

## SQL Espacial

# Funciones de acceso a las propiedades de las geometrías

forma **SIG**

La plataforma de aprendizaje en SIG Libre



SERVEI DE SISTEMES  
D'INFORMACIÓ GEOGRÀFICA  
I TELEDETECCIÓ  
Universitat de Girona



**UdG Formació**

FUNDACIÓ UNIVERSITAT DE GIRONA:  
INNOVACIÓ I FORMACIÓ

Edita: Servicio de SIG y Teledetección (SIGTE) de la Universitat de Girona

Año: 2014

Contenidos elaborados por: Toni Hernández Vallès

Este documento está sujeto a la licencia Creative Commons BY-NC-SA, es decir, sois libres de copiar, distribuir y comunicar esta obra, bajo las siguientes condiciones:



**Atribución** — Debe reconocer los créditos de la obra de la manera especificada por el autor o el licenciante (pero no de una manera que sugiera que tiene su apoyo o que apoyan el uso que hace de su obra).



**No Comercial** — No puede utilizar esta obra para fines comerciales.



**Compartir bajo la Misma Licencia** — Si altera o transforma esta obra, o genera una obra derivada, sólo puede distribuir la obra generada bajo una licencia idéntica a ésta.

## Funciones de acceso a las propiedades de las geometrías

Este grupo de funciones devuelve información, perteneciente o relativa, a un objeto de tipo *geometry*. Vamos a clasificar estas funciones en dos grupos. Las que retornan un tipo de datos distinto a *geometry* y las que retornan un tipo de datos *geometry*.

### **Funciones que no devuelven un tipo de datos *geometry***

#### **GeometryType(geometry)**

Devuelve el tipo de geometría en formato texto (*point*, *linestring*, *multipoint*, etc)

#### **St\_CoordDim(geometry)**

Devuelve la dimensión de las coordenadas de una geometría.

#### **St\_Dimension(geometry)**

Devuelve la dimensión inherente a la geometría, la cual debe ser menor o igual a la dimensión de las coordenadas.

#### **St\_IsClosed(geometry)**

Devuelve cierto si el punto de inicio y final de un LINESTRING son coincidentes

#### **St\_Empty(geometry)**

Devuelve cierto si la geometría está vacía.

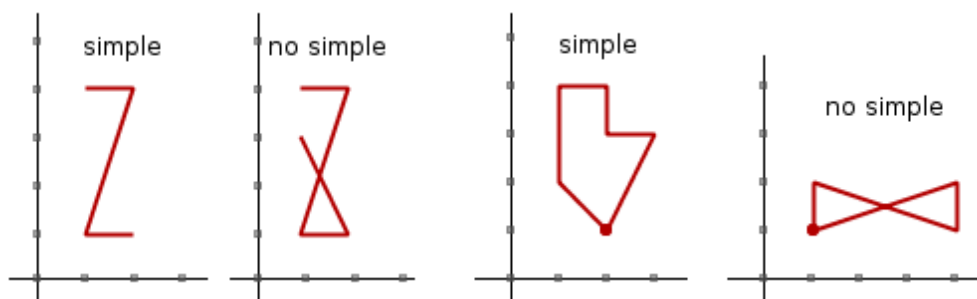
#### **St\_IsSimple(geometry)**

Se utiliza sobre geometrías de tipo *multipoint*, *linestring* y *multilinestring*. El resto de geometrías (*point*, *polygon* y *multipolygon*) son, por definición, geometrías simples. Devuelve cierto si la geometría no presenta anomalías. Estas anomalías son compatibles con las

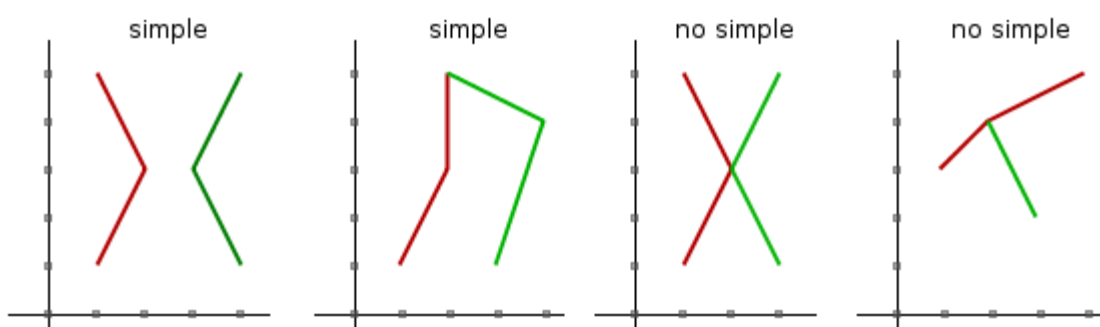
especificaciones del *Open Geospatial Consortium* (OGC). Cada tipo de geometría puede presentar las siguientes anomalías:

Multipoint: No son *simples* únicamente si contiene más de un punto con las mismas coordenadas.

