
PostGIS – Fonctions spatiales de base

Nicolas Ribot - Licence GNU FDL - Version 1.4

Fonctions de gestion

- **postgis_full_version**: Affiche la version courante de Postgis, GEOS, PROJ4 et PostgreSQL
- **UpdateGeometrySRID**: Met à jour le SRID de toutes les entités d'une colonne géographique, de la table geometry_columns, et de la contrainte sur le SRID de la table

AddGeometryColumn

DropGeometryColumn

DropGeometryTable

PostGIS_Full_Version

PostGIS_GEOS_Version

PostGIS_LibXML_Version

PostGIS_Lib_Build_Date

PostGIS_Lib_Version

PostGIS_PROJ_Version

PostGIS_Scripts_Build_Date

PostGIS_Scripts_Installed

PostGIS_Scripts_Released

PostGIS_Version

Populate_Geometry_Columns

UpdateGeometrySRID



Constructeurs de géométries

- ❖ **ST_GeomFromText** : Construit une géométrie à partir de sa représentation WKT.

```
select st_geomFromText(
    'POINT(2.5 43)', 4326);
```

- ❖ **ST_GeometryFromText**: Alias pour st_geomFromText
- ❖ **ST_GeomFromWKB**: Construit une géométrie à partir de sa représentation WKB.
- ❖ **ST_makeBox2D, ST_makeBox3D**: Construit une bbox à partir de points.
- ❖ **ST_MakePoint/ST_Point**: Construit un point à partir de coordonnées

```
SELECT ST_MakePoint(1,2);
```

- ❖ **ST_MakeLine**: Construit une ligne a partir de points

```
SELECT ST_AsText (
    ST_MakeLine (
        ST_MakePoint (1, 2) ,
        ST_MakePoint (3, 4) ) ) ;
```

- ❖ **ST_MakePolygon**: Construit un polygone a partir d'une ligne fermée

Constructeurs de géométries (suite)

ST_BdPolyFromText	ST_GeomFromText	ST_MPointFromText
ST_BdMPolyFromText	ST_GeomFromWKB	ST_MPolyFromText
ST_GeogFromText	ST_LineFromMultiPoint	ST_Point
ST_GeographyFromText	ST_LineFromText	ST_PointFromText
ST_GeogFromWKB	ST_LineFromWKB	ST_PointFromWKB
ST_GeomCollFromText	ST_LinestringFromWKB	ST_Polygon
ST_GeomFromEWKB	ST_MakeBox2D	ST_PolygonFromText
ST_GeomFromEWKT	ST_3DMakeBox	ST_WKBToSQL
ST_GeometryFromText	ST_MakeLine	ST_WKTTToSQL
ST_GeomFromGML	ST_MakeEnvelope	
ST_GeomFromGeoJSON	ST_MakePolygon	
N	ST_MakePoint	
ST_GeomFromKML	ST_MakePointM	
ST_GMLToSQL	ST_MLineFromText	

Accessseurs de géométries

- ❖ **GeometryType/ST_GeometryType**: Retourne le type de la géométrie ('POINT', 'POLYGONE'), respectivement ('ST_POINT', 'ST_POLYGONE')
- ❖ **ST_Boundary**: Retourne l'ensemble formant la frontière d'un objet

```
select st_boundary(st_point(0, 0));
```

```
select st_boundary(
```

```
    st_geomFromText('LINESTRING(0 0, 1 1, 2 2)', -1));
```

```
select astext (
```

```
    st_boundary (
```

```
        st_polygonFromText ('POLYGON((0 0, 6 0, 6 6,
0 6, 0 0), (1 1, 5 1, 5 5, 1 5, 1 1))')));
```

- ❖ **st_CoordDim**: Retourne la dimension des coordonnées de la géométrie
- ❖ **ST_EndPoint/ST_StartPoint**: Retourne le dernier, respectivement le premier point d'une LINESTRING
- ❖ **ST_Envelope**: Retourne une géométrie représentant l'enveloppe englobant la géométrie.
- ❖ **ST_ExteriorRing**: Retourne une LINESTRING représentant l'extérieur du POLYGONE
- ❖ **ST_GeometryN**: Retourne la Nieme géométrie (commençant à 1) d'une collection (GeomCol, MULTI*)
- ❖ **ST_InteriorRingN**: Retourne la Nieme LINESTRING représentant l'intérieur d'un POLYGONE

