

Tutorial PostGIS, 3.x

Para analistas SIG, geografía y ciencias terrestres...



PostgreSQL



5/15/2024

PRITS

Iván Santiago

isantiago@prits.pr.gov



Tabla de contenido

1. Introducción:.....	6
Sobre PostgreSQL.....	6
PostGIS.....	7
Objetivos de este libro.....	8
Teoría de bases de datos relacionales.....	9
1A. Bases de datos relacionales.....	9
1B: Lenguaje Structured Query Language (SQL):.....	10
1C: SQL Espacial.....	13
Integración con QGIS.....	13
1D: Tipos de datos ¿Cómo se codifican y guardan los datos?	14
Consultas básicas en SQL.....	15
SQL básico.....	15
Queries básicos en pgAdmin 4.....	17
Select Distinct	20
Realizar cálculos.....	21
Uso de Alias para identificar campos/columnas.....	21
Los alias pueden definirse sin usar la palabra AS.	22
Cláusula WHERE	22
Operadores lógicos en la cláusula WHERE	24
Operadores AND, OR	24
Operadores IN, BETWEEN y LIKE.....	25
ORDER BY, LIMIT	27
Offset	29
Orden de precedencia al evaluar condicionalmente	29
GROUP BY: queries con datos agregados	30
Queries simples con geodatos.....	31
Usar Alias para identificar tablas temporalmente.....	32
Orden de operaciones SQL.....	33
Metadatos de las tablas con geometrías.....	34
Hacer un listado de columnas de una tabla.....	35
Modificar datos: DROP, CREATE, INSERT, ALTER, UPDATE	35
DROP.....	35



Tutorial PostGIS, 3.x

CREATE.....	36
INSERT	36
ALTER.....	36
UPDATE.....	37
Prácticas con SQL Básico.....	37
SQL espacial para geometrías vectoriales.....	44
Operaciones espaciales: Adyacencia, continencia, intersección, distancia.....	44
El modelo <i>Simple Features</i> del Open Geospatial Consortium	45
Dimensión de las geometrías	45
Interior, contorno y exterior de las geometrías	46
Predicados para las relaciones topológicas	47
Desglose de relaciones topológicas por geometría	48
Usar QGIS para ejemplos de relaciones topológicas	48
ST_Disjoint.....	50
ST_Touches.....	52
ST_Crosses.....	55
ST_Within/ST_Contains.....	56
ST_Overlaps.....	58
ST_Intersects.....	60
ST_Covers & ST_Coveredby.....	62
ST_DWithin.....	62
Queries con geometrías:.....	63
ST_Touches.....	63
Visualizar las geometrías en PostGIS.....	65
Mostrar área, perímetro y tipo de geometría	66
ST_Length.....	67
Round()	68
ST_Distance.....	69
ST_XMin, ST_YMin.....	70
ST_Transform	71
Group by (ejemplo trivía)	72
Spatial Joins.....	72
Left Join.....	73
Subqueries:.....	75
ST_Disjoint y Subqueries.....	75



Tutorial PostGIS, 3.x

Cross Join Lateral, ST_Distance.....	76
Ordenar por distancias usando el operador <-> en ORDER BY.....	77
Prácticas:.....	79
Consideraciones importantes antes del análisis geográfico y geoprocesos.....	85
Índices espaciales.....	85
Predicados espaciales que usan índice geoespacial por defecto.....	85
Creación de tablas a partir de un query	85
Homogenizar geometrías devueltas al sobreponer geodatos.....	88
Utilización de función extra: STX_Extract	90
Tolerancia en el análisis espacial.....	92
Análisis geoespacial.....	95
Intersección.....	95
Erase	97
Qué es dependencia funcional	98
Unión de layers	100
Identity.....	104
Update.....	108
Clip.....	109
Buffer.....	111
Selección por distancia.....	112
Append.....	113
Dissolve.....	113
Interpolación areal.....	114
Prácticas.....	117
Geodatos ráster en PostGIS	123
Descarga de datos para el ejercicio:.....	126
Apéndice A-1: Inventario de relaciones topológicas	127
Apéndice A-2: Relaciones topológicas clasificadas por predicado	148
Patrón de 9 caracteres para matriz DE-9IM.....	148
ST_Equals.....	149
ST_Disjoint.....	150
ST_Touches.....	152
ST_Crosses.....	158
ST_Within.....	164



Tutorial PostGIS, 3.x

ST_Contains	168
ST_Overlaps.....	170
ST_Intersects.....	173
Operador &&.....	173
ST_Covers	173
ST_CoveredBy.....	177
Referencias.....	181



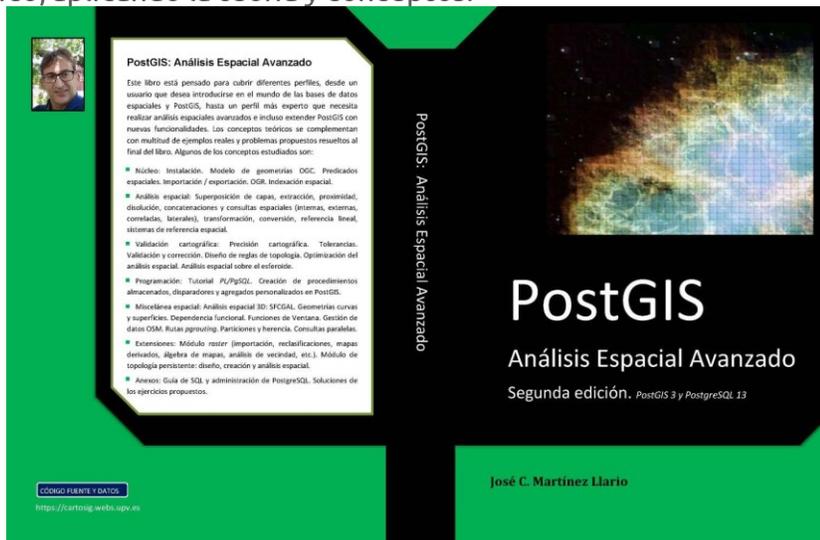
1. Introducción:

Este tutorial está dirigido a personas con experiencia en el uso de programas de procesamiento de datos geográficos (conocidos por Sistemas de Información Geográfica) en su modalidad de interfaces gráficas.

Este tutorial le dará la oportunidad de aprender el uso del lenguaje SQL espacial dentro del gestor de bases de datos (RDBMS) PostgreSQL. Éste, al igual que otros RDBMS, maneja tipos de datos numéricos, de texto y otros tipos de datos especiales para resguardar datos geográficos.

