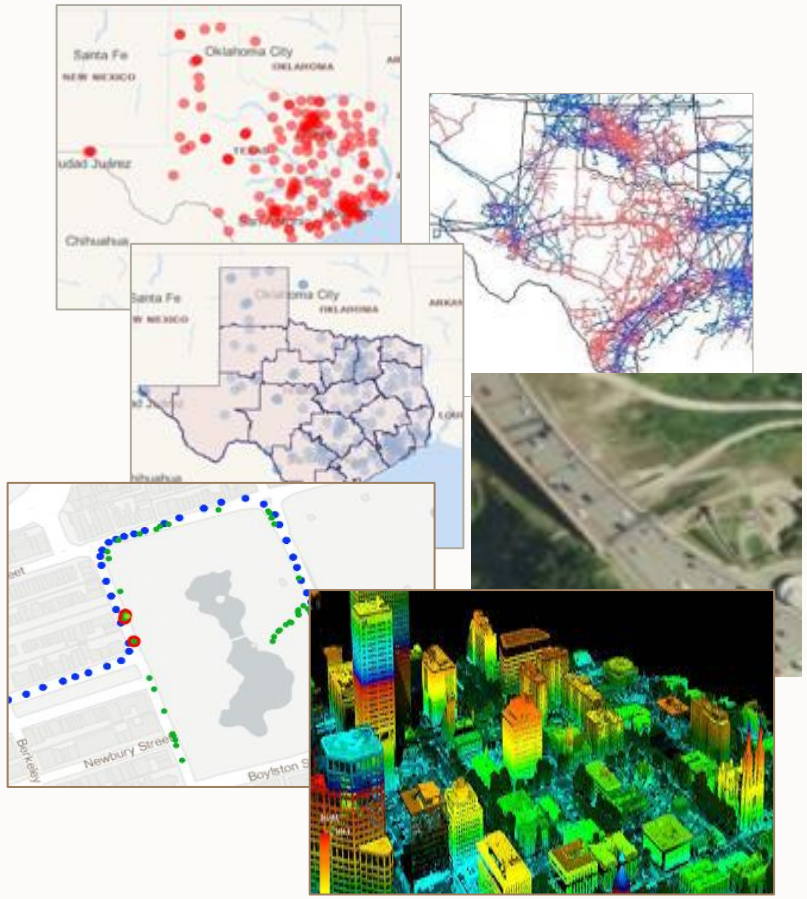
- SQL, PL/SQL
- Java, Python, .NET, Node.js
- REST

Tools

- Oracle Spatial Studio (No-code visual interface)
- SQL Developer (IDE)

Core functionality of the Oracle Database

Native geospatial data management



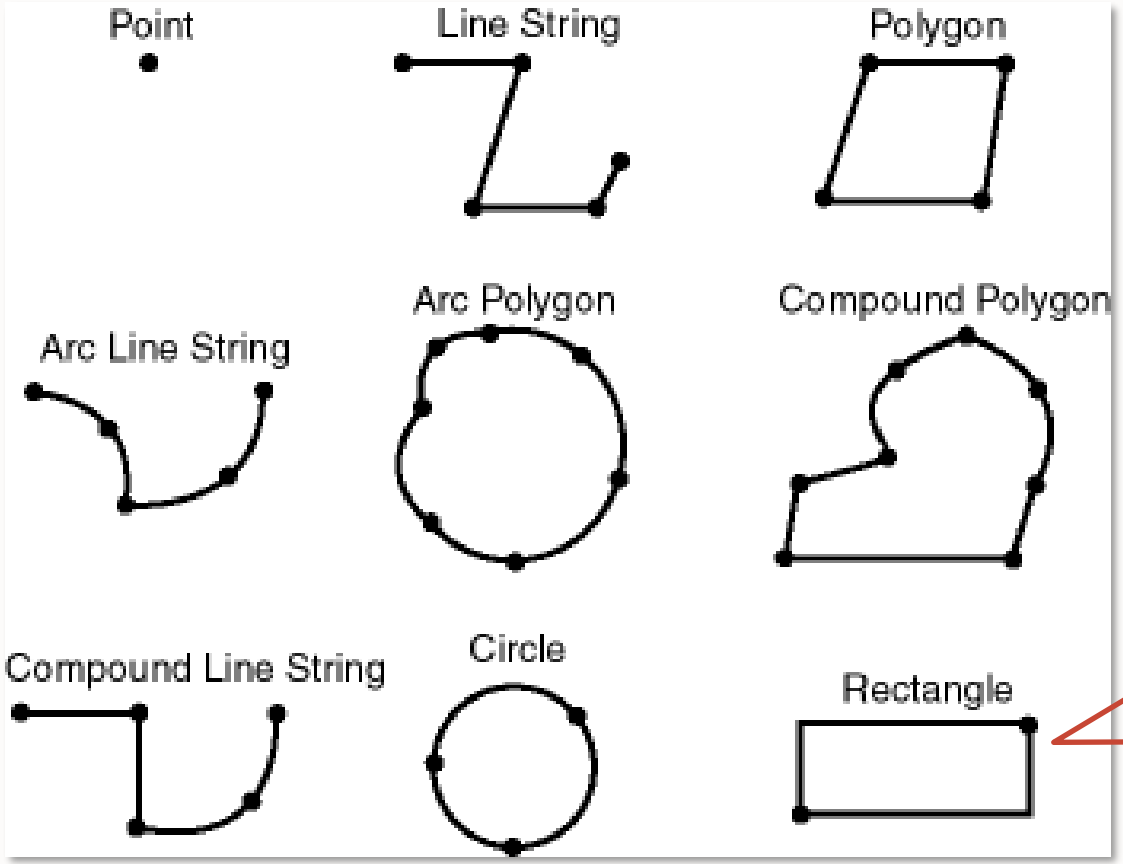
Analyze and relate information based on location

- Which properties are contained in a flood zone?
- Which schools are within a mile of a hurricane path?
- Where are the 5 nearest maintenance facilities?
- Where can we deliver within 35 minutes?
- What is the vegetation index of certain regions in a given country or state?

How do (raw) spatial data look?

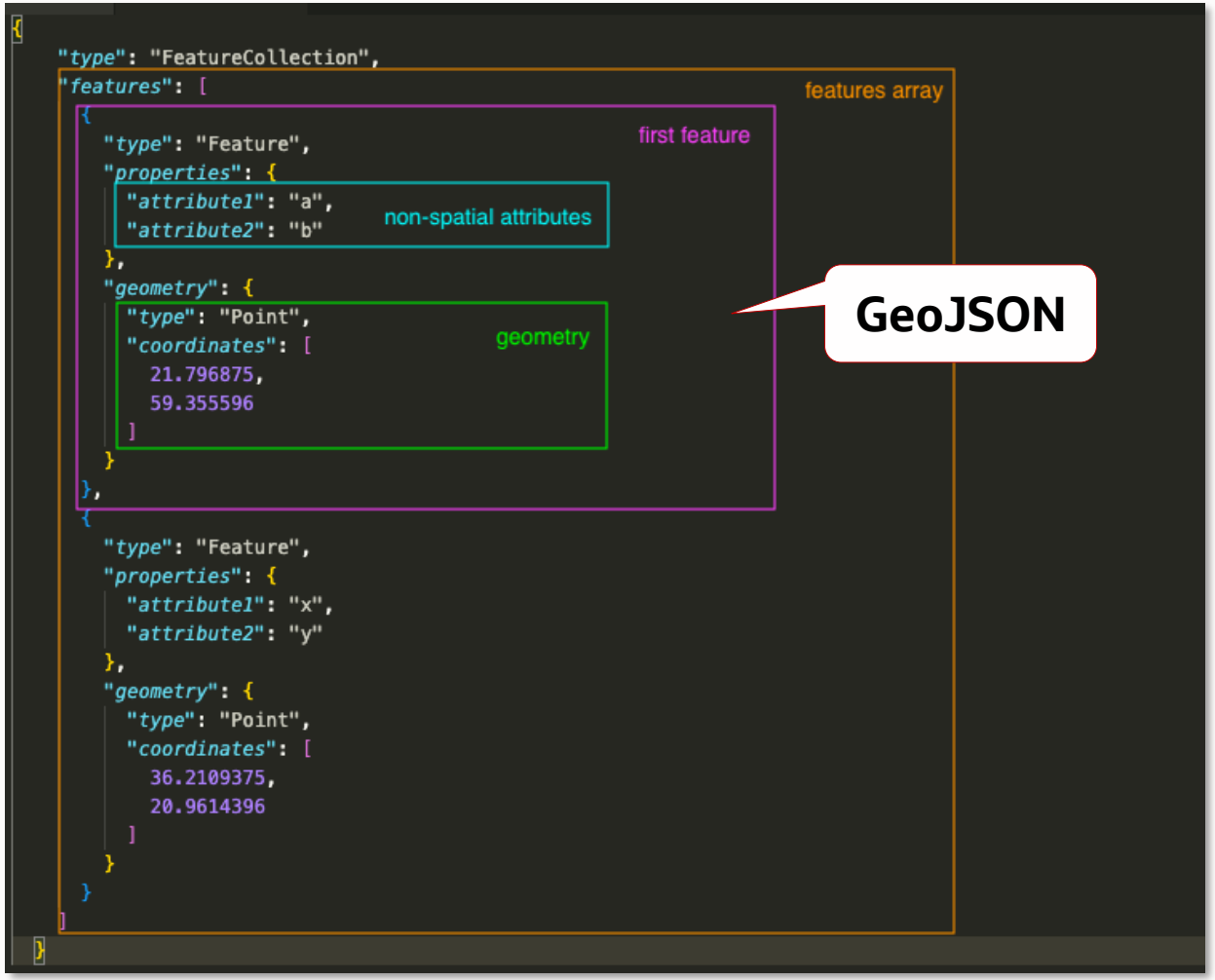
Simple Features

Open Geospatial Consortium Standards: Simple Feature Access - www.ogc.org/standards/sfa



APEX 23.2 supports
“Optimized Rectangles”

Simple Features



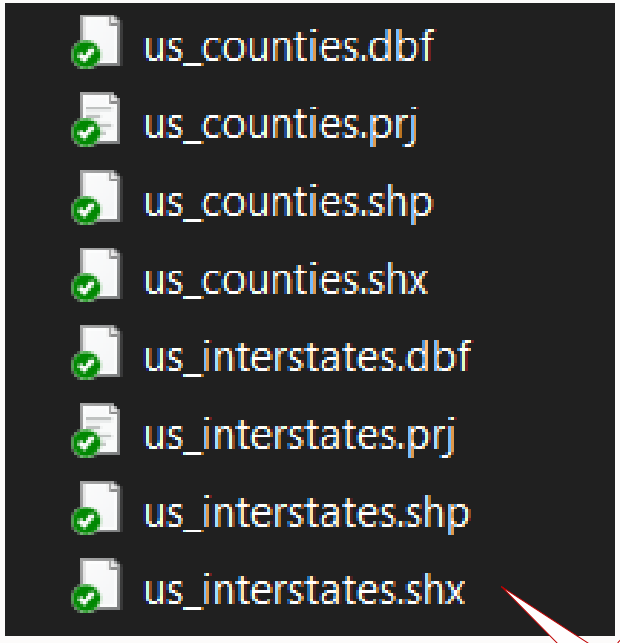

```
{
  "type": "FeatureCollection",
  "features": [
    {
      "type": "Feature",
      "properties": {
        "attribute1": "a",
        "attribute2": "b"
      },
      "geometry": {
        "type": "Point",
        "coordinates": [
          21.796875,
          59.355596
        ]
      }
    },
    {
      "type": "Feature",
      "properties": {
        "attribute1": "x",
        "attribute2": "y"
      },
      "geometry": {
        "type": "Point",
        "coordinates": [
          36.2109375,
          20.9614396
        ]
      }
    }
  ]
}
```

Simple Features

```
STORE_NAME, STORE_TYPE, POSTAL_CODE, LONGITUDE, LATITUDE
Marshall,WHOLESALE,75672,-94.35579,32.50555
Austin-2,RETAIL,78745,-97.75902,30.20538
Grand Prairie,RETAIL,75050,-96.99732,32.76199
Pasadena,WHOLESALE,77505,-95.15763,29.64587
Sulphur Springs,RETAIL,75482,-95.5972,33.12357
San Angelo-1,RETAIL,76901,-100.49426,31.44721
Houston-6,WHOLESALE,77025,-95.43044,29.67853
Mount Pleasant,WHOLESALE,75455,-94.98048,33.17527
Palestine,RETAIL,75802,-95.63011,31.77344
```

Lon/Lat

Simple Features



ESRI Shape Files

Simple Features

```
POINT(6 10)
LINESTRING(3 4,10 50,20 25)
POLYGON((1 1,5 1,5 5,1 5,1 1),(2 2, 3 2, 3 3, 2 3,2 2))
MULTIPOINT((3.5 5.6),(4.8 10.5))
MULTILINESTRING((3 4,10 50,20 25),(-5 -8,-10 -8,-15 -4))
MULTIPOLYGON(((1 1,5 1,5 5,1 5,1 1),(2 2, 3 2, 3 3, 2 3,2 2)),((3 3,6 2,6 4,3 3)))
GEOMETRYCOLLECTION(POINT(4 6),LINESTRING(4 6,7 10))
POINT ZM (1 1 5 60)
POINT M (1 1 80)
POINT EMPTY
MULTIPOLYGON EMPTY
```

