

[Home \(/\)](#) / [Database \(/en/database/\)](#) / [Oracle Database Online Documentation 12c Release 1 \(12.1\) \(../index.html\)](#) / [Unstructured Data and Content Management \(../nav/portal_7.htm\)](#)

Spatial and Graph Developer's Guide

2.2 SDO_GEOMETRY Object Type

With Spatial and Graph, the geometric description of a spatial object is stored in a single row, in a single column of object type SDO_GEOMETRY in a user-defined table. Any table that has a column of type SDO_GEOMETRY must have another column, or set of columns, that defines a unique primary key for that table. Tables of this sort are sometimes referred to as spatial tables or spatial geometry tables.

Oracle Spatial and Graph defines the object type SDO_GEOMETRY as:

```
CREATE TYPE sdo_geometry AS OBJECT ( SDO_GTYPE NUMBER, SDO_SRID NUMBER, SDO_POINT SDO_POINT_TYPE, SDO_ELEM_INFO SDO_ELEM_INFO_ARRAY, SDO_ORDINATES SDO_ORDINATE_ARRAY );
```

Oracle Spatial and Graph also defines the SDO_POINT_TYPE, SDO_ELEM_INFO_ARRAY, and SDO_ORDINATE_ARRAY types, which are used in the SDO_GEOMETRY type definition, as follows:

```
CREATE TYPE sdo_point_type AS OBJECT ( X NUMBER, Y NUMBER, Z NUMBER); CREATE TYPE sdo_elem_info_array AS VARRAY (1048576) of NUMBER; CREATE TYPE sdo_ordinate_array AS VARRAY (1048576) of NUMBER;
```

Because the maximum SDO_ORDINATE_ARRAY size is 1,048,576 numbers, the maximum number of vertices in an SDO_GEOMETRY object depends on the number of dimensions per vertex: 524,288 for two dimensions, 349,525 for three dimensions, and 262,144 for four dimensions.

The sections that follow describe the semantics of each SDO_GEOMETRY attribute, and then describe some usage considerations (Usage Considerations (sdo_geometry-object-type.htm#GUID-B05F7009-3B50-48DF-9759-FC463EED822C)).

The SDO_GEOMETRY object type has methods that provide convenient access to some of the attributes. These methods are described in SDO_GEOMETRY Methods (sdo_geometry-methods.htm) .

Some Spatial and Graph data types are described in locations other than this section:

- [Data Types for Geocoding \(data-types-geocoding.htm#GUID-41EA1F45-3978-496E-93A5-295E9A6328BA\)](#) describes data types for geocoding.
- *Oracle Spatial and Graph GeoRaster Developer's Guide* ([../GEORS/geor_operations.htm#GEORS200](#)) describes data types for Oracle Spatial and Graph GeoRaster.
- *Oracle Spatial and Graph Topology Data Model and Network Data Model Graph Developer's Guide* ([../TOPOL/topology-data-model-overview.htm#TOPOL120](#)) describes data types for the Oracle Spatial and Graph topology data model.

2.2.1 SDO_GTYPE

The SDO_GTYPE attribute indicates the type of the geometry. Valid geometry types correspond to those specified in the *Geometry Object Model for the OGIS Simple Features for SQL* specification (with the exception of Surfaces). The numeric values differ from those given in the OGIS specification, but there is a direct correspondence between the names and semantics where applicable.

The SDO_GTYPE value is 4 digits in the format *DLTT*, where:

- *D* identifies the number of dimensions (2, 3, or 4)
- *L* identifies the linear referencing measure dimension for a three-dimensional linear referencing system (LRS) geometry, that is, which dimension (3 or 4) contains the measure value. For a non-LRS geometry, or to accept the Spatial and Graph default of the last dimension as the measure for an LRS geometry, specify 0. For information about the linear referencing system (LRS), see [Linear Referencing System \(linear-referencing-system.htm\)](#).
- *TT* identifies the geometry type (00 through 09, with 10 through 99 reserved for future use).

Table 2-1 ([sdo_geometry-object-type.htm#GUID-4D84CE67-67E4-4A84-8255-C586E765A94B__G1013735](#)) shows the valid SDO_GTYPE values. The Geometry Type and Description values reflect the OGIS specification.

Table 2-1 Valid SDO_GTYPE Values

Value	Geometry Type	Description
DL00	UNKNOWN_GEOMETRY	Spatial and Graph ignores this geometry.
DL01	POINT	Geometry contains one point.