Durante el transcurso de este tutorial, podrá notar las capacidades para resumir y analizar información geográfica de una manera diferente a cualquier otro SIG con interfaz gráfica.

Los ejemplos de geoprocésamiento y teoría cartográfica (conceptos de escala, tolerancias en el geoprocésamiento, y otros) están basados en el libro **PostGIS Análisis Espacial Avanzado 2ª edición, 2020** del profesor **José C. Martínez Llario**, de la **Universidad Politécnica de Valencia**, España. Usamos los ejemplos de código de este libro y los adaptamos usando datos de Puerto Rico, aplicando la teoría y conceptos.



Si desean profundizar sobre PostGIS y PostgreSQL, el libro del Prof. Martínez le dará información detallada sobre otros temas de importancia.

Sobre PostgreSQL

PostgreSQL comenzó a desarrollarse a mediados de los años 80 en la Universidad de California en Berkeley. El origen del nombre proviene de su predecesor, el gestor de bases de datos relacionales "Ingres", el cual fue desarrollándose desde 1977 al 1985. Se decidió ponerle el nombre post-Ingres, quedando finalmente como Postgres en 1986.

Postgres experimentó desarrollos durante 8 años. Luego fue comercializado con el nombre de Illustra. Éste fue adquirido por Informix, que a su vez fue comprado por IBM en 2001.

A partir de un trabajo de investigación académica de 1995, Postgres salió de la academia y empezó una nueva vida en la comunidad open source. Luego de ocho años más, Postgres fue



Tutorial PostGIS, 3.x

transformado radicalmente por la comunidad open source, integrando consistencia y uniformidad al código de base, entre otras cosas.

Postgres fue ganando reputación por su estabilidad. Fue durante este periodo que su nombre cambió a PostgreSQL (*póstgre-skiué*). Este manejador de bases de datos relacionales fue publicado desde la versión 6.0. Hasta hoy, este programa ha continuado su desarrollo por parte de la comunidad open source.

PostGIS

PostGIS surgió a partir de la necesidad de manejar datos geográficos que estaban en archivos separados pero que debían ser modificados frecuentemente. Esta necesidad propició la idea de manejar estos datos geográficos *dentro* de una base de datos de manera *versionada*. Las bases de datos están diseñadas para ambientes compartidos, donde hay dos o más usuarios accediendo a y modificando los mismos datos, pero en versiones diferentes.

La elección de usar el manejador de bases de datos (RDBMS) PostgreSQL fue un golpe de suerte en ese momento (2001), según indican sus desarrolladores iniciales (refrations.net). Este RDBMS tiene la capacidad de crear '*custom data types*', lo cual dejó el camino libre para desarrollar un modelo de almacenamiento de datos geométricos para fines de almacenamiento, manejo y análisis. El documento 'Simple Features for SQL' del OpenGIS Consortium dio las bases para la creación del primer intento de crear un tipo de dato geométrico para GIS.

Durante el transcurso del desarrollo tuvieron que descartar el primer modelo por su lentitud y decidieron crear su propio 'geometry' data type, el cual resultó ser mucho más rápido y eficiente que el primer modelo. Finalmente, se aprovechó la existencia de un tipo de índice R-Tree dentro de PostgreSQL, el cual serviría para acelerar las búsquedas en términos de milisegundos en tablas espaciales con millones de récords.

Para fines de mayo de 2001, se publicó la primera versión de PostGIS 0.1. Desde esa fecha, este producto se ha mantenido dentro de la comunidad open source y al presente va por la versión 2.x, integrando tanto los modelos vectoriales y matriciales de almacenamiento de datos geográficos. Además, desde la versión 2.0 se retomó el asunto de guardar datos vectoriales usando un modelo de topología persistente, diferente al modelo Simple Features del OGC. Además, en la versión 2 se integró el modelo ráster dentro del conjunto de funciones de PostGIS. Esto permite usar funciones analíticas vectoriales y ráster dentro de PostGIS. La versión 3.0 fue publicada a fines de 2019 y se ha estado concentrando en mejoras al desempeño de distintas funciones. La versión 3.x integra nuevas funciones para ráster y adoptar las bibliotecas de programación [GDAL](#).

Fuentes:

<http://www.refrations.net/products/postgis/history/> recuperado 30 nov, 2023

<https://www.postgresql.org/about/> recuperado 30 nov, 2023

<https://carto.com/blog/postgis-3-release> recuperado 30 nov, 2023

<https://www.crunchydata.com/blog/postgis-3.2-new-and-improved> recuperado 30 nov, 2023



Objetivos de este libro

Este tutorial se enfocará en dar algunos ejemplos básicos de consultas **Structured Query Language (SQL) para datos geográficos**. Dedicaremos una parte a mostrarles funciones básicas de SQL no espacial y luego pasaremos a mostrarles cómo aplicar funciones de **SQL espacial**.

- No se tratarán aspectos de administración de bases de datos, resguardo, creación de cuentas, creación de '*schemas*', estadísticas, y otras tareas administrativas. Estas tareas extenderían demasiado este taller.
- El objetivo es dar nociones básicas del lenguaje SQL. Usaremos la interfaz gráfica de administración de bases de datos [DBeaver](#) para redactar y ejecutar las sentencias SQL (queries).
- Trabajaremos funciones SQL para el manejo y análisis de datos geográficos vectoriales en PostGIS.
- En caso de ser necesario, integraremos el uso del programa QGIS para visualización de estos datos.



Teoría de bases de datos relacionales

1A. BASES DE DATOS RELACIONALES

Pregunta funcional: ¿Cuándo una tabla o serie de tablas debe convertirse en una base de datos?

Podemos usar hojas de cálculo e incluso programas de bancos de datos como MS Access y SQLite para manejar datos de cierta complejidad y volumen. Sin embargo, estos programas mencionados no son aptos para ambientes compartidos (en una empresa, agencia, en la Internet), donde dos o más usuarios están accediendo y modificando datos simultáneamente.

Una de las contestaciones a esta pregunta tiene que ver con el propósito de uso de los datos, el volumen, complejidad del manejo de los datos. Otro aspecto clave es determinar si varias personas o usuarios deben acceder a los datos simultáneamente (ambientes compartidos). Si los datos a usarse tienen estas características mencionadas, debe considerar el uso de un programa de manejo de bases de datos (RDBMS en inglés).

Los datos en tablas de hojas de cálculo y en una base de datos, se manejan de forma diferente. En una hoja de cálculo puede haber una o varias tablas, pero estas no necesariamente están relacionadas entre filas o columnas.