Well-Known Text (WKT)

Simple Features

```kml
<?xml version="1.0" encoding="utf-8"?>
<kml xmlns="http://www.opengis.net/kml/2.2">
  <Document>
    <Placemark>
      <name>Portland</name>
      <Point>
        <coordinates>-122.681944,45.52,0</coordinates>
      </Point>
    </Placemark>
    <Placemark>
      <name>Rio de Janeiro</name>
      <Point>
        <coordinates>-43.196389,-22.908333,0</coordinates>
      </Point>
    </Placemark>
    <Placemark>
      <name>Istanbul</name>
      <Point>
        <coordinates>28.976018,41.01224,0</coordinates>
      </Point>
    </Placemark>
    <Placemark>
      <name>Reykjavik</name>
      <Point>
        <coordinates>-21.933333,64.133333,0</coordinates>
      </Point>
    </Placemark>
```

KML

Simple Features

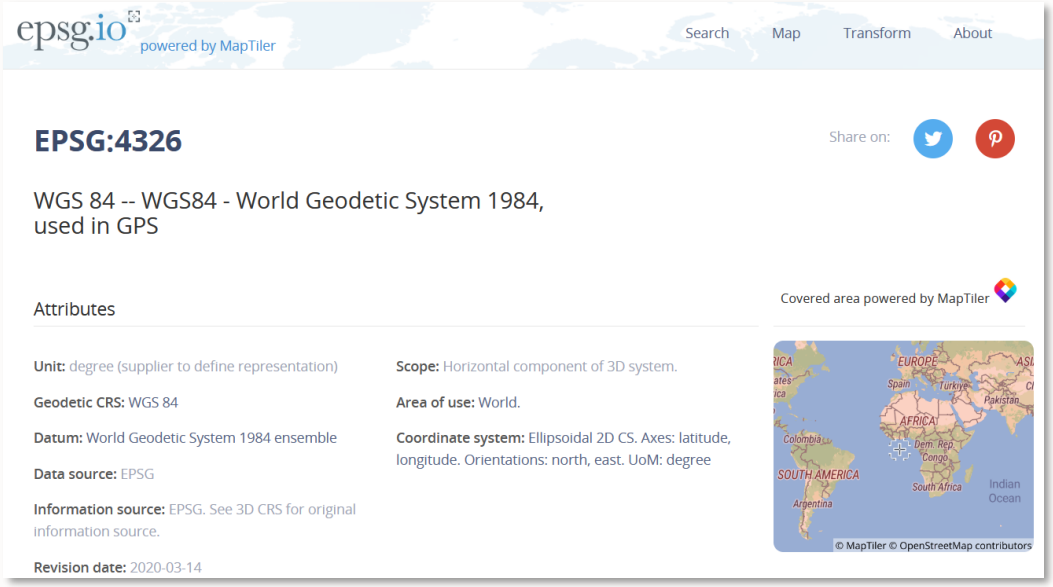
```
<gml:Polygon>
         <gml:outerBoundaryIs>
                 <gml:LinearRing>
                         <gml:coordinates>0,0 100,0 100,100 0,100 0,0</gml:coordinates>
                 </gml:LinearRing>
         </gml:outerBoundaryIs>
      </gml:Polygon>
      <gml:Point>
         <gml:coordinates>100,200</gml:coordinates>
      </gml:Point>
      <gml:LineString>
         <gml:coordinates>100,200 150,300</gml:coordinates>
      </gml:LineString>
```

GML

How are spatial data mapped to the earth?

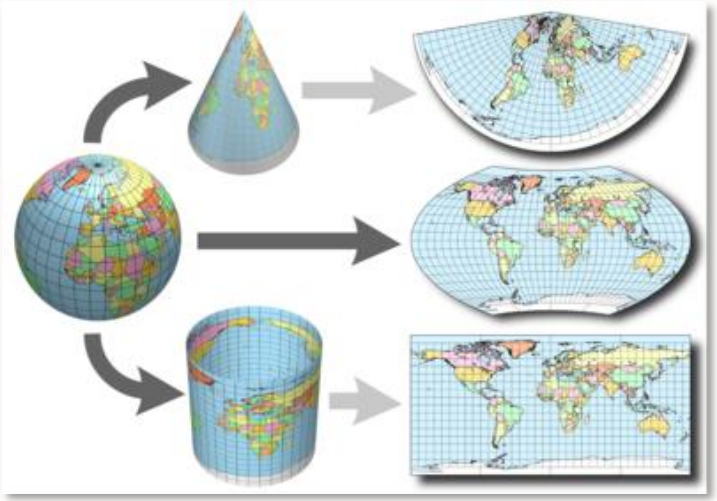
Coordinate reference systems

- Identified by their **SRID** (SR = Spatial reference)
- Most commonly known is WGS 84 (GPS) with SRID = 4326
- Often used in APEX is also SRID = 3857 (WGS 84 / Pseudo-Mercator -- Spherical Mercator, Google Maps, OpenStreetMap, Bing, ArcGIS, ESRI)



Source: epsg.io

Geodetic vs. projected coordinate reference systems



Source: www.earthdatascience.org/courses/use-data-open-source-python/intro-vector-data-python/spatial-data-vector-shapefiles/geographic-vs-projected-coordinate-reference-systems-python/

Coordinate Systems Definitions

Format: OGC Well Known Text (WKT)

```
GEOGCS["WGS 84",
    DATUM["WGS_1984",
        SPHEROID["WGS
84",6378137,298.257223563,
            AUTHORITY["EPSG","7030"]],
        AUTHORITY["EPSG","6326"]],
    PRIMEM["Greenwich",0,
        AUTHORITY["EPSG","8901"]],
    UNIT["degree",0.0174532925199433,
        AUTHORITY["EPSG","9122"]],
    AUTHORITY["EPSG","4326"]]
```

```
PROJCS["WGS 84 / Pseudo-Mercator",
    GEOGCS["WGS 84",
        DATUM["WGS_1984",
            SPHEROID["WGS 84",6378137,298.257223563,
                AUTHORITY["EPSG","7030"]],
            AUTHORITY["EPSG","6326"]],
        PRIMEM["Greenwich",0,
            AUTHORITY["EPSG","8901"]],
        UNIT["degree",0.0174532925199433,
            AUTHORITY["EPSG","9122"]],
        AUTHORITY["EPSG","4326"]],
    PROJECTION["Mercator_1SP"],
    PARAMETER["central_meridian",0],
    PARAMETER["scale_factor",1],
    PARAMETER["false_easting",0],
    PARAMETER["false_northing",0],
    UNIT["metre",1,
        AUTHORITY["EPSG","9001"]],
    AXIS["Easting",EAST],
    AXIS["Northing",NORTH],
    EXTENSION["PROJ4",
        "+proj=merc +a=6378137 +b=6378137
        +lat_ts=0 +lon_0=0 +x_0=0 +y_0=0 +k=1
        +units=m +nadgrids=@null +wktext +no_defs"],
    AUTHORITY["EPSG","3857"]]
```

How are spatial data managed in the Oracle Database?

Manage simple features in the Oracle Database

SDO_GEOMETRY data type

ID	Type
2001	Point
2002	Line
2003	Polygon

COLUMN_NAME	DATA_TYPE
WAREHOUSE_NAME	VARCHAR2(30 BYTE)
LAT	NUMBER
LON	NUMBER
GEOMETRY	SDO_GEOMETRY

```
SDO_GEOMETRY(
    [geometry type]              -- ID for points/lines/polygons
    , [coordinate system]        -- ID of coordinate system
    , [point coordinate]         -- used for points only
    , [line/polygon info]        -- used for lines/polygons only
    , [line/polygon coordinates] -- used for lines/polygons only
)
```

SRID

ID	Coordinate System
4326	Latitude/Longitude
3857	World Mercator

```
SDO_GEOMETRY(
    2001                         -- 2D point
    , 4326                       -- Coordinate system
    , SDO_POINT_TYPE(
      -100.123, 20.456, NULL)    -- lon/lat values
    , NULL                       -- Not used for points
    , NULL                       -- Not used for points
)
```

Inside the Oracle Database
SDO_GEOMETRY metadata

```
INSERT INTO USER_SDO_GEOM_METADATA VALUES (
  <table name>,
  <geometry column name>,
  SDO_DIM_ARRAY(
    SDO_DIM_ELEMENT('X',<min x>,<max x>,<tolerance>),
    SDO_DIM_ELEMENT('Y',<min y>,<max y>,<tolerance>)),
  <coordinate system id>
  );
```

```
INSERT INTO USER_SDO_GEOM_METADATA VALUES (
  <table name>,
  <geometry column name>,
  SDO_DIM_ARRAY(
    SDO_DIM_ELEMENT('X', -180, 180, 0.005),
    SDO_DIM_ELEMENT('Y',-90, 90, 0.005)),
  4326
  );
```

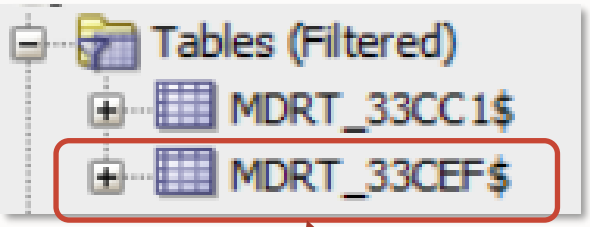
```
begin
  apex_spatial.insert_geom_metadata (
    p_table_name  => '<table_name>',
    p_column_name => '<geom_column_name>',
    p_diminfo     => sdo_dim_array(
      sdo_dim_element('X',-180,180,1),
      sdo_dim_element('Y',-90,90,1) ),
    p_srid        => apex_spatial.c_wgs_84 );
end;
/
```

APEX

Inside the Oracle Database

R-Tree index on SDO_GEOMETRY columns

```
CREATE INDEX WAREHOUSES_SIDX ON
       WAREHOUSES (
            GEOMETRY
       )
            INDEXTYPE IS MDSYS.SPATIAL_INDEX_V2;
```



Additional parameters to optimize index creation, maintenance and usage, e.g.: `PARAMETERS ('LAYER_GTYPE=POINT')`

```
select * from user_sdo_index_metadata  where sdo_table_name = 'WAREHOUSES';
```

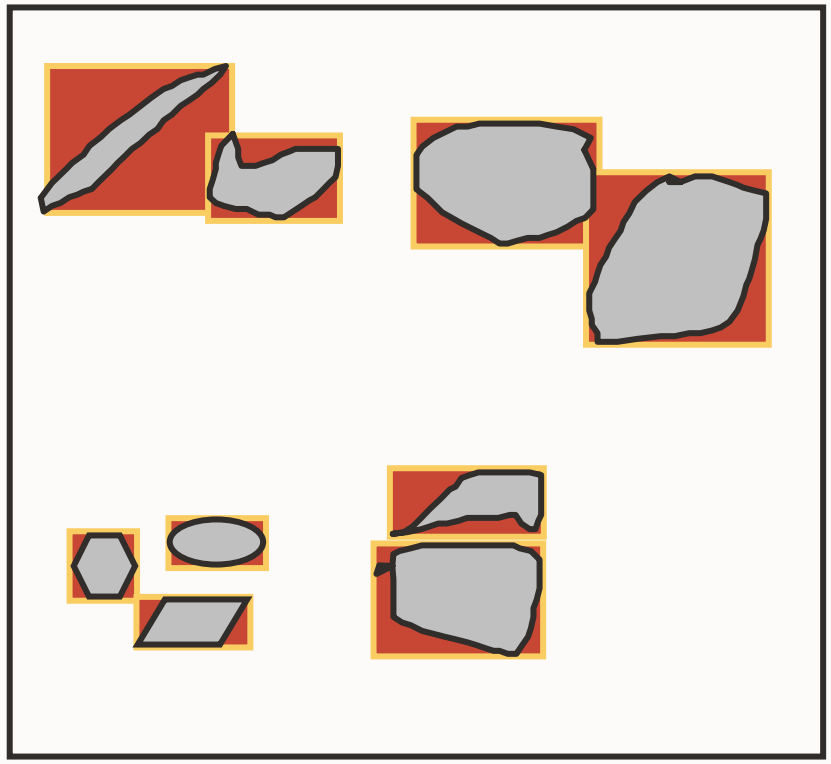
t Output x | Query Result x

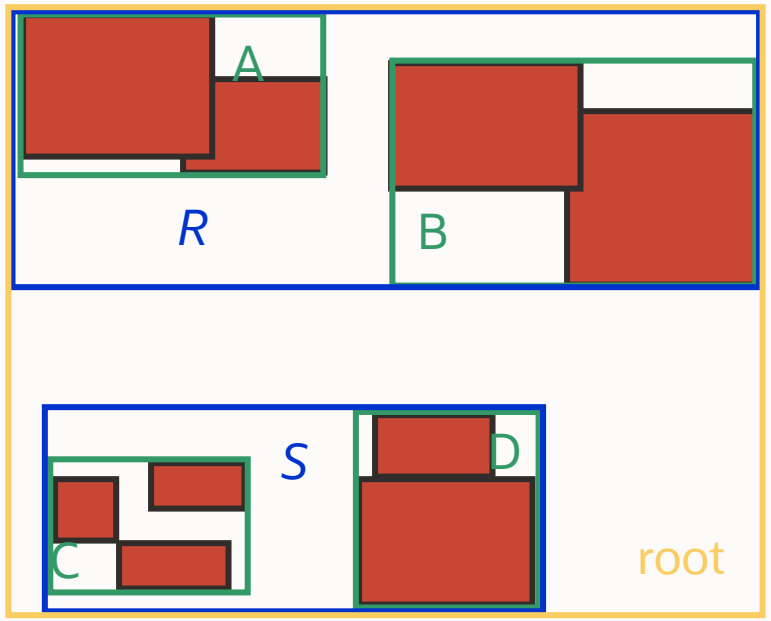
SQL | All Rows Fetched: 1 in 0,418 seconds

SDO_INDEX_OWNER	SDO_INDEX_TYPE	SDO_INDEX_NAME	SDO_TABLE_NAME	SDO_COLUMN_NAME	SDO_INDEX_TABLE
SPATIALUSER	RTREE	WAREHOUSES_SIDX	WAREHOUSES	GEOMETRY	MDRT_33CEF$

Inside the Oracle Database

R-Tree index on SDO_GEOMETRY columns





How can you access and use spatial data?