Value	Geometry Type	Description
DL02	LINE or CURVE	Geometry contains one line string that can contain straight or circular arc segments, or both. (LINE and CURVE are synonymous in this context.)
DL03	POLYGON or SURFACE	Geometry contains one polygon with or without holes, ^{Foot 1} (#n_1) or one surface consisting of one or more polygons. In a three-dimensional polygon, all points must be on the same plane.
DL04	COLLECTION	Geometry is a heterogeneous collection of elements. COLLECTION is a superset that includes all other types.
DL05	MULTIPOINT	Geometry has one or more points. (MULTIPOINT is a superset of POINT.)
DL06	MULTILINE or MULTICURVE	Geometry has one or more line strings. (MULTILINE and MULTICURVE are synonymous in this context, and each is a superset of both LINE and CURVE.)
DL07	MULTIPOLYGON or MULTISURFACE	Geometry can have multiple, disjoint polygons (more than one exterior boundary). or surfaces (MULTIPOLYGON is a superset of POLYGON, and MULTISURFACE is a superset of SURFACE.)
DL08	SOLID	Geometry consists of multiple surfaces and is completely enclosed in a three-dimensional space. Can be a cuboid or a frustum.
DL09	MULTISOLID	Geometry can have multiple, disjoint solids (more than one exterior boundary). (MULTISOLID is a superset of SOLID.)

Footnote 1

For a polygon with holes, enter the exterior boundary first, followed by any interior boundaries.

The *D* in the Value column of Table 2-1 ([sdo_geometry-object-type.htm#GUID-4D84CE67-67E4-4A84-8255-C586E765A94B__G1013735](#)) is the number of dimensions: 2, 3, or 4. For example, an SDO_GTYPE value of 2003 indicates a two-dimensional polygon. The number of dimensions reflects the number of ordinates used to represent each vertex (for example, X,Y for two-dimensional objects).

In any given layer (column), all geometries must have the same number of dimensions. For example, you cannot mix two-dimensional and three-dimensional data in the same layer.

The following methods are available for returning the individual *DLTT* components of the SDO_GTYPE for a geometry object: `Get_Dims`, `Get_LRS_Dim`, and `Get_Gtype`. These methods are described in [SDO_GEOMETRY Methods \(sdo_geometry-methods.htm\)](#) .

For more information about SDO_GTYPE values for three-dimensional geometries, see Table 1-1 ([three-dimensional-spatial-objects.htm#GUID-6F3A9C6E-3617-47D9-AE61-CF9F1770555A__BABFDHHDH](#)) in [Three-Dimensional Spatial Objects \(three-dimensional-spatial-objects.htm\)](#) .

2.2.2 SDO_SRID

The SDO_SRID attribute can be used to identify a coordinate system (spatial reference system) to be associated with the geometry. If SDO_SRID is null, no coordinate system is associated with the geometry. If SDO_SRID is not null, it must contain a value from the SRID column of the SDO_COORD_REF_SYS table (described in [SDO_COORD_REF_SYS Table \(coordinate-systems-data-structures.htm#GUID-951C650B-5966-408C-B5E9-28142C43C141\)](#)), and this value must be inserted into the SRID column of the USER_SDO_GEOM_METADATA view (described in [Geometry Metadata Views \(geometry-metadata-views.htm#GUID-AEA8FA01-880B-4865-8074-3B6FBD17A5E1\)](#)).

All geometries in a geometry column must have the same SDO_SRID value if a spatial index will be built on that column.

For information about coordinate systems, see [Coordinate Systems \(Spatial Reference Systems\) \(coordinate-systems-spatial-reference-systems.htm\)](#) .

2.2.3 SDO_POINT

The SDO_POINT attribute is defined using the SDO_POINT_TYPE object type, which has the attributes X, Y, and Z, all of type NUMBER. (The SDO_POINT_TYPE definition is shown in [SDO_GEOMETRY Object Type \(sdo_geometry-object-type.htm#GUID-683FF8C5-A773-4018-932D-2AF6EC8BC119\)](#) .) If the SDO_ELEM_INFO and SDO_ORDINATES arrays are both null, and the SDO_POINT attribute is non-null, then the X, Y, and Z values are considered to be the coordinates for a point geometry. Otherwise, the SDO_POINT attribute is ignored by Spatial and Graph. You should store point geometries in the SDO_POINT attribute for optimal storage; and if you have only point geometries in a layer, it is strongly recommended that you store the point geometries in the SDO_POINT attribute.