En programas de manejo de bases de datos *relacionales*, los datos están separados por temas en distintas tablas. En términos de **teoría de conjuntos y lógica de primer orden, una tabla es una relación** (Viescas, 2018). Estas tablas contienen datos sobre un tema. Esto se hace para evitar redundancia o repetición además de darle organización a los datos. Sin embargo, al estar separadas estas tablas, será necesario el uso de identificadores, los cuales sirven para poder reconocer récords y relacionarlos entre tablas. Esto se hace mediante el mecanismo de identificadores comunes (*primary keys y foreign keys*).

Las personas que han usado programas SIG ya están acostumbrados a enlazar al menos dos tablas usando la función **Join** en ArcGIS o en QGIS, entre otros programas. Esta función **Join** generalmente enlaza una fila (récord) con otra fila en otra tabla. En otros tipos de **Join espacial**, puede relacionarse un identificador con múltiples elementos. En ese caso se producen múltiples instancias del mismo objeto, una para cada fila que contiene información diferente. En otros casos estos programas proveen el uso de relaciones (*Relate*), donde la correspondencia (cardinalidad) entre identificadores ya no es 1 a 1 sino 1 a múltiples.



1B: LENGUAJE STRUCTURED QUERY LANGUAGE (SQL):

Qué es SQL

SQL (Structured Query Language) es un lenguaje de *dominio específico*, esto es, un lenguaje de programación **dedicado a la creación-destrucción, manipulación-actualización y búsqueda de datos** en bases de datos y objetos relacionales.

Orígenes:

El lenguaje SQL está basado en el álgebra de conjuntos. Esta rama de las matemáticas fue desarrollada por [Georg Cantor](#) a lo largo de la última mitad del siglo XIX. En 1968, [el matemático holandés D. L. Childs](#) publicó en la Universidad de Michigan la tesis "[Feasibility of a set-theoretical data structure](#)". En esta tesis, Childs afirma que se puede explicar todo tipo de preguntas solo con **tres funciones básicas: selección, relación y reagrupamiento**. Esta invención dio lugar al lenguaje SQL.

Para la misma época y por distintos caminos, [E.F. Codd](#) inventaba un sistema de almacenamiento de datos más efectivo que el de tipo secuencial indexado y el de tipo jerárquico. El nuevo tipo de base de datos utilizaría las relaciones entre conjuntos para contestar preguntas sobre los datos.

Elementos principales del lenguaje SQL:

El lenguaje SQL está dividido tres sub-lenguajes.

- **Lenguaje de manipulación de datos (Data Manipulation Language o DML)**
Comandos para **devolver, modificar o insertar datos** en las tablas, una vez estos objetos sean creados.
- **Lenguaje para definición de datos (Data Definition Language o DDL)** contiene instrucciones y comandos para **generar o borrar bases de datos, crear/borrar tablas y objetos**.
- **Lenguaje de control de datos (Data Control Language o DCL)**. Estos comandos se usan para el **manejo de cuentas, control de acceso mediante permisos y control de las transacciones**. Ejemplos de estos comandos son Create Role, Grant/Revoke, Commit, Rollback.

Ejemplos de Data Manipulation Language:

SELECT. Este es el más conocido y se usa para extraer datos de una o múltiples tablas.

SELECT * from localidades;

Esta instrucción, sentencia o consulta **devuelve todos los registros (filas) y campos (columnas) de la tabla localidades**.

Podemos reducir la cantidad de datos a devolver estableciendo condiciones, trayendo solo los que nos interesan. Por ejemplo,

```
SELECT nombre FROM localidades -- trae filas del campo nombre de la tabla localidades  
WHERE nombre LIKE 'San%'; -- WHERE se usa para establecer condiciones
```

Esta consulta nos **devuelve solamente los nombres de localidades que comiencen con la secuencia de caracteres "San", seguido de cualquier otra secuencia de caracteres**. Esto puede incluir San Juan, Santa Isabel, etc....



Otro ejemplo puede ser:

```
SELECT * FROM localidades -- trae todas las filas y campos de la tabla localidades  
WHERE población > 1000; -- donde la población sea menor de 1,000
```

En esta consulta, **se devolverán los datos de la tabla localidades con todas las columnas donde la población supere los mil habitantes.**

INSERT

Este comando se usa para añadir registros a las tablas ya existentes. Por ejemplo:

```
INSERT INTO localidades -- inserta filas en la tabla existente localidades  
(barrio, población) -- en los campos barrio y población  
VALUES ('Santolaya', 1500); -- los valores 'Santolaya' y 1,000
```

Al insertar datos dejando columnas sin datos, estos adoptarán el valor "NULL". Esto quiere decir que esta celda no tiene valor, no tiene dato. Esto tendrá consecuencias más adelante al tratar de hacer operaciones matemáticas, por ejemplo. No tendrán valor "NULL" aquellos campos con tipo de dato numérico *serial*, que se llenan automáticamente al insertar una fila.

UPDATE

Este comando se usa para **modificar el contenido de las tablas.**

```
UPDATE localidades -- actualiza la tabla existente localidades  
SET nombre = 'Santa Olaya' -- escribe 'Santa Olaya' en el campo nombre  
WHERE nombre = 'Santolaya'; -- en aquellas filas donde el nombre sea 'Santolaya'
```

En este ejemplo, estamos cambiando el nombre 'Santolaya' por el nombre 'Santa Olaya'.

Además, se puede **recalcular** sobre registros existentes

```
UPDATE localidades -- actualiza la tabla existente localidades  
SET población = población + 300 -- sumar 300 a la población existente  
WHERE nombre = 'Santa Olaya'; -- donde el nombre sea 'Santa Olaya'
```

Aquí se está sumando 300 a la población existente solo para la localidad 'Santa Olaya'.

DELETE

Se usa para **borrar registros** de una tabla.

```
DELETE FROM menú -- borra filas de la tabla existente localidades  
WHERE nombre_plato = 'Arroz con guinea' -- donde el nombre del plato sea 'Arroz con guinea'
```

Aquí en esta sentencia, borraremos el registro o registros de la tabla menú que contengan la cadena de caracteres 'Arroz con guinea' en la columna nombre_plato. Note que **las cadenas de caracteres van encerradas en comillas simples.**

Ejemplos de Data Definition Language:

```
CREATE DATABASE base_de_datos -- crear una base de datos
```

```
CREATE TABLE localidades -- crear tabla localidades
```

```
(gid serial primary key, nombre varchar, poblacion integer); -- con los campos:
```

```
/* (gid: tipo de dato serial, con identificador primario, nombre: tipo de dato varchar o de texto y  
de longitud variable, poblacion: tipo de dato entero) las columnas van encerradas entre  
paréntesis ().*/
```

```
-- Los símbolos /* y */ son para hacer comentarios multilinea en PostgreSQL.
```

Crear la tabla llamada 'localidades' con una serie de columnas o atributos con diferentes tipos de dato numéricos (integer) o de texto (varchar).