Linestring: Una geometría de tipo *linestring* es *simple* si no pasa por un mismo punto más de una vez. Existe sin embargo una excepción a esta regla. Si un *linestring* pasa dos veces por el punto final de la geometría, entonces se considera una geometría *simple* además de cerrada. También nos podemos referir a este tipo de geometrías como anillos lineales.



Multilinestring: Son geometrías simples si cada uno de sus elementos (*linestring*) son *simples* y solo la intersección entre dos elementos tiene lugar en los puntos que se encuentran en los límites (*boundary*) de los dos elementos. La siguiente imagen muestra los distintos elementos de una misma geometría en color verde y rojo.

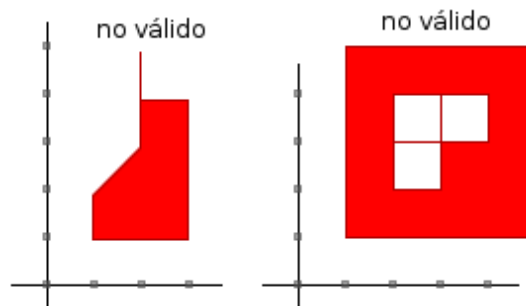


### St\_IsValid(geometry)

Generalmente se utiliza sobre geometrías de tipo *polygon* y *multipolygon*. Devuelve cierto si la geometría no presenta anomalías. Estas anomalías son compatibles con las especificaciones del *Open Geospatial Consortium* (OGC). Estas geometrías pueden presentar las siguientes

anomalías:

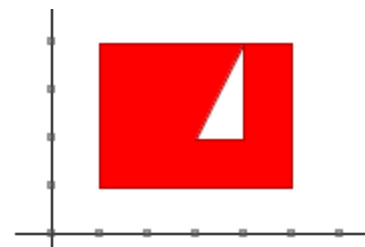
Polygon: Una geometría de tipo *polygon* es *válida* si no existen dos anillos en los límites (*boundary*) del polígono que se cruzan. El límite del polígono puede interseccionar con el polígono propiamente, en un punto pero solo de manera tangencial. Los anillos interiores deben estar totalmente incluidos dentro del anillo exterior (sin agujeros definidos fuera del polígono). Tampoco pueden aparecer líneas de corte ni crestas puntiagudas.



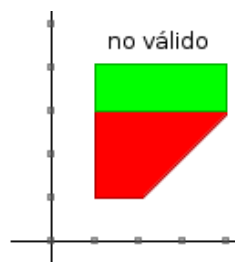
El polígono que aparece en la siguiente imagen será válido si se define con un agujero. En caso de definirse sin el agujero será considerado un polígono no válido.

No válido: Polygon((1 1, 1 4, 4 4, 3 2, 4 2, 4 4, 5 4, 5 1, 1 1))

Válido: Polygon((1 1, 1 4, 5 4, 5 1, 1 1), (4 4, 3 2, 4 2, 4 4))



Multipolygon: Una geometría de tipo *multipolygon* es válida si todos sus elementos (*polygon*) son válidos y además no existen dos anillos interiores de elementos distintos que interseccionan. Los límites de dos elementos pueden estar en contacto pero solo en un número finito de puntos. La siguiente imagen muestra los distintos elementos de una misma geometría en color verde y rojo.



**St\_IsRing(geometry)**

Devuelve cierto si la geometría de tipo LINESTRING está cerrada (St\_IsClosed) y es sencilla (St\_IsSimple).

**St\_IsValidReason(geometry)**

Devuelve una explicación sencilla (en formato de texto) de por qué razón una geometría no es válida.

**St\_NPoints(geometry)**

Devuelve el número de nodos de una geometría.

**St\_NRings(geometry)**

Devuelve el número de anillos de una geometría.

**St\_NumGeometries(geometry)**

Devuelve el número de geometrías de una geometría de tipo *geometrycollection*. En caso contrario devuelve un valor nulo.

**St\_NumInteriorRings(geometry)**

Devuelve el número de anillos internos de un polígono. También funciona para geometrías de tipo *multipolygon* en cuyo caso devolverá el número de anillos internos del primer polígono de la geometría.

**St\_NumPoints(geometry)**

Devuelve el número de puntos de una geometría de tipo LINESTRING.

**St\_Srid(geometry)**

Devuelve el código SRID del sistema de referencia de la geometría.

**St\_X(geometry)**

Devuelve la coordenada X de una geometría de tipo POINT.

**St\_Y(geometry)**

Devuelve la coordenada Y de una geometría de tipo POINT.