[Point \(point.htm\)](#) illustrates a point geometry and provides examples of inserting and querying point geometries.

Note:

2.2.4 SDO_ELEM_INFO

The SDO_ELEM_INFO attribute is defined using a varying length array of numbers. This attribute lets you know how to interpret the ordinates stored in the SDO_ORDINATES attribute (described in SDO_ORDINATES (sdo_geometry-object-type.htm#GUID-79D7620B-5526-40FD-9604-9642B79908F2)).

Each triplet set of numbers is interpreted as follows:

- **SDO_STARTING_OFFSET** -- Indicates the offset within the SDO_ORDINATES array where the first ordinate for this element is stored. Offset values start at 1 and not at 0. Thus, the first ordinate for the first element will be at SDO_GEOMETRY.SDO_ORDINATES(1). If there is a second element, its first ordinate will be at SDO_GEOMETRY.SDO_ORDINATES(*n*), where *n* reflects the position within the SDO_ORDINATE_ARRAY definition (for example, 19 for the 19th number, as in Polygon with a Hole (polygon-hole.htm)).
- **SDO_ETYPE** -- Indicates the type of the element. Valid values are shown in Table 2-2 (sdo_geometry-object-type.htm#GUID-270AE39D-7B83-46D0-9DD6-E5D99C045021__BGHDGCCE) .

SDO_ETYPE values 1, 2, 1003, and 2003 are considered *simple elements*. They are defined by a single triplet entry in the SDO_ELEM_INFO array. For SDO_ETYPE values 1003 and 2003, the first digit indicates *exterior* (1) or *interior* (2):

1003: exterior polygon ring (must be specified in counterclockwise order)

2003: interior polygon ring (must be specified in clockwise order)

Note:

The use of 3 as an SDO_ETYPE value for polygon ring elements in a single geometry is discouraged. You should specify 3 only if you do not know if the simple polygon is exterior or interior, and you should then upgrade the table or layer to the current format using the SDO_MIGRATE.TO_CURRENT (sdo_migrate-to_current.htm) procedure, described in SDO_MIGRATE Package (Upgrading) (sdo_migrate-package-upgrading.htm) .

You cannot mix 1-digit and 4-digit SDO_ETYPE values in a single geometry.

SDO_ETYPE values 4, 1005, 2005, 1006, and 2006 are considered *compound elements*. They contain at least one header triplet with a series of triplet values that belong to the compound element. For 4-digit SDO_ETYPE values, the first digit indicates *exterior* (1) or *interior* (2):

1005: exterior polygon ring (must be specified in counterclockwise order)

2005: interior polygon ring (must be specified in clockwise order)

1006: exterior surface consisting of one or more polygon rings

2006: interior surface in a solid element

1007: solid element

The elements of a compound element are contiguous. The last point of a subelement in a compound element is the first point of the next subelement. The point is not repeated.

- **SDO_INTERPRETATION** -- Means one of two things, depending on whether or not **SDO_ETYPE** is a compound element.

If SDO_ETYPE is a compound element (4, 1005, or 2005), this field specifies how many subsequent triplet values are part of the element.

If the SDO_ETYPE is not a compound element (1, 2, 1003, or 2003), the interpretation attribute determines how the sequence of ordinates for this element is interpreted. For example, a line string or polygon boundary may be made up of a sequence of connected straight line segments or circular arcs.

Descriptions of valid SDO_ETYPE and SDO_INTERPRETATION value pairs are given in Table 2-2 ([sdo_geometry-object-type.htm#GUID-270AE39D-7B83-46D0-9DD6-E5D99C045021__BGHDGCCE](#)) .

If a geometry consists of more than one element, then the last ordinate for an element is always one less than the starting offset for the next element. The last element in the geometry is described by the ordinates from its starting offset to the end of the SDO_ORDINATES varying length array.

For compound elements (SDO_ETYPE values 4, 1005, or 2005), a set of n triplets (one for each subelement) is used to describe the element. It is important to remember that subelements of a compound element are contiguous. The last point of a subelement is the first point of the next subelement. For subelements 1 through $n-1$, the end point of one subelement is the same as the starting point of the next subelement. The starting point for subelements 2... $n-2$ is the same as the end point of subelement 1... $n-1$. The last ordinate of subelement n is either the starting offset minus 1 of the next element in the geometry, or the last ordinate in the SDO_ORDINATES varying length array.