Tutorial PostGIS, 3.x

ALTER TABLE localidades ADD COLUMN geom geometry (point, 6566);
Añadir la columna 'geom', en la tabla existente *localidades*, la cual guardará las coordenadas de los lugares registrados. Estas se registrarán usando **geometría puntual**, en el **sistema de referencia espacial** con código **6566** (State Plane Coordinate System, Puerto Rico, NAD83(2011)).

USE poblacion_2010;
Este se usa para indicar **cuál es la base de datos** a consultar o hacer modificaciones.

DROP TABLE barrios_exclusivos
Este comando sirve para **borrar tablas** y objetos de la base de datos.

DROP DATABASE base_de_datos
También puede usarse para **borrar** la **base de datos**.

Ejemplos de Data Control Language (DCL)

Crear una cuenta o usuario nuevo en una base de datos dentro de un grupo o rol ya establecido:

```
CREATE USER estudiante1 WITH PASSWORD 'estudiante1';
```

Dar acceso a conectarse a una base de datos:

```
GRANT CONNECT ON DATABASE gisdb TO estudiante1;
```

Dar privilegio de entrada/uso dentro de un 'schema'

```
GRANT USAGE IN SCHEMA pr_geodata TO estudiante1;
```

Dar privilegio de lectura a un usuario, y a una tabla:

```
GRANT SELECT ON g01_biota_bosques_auxiliares ;
```

Más adelante estaremos explicando otras funciones importantes de SQL tales como las funciones de agrego de datos, unión de tablas, y algunas de las funciones geoespaciales.

Referencias:

Fundamentos de SQL (conciso): <https://www.thoughtco.com/sql-fundamentals-1019780> recuperado en 14 febrero de 2024.

Historia

Breve historia (orígenes-fundamentos matemáticos) del lenguaje SQL:
http://www.decideo.fr/Petite-histoire-des-bases-relationnelles-et-du-langage-SQL_a6377.html recuperado en 14 febrero 2024.

Breve historia: Hernández, M., Viescas J.L, *SQL Queries for mere mortals*, 2nd Ed, Sep. 2007, pp. 53-65.

Genealogía de los RDBMS: <http://fadace.developpez.com/sbgdcmp/story/> recuperado el 14 febrero 2024.



Tutorial PostGIS, 3.x

1C: SQL ESPACIAL

Orígenes

PostGIS fue publicado por la empresa consultora informática canadiense [Refractions Research](http://www.refrations.net/) en el año 2001. En ese momento solo contaba con varias funciones de manejo de datos geométricos (geometry data type). La publicación de un estándar abierto de este tipo de datos geométricos en 1998 hizo que el trabajo de mejoras y actualizaciones de PostGIS pudiera continuar hasta estos días.

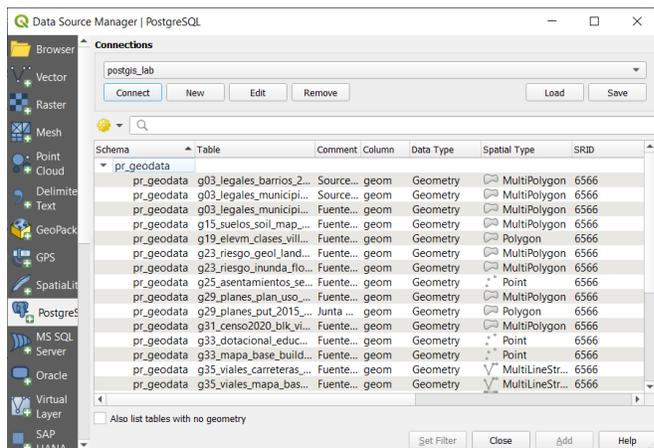
Referencias:

Martínez Llario, José C., [PostGIS 2 Análisis Espacial Avanzado](http://cartosig.upv.es/), 2012-13. [http://cartosig.upv.es.](http://cartosig.upv.es/) Universidad Politécnica de Valencia, España.

<http://www.refrations.net/>, recuperado en diciembre 1, 2023

INTEGRACIÓN CON QGIS

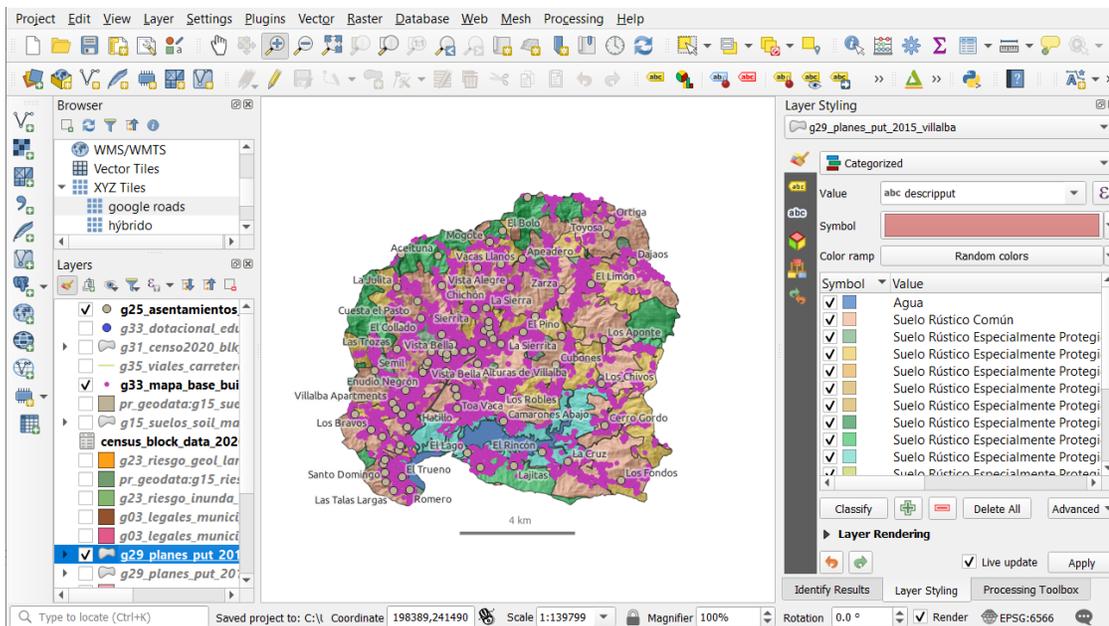
QuantumGIS (QGIS) será el programado que utilizaremos para ver algunos de los resultados de las consultas SQL que vamos a realizar en PostGIS. De hecho, QGIS surgió por la necesidad de tener un visualizador de los datos geoespaciales guardados en PostGIS. Si no tiene el programa QGIS instalado, descárguelo gratis desde la dirección <http://qgis.org/en/site/forusers/download.html>. Descargue e instale, según su sistema operativo: Windows 32bit o 64bit, Mac, o las distintas variedades de Linux.