**St\_Z(geometry)**

Devuelve la coordenada Z de una geometría de tipo POINT.

**Funciones que devuelven un objeto espacial****St\_Boundary(geometry)**

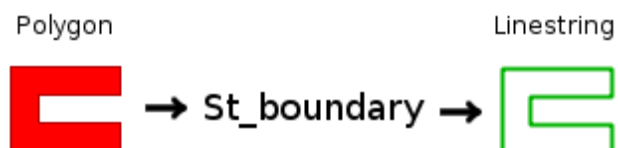
Devuelve el límite (frontera) de la geometría. Un polígono sin agujeros devolverá una *LineString*, mientras que un polígono con agujeros devolverá un *MultiLineString*.

En los siguientes ejemplos hemos utilizado la función `St_AsText(geometry)` para que el resultado de la consulta se muestre en formato WKT.

Ejemplo:

```
SELECT St_Astext(St_boundary(St_GeomFromText('polygon((1 1, 5 1, 5 2, 2 2, 2 3, 5 3, 5 4, 1 4, 1 1))'))));
```

Resultado: LINESTRING(1 1,5 1,5 2,2 2,2 3,5 3,5 4,1 4,1 1).



**St\_PointN(geometry, N)**

Devuelve el enésimo punto de una geometría de tipo LINESTRING.

Ejemplo:

```
SELECT St_AsText(ST_PointN(ST_GeomFromText('LINESTRING(1 2, 3 2, 1 2)'),2));
```

Resultado: Devuelve el punto número 2 de la geometría. POINT(3 2)

**St\_StartPoint(geometry)**

Obtiene el primer punto de una geometría de tipo LINESTRING.

Ejemplo:

```
SELECT ST_AsText(St_StartPoint(St_GeomFromText('LINESTRING(1 2, 3 2, 4 2)')));
```

Resultado: POINT(1 2)

**St\_EndPoint(geometry)**

Obtiene el último punto de una geometría de tipo LINESTRING.

Ejemplo:

```
SELECT St_AsText(St_StartPoint(St_GeomFromText('LINESTRING(1 2, 3 2, 4 2)')));
```

Resultado: POINT(4 2)

**St\_Envelope(geometry)**

Devuelve el polígono correspondiente al mínimo rectángulo (*bounding box*) que contiene a la geometría indicada. Las líneas totalmente verticales u horizontales devuelven un st\_envelope de tipo *linestring*.

Ejemplo:

```
SELECT St_envelope(St_GeomFromText('Polygon((1 4, 4 6,2 1,5 3, 1 4))'));
```



Resultado:



**St\_ExteriorRing(geometry)**

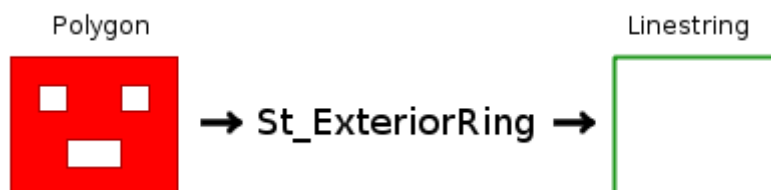
**St\_ExteriorRing(geometry)**

Devuelve una geometría de tipo *LineString* del anillo exterior de una geometría de tipo POLYGON.

Ejemplo:

```
SELECT St_Exteriorring(St_geomfromtext('polygon((2 1,2 6, 8 6,8 1,2 1),(3 5,4 5,4 4,3 4, 3 5),(6 5,7 5, 7 4, 6 4, 6 5),(4 3, 6 3, 6 2, 4 2, 4 3))'));
```

Resultado:



**St\_InteriorRingN(geometry, N)**

Devuelve un *LineString* del 'enésimo' anillo de una geometría de tipo POLYGON. El siguiente ejemplo muestra los distintos resultados (1, 2 y 3) para los distintos 'N' anillos del polígono inicial.

Ejemplo:

```
SELECT St_InteriorRingN(St_geomfromtext('polygon((2 1,2 6, 8 6,8 1,2 1),(3 5,4 5,4  
4,3 4, 3 5),(6 5,7 5, 7 4, 6 4, 6 5),(4 3, 6 3, 6 2, 4 2, 4 3))'), N);
```

Resultado: Donde el parámetro 'N' se ha substituido por los valores 1, 2 y 3.





SERVEI DE SISTEMES  
D'INFORMACIÓ GEOGRÀFICA  
I TELEDETECCIÓ  
Universitat de Girona



**UdGFormació**

FUNDACIÓ UNIVERSITAT DE GIRONA:  
INNOVACIÓ I FORMACIÓ

[www.sigte.udg.edu/formasig](http://www.sigte.udg.edu/formasig)

[formasig@sigte.org](mailto:formasig@sigte.org)