Flip the (power) switch

Recommendation: Turn on **Vector Performance Accelerator**

```
show parameter spatial_vector_acceleration

alter session set spatial_vector_acceleration=true
-- or
alter system set spatial_vector_acceleration=true [scope=both]
```

Documentation:
docs.oracle.com/en/database/oracle/oracle-database/19/refrn/SPATIAL_VECTOR_ACCELERATION.html

Oracle Spatial SQL API

Spatial operators and functions (from basic to advanced/from general purpose to specialized)

SDO_ANYINTERACT
SDO_CONTAINS
SDO_COVEREDBY
SDO_COVERS
SDO_EQUAL
SDO_FILTER
SDO_INSIDE
SDO_JOIN
SDO_NN
SDO_NN_DISTANCE
SDO_ON
SDO_OVERLAPBDYDISJOINT
SDO_OVERLAPBDYINTERSECT
SDO_OVERLAPS
SDO_POINTINPOLYGON
SDO_RELATE
SDO_TOUCH
SDO_WITHIN_DISTANCE

SDO_AGGR_CENTROID
SDO_AGGR_CONCAT_LINES
SDO_AGGR_CONCAVEHULL
SDO_AGGR_CONVEXHULL
SDO_AGGR_LRS_CONCAT
SDO_AGGR_MBR
SDO_AGGR_SET_UNION
SDO_AGGR_UNION

SDO_GEOM.RELATE
SDO_GEOM.SDO_ALPHA_SHAPE
SDO_GEOM.SDO_ARC_DENSIFY
SDO_GEOM.SDO_AREA
SDO_GEOM.SDO_BUFFER
SDO_GEOM.SDO_CENTROID
SDO_GEOM.SDO_CLOSEST_POINTS
SDO_GEOM.SDO_CONCAVEHULL
SDO_GEOM.SDO_CONCAVEHULL_BOUNDARY
SDO_GEOM.SDO_CONVEXHULL
SDO_GEOM.SDO_DIAMETER
SDO_GEOM.SDO_DIAMETER_LINE
SDO_GEOM.SDO_DIFFERENCE
SDO_GEOM.SDO_DISTANCE
SDO_GEOM.SDO_INTERSECTION
SDO_GEOM.SDO_LENGTH
SDO_GEOM.SDO_MAX_MBR_ORDINATE
SDO_GEOM.SDO_MAXDISTANCE
SDO_GEOM.SDO_MAXDISTANCE_LINE

SDO_GEOM.SDO_MBC
SDO_GEOM.SDO_MBC_CENTER
SDO_GEOM.SDO_MBC_RADIUS
SDO_GEOM.SDO_MBR
SDO_GEOM.SDO_MIN_MBR_ORDINATE
SDO_GEOM.SDO_POINTONSURFACE
SDO_GEOM.SDO_SELF_UNION
SDO_GEOM.SDO_TRIANGULATE
SDO_GEOM.SDO_UNION
SDO_GEOM.SDO_VOLUME
SDO_GEOM.SDO_WIDTH
SDO_GEOM.SDO_WIDTH_LINE
SDO_GEOM.SDO_XOR
SDO_GEOM.VALIDATE_GEOMETRY_WITH_CONTEXT
SDO_GEOM.VALIDATE_LAYER_WITH_CONTEXT
SDO_GEOM.WITHIN_DISTANCE

Inside the Oracle Database

Query spatial data using spatial operators and functions

```
SELECT
     STORE_NAME,
     STORE_TYPE
 FROM
     STORES     A,
     WAREHOUSES B
 WHERE
      B.WAREHOUSE_NAME = 'Dallas Warehouse'
 AND SDO_WITHIN_DISTANCE(
       GET_GEOMETRY(A.LONGITUDE, A.LATITUDE),
       B.GEOMETRY,
       'distance=20 unit=mile') = 'TRUE'
```

Proximity

```
SELECT
      A.STORE_NAME,
      A.STORE_TYPE
  FROM
      STORES  A,
      REGIONS B
  WHERE REGION = 'REGION-02'
  AND SDO_INSIDE(
        GET_GEOMETRY(A.LONGITUDE, A.LATITUDE),
        B.GEOMETRY) = 'TRUE';
```

Containment/Topological relationship

Inside the Oracle Database

Validating and Rectifying Spatial Data in the Oracle Database

April 15, 2020 | 3 minute read

Hans Viehmann
Product Manager EMEA

It always amazes me, how many of the complaints about unexpected results or errors in spatial queries we hear about can be traced back to invalid geometries. In our documentation, should you ever bother to read it, we are actually very clear about this:

"You should validate all geometry data, and fix any validation errors, before performing any spatial operations on the data. The recommended procedure for loading and validating spatial data is as follows:

1. Load the data, using a method described in Bulk Loading or Transactional Insert Operations Using SQL.
2. Use the SDO_GEOM.VALIDATE_GEOMETRY_WITH_CONTEXT function or the SDO_GEOM.VALIDATE_LAYER_WITH_CONTEXT procedure on all spatial data loaded into the database.
3. For any geometries with the wrong orientation or an invalid ETYPE or GTYPE value, use SDO_MIGRATE.TO_CURRENT on these invalid geometries to fix them.
4. For any geometries that are invalid for other reasons, use SDO_UTIL.RECTIFY_GEOMETRY to fix these geometries."

The need to eliminate invalid geometries is probably immediately obvious. If you have, say, a self-crossing polygon, how would you know what is inside and what is outside? What makes this trickier is that across the industry, tools and solutions have varying levels of strictness when it comes to tolerating errors in spatial data. This means that an incorrect geometry may have been loaded into a GIS tool without problems, but once it is saved to the database, it could raise errors when it is used in a subsequent query. Moreover, the Oracle Database has become less forgiving from one release to the next in this regard, so that an invalid geometry may have gone unnoticed in 11gR2 while in 19c it would cause issues.

blogs.oracle.com/oraclespatial/post/validating-and-rectifying-spatial-data-in-the-oracle-database

Oracle Spatial SQL API

SDO_UTIL.AFFINETRANSFORMS

SDO_UTIL.APPEND

SDO_UTIL.BEARING_TILT_FOR_POINTS

SDO_UTIL.CIRCLE_POLYGON

SDO_UTIL.CONCAT_LINES

SDO_UTIL.CONVERT_UNIT

SDO_UTIL.CONVERT3007TO3008

SDO_UTIL.DELETE_SDO_GEOM_METADATA

SDO_UTIL.DENSIFY_GEOMETRY

SDO_UTIL.DROP_WORK_TABLES

SDO_UTIL.ELLIPSE_POLYGON

SDO_UTIL.EXPAND_GEOM

SDO_UTIL.EXTRACT

SDO_UTIL.EXTRACT_ALL

SDO_UTIL.EXTRACT3D

SDO_UTIL.EXTRUDE

SDO_UTIL.FROM_GEOJSON

SDO_UTIL.FROM_GML311GEOMETRY

SDO_UTIL.FROM_GMLGEOMETRY

SDO_UTIL.FROM_JSON

SDO_UTIL.FROM_KMLGEOMETRY

SDO_UTIL.FROM_WKBGEOMETRY

SDO_UTIL.FROM_WKTGEOMETRY

SDO_UTIL.GEO_SEARCH

SDO_UTIL.GET_2D_FOOTPRINT

SDO_UTIL.GET_COORDINATE

SDO_UTIL.GETFIRSTVERTEX

SDO_UTIL.GETLASTVERTEX

SDO_UTIL.GETNUMELEM

SDO_UTIL.GETNUMVERTICES

SDO_UTIL.GETNURBSAPPROX

SDO_UTIL.GETVERTICES

SDO_UTIL.INITIALIZE_INDEXES_FOR_TTS

SDO_UTIL.INSERT_SDO_GEOM_METADATA

SDO_UTIL.INTERIOR_POINT

SDO_UTIL.POINT_AT_BEARING

SDO_UTIL.POLYGONTOLINE

SDO_UTIL.RECTIFY_GEOMETRY

SDO_UTIL.REMOVE_DUPLICATE_VERTICES

SDO_UTIL.REVERSE_LINESTRING

SDO_UTIL.SIMPLIFY

SDO_UTIL.SIMPLIFYVW

SDO_UTIL.THEME3D_GET_BLOCK_TABLE

SDO_UTIL.THEME3D_HAS_LOD

SDO_UTIL.THEME3D_HAS_TEXTURE

SDO_UTIL.TILE_GEOMETRY

SDO_UTIL.TO_GEOJSON

SDO_UTIL.TO_GML311GEOMETRY

SDO_UTIL.TO_GMLGEOMETRY

SDO_UTIL.TO_JSON

SDO_UTIL.TO_JSON_VARCHAR

SDO_UTIL.TO_KMLGEOMETRY

SDO_UTIL.TO_WKBGEOMETRY

SDO_UTIL.TO_WKTGEOMETRY

SDO_UTIL.VALIDATE_3DTHEME

SDO_UTIL.VALIDATE_SCENE

SDO_UTIL.VALIDATE_VIEWFRAME

SDO_UTIL.VALIDATE_WKBGEOMETRY

SDO_UTIL.VALIDATE_WKTGEOMETRY

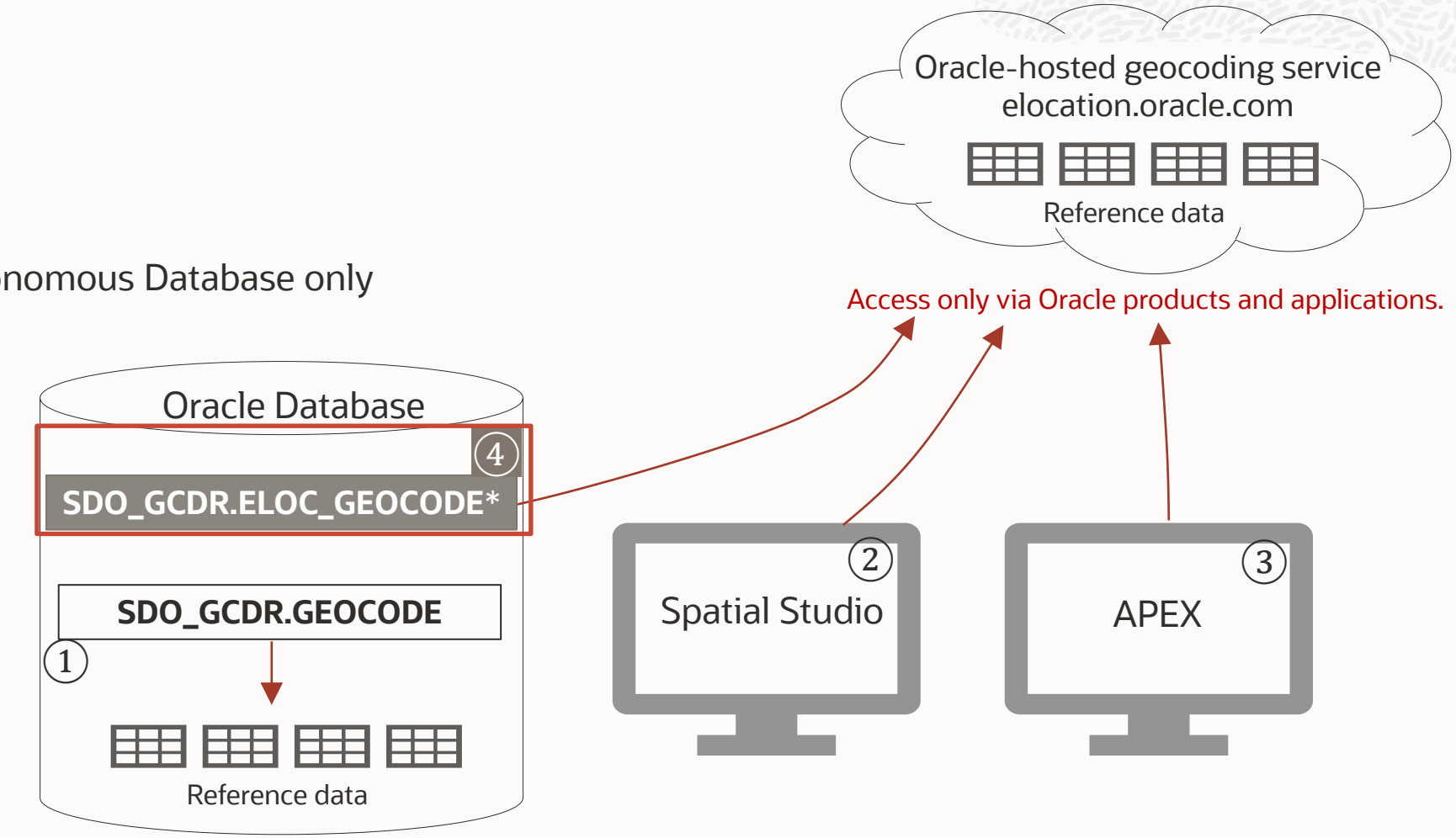
And many more

Turn textual data into spatial data

Geocoding

Geocoding options - Overview

* Autonomous Database only



Geocoding using SQL API for Autonomous Database

Basics

Format

```
SDO_GCDR.ELOC_GEOCODE(
     street      IN VARCHAR2,
     city        IN VARCHAR2,
     region      IN VARCHAR2,
     postal_code IN VARCHAR2,
     cc2         IN VARCHAR2,
     match_mode  IN VARCHAR2 default 'DEFAULT');
```