Ventana para añadir tablas geoespaciales (layers) al visor de geodatos de QGIS



Tutorial PostGIS, 3.x



Visualización en QGIS: Algunos de los layers guardados en PostgreSQL/PostGIS.

1D: TIPOS DE DATOS ¿CÓMO SE CODIFICAN Y GUARDAN LOS DATOS?

La tabla de atributos se compone principalmente de tres [tipos de datos](#), dos de ellos son los más comunes:

- **Texto** (*character, string*): letras, palabras, frases, oraciones, códigos **alfanuméricos**, identificadores.
No se usan para operaciones matemáticas. Generalmente se manipulan con funciones de texto como concatenaciones, extracción, etc. Puede usarse ordenamiento (*sorting*).
- **Cifras, números** enteros, decimales, binarios, fechas.
En estos es común el ordenamiento y operaciones matemáticas.
- **Objetos**, (datos en formato que solo puede interpretar la computadora mediante instrucciones)
Ciertas bases de datos pueden guardar las coordenadas de un punto, línea, área, celda(s) en un campo de una tabla. Usualmente se usa el tipo de dato numérico **“binario”** para guardarlos.

Sea prudente al momento de escoger un tipo de representación numérica.

- Evite usar números con decimales cuando sepa de antemano que todos los números del conjunto de datos son enteros.
- Use el menor espacio posible para los atributos de texto. Si va a guardar un código que no pasa de tres espacios, no use el espacio por defecto de algunos programas (50 espacios)

Al hacer esto, se economiza espacio en disco y el rendimiento del programa se mantendrá óptimo.



Consultas básicas en SQL

SQL BÁSICO

En esta parte vamos a practicar sentencias SQL simples. Un enunciado SQL básico de selección de datos está compuesto de:

SELECT *columna1, columna2, columna3,... columna_n, o un asterisco * para todas las columnas y récords.*

FROM *para indicar la tabla o tablas que vamos a usar*

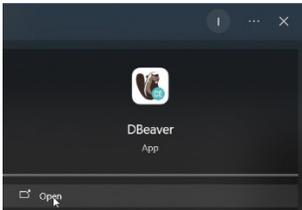
; *punto y coma para cerrar la sentencia SQL*

Ejemplo para practicar:

- Abra una sesión del programa **DBeaver**, vaya a la caja de texto **Search** de Windows y escriba **dbeaver**



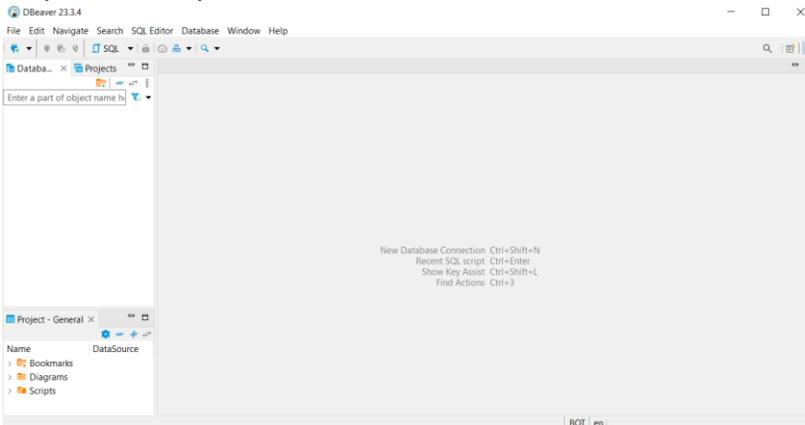
- Abra la aplicación haciendo **click** en la opción **Open** de **pgAdmin 4**.



- Espere que el *splash screen* termine de hacer la carga y aparezca la aplicación.



Así debe aparecer la aplicación DBeaver



- Haga **click** en el **triángulo negro** del botón **New Database Connection (Ctrl+Shift+N)**



- Escoja (click en) la opción **PostgreSQL**



Tutorial PostGIS, 3.x

- En la forma **Connect to a database**, siga la secuencia con los siguientes parámetros:

1: Escriba la dirección donde se encuentra la base de datos. Si está tomando el adiestramiento de forma presencial, se le dirá cuál es la dirección. Si está siguiendo el libro desde su domicilio o trabajo, la dirección debe ser **localhost**.

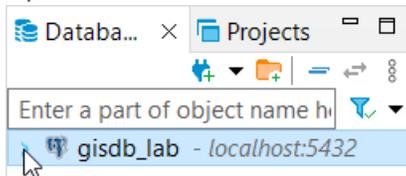
2: La base de datos es **gisdb_lab**.

3y4: Escriba el nombre del usuario de la base de datos. Si está tomando el adiestramiento de forma presencial, se le indicará el **nombre de usuario** y contraseña. **Si usted está siguiendo el libro desde su domicilio o trabajo**, la dirección (**host**) debe ser **localhost**

5: Haga **click** en el botón **Test Connection** para asegurarse que se puede conectar a la base de datos. Si todo está bien, aparecerá la forma **Connection test** y solo necesitará hacer **click** en el botón **OK**. De lo contrario, revise los parámetros antes mencionados.

6: Haga **click** en el botón **Finish** en la forma **Connections Settings** para cerrarla.

- Para los que no estén de forma presencial en el adiestramiento, es su decisión guardar o no el password en la interfaz de conexión de DBeaver.
- Deberá estar seguro de cuál es el password del usuario postgres al momento que instaló el PostgreSQL y PostGIS.
- En la lista arborescente del panel **Database Navigator**, haga **click** en el **nodo** al lado izquierdo del nombre de la base de datos **gisdb_lab**.



Si no guardó el password de usuario, aparecerá una forma para que escriba el password.