Accesseurs de géométries (suite)

- ❖ **ST_IsClosed**: Vrai si le premier et le dernier point d'une LINESTRING sont les mêmes
- ❖ **ST_IsEmpty**: Vrai si cette géométrie est vide (GEOMETRYCOLLECTION(EMPTY))
- ❖ **ST_IsRing**: Retourne vrai si la géométrie est un ring (linestring fermée)
- ❖ **ST_IsSimple**: Retourne vrai si la géométrie est simple (ligne ne se croisant pas, par exemple)
- ❖ **ST_IsValid**: Vrai si cette géométrie est bien formée
- ❖ **ST_IsValidReason**: Retourne la raison de l'invalidité de la géométrie.

- ❖ **ST_NPoints**: Retourne le nombre total de sommets dans la géométrie.
- ❖ **ST_NRings**: Si la géométrie est un POLYGON ou MULTIPOLYGON, retourne le nombre total de rings.
- ❖ **ST_NumGeometries**: Si la géométrie est une GEOMETRYCOLLECTION ou un MULTI*, retourne le nombre total de géométries dans la collection.
- ❖ **ST_NumInteriorRings**: Si la géométrie est un POLYGON ou un MULTIPOLYGON, retourne le nombre de rings intérieurs dans le **premier polygone** de la géométrie.

- ❖ **ST_PointN**: Retourne le Nieme point dans une ST_LINESTRING ou ST_CIRCULARSTRING.
- ❖ **ST_Summary**: Retourne un résumé de la géométrie

```
select st_summary(the_geom)  
from departements;
```
- ❖ **ST_X, ST_Y, ST_Z**: Retourne les coordonnées X, Y, Z, respectivement, du POINT

Accesseurs de géométries (suite)

GeometryType	ST_IsValidReason	ST_X
ST_Boundary	ST_IsValidDetail	ST_XMax
ST_CoordDim	ST_M	ST_XMin
ST_Dimension	ST_NDims	ST_Y
ST_EndPoint	ST_NPoints	ST_YMax
ST_Envelope	ST_NRings	ST_YMin
ST_ExteriorRing	ST_NumGeometries	ST_Z
ST_GeometryN	ST_NumInteriorRings	ST_ZMax
ST_GeometryType	ST_NumInteriorRing	ST_Zmflag
ST_InteriorRingN	ST_NumPatches	ST_ZMin
ST_IsClosed	ST_NumPoints	
ST_IsCollection	ST_PatchN	
ST_IsEmpty	ST_PointN	
ST_IsRing	ST_SRID	
ST_IsSimple	ST_StartPoint	
ST_IsValid	ST_Summary	

Editeurs de géométries

- ❖ **ST_AddPoint/ST_RemovePoint**: Ajoute/supprime un point à une LINESTRING avant le/au point de position donnée (0-based index).
- ❖ **ST_Affine, ST_Rotate, ST_RotateX, ST_RotateY, ST_Scale, ST_Transcale**: Effectue des opérations math. de transformation affine, rotation mise à l'échelle, translation, etc.
- ❖ **ST_Force2D, ST_Force3D, ST_Force3DM**: Force la dimension de la géométrie
- ❖ **ST_ForceRHR**: Force l'orientation des sommets du POLYGON selon la règle de la main droite (sens horaire)