Copy

or

```
SDO_GCDR.ELOC_GEOCODE(address IN VARCHAR2);
```

Copy

or

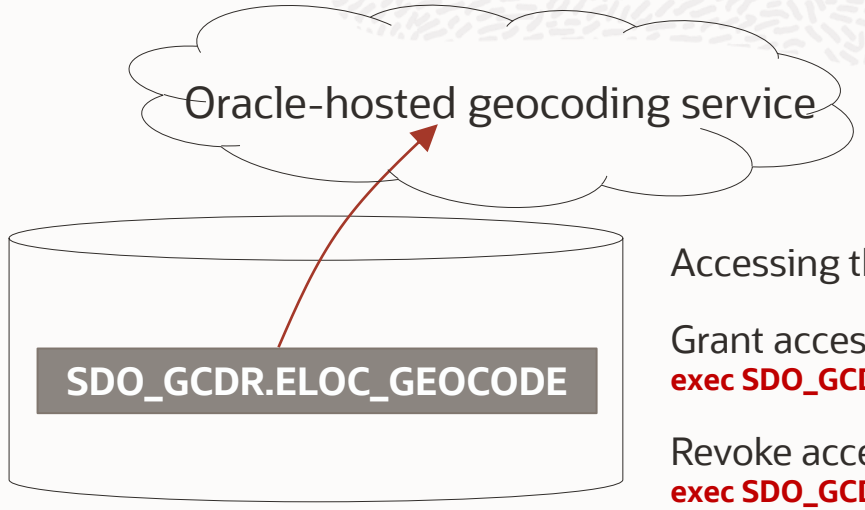
```
SDO_GCDR.ELOC_GEOCODE(
     longitude IN NUMBER,
     latitude  IN NUMBER);
```

Copy

Description

Geocodes a formatted (address parts in separate fields) or an unformatted (complete address in a single string field) address and returns the standardized address with geographic coordinates and geocoding metadata in JSON format.

For longitude and latitude input, the function reverse geocodes the location and returns the address in JSON format.



Accessing the service requires:

Grant access to a Database user via
exec SDO_GCDR.ELOC_GRANT_ACCESS('SCOTT');

Revoke access via
exec SDO_GCDR.ELOC_REVOKE_ACCESS('SCOTT');

- No local reference data is needed
- Dependency on public internet access from the database
- Currently only on ADB-Shared
- Blog: blogs.oracle.com/database/post/new-in-database-geocoder-for-autonomous-database-shared
- Doc: docs.oracle.com/en/database/oracle/oracle-database/19/spatl/SDO_GCDR-reference.html

Geocoding using SQL API for Autonomous Database

Usage

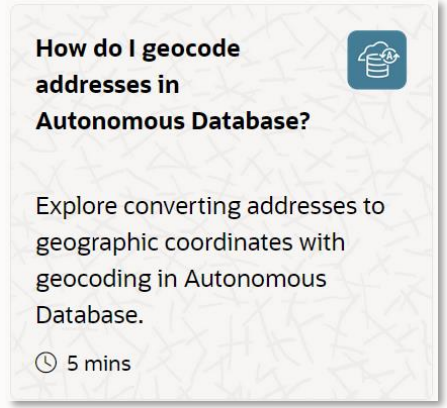
```
-- Sample address data
street,city,region,postal_code,country
De Nieuwe Poort 20,Amersfoort,PA,3812,NL
Bötzower Str. 24,Berlin,BE,10407,DE
Bd Carl-Vogt 67,Genève,,1205,CH
100 N Renfrew St,Vancouver,BC,V5K 4W3,CA
2 Chome-5-8 Kitaaoyama,Tokyo,,107-0061,JP
123 Beacon St,Boston,MA,02116,US


-- Structured input returning JSON or SDO_GEOMETRY
select sdo_gcdr.eloc_geocode(street, city, region,
postal_code, country, 'RELAX_POSTAL_CODE') from
sample_addresses;
select sdo_gcdr.eloc_geocode_as_geom(street, city,
region, postal_code, country, 'RELAX_POSTAL_CODE') from
sample_addresses;


-- Unstructured input returning JSON or SDO_GEOMETRY
select sdo_gcdr.eloc_geocode(address_line) from
sample_addresses;
select sdo_gcdr.eloc_geocode_as _geom(address_line)
from sample_addresses;
```

- Returned values
  - **ELOC_GEOCODE** ☞ JSON document containing one or more best matches
    - Additional address information, such as a municipality, side of the street, or to fill missing address data
    - Match vector to tell you about the accuracy of the geocoded address
  - **ELOC_GEOCODE_AS_GEOM** ☞ SDO_GEOMETRY object for the best match

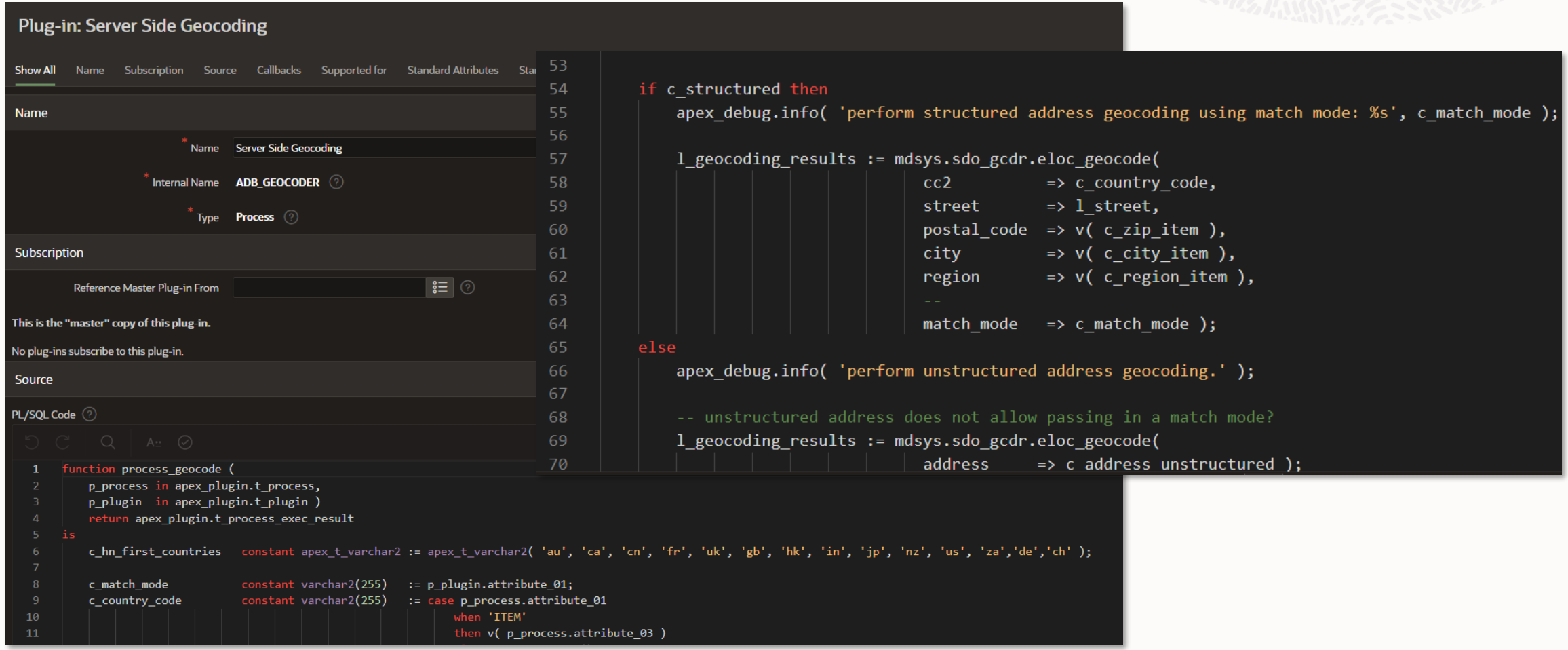
- Oracle LiveLabs Sprint



How to use the ADB Geocoder SQL API in APEX?

APEX using the ADB Geocoder SQL API

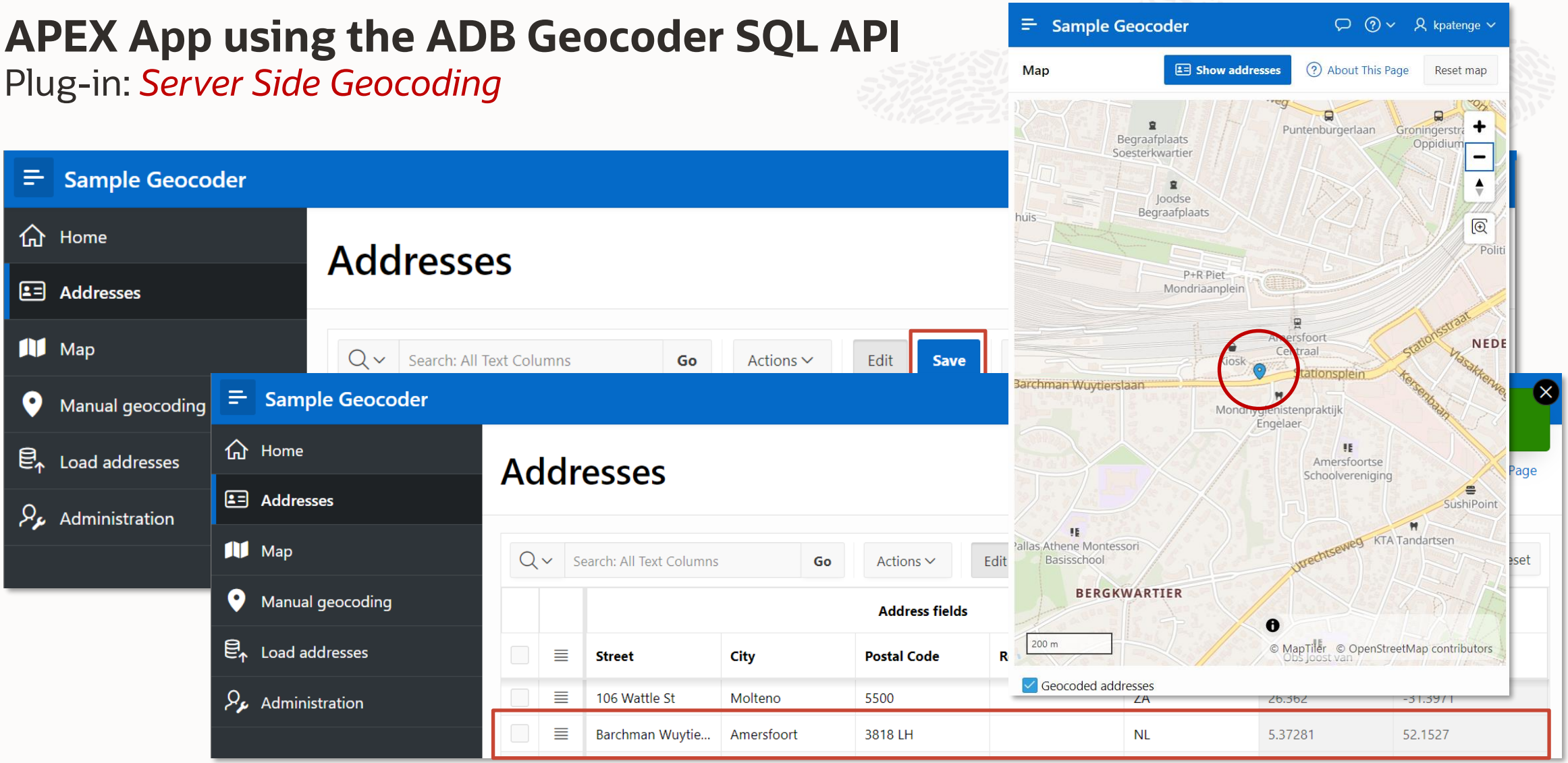
Solution: APEX Plug-in that uses the SQL API