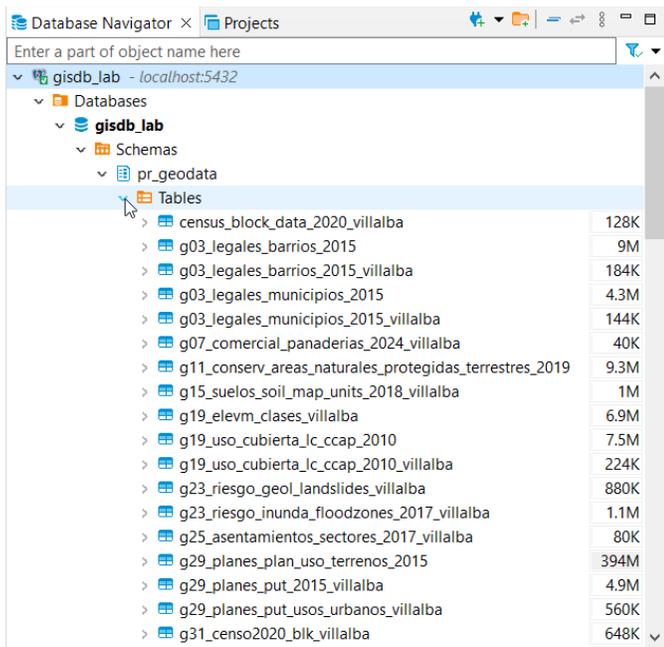


Tutorial PostGIS, 3.x

- ❑ **Expanda** el nodo **Databases** y luego **expanda** el nodo **gisdb_lab**



- ❑ Continuando, **expanda** el nodo **Schemas**, luego el nodo **pr_geodata** y finalmente el nodo **Tables**.

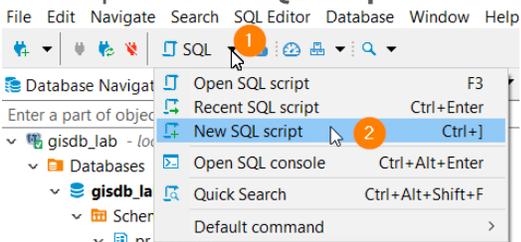


Al lado derecho de cada **tabla o "relación"** como se les conocen también a las tablas, verá el espacio en disco que ocupa cada tabla.

Las tablas que vamos a usar están bajo el nodo **Schemas > pr_geodata > Tables**

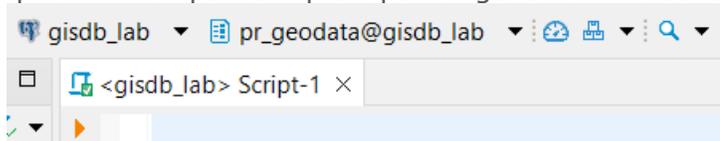
QUERIES BÁSICOS EN PGADMIN 4

- ❑ Una vez esté disponible la base de datos para estas prácticas, en la interfaz gráfica de **DBeaver** deberá hacer **click en el triángulo negro** del botón **SQL** y luego **click** en la opción **New SQL script**



De **igual** efecto será ir al **menú principal**, hacer **click** en **SQL Editor** y escoger la opción **New SQL Script**.

- ❑ Fíjese que en la parte superior de la ventana **<gisdb_lab> Script-1** aparece el nombre de la base de datos activa (**gisdb_lab**), así como el nombre del Schema activo (**pr_geodata**). **Eso es importante**. De lo contrario, estará ubicado en otro Schema y tendrá que escribir el path completo para llegar a la tabla cuando vaya a escribir los queries.





Tutorial PostGIS, 3.x

Vamos a realizar la primera **búsqueda simple** (query) para comenzar a interactuar con una de las tablas de esta base de datos. Primero, vamos a mostrar el diccionario de datos de la tabla **census_block_data_2020_villalba**.

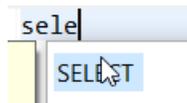
Esta es la **estructura y descripción** de las columnas de esta tabla (no es geodato).

Columna	Tipo de dato	Descripción
gid	integer	Identificador (ID) interno
logrecno	integer	número lógico de récord (ID censal)
sumlev	character varying	Summary level. Nivel geográfico (bloque, grupo de bloque, tract, barrio, subbarrio, municipio, PR, USA)
geoids	character varying	ID compuesto para bloque censal 2020
cousub	character varying	ID para barrios
barrio	character varying	Nombre del barrio
tract	character varying	ID Tract, sector censal
blkgrp	character varying	ID Block group, grupo de bloques censales
block	character varying	ID Bloque censal
name	character varying	Nombre del bloque censal
vtd	character varying	Voting District: Unidad electoral, de la Comisión Estatal de Elecciones. Los límites no necesariamente concuerdan con el mapa 2020 de la CEE.
pop100	integer	Conteo 100% de población, 2020
hu100	integer	Conteo 100% de viviendas, 2020
h0010002	integer	Viviendas ocupadas, 2020
h0010003	integer	Viviendas vacantes, 2020

DBEaver facilita la escritura de queries, ya que al escribir se presentan opciones para completar la cláusula, función, tabla, etc.

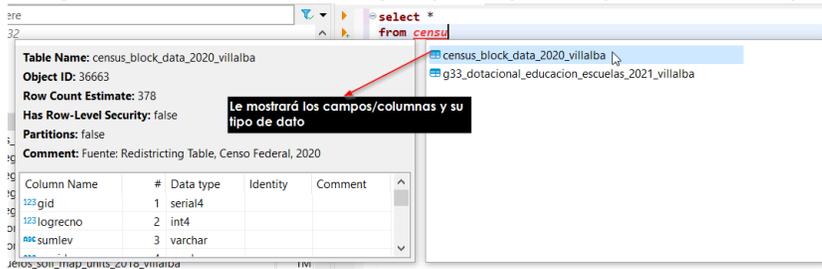
- En la pestaña **<gisdb_lab Script-1**, escriba la siguiente sentencia o query:

sele



use la tecla tab para que le ayude a completar la palabra SELECT

- Escriba ***** y luego presione la tecla **enter**
- Luego escriba **FROM censu** y espere que le aparezcan las opciones de tablas



- Use la tecla tab** para **escoger** la tabla **census_block_data_2020_villalba**. Fijese que el DBEaver le otorgó un **alias** a esta tabla **"cbdv"** para facilitar otras operaciones.