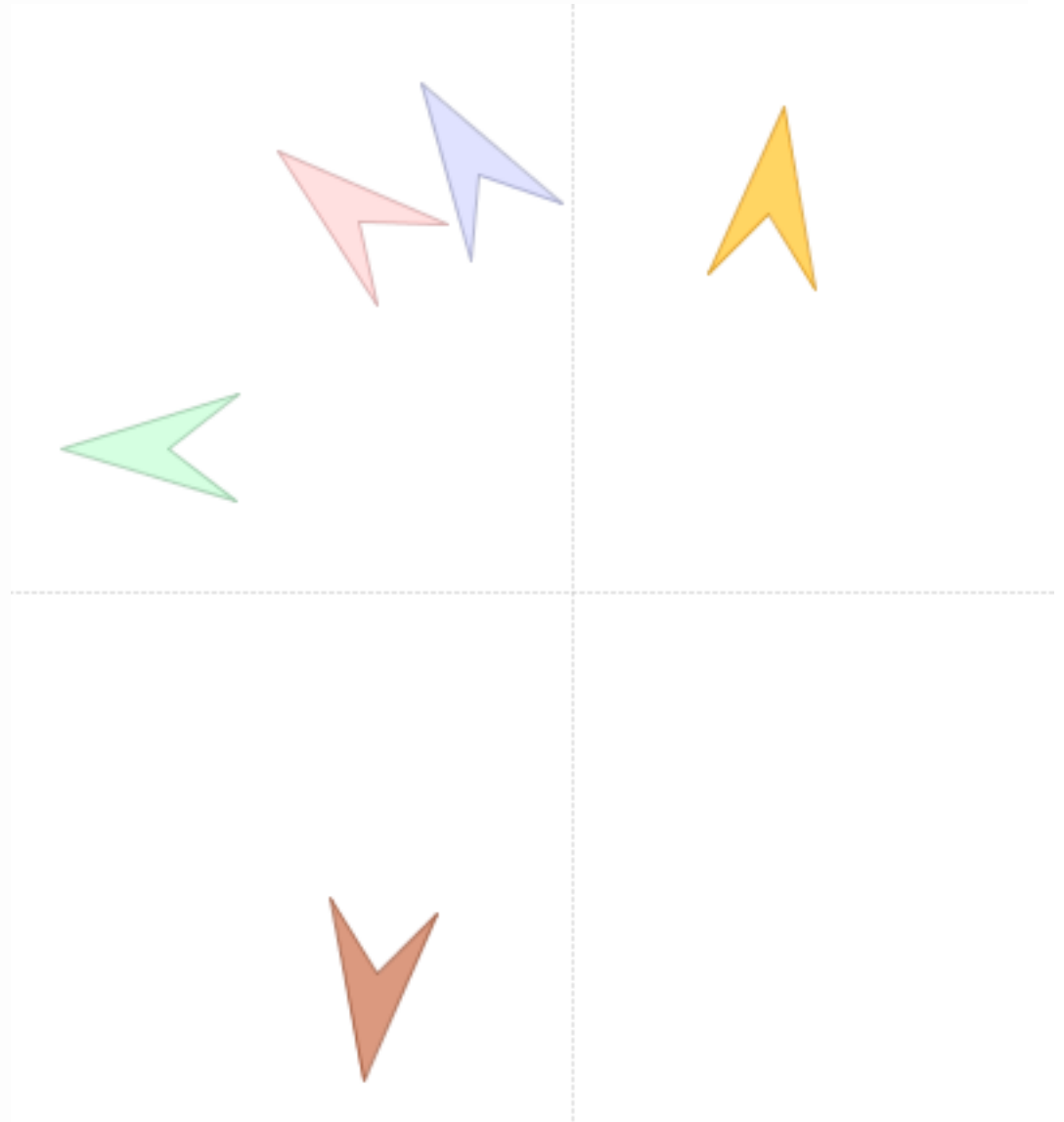


Editeurs de géométries (suite)



ST_rotate

```
select st_rotate(  
'POLYGON((1.4 3.2,0.9 2.1,1.3 2.5,1.6 2,1.4  
3.2))'::geometry, 3.149);
```



- ❖ **ST_LineMerge**: Assemble des LINESTRING entre elles provenant d'une MULTILINESTRING
- ❖ **ST_Multi**: Force la géométrie à être de la forme MULTI*.
- ❖ **ST_Force2D, ST_Force3D, ST_Force3DM**: Force la dimension de la géométrie
- ❖ **ST_ForceRHR**: Force l'orientation des sommets du POLYGONE selon la règle de la main droite (sens horaire)
- ❖ **ST_Reverse**: Inverse l'ordre les sommets de la géométrie



- ❖ **ST_Transform**: Change le système de coordonnées de la géométrie passée en argument et retourne une nouvelle géométrie

```
ST_Transform(geometry geom, integer srid)
```

```
SELECT ST_Transform(the_geom, 4326) FROM ma_table;
```

- ❖ **ST_SetSRID**: Met à jour le SRID de la géométrie
- ❖ **ST_Segmentize**: Retourne la geometry modifiée de telle sorte que la longueur maximum d'un segment soit inférieur ou égale à la distance donnée: des points seront alors ajoutés