Plug-in by Carsten Czarski (APEX Dev Team)

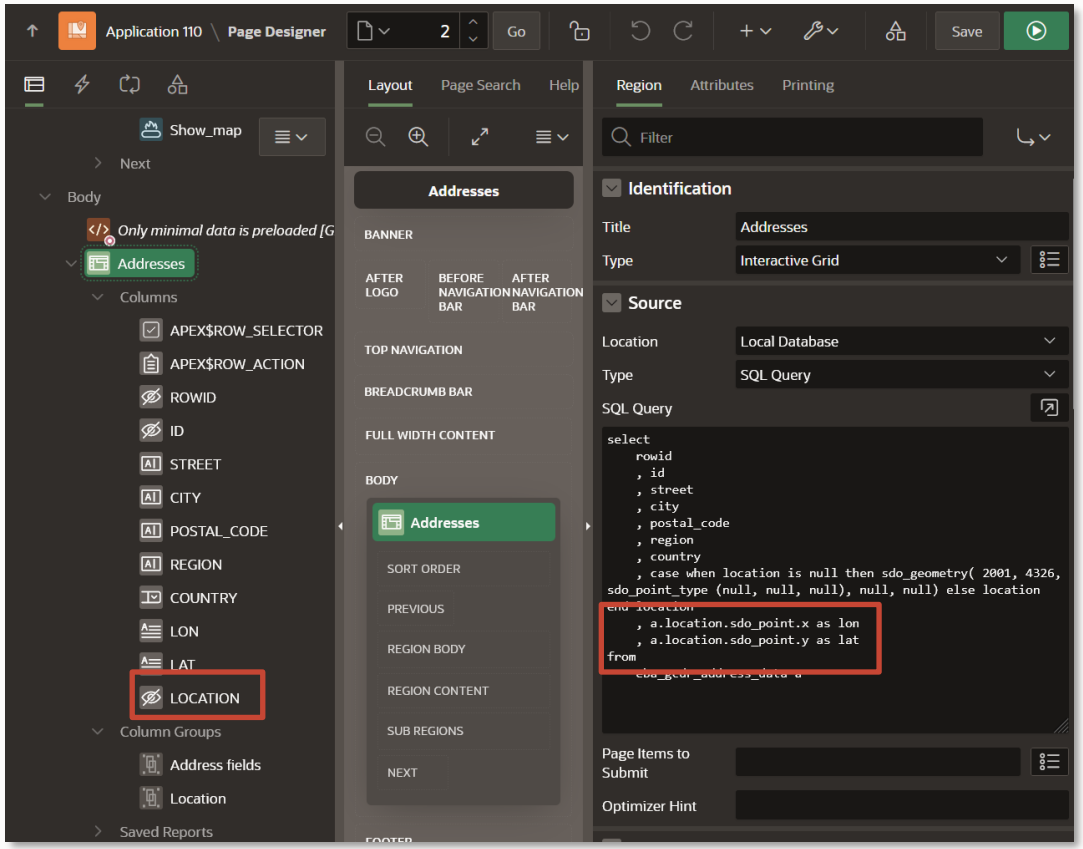
APEX App using the ADB Geocoder SQL API

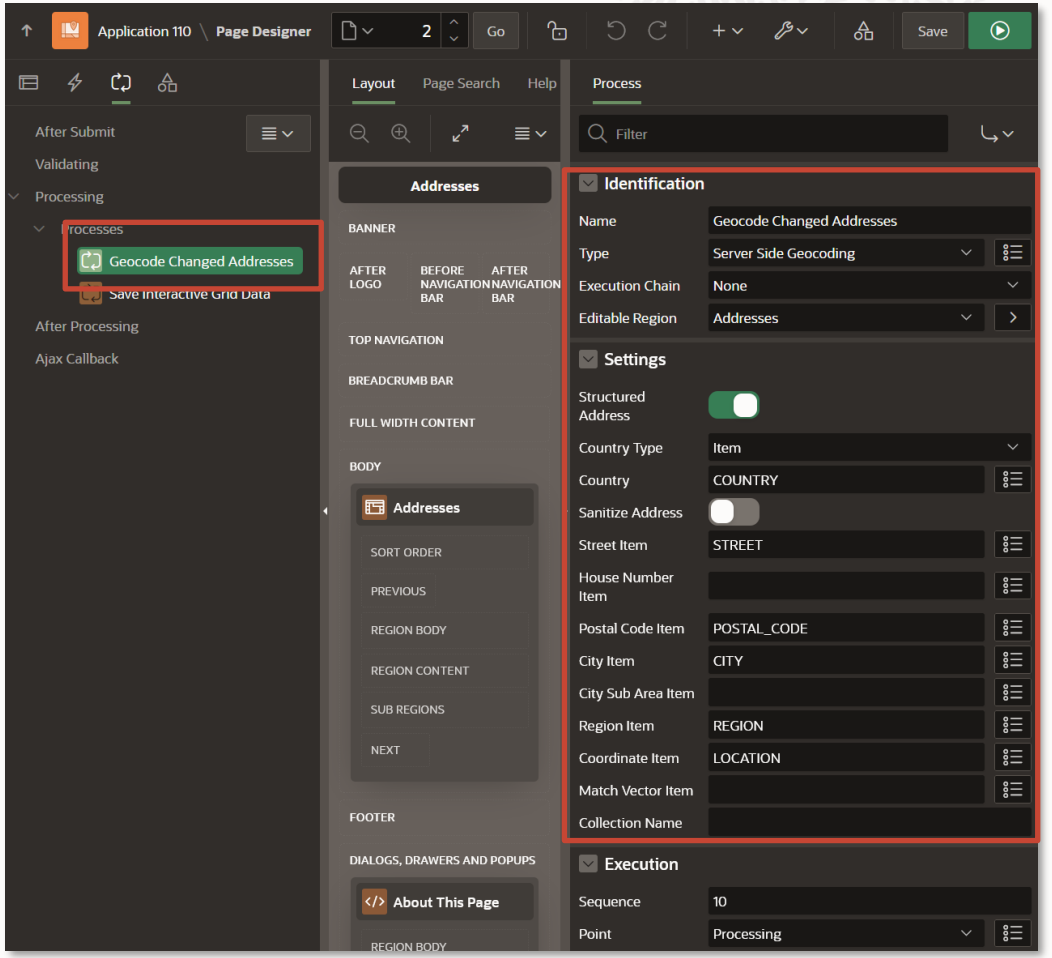
Plug-in: *Server Side Geocoding*



APEX App using the ADB Geocoder SQL API

Plug-in: *Server Side Geocoding*



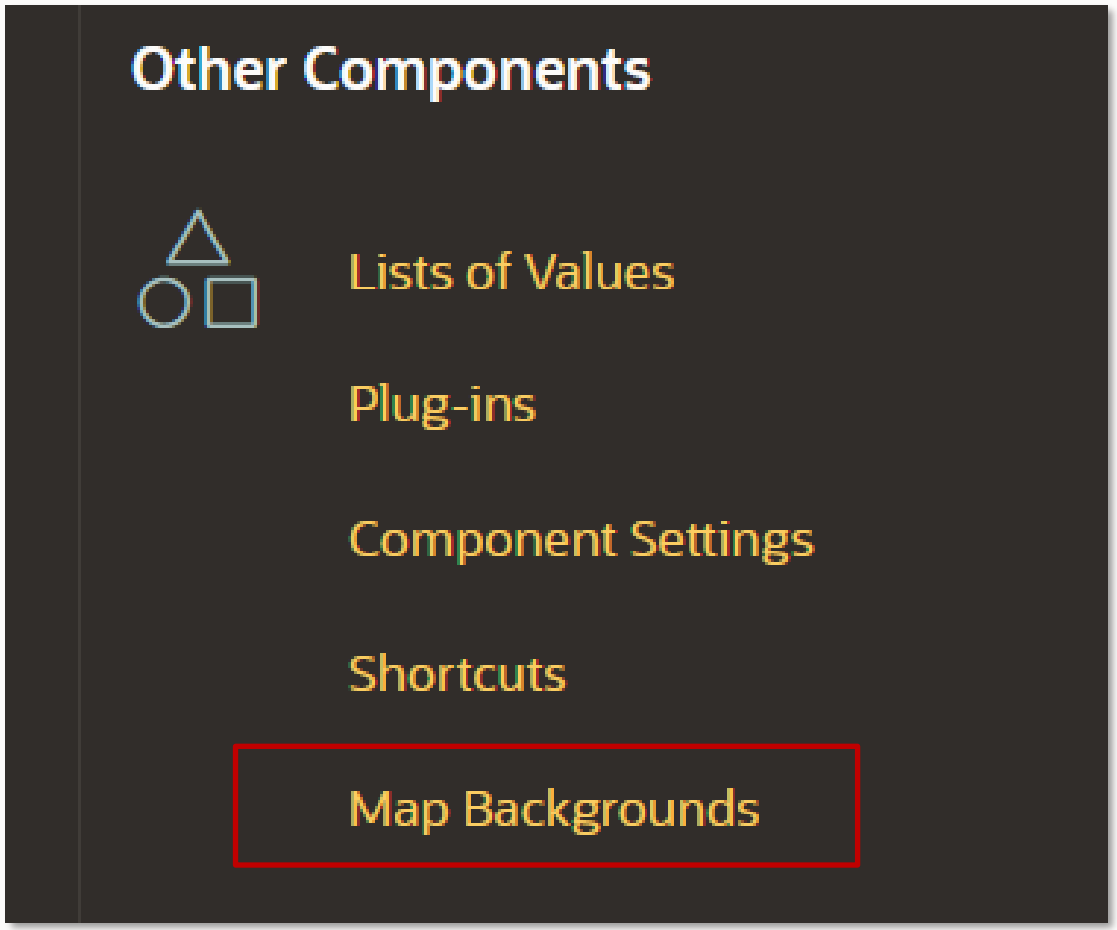


Custom Map Backgrounds in APEX?

New feature in APEX 23.2

Define Map Backgrounds

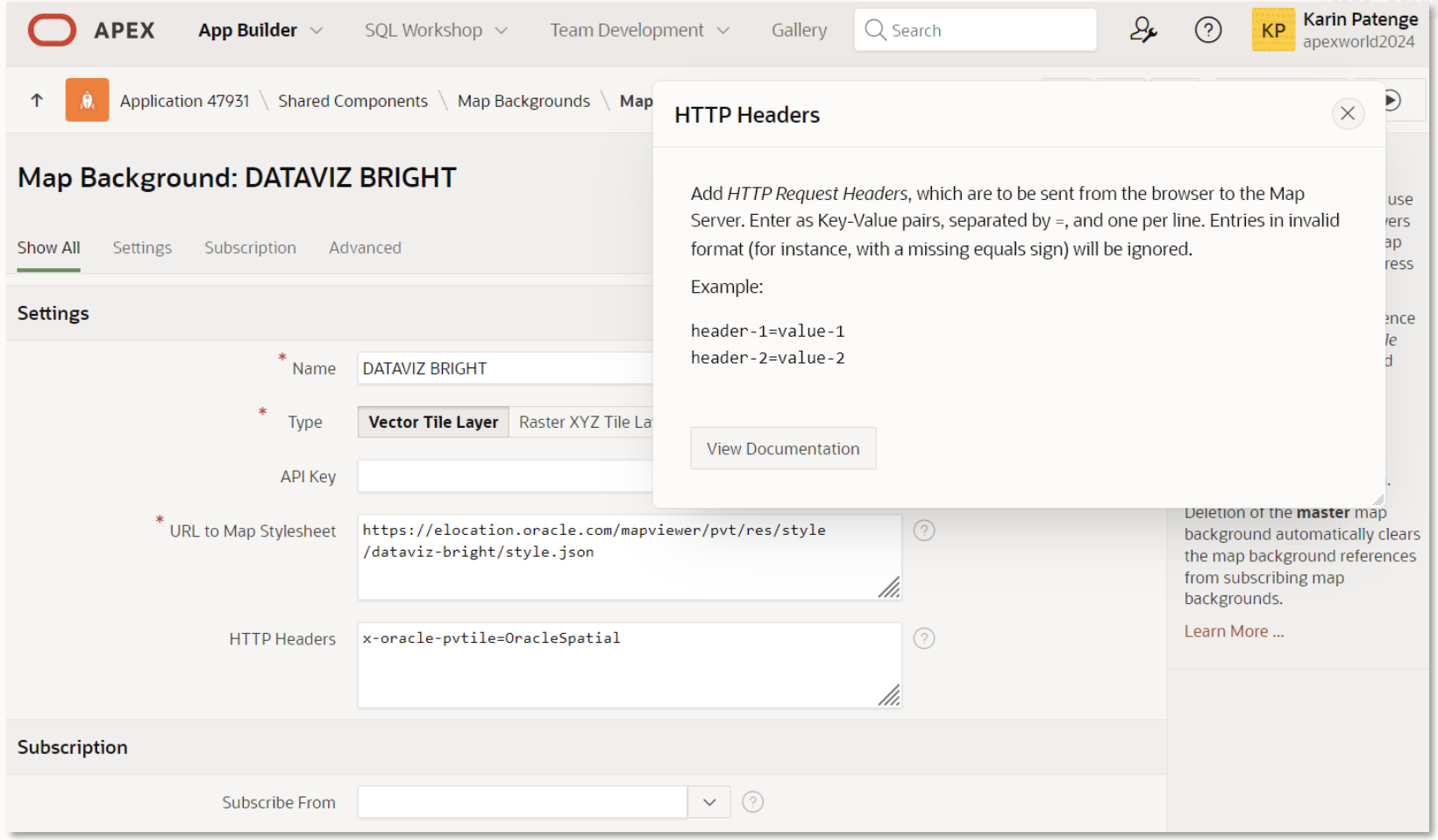
Shared Components



- You have three options:
  - Vector Tile Layer
  - Raster XYZ Tile Layer
  - OGC WMS
- Map backgrounds can be used in
  - Map regions
  - Display map
  - Geocoded address items
- Note:
  - OGC WMS as Map Background
    - APEX automatically adds a few URL parameters. You can omit those in the definition