Tutorial PostGIS, 3.x

Termine la sentencia escribiendo ; al final.

```
*<gisdb_lab> Script-1 ×  
  
select *  
from census_block_data_2020_villalba cbdv;
```

- Haga **click** en el botón **Execute SQL Query** o presione las teclas **ctrl+enter** para ejecutar esta sentencia/query

```
*<gisdb_lab> Script-1 ×  
  
select *  
from census_block_data_2020_villalba cbdv;  
Execute SQL query (Ctrl+Enter)
```

Podrá ver los resultados de esta forma en la interfaz gráfica de DBeaver:

gid	logrecono	sumlev	geoids
1	1	89,707 750	721497201001000
2	2	89,708 750	721497201001001
3	3	89,709 750	721497201001002
4	4	89,710 750	721497201001003
5	5	89,711 750	721497201001004
6	6	89,712 750	721497201001005
7	7	89,713 750	721497201001006
8	8	89,714 750	721497201001007
9	9	89,715 750	721497201001008
10	10	89,716 750	721497201001009
11	11	89,717 750	721497201001010
12	12	89,718 750	721497201001011
13	13	89,719 750	721497201001012
14	14	89,720 750	721497201001013
15	15	89,721 750	721497201001014
16	16	89,722 750	721497201001015
17	17	89,723 750	721497201001016

- Bajo la pestaña **census_block_data_2020_villalba 1**, use las barras de navegación hacia los lados y de arriba y abajo para explorar este resultado

Este query que acabamos de escribir y ejecutar quiere decir que pedimos a **DBeaver** que nos dé **todas** las columnas y **todos** récords/filas de la tabla **census_block_data_2020_villalba**. Esta tabla contiene datos básicos de población y vivienda para el Censo de 2020, además de una serie de campos de niveles geográficos.

- Haga **click** en el botón **Calculate total row count** inmediatamente debajo de los resultados, y verá que **la tabla contiene 378 récords** (Total rows).

Refresh Save Cancel
200 378 Connected to 'gisdb_lab'
Calculate total row count Sel: 0 | 0



¿Qué quiere decir SELECT *?

Un detalle importante que aclarar desde el principio es que **el uso de SELECT * aunque simple, no es recomendado**. Esta **instrucción trae todos los datos a través de la red, y si el contenido de una o más tablas es grande (miles o millones de récords), esto va a afectar el desempeño de la búsqueda**. Imagínese, traer esa cantidad de datos, multiplicado por el número de usuarios de una base de datos.

Ya que se mostró el contenido de la tabla `census_block_data_2020_villalba`, haremos una serie de búsquedas usando esta tabla.

SELECT DISTINCT

Como pudimos ver en los resultados de la búsqueda pasada, hay columnas con valores repetidos. Por ejemplo, los ID y nombres de barrios están repetidos porque la tabla tiene **datos a nivel de bloque censal**. Por lo tanto, **los ID (columna: `geoids`) de bloques no deben estar repetidos**. Podemos usar la opción “**DISTINCT**” para que nos devuelva una lista única de barrios y sus identificadores. Use la tabla de diccionario de datos de la tabla `census_block_data_2020_villalba` en la página anterior.

¿Cuál es el nombre de la columna de nombres de barrios? _____
¿Cuál es el nombre de la columna de ID de barrios? _____

- Vaya a la caja de texto del **Query Script** y **borre el contenido del query escrito anteriormente**.
- Ya que vamos a traer más de una columna, debemos **separar cada columna usando una coma**. Escriba lo siguiente:
SELECT DISTINCT `cousub`, `barrio`
FROM `census_block_data_2020_villalba`;
-- Los comentarios en las líneas de código son para su beneficio. No los tiene que copiar.
- Presione **ctrl+enter** o haga **click** en el botón **Execute**.

```
select distinct cousub, barrio -- trae datos sin repetir de columnas cousub y barrio
from census_block_data_2020_villalba; -- de esta tabla
```

	<code>cousub</code>	<code>barrio</code>
1	33898	Hato Puerco Arriba
2	86917	Villalba Abajo
3	33855	Hato Puerco Abajo
4	86874	Villalba CUT
5	13430	Caonillas Abajo
6	86960	Villalba Arriba
7	13516	Caonillas Arriba
8	84595	Vacas

8 row(s) fetched - 0.006s, on 2024-02-13 at 16:40:12

Aquí podemos ver que hay **ocho barrios** en el **Municipio de Villalba**. Los nombres de barrios responden a **referencias históricas** del uso de territorios, asentamientos, flora o fauna.

Los barrios **Hato Puerco** y **Vacas** fueron lugares de crianza de **ganado porcino** y de **ganado vacuno** en siglos pasados. En el caso del nombre



Tutorial PostGIS, 3.x

Caonillas, este nombre tiene origen taíno y está relacionado a la búsqueda de **oro**, ya que ese nombre está relacionado a lugares donde se buscaba oro en cuerpos de agua. **Villalba CUT** es el **centro urbano tradicional**. La **fundación** de este **municipio** es relativamente reciente, a principios del siglo XX, el **12 de abril de 1917**.

REALIZAR CÁLCULOS

El lenguaje SQL tiene múltiples funciones estándar para operaciones matemáticas básicas como suma, resta, multiplicación, división, exponenciación, y funciones estadísticas básicas, como promedio y otras. Para más detalles sobre las funciones y requisitos para su uso, vaya al siguiente enlace: <https://www.postgresql.org/docs/current/functions-math.html>. Más adelante discutiremos algunas funciones matemáticas agregadas.

Vamos a realizar una **sumatoria de población y vivienda** para todos los bloques censales del Municipio de Villalba.

- En la **caja de texto** de **Query Script**, **borre** el query anterior y **escriba** lo siguiente:

```
SELECT SUM(pop100), SUM(hu100)
FROM census_block_data_2020_villalba;
```

	cousub	barrio
1	33898	Hato Puerto Arriba
2	86917	Villalba Abajo
3	33855	Hato Puerto Abajo
4	86874	Villalba CUT
5	13430	Caonillas Abajo
6	86960	Villalba Arriba
7	13516	Caonillas Arriba
8	84595	Vacas

Observe que **solo se devuelve una fila**. Además notará que los nombres de las columnas tienen el mismo nombre.

Esto no es recomendado, así que en el próximo ejercicio vamos a darles un nombre a las columnas calculadas.

USO DE ALIAS PARA IDENTIFICAR CAMPOS/COLUMNAS

Como se mencionó anteriormente, las columnas del query anterior tiene el mismo nombre. Otras veces, queremos presentar nombres más descriptivos, según sea el caso.

- NO borre** el query anterior. Solo vamos a cambiarlo un poco.
- Modifique el query anterior. **Añada** las palabras **AS población2020** luego de SUM(POP100). **Añada** las palabras **AS viviendas2020** luego de SUM(HU100)



Tutorial PostGIS, 3.x

```

* <gisdb_lab> Script-1 x
select sum(pop100) as población2020,
sum(hu100) as viviendas2020
from census_block_data_2020_villalba;

```

Grid	123 población2020	123 viviendas2020
1	22,093	8,875

No debe dejar espacios entre *población 2020* ni entre *vivienda 2020* porque DBeaver devolverá un mensaje de error.

```

* <gisdb_lab> Script-1 x
select sum(pop100) as población 2020,
sum(hu100) as viviendas 2020
from census_block_data_2020_villalba;

```

SQL Error [42601]: ERROR: syntax error at or near "2020" Position: 33

Esto es como definir otro alias sin referencia a columna alguna.