The current size of a varying length array can be determined by using the function `varray_variable.Count` in PL/SQL or `OCICollSize` in the Oracle Call Interface (OCI).

The semantics of each SDO_ETYPE element and the relationship between the SDO_ELEM_INFO and SDO_ORDINATES varying length arrays for each of these SDO_ETYPE elements are given in Table 2-2 ([sdo_geometry-object-type.htm#GUID-270AE39D-7B83-46D0-9DD6-E5D99C045021__BGHDGCCE](#)) .

Table 2-2 Values and Semantics in SDO_ELEM_INFO

SDO_ETYPE	SDO_INTERPRETATION	Meaning
0	(any numeric value)	Type 0 (zero) element. Used to model geometry types not supported by Oracle Spatial and Graph. For more information, see Type 0 (Zero) Element (type-0-zero-element.htm) .
1	1	Point type.
1	0	Orientation for an oriented point. For more information, see Oriented Point (oriented-point.htm) .
1	$n > 1$	Point cluster with n points.
2	1	Line string whose vertices are connected by straight line segments.
2	2	<p>Line string made up of a connected sequence of circular arcs.</p> <p>Each circular arc is described using three coordinates: the start point of the arc, any point on the arc, and the end point of the arc. The coordinates for a point designating the end of one arc and the start of the next arc are not repeated. For example, five coordinates are used to describe a line string made up of two connected circular arcs. Points 1, 2, and 3 define the first arc, and points 3, 4, and 5 define the second arc, where point 3 is only stored once.</p>
2	3	NURBS (non-uniform rational B-spline) curve. For more information, see NURBS Curve Support in Oracle Spatial and Graph (nurbs-curve-support-oracle-spatial-and-graph.htm) .
1003 or 2003	1	Simple polygon whose vertices are connected by straight line segments. You must specify a point for each vertex; and the last point specified must be exactly the same point as the first (within the tolerance value), to close the polygon. For example, for a 4-sided polygon, specify 5 points, with point 5 the same as point 1.

SDO_ETYPE	SDO_INTERPRETATION	Meaning
1003 or 2003	2	<p>Polygon made up of a connected sequence of circular arcs that closes on itself. The end point of the last arc is the same as the start point of the first arc.</p> <p>Each circular arc is described using three coordinates: the start point of the arc, any point on the arc, and the end point of the arc. The coordinates for a point designating the end of one arc and the start of the next arc are not repeated. For example, five coordinates are used to describe a polygon made up of two connected circular arcs. Points 1, 2, and 3 define the first arc, and points 3, 4, and 5 define the second arc. The coordinates for points 1 and 5 must be the same (tolerance is not considered), and point 3 is not repeated.</p>
1003 or 2003	3	<p>Rectangle type (sometimes called <i>optimized rectangle</i>). A bounding rectangle such that only two points, the lower-left and the upper-right, are required to describe it. The rectangle type can be used with geodetic or non-geodetic data. However, with geodetic data, use this type only to create a query window (not for storing objects in the database).</p> <p>For information about using this type with geodetic data, including examples, see Geodetic MBRs (geodetic-coordinate-support.htm#GUID-AFB77449-C15A-4A64-9A66-384C16413C98) . For information about creating three-dimensional optimized rectangles, see Three-Dimensional Optimized Rectangles (three-dimensional-optimized-rectangles.htm) .</p>
1003 or 2003	4	<p>Circle type. Described by three distinct non-colinear points, all on the circumference of the circle.</p>
4	$n > 1$	<p>Compound line string with some vertices connected by straight line segments and some by circular arcs. The value n in the Interpretation column specifies the number of contiguous subelements that make up the line string.</p> <p>The next n triplets in the SDO_ELEM_INFO array describe each of these subelements. The subelements can only be of SDO_ETYPE 2. The last point of a subelement is the first point of the next subelement, and must not be repeated.</p> <p>See Compound Line String (compound-line-string.htm) and Figure 2-5 (compound-line-string.htm#GUID-C1BC41F3-90B6-456B-B1FF-0BFED41FF32__I1006698) for an example of a compound line string geometry.</p>

