**Comparison of different intersections in POSTGIS**

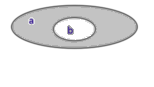
There are multiple methods to do intersection in postgis. Each method can be used by predefined functions in postgis such as ST\_Intersection,ST\_Intersects,ST\_CoveredBy and ST\_Within.Using ST\_Intersection is slower than other functions. If the intersection of two geometries are known and it is not needed to be computed,then the costly and time consuming ST\_Intersection can be skipped.These are the following example cases which the intersection of the geometries are known:

1. Geometry a is covered by geometry b 🡪 intersection is geometry a.



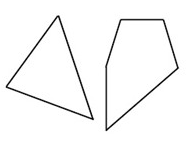
Geometry a is covered by geometry b.

1. Geometry b is covered by geometry a 🡪 intersection is geometry b.



Geometry b is covered by geometry a.

1. Geometry a does not intersect geometry b 🡪 intersection is empty geometry



b

a

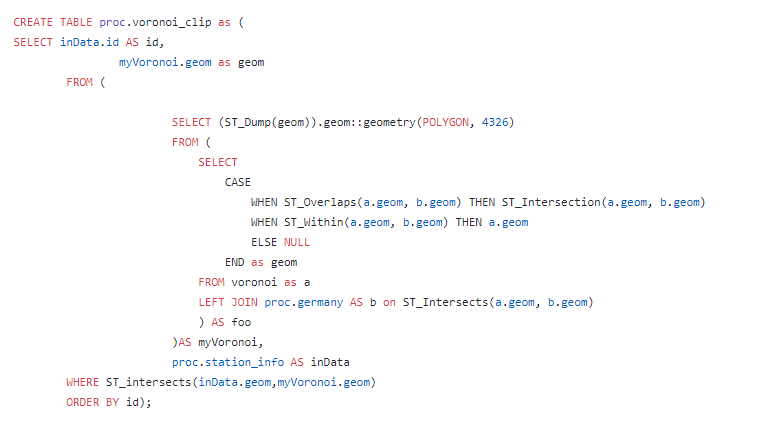
Geometry a does not intersect geometry b.

There can be many applications of these functions.The following example uses ST\_CoveredBy function of postgis instead of ST\_Within because the aim is to extract the portion of each parcel that falls in a neighbourhood. The main difference of ST\_Within and ST\_CoveredBy is that, using ST\_CoveredBy allows geometries to be wholly within the boundary of another and computation speed is faster than ST\_Within. Here are the sample SQL statements which shows the use of the ST\_CoveredBy function.



Using ST\_CoveredBy function and ST\_Intersection.

The second example can be shown in the clipping of the Voronoi which we use in our SQL commands to clip the Voronoi around the Germany boundary. The following SQL codes shows the clipping process.



Using ST\_Overlaps and ST\_Within for clipping the voronois.

By using ST\_Within the voronois which lie completely inside the Germany boundary are selected and the ST\_Intersection function is used only when two geometries (Voronoi and Germany boundary) spatially overlaps.

The difference of speed of intersection can be clarified as Paul Ramsey used in his example.

A test data contains two separated tables. First table has 80 large polygons and another table has 8000 small polygons. Additionally each large polygon contains about 100 small polygons. Without the Preparation of geometry using ST\_Intersects takes about 40 seconds whereas with prepared geometry using functions such as ST\_Intersects(),ST\_Contains(),ST\_Covers() and ST\_ContainsProperly() only takes 8 seconds.

References :

1. <https://postgis.net/2014/03/14/tip_intersection_faster/>

2. <http://blog.cleverelephant.ca/2008/10/postgis-performance-prepared-geometry.html>

3. <https://postgis.net/docs/ST_Overlaps.html>

4. <https://postgis.net/docs/ST_Within.html>

5. <https://postgis.net/docs/ST_CoveredBy.html>