ST_AddPoint	ST_CollectionHomogenize	ST_SetSRID
ST_Affine		ST_SnapToGrid
ST_Force_2D	ST_Multi	ST_Snap
ST_Force_3D	ST_RemovePoint	ST_Transform
ST_Force_3DZ	ST_Reverse	ST_Translate
ST_Force_3DM	ST_Rotate	ST_TransScale
ST_Force_4D	ST_RotateX	
ST_Force_Collection	ST_RotateY	
ST_ForceRHR	ST_RotateZ	
ST_LineMerge	ST_Scale	
ST_CollectionExtract	ST_Segmentize	
	ST_SetPoint	



Sorties en différents formats

- ❖ **ST_Binary** : Retourne la représentation WKB de la géométrie
- ❖ **ST_AsGeoJson** : Retourne la représentation GeoJson de la géométrie
- ❖ **ST_AsGML** : Retourne la représentation GML 2 ou 3 de la géométrie
- ❖ **ST_AsKML** : Retourne la représentation KML de la géométrie, avec contrôle de la sortie (précision, version)
- ❖ **ST_AsSVG** : Retourne la représentation SVG (data path) de la géométrie
- ❖ **ST_AsGeoHash** : Retourne la représentation GeoHash de la géométrie





Sorties en différents formats

ST_AsBinary

ST_AsLatLonText

ST_AsEWKB

ST_AsEWKT

ST_AsGeoJSON

ST_AsGML

ST_AsHEXEWKB

ST_AsKML

ST_AsSVG

ST_AsX3D

ST_GeoHash

ST_AsText



Relations spatiales et mesures

- ❖ **ST_Azimuth**: Retourne l'angle en radian formé par le vecteur composé des deux POINTs
- ❖ **ST_LineCrossingDirection**: Etant données deux LINESTRING, retourne un nombre entre -3 et 3 donnant le type de croisement, 0 voulant dire pas de croisement:
 - ❖ 0: LINE NO CROSS
 - ❖ -1: LINE CROSS LEFT
 - ❖ 1: LINE CROSS RIGHT
 - ❖ -2: LINE MULTICROSS END LEFT
 - ❖ 2: LINE MULTICROSS END RIGHT
 - ❖ -3: LINE MULTICROSS END SAME FIRST LEFT
 - ❖ 3: LINE MULTICROSS END SAME FIRST RIGHT

- **ST_Distance**: Retourne la distance cartésienne minimale entre deux géométries.
- **ST_MaxDistance**: Retourne la distance cartésienne maximale entre deux géométries.
- **ST_DistanceSphere**: Retourne la distance en mètres de deux points situés sur la sphère terrestre (rayon 6370986 m)
- **ST_DistanceSpheroid**: Retourne la distance en mètres de deux points situés sur le sphéroïde donné
- **ST_DWithin**: Retourne vrai si deux géométries sont plus proche que la distance donnée.



- ❖ **ST_OrderingEquals** : Retourne vrai si les deux géométries sont égales et leurs points sont dans le même ordre
- ❖ **ST_Relate** :
 - ❖ Retourne vrai si les deux géométries ont une relation spatiale entre leurs intérieurs, extérieurs, frontières correspondant à la définition données dans la matrice d'intersection.
 - ❖ Si aucune matrice d'intersection n'est passée, retourne la matrice d'intersection maximale qui relie les deux géométries



- ❖ Renvoie la surface en unités carrées (*) d'une géométrie polygonale

- ❖ Prototype:

`ST_Area(geometry geom)`

`SELECT ST_Area(the_geom) FROM ma_table;`

- ❖ Voir aussi:

- ❖ ST_Perimeter

- ❖ Renvoie la géométrie du centroïde d'une géométrie polygonale
- ❖ Prototype:
`ST_Centroid(geometry geom)`

`SELECT ST_Centroid(the_geom) FROM ma_table;`
- ❖ Attention, le centroïde n'est pas toujours situé sur la surface
- ❖ Voir aussi:
- ❖ `ST_PointOnSurface`

- ❖ Renvoie la longueur(*) d'une géométrie linéaire
- ❖ Prototype:
`ST_Length(geometry geom)`

`SELECT ST_Length(the_geom) FROM ma_table;`
- ❖ Voir aussi:
 - ❖ ST_Length_Sphere/spheroid (distance en mètres)
 - ❖ ST_Perimeter

(*) Distance en unité des données, dépendant du système de projection.

