DZ_GML

- Release:
- Commit Date: Mon Oct 10 16:40:29 2016 -0400

Utilities for the exchange of geometries between Oracle Spatial and OGC GML 3.x formats.

Summary

DZ_GML

Functions

dz_gml_main.sdo2geogml	Wrapper around MDSYS.SDO_UTIL.TO_GMLGEOMETRY and MDSYS.SDO_UTIL.TO_GML311GEOMETRY for conversion of Oracle Spatial SDO_GEOMETRY into GML geospatial tags allowing GML 3.2 output and more OGC compliant srs information.
dz_gml_main.fetch_gml_namespace	Direct exposure of the GML version number to namespace utility to test if your gml version is supported as you expect by the DZ_GML package.
dz_gml_main.geogml2sdo	Function for conversion of GML geospatial tags into Oracle Spatial SDO_GEOMETRY geometry objects.

Functions

dz_gml_main.sdo2geogml

Wrapper around MDSYS.SDO_UTIL.TO_GMLGEOMETRY and MDSYS.SDO_UTIL.TO_GML311GEOMETRY for conversion of Oracle Spatial SDO_GEOMETRY into GML geospatial tags allowing GML 3.2 output and more OGC compliant srs information.

Parameters

p_input	input SDO_GEOMETRY. Only geometry types supported by SDO_UTIL GML packages are supported.
p_pretty_print	nonfunctional at this time.
p_2d_flag	set to TRUE to remove all 3D and LRS information from input geometry before conversion.
p_output_srid	set to desired output coordinate reference system. srsName will be populated as defined in dz_gml_util.srid2srs utility function.
p_geometry_format	hint to push logic to use TO_GMLGEOMETRY or TO_GML311GEOMETRY. Default is to assume output should be GML 3.2. The only need for this parameter is when you do really want old GML 2.0.
p_prune_number	nonfunctional at this time, the idea here would be to remove large amounts of precision from the source Oracle coordinate numbers.
p_output_srs	nonfunctional at this time, the idea would be to allow an URN as input to replace or override p_output_srid parameter.
p_axes_latlong	nonfunctional at this time, the idea would be to swap around the longitude for latitude in the output to match desired WFS specification.
p_gml_id	The gml:id value to add to GML 3.2 output. The default value is “1”.

Returns

CLOB text in GML format

Notes

- This function is the flip-side of geogml2sdo and never had a production implementation in my work so its a bit of a place holder. Ideally the logic to unpack SDO into GML would be done in PLSQL and the dependence on the SDO_UTIL utilities removed. I just never have had the need.

dz_gml_main.fetch_gml_namespace

Direct exposure of the GML version number to namespace utility to test if your gml version is supported as you expect by the DZ_GML package. To verify, execute “SELECT dz_gml_main.fetch_gml_namespace(3.2) FROM dual;” Replace 3.2 with the version of GML you wish to test is supported.

Parameters

p_input	GML version desired for conversion

Returns

GML namespace text value

Notes

- Fairly simple logic currently, if less than 3.2 then xmlns:gml=”http://www.opengis.net/gml” if more than 3.2 and less than 3.3 then xmlns:gml=”http://www.opengis.net/gml/3.2” if more than 3.3 and less than 3.4 then xmlns:gml=”http://www.opengis.net/gml/3.3” else err

dz_gml_main.geogml2sdo

Function for conversion of GML geospatial tags into Oracle Spatial SDO_GEOMETRY geometry objects. This utility does not utilize the java SDO_UTIL.FROM_GMLGEOMETRY or SDO_UTIL.FROM_GML311GEOMETRY utilities in any fashion. Being a pure PL/SQL conversion allows more flexibility in the interpretation of more modern forms of GML.

Parameters

p_input

input GML geometry as SYS.XMLTYPE or CLOB. All input must be able to be parsed as Oracle SYS.XMLTYPE so users are encouraged to do that step themselves to avoid issues. The XML snippet should be presented as the

geometry alone, without any parent tags

the same as SDO_UTIL.FROM_GMLGEOMETRY and SDO_UTIL.FROM_GML311GEOMETRY expect.

p_gml_version

nonfunctional at this time. The parameter was intended as a hint when parsing GML when the version is unclear. May still be needed in the future.

p_srid

override for output SDO_SRID value. The srid is normally extracted from the srsName on the GML object. Use this parameter to force to a given value and skip the logic to search for the value.

p_num_dims

the number of dimensions is required to properly unpack GML coordinates. This value is normally provided in the srsDimension attribute. Set this parameter to force to a given number and skip the logic to search for the value. Setting this to 2 when you know you just have 2D geometries will provide a modest performance boost.

p_axes_latlong

Set to TRUE if your input GML has longitude and latitude reversed (e.g. WFS 1.1 and 2.0).

Returns

CLOB text in WKT or EWKT format

Notes

- For more control over conversion attempts which generate specific errors utilize the procedure versions which return an error code and status message.