SDO_ETYPE	SDO_INTERPRETATION	Meaning
1005 or 2005	$n > 1$	<p>Compound polygon with some vertices connected by straight line segments and some by circular arcs. The value n in the Interpretation column specifies the number of contiguous subelements that make up the polygon.</p> <p>The next n triplets in the SDO_ELEM_INFO array describe each of these subelements. The subelements can only be of SDO_ETYPE 2. The end point of a subelement is the start point of the next subelement, and it must not be repeated. The start and end points of the polygon must be exactly the same point (tolerance is ignored).</p> <p>See Compound Polygon (compound-polygon.htm) for an example of a compound polygon geometry.</p>
1006 or 2006	$n > 1$	<p>Surface consisting of one or more polygons, with each edge shared by no more than two polygons. A surface contains an area but not a volume. The value n in the Interpretation column specifies the number of polygons that make up the surface.</p> <p>The next n triplets in the SDO_ELEM_INFO array describe each of these polygon subelements.</p> <p>A surface must be three-dimensional. For an explanation of three-dimensional support in Spatial and Graph, see Three-Dimensional Spatial Objects (three-dimensional-spatial-objects.htm) .</p>
1007	$n = 1$ or 3	<p>Solid consisting of multiple surfaces that are completely enclosed in a three-dimensional space, so that the solid has an interior volume. A solid element can have one exterior surface defined by the 1006 elements and zero or more interior boundaries defined by the 2006 elements. The value n in the Interpretation column must be 1 or 3.</p> <p>Subsequent triplets in the SDO_ELEM_INFO array describe the exterior 1006 and optional interior 2006 surfaces that make up the solid element.</p> <p>If n is 3, the solid is an <i>optimized box</i>, such that only two three-dimensional points are required to define it: one with minimum values for the box in the X, Y, and Z dimensions and another with maximum values for the box in the X, Y, and Z dimensions. For example: <code>SDO_GEOMETRY(3008, NULL, NULL, SDO_ELEM_INFO_ARRAY(1,1007,3), SDO_ORDINATE_ARRAY(1,1,1, 3,3,3))</code></p> <p>For an explanation of three-dimensional support in Spatial and Graph, see Three-Dimensional Spatial Objects (three-dimensional-spatial-objects.htm) .</p>

2.2.5 SDO_ORDINATES

The SDO_ORDINATES attribute is defined using a varying length array (1048576) of NUMBER type that stores the coordinate values that make up the boundary of a spatial object. This array must always be used in conjunction with the SDO_ELEM_INFO varying length array. The values in the array are ordered by dimension. For example, a polygon whose boundary has four two-dimensional points is stored as {X1, Y1, X2, Y2, X3, Y3, X4, Y4, X1, Y1}. If the points are three-dimensional, then they are stored as {X1, Y1, Z1, X2, Y2, Z2, X3, Y3, Z3, X4, Y4, Z4, X1, Y1, Z1}. The number of dimensions associated with each point is stored as metadata in the xxx_SDO_GEOM_METADATA views, described in Geometry Metadata Views (geometry-metadata-views.htm#GUID-AEA8FA01-880B-4865-8074-3B6FBD17A5E1) .

The values in the SDO_ORDINATES array must all be valid and non-null. There are no special values used to delimit elements in a multielement geometry. The start and end points for the sequence describing a specific element are determined by the STARTING_OFFSET values for that element and the next element in the SDO_ELEM_INFO array, as explained in SDO_ELEM_INFO (sdo_geometry-object-type.htm#GUID-270AE39D-7B83-46D0-9DD6-E5D99C045021) . The offset values start at 1. SDO_ORDINATES(1) is the first ordinate of the first point of the first element.

2.2.6 Usage Considerations

You should use the SDO_GTYPE values as shown in SDO_GTYPE (sdo_geometry-object-type.htm#GUID-4D84CE67-67E4-4A84-8255-C586E765A94B) ; however, Spatial and Graph does not check or enforce all geometry consistency constraints. Spatial and Graph does check the following:

- For SDO_GTYPE values *d001* and *d005*, any subelement not of SDO_ETYPE 1 is ignored.
- For SDO_GTYPE values *d002* and *d006*, any subelement not of SDO_ETYPE 2 or 4 is ignored.
- For SDO_GTYPE values *d003* and *d007*, any subelement not of SDO_ETYPE 3 or 5 is ignored. (This includes SDO_ETYPE variants 1003, 2003, 1005, and 2005, which are explained in SDO_ELEM_INFO (sdo_geometry-object-type.htm#GUID-270AE39D-7B83-46D0-9DD6-E5D99C045021)).

The SDO_GEOM.VALIDATE_GEOMETRY_WITH_CONTEXT (sdo_geom-validate_geometry_with_context.htm) function can be used to evaluate the consistency of a single geometry object or of all geometry objects in a specified feature table.