- ❖ Renvoie la BBOX **agrégée** d'un ensemble de géométries

- ❖ Prototype:

`ST_Extent(geometry_set geom)`

`SELECT ST_Extent(the_geom) FROM ma_table;`

- ❖ Voir aussi:

- ❖ ST_Box2D et ST_Box3D

- ❖ ST_Expand

- ❖ ST_Enveloppe

- ❖ ST_Estimated_Extent

ST_3DClosestPoint	ST_Crosses	ST_Length_Spheroid
ST_3DDistance	ST_LineCrossingDirection	ST_Length2D_Spheroid
ST_3DDWithin	ST_Disjoint	ST_3DLength_Spheroid
ST_3DDFullyWithin	ST_Distance	ST_LongestLine
ST_3DIntersects	ST_HausdorffDistance	ST_OrderingEquals
ST_3DLongestLine	ST_MaxDistance	ST_Overlaps
ST_3DMaxDistance	ST_Distance_Sphere	ST_Perimeter
ST_3DShortestLine	ST_Distance_Spheroid	ST_Perimeter2D
ST_Area	ST_DFullyWithin	ST_3DPerimeter
ST_Azimuth	ST_DWithin	ST_PointOnSurface
ST_Centroid	ST_Equals	ST_Project
ST_ClosestPoint	ST_HasArc	ST_Relate
ST_Contains	ST_Intersects	ST_RelateMatch
ST_ContainsProperly	ST_Length	ST_ShortestLine
ST_Covers	ST_Length2D	ST_Touches
ST_CoveredBy	ST_3DLength	ST_Within



Fonction de traitement géométrique

- ❖ Renvoie une géométrie généralisée (algo: Douglas Peuker)

- ❖ Prototype:

```
ST_Simplify(geometry geom, integer tolerance)
```

```
SELECT ST_Simplify(the_geom, 100) FROM ma_table;
```

- ❖ Voir aussi:

- ❖ ST_SnapToGrid

- ❖ ST_SimplifyPreserveTopology

```
SELECT ST_Simplify(the_geom, 0.5) FROM pays;
```



- ❖ Renvoie la géométrie résultant de l'intersection entre deux géométries données

- ❖ Prototype:

`ST_Intersection(geometry geom_A, geometry geom_B)`

```
SELECT ST_Intersection(
  (SELECT the_geom FROM ma_table WHERE gid='4'),
  (SELECT the_geom FROM ma_table WHERE gid='2')
);
```

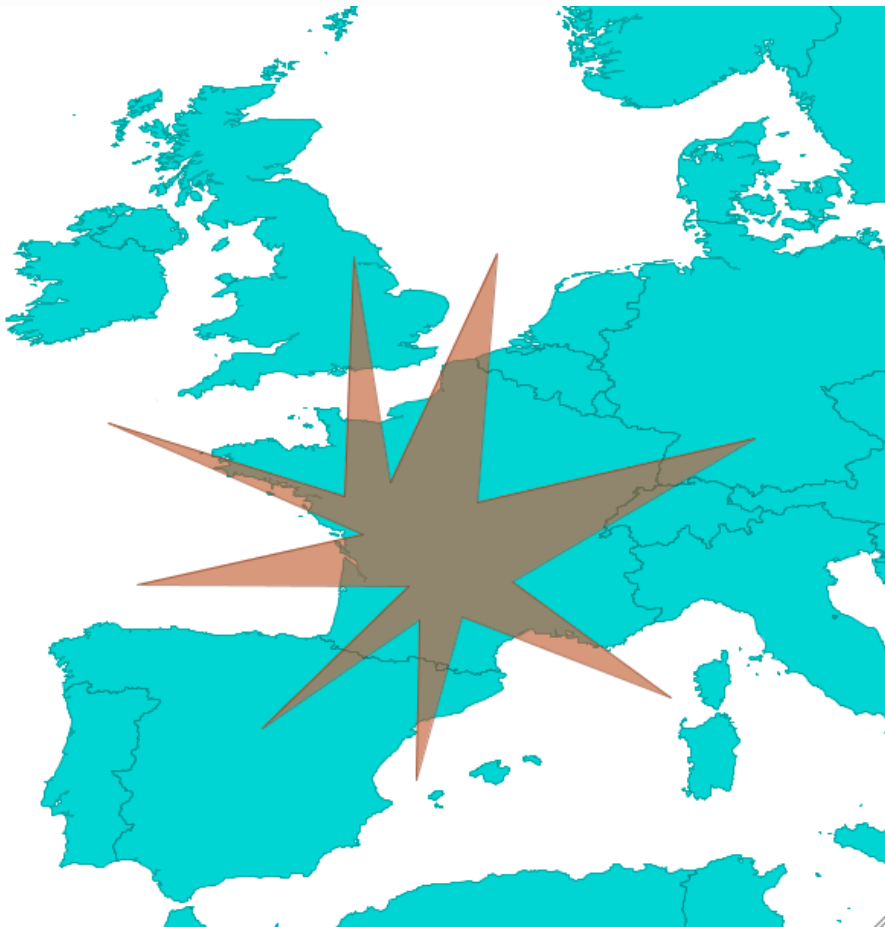
- ❖ Voir aussi:

- ❖ ST_Union

- ❖ ST_Difference, ST_sym_difference

ST_Intersection

```
SELECT ST_intersection(pays.geometry,  
etoiles.geometry) as geom  
FROM pays, etoiles  
where pays.country='France';
```



- ❖ Renvoie une géométrie représentant un buffer de X unités(*) autour d'une géométrie donnée. (unités des données)

- ❖ Prototype:

```
ST_Buffer(geometry geom, double buffer_distance,  
[integer number_of_segments])
```

```
SELECT ST_Buffer(the_geom, 30) FROM ma_table;
```

- ❖ Nota:

- ❖ La valeur du buffer peut être négative (buffer inverse)

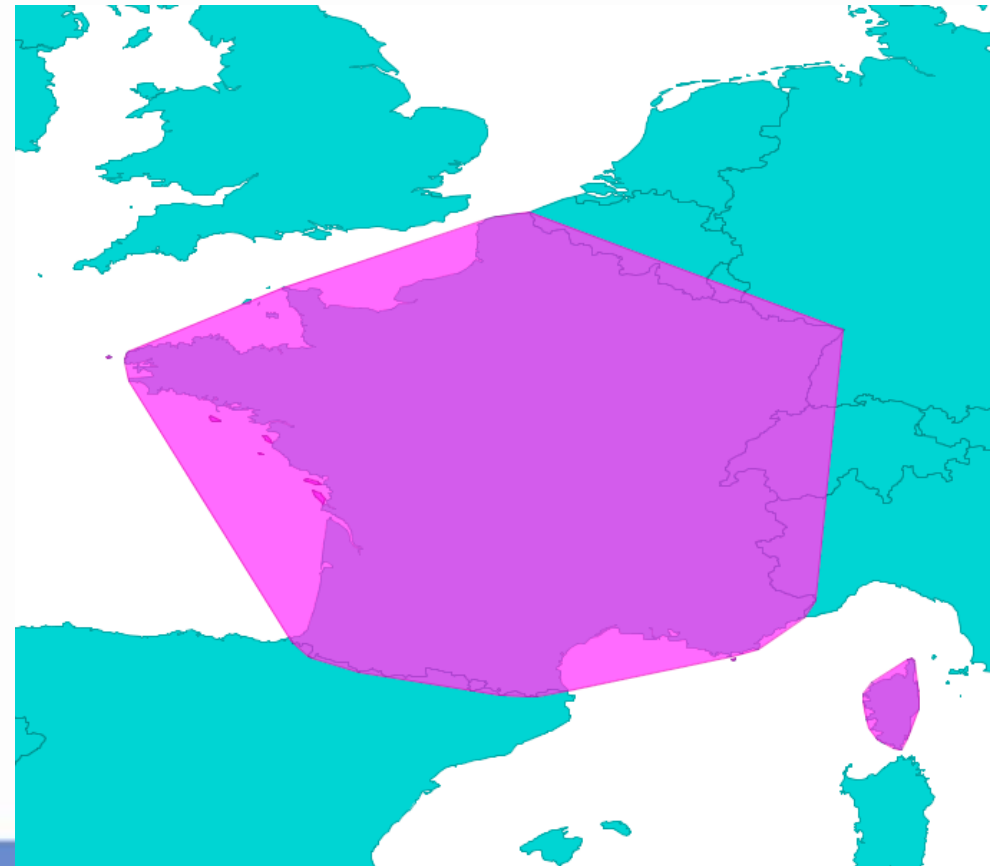
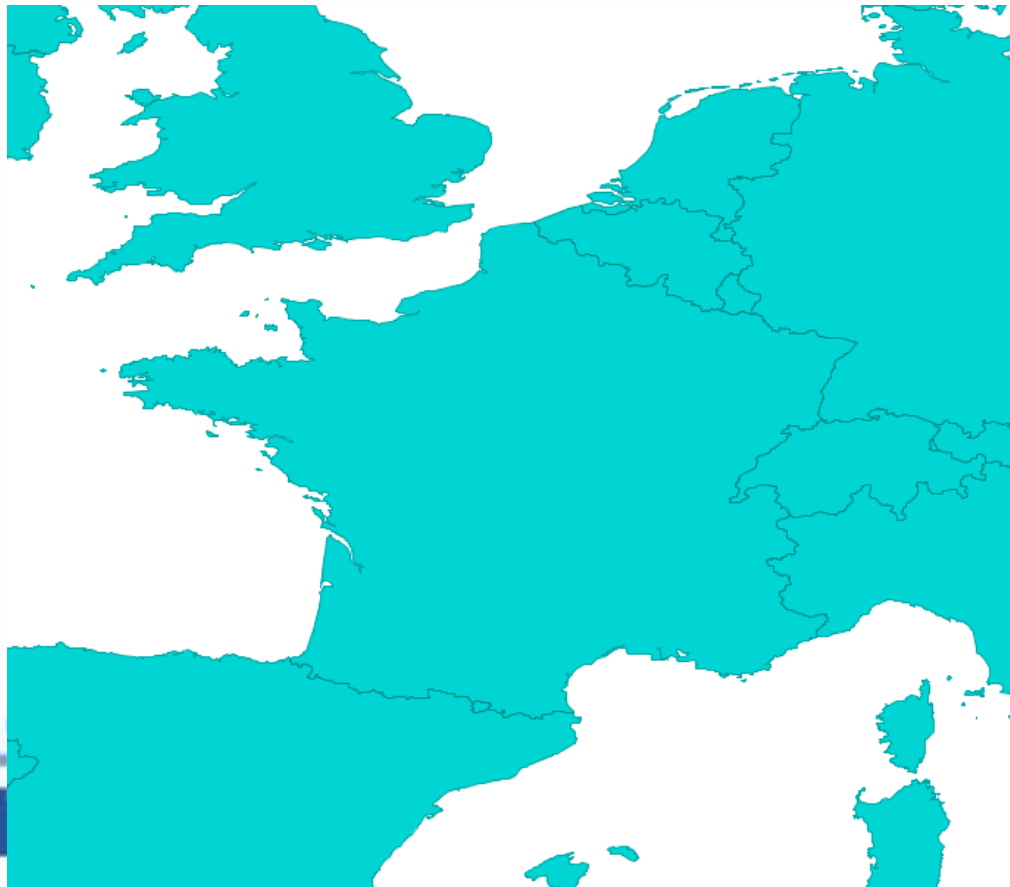
```
SELECT ST_buffer(pays.geometry, 0.1) as geom  
FROM pays  
where pays.country='France';
```