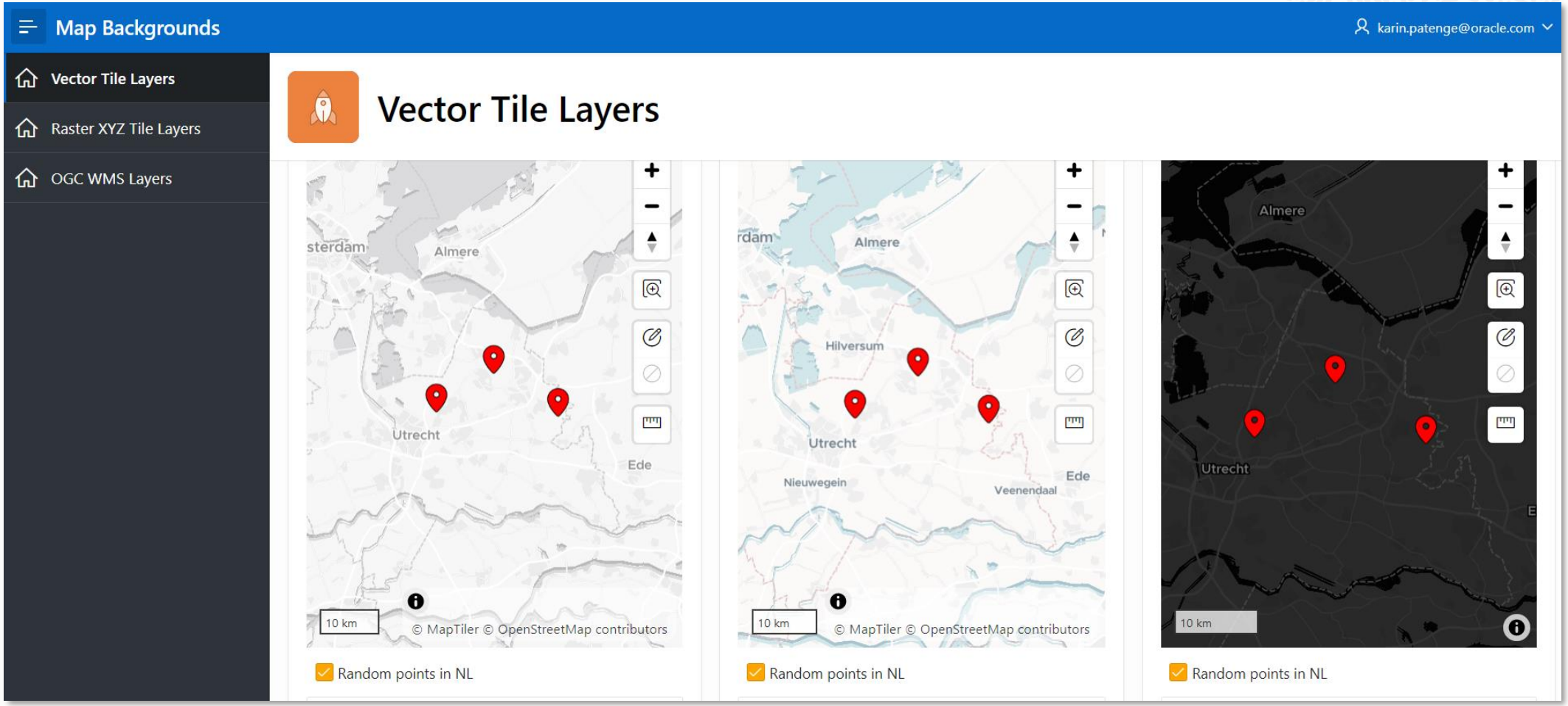
Custom Map Backgrounds

Example: Vector Tile Layer Definition



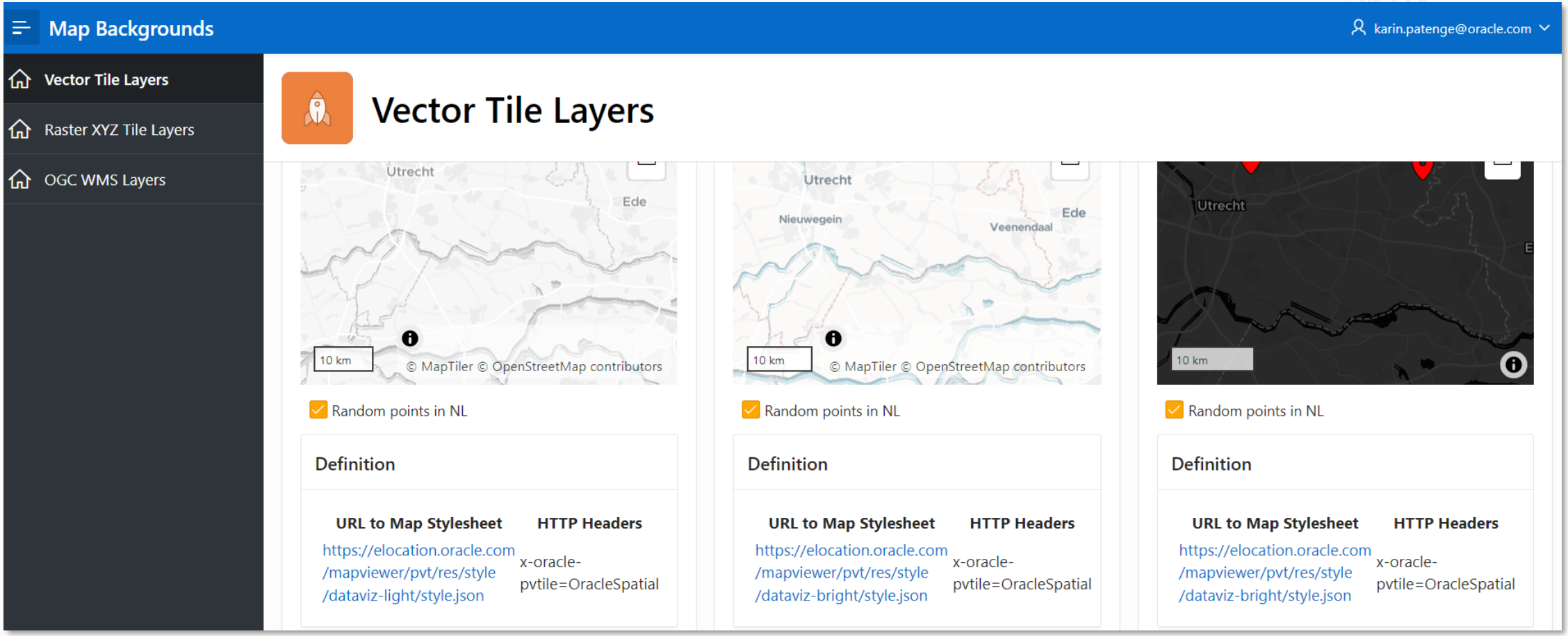
Custom Map Backgrounds

Example: Vector Tile Layer in Use (1)



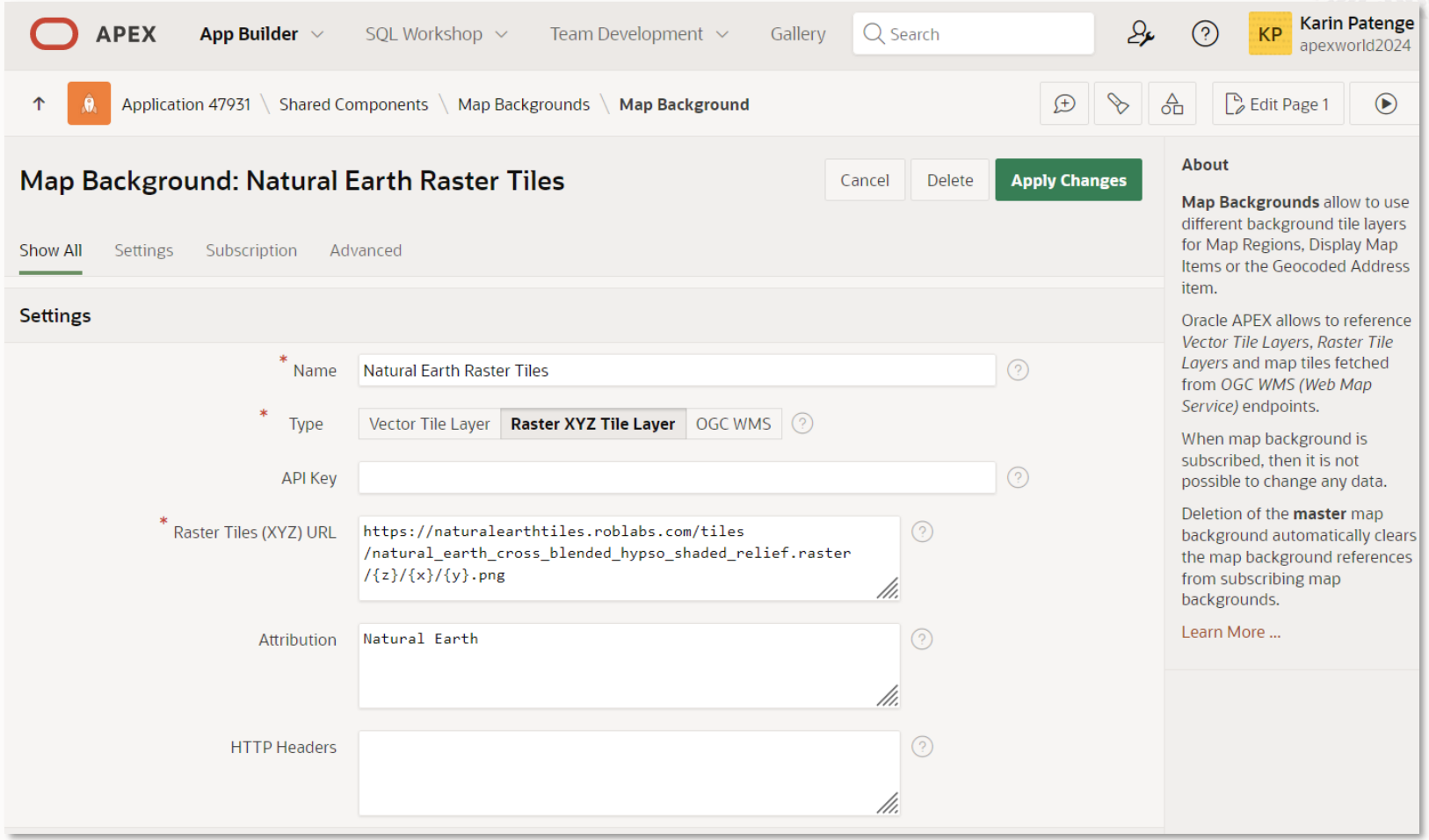
Custom Map Backgrounds

Example: Vector Tile Layer in Use (2)



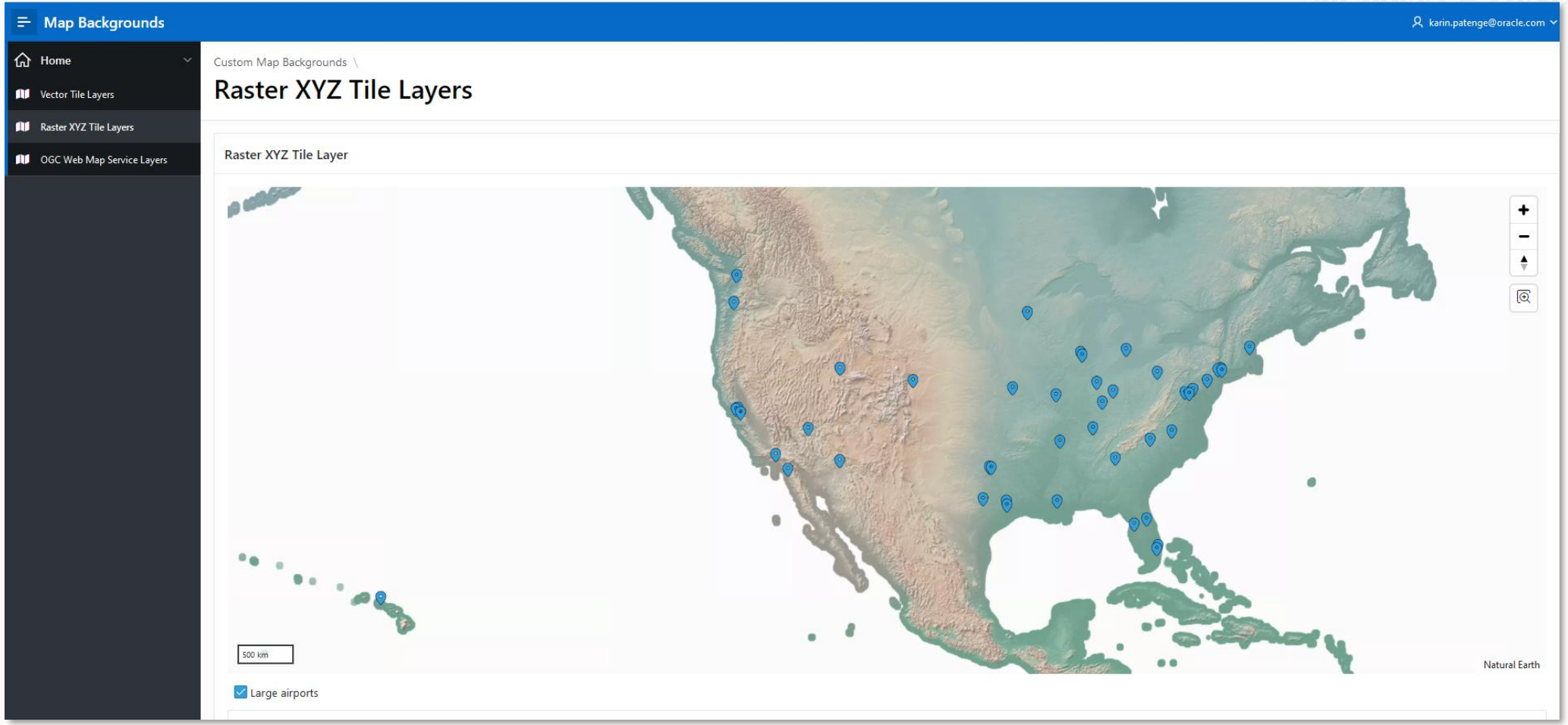
Custom Map Backgrounds

Example: Raster XYZ Layer Definition



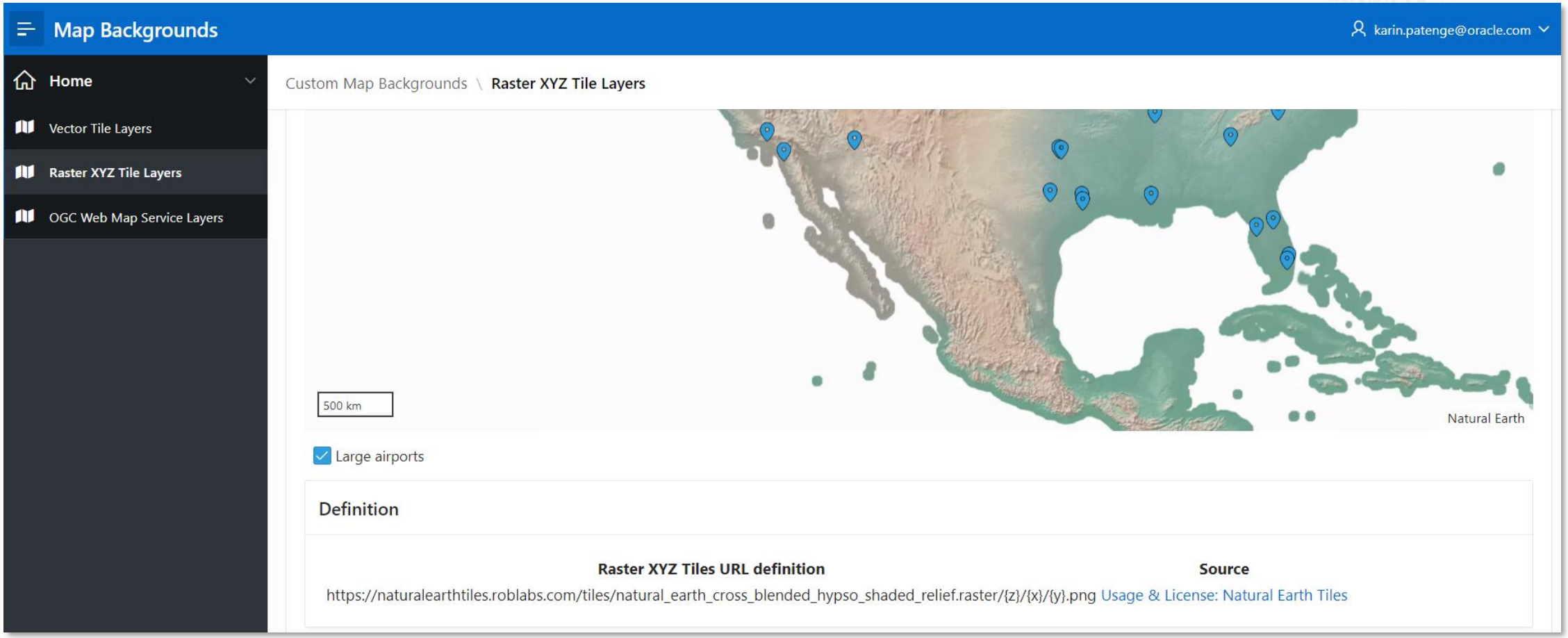
Custom Map Backgrounds

Example: Raster XYZ Layer in Use (1)



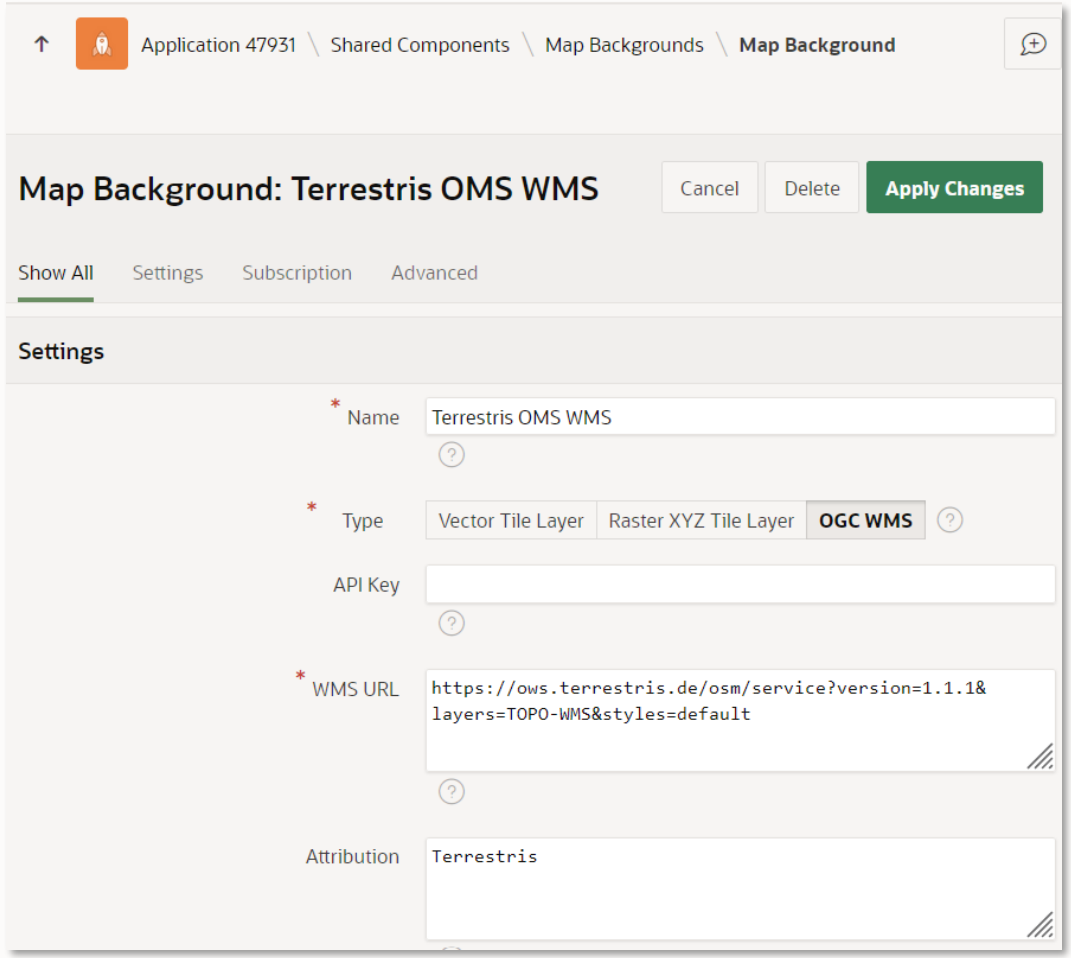
Custom Map Backgrounds

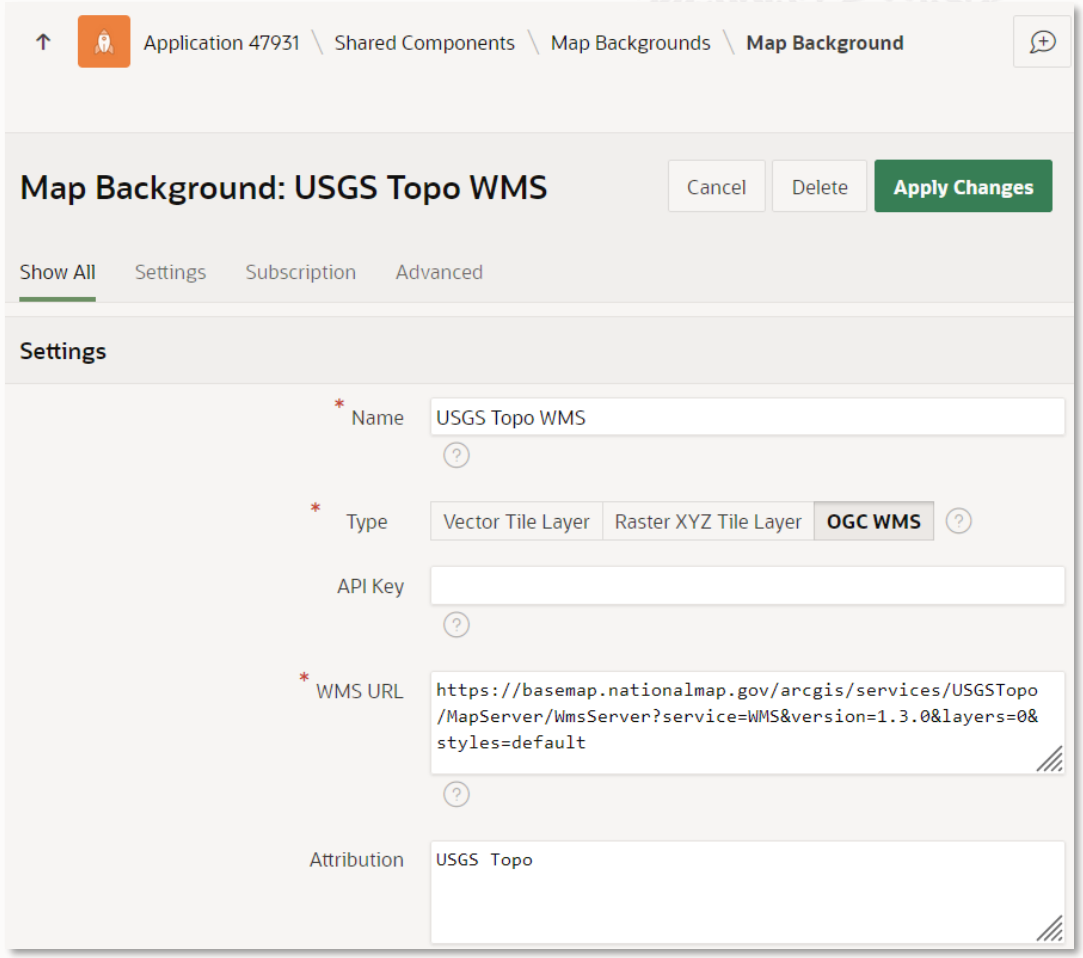
Example: Raster XYZ Layer in Use (2)