- ❖ **ST_Collect**: Retourne une géométrie de type collection (GeometryCollection ou MULTI*) à partir de la liste de géométries donnée (fonctionne avec une colonne, deux géométries ou un tableau de géométries).
- ❖ **ST_ConvexHull**: Retourne l'enveloppe convexe de la géométries donnée
- ❖ **ST_Dump**: Retourne un ensemble de géométries constituant la géométrie donnée (Collection)
- ❖ **ST_Polygonize**: Retourne une GEOMETRYCOLLECTION contenant des polygones possiblement formés à partir du réseau de lignes constitué par la géométrie donnée.

ConvexHull

```
SELECT ST_convexHull(geometry) as geom  
FROM metro;
```



ST_Buffer	ST_MakeValid	polygon
ST_BuildArea	ST_MemUnion	ST_Split
ST_Collect	ST_MinimumBoundingCircle	ST_SymDifference
ST_ConcaveHull		ST_Union
ST_ConvexHull	ST_Polygonize	ST_UnaryUnion
ST_CurveToLine	ST_Node	
ST_Difference	ST_OffsetCurve	
ST_Dump	ST_RemoveRepeatedPoints	
ST_DumpPoints	ST_SharedPaths	
ST_DumpRings	ST_Shift_Longitude	
ST_FlipCoordinates	ST_Simplify	
ST_Intersection	ST_SimplifyPreserveTo	
ST_LineToCurve		