Custom Map Backgrounds

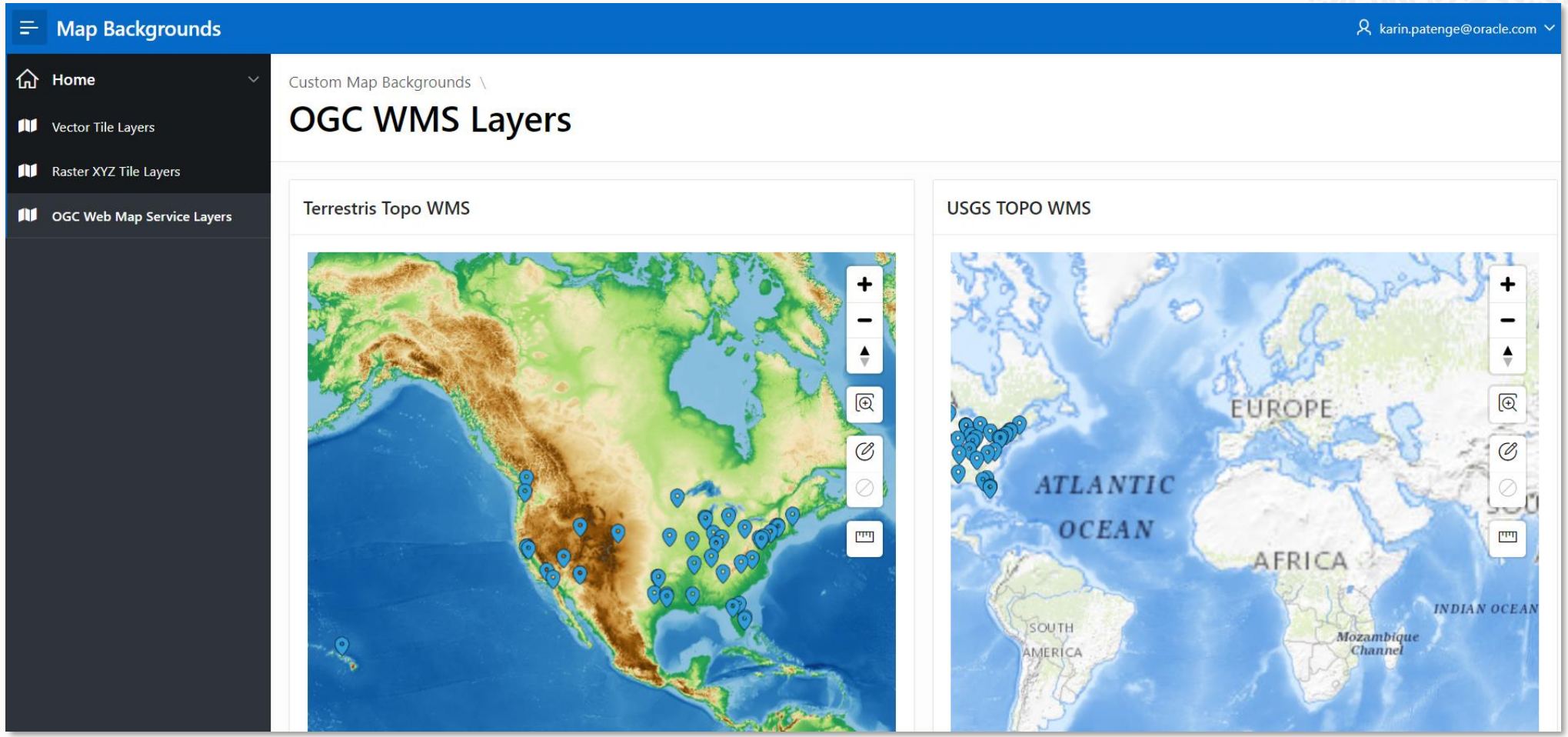
Example: OGC Web Map Service (WMS) Definition





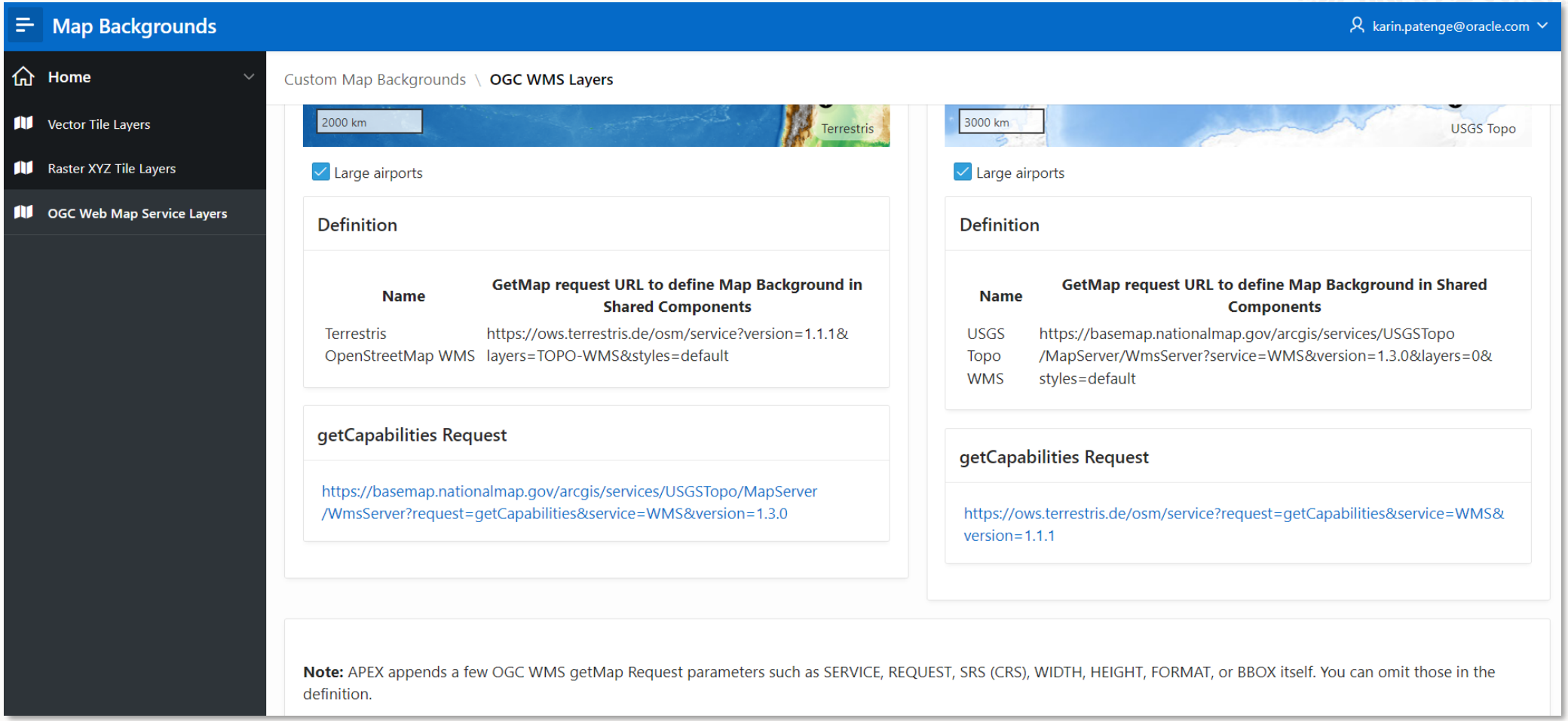
Custom Map Backgrounds

Example: OGC Web Map Service (WMS) in Use (1)



Custom Map Backgrounds

Example: OGC Web Map Service (WMS) in Use (2)



Demo

1. Using the ADB Geocoder in APEX
2. Useful code samples
3. Map Backgrounds in APEX

Autonomous Database

Self-paced LiveLabs Workshop: Get Started with Maps and Spatial in APEX

apexapps.oracle.com/pls/apex/r/dbpm/livelabs/view-workshop?wid=936

More Information

Make sure to always apply
the latest Spatial Patch Bundle

MOS Note ID 2514624.1

Where can you learn more about Spatial?

General information

- www.oracle.com/database/spatial

Spatial on Oracle LiveLabs

- https://apexapps.oracle.com/pls/apex/f?p=133:100:::::SEARCH:spatial

Documentation

- docs.oracle.com/en/database/oracle/apex/23.2/aeapi/APEX_SPATIAL.html
- docs.oracle.com/en/database/oracle/oracle-database/19/spatial-and-graph.html (version 19c)

Reference Architecture Center

- Build a geospatial platform on Oracle Autonomous Database
  docs.oracle.com/solutions/?q=geospatial

AskTOM Spatial Office Hours

- Spatial for DBAs, Analysts and Developers (Every last Tuesday of a month)

Blogs

- blogs.oracle.com/database/category/db-spatial
- medium.com/tag/oracle-spatial
- blogs.oracle.com/oraclespatial (legacy)
- oracle-spatial.blogspot.com (legacy)

YouTube

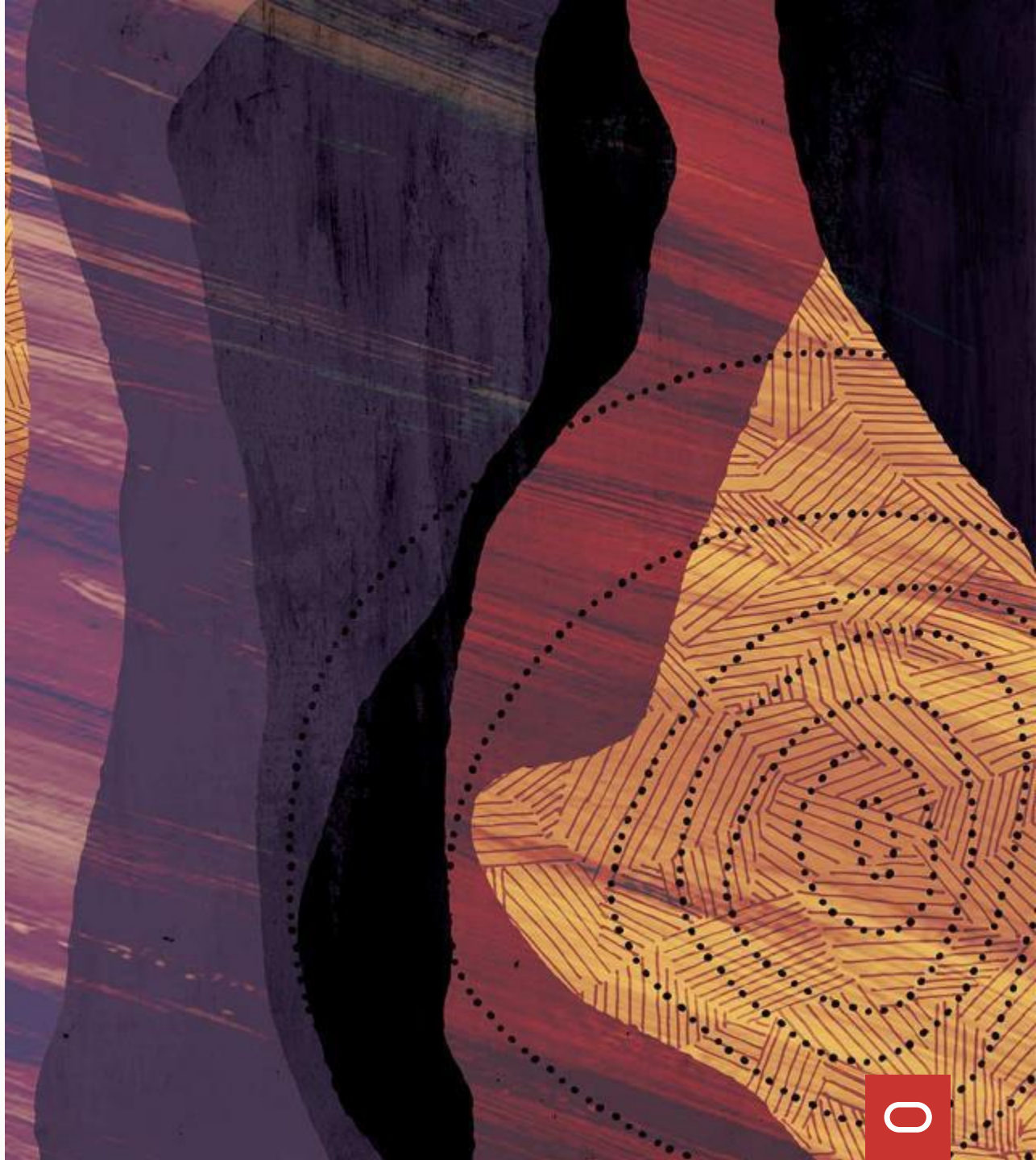
- Oracle Spatial and Oracle Graph channel
  bit.ly/Spatial-Graph-YouTube
- Spatial AskTOM OH Playlist
  www.youtube.com/playlist?list=PL3ZqpALcm8HP5glGHJfYLvOzQmjn9QEkn

LinkedIn

- Oracle Spatial and Graph group
  www.linkedin.com/groups/1848520/

Thank you

karin.patenge@oracle.com




ORACLE