Si quiere dejar un espacio o quiere usar mayúscula inicial, deberá encerrar entre comillas el alias. "Población 2020". Lo mismo aplicará a "Vivienda 2020", como aparece aquí a la derecha. Puede modificar el query anterior para que practique. **Esta práctica no es recomendable para dar nombre a las columnas de una tabla permanente.** De lo contrario, tendrá que utilizar comillas "Columna etc." cada vez que vaya a referirse a una columna con espacios y mayúsculas.

```

* <gisdb_lab> Script-1 x
select sum(pop100) as "población 2020",
sum(hu100) as "viviendas 2020"
from census_block_data_2020_villalba;

```

Grid	123 población 2020	123 viviendas 2020
1	22,093	8,875

LOS ALIAS PUEDEN DEFINIRSE SIN USAR LA PALABRA AS.

- Practique con el mismo query, esta vez borre las palabras AS:

```

SELECT SUM(POP100) población2020,
SUM(HU100) vivienda2020
FROM census_block_data_2020_villalba;

```

```

* <gisdb_lab> Script-1 x
select sum(pop100) población2020,
sum(hu100) vivienda2020
from census_block_data_2020_villalba;

```

Grid	123 población2020	123 vivienda2020
1	22,093	8,875

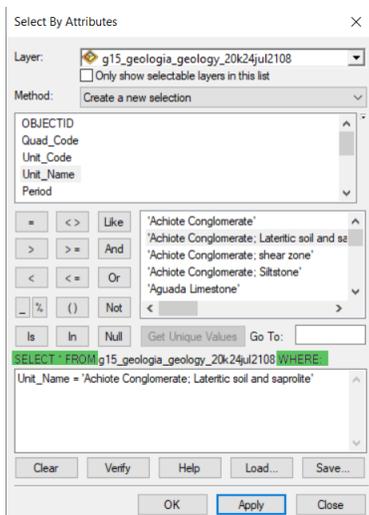
Note también que **la columna del query final NO requiere escribir la coma.** Tampoco afecta el query si luego de cada coma hay un "enter" o CR/LF (carriage return/line feed).

CLÁUSULA WHERE

Mediante la cláusula **WHERE** es que comenzaremos a escoger **subconjuntos** de datos en una o más tablas. Esta es la cláusula tradicionalmente "visible" en los programas SIG con interfaz gráfica.



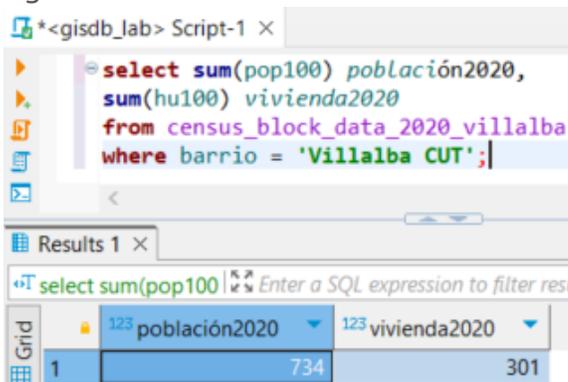
Tutorial PostGIS, 3.x



Este es un ejemplo de la función **Select by Attribute** de ArcMap (Esri). Más abajo en esa forma, resalto en color verde el query y la fracción **WHERE**, la cual es la que se ejecuta con esta función. Para unir tablas, será hacer un join aparte en otra función.

Hasta ahora, hemos visto **SELECT columna1, columna2, col3... FROM tabla**. A esto vamos a añadir la cláusula **WHERE** para establecer condiciones para extraer subconjuntos de datos.

- Vaya a la pestaña **SQL Editor <gisdb_lab> Script-1** y modifique el query de la siguiente manera:



Este query ejecuta una suma de todos los bloques censales que componen el barrio pueblo o la zona del centro urbano tradicional/histórico identificados como 'Villalba CUT'. El resultado es 734 personas y 301 viviendas.

- La última columna: vivienda2020 no debe tener coma final. De lo contrario devolverá un error.



OPERADORES LÓGICOS EN LA CLÁUSULA WHERE

A continuación presentamos una lista de operadores para usar dentro de la cláusula WHERE.

Operador	Descripción
=	Igualdad
>	Mayor que
<	Menor que
>=	Mayor o igual
<=	Menor o igual
<> o !=	Desigualdad
AND	Operador lógico AND (excluyente)
OR	Operador lógico OR (incluyente)
<u>IN</u>	Devuelve TRUE si el valor para con cualquier valor en una lista.
<u>BETWEEN</u>	Devuelve TRUE si el valor está entre un conjunto o entre una gama de valores
<u>LIKE</u>	Devuelve TRUE si un valor para con un patrón, usualmente de caracteres
<u>IS NULL</u>	Devuelve TRUE si el valor es NULL (ausencia de datos)
NOT	Negar el resultado de otras operaciones

Practicemos un par de ejemplos con WHERE.

Pregunta: ¿Cuáles son los bloques censales que no tienen población? Añada los barrios para dar una idea somera de dónde están.

- Vaya a la pestaña **SQL Editor <gisdb_lab> Script-1** y escriba lo siguiente:

```

select name as bloque, barrio, pop100 -- trae columnas name alias bloque, barrio y pop100
from census_block_data_2020_villalba -- de esta tabla
where pop100 = 0; -- condición: filas con valor de población (pop100)=0

```

Este query devuelve **46 filas** que corresponden a bloques censales **que no tenían población para el Censo 2020**, además de los **barrios** donde estos bloques están ubicados.

OPERADORES AND, OR

Este operador puede confundirse con nuestra manera de entender “y” si pensamos literalmente en su traducción. No es adición. **AND** significa buscar **cuáles elementos cumplen simultáneamente dos o más condiciones**. Por ejemplo, hay un solo día en un año



Tutorial PostGIS, 3.x

que es 12 y a la vez está en el mes de diciembre. Se puede añadir el año y *restringimos* aún más nuestra búsqueda.

Pregunta: ¿Cuáles son los bloques censales que tienen población ≥ 0 y que están dentro del barrio Caonillas Arriba?

- Vaya a la pestaña **SQL Editor <gisdb_lab> Script-1** y escriba lo siguiente:

```

select name as bloque, barrio, pop100 -- trae columnas name alias bloque, barrio y pop100
from census_block_data_2020_villalba -- de esta tabla
where pop100 >= 0 -- condiciones: filas con valor de población (pop100)=0 ...
and barrio = 'Caonillas Arriba'; -- Y que el barrio sea 'Caonillas Arriba'

```

Grid	bloque	barrio	pop100
1	Block 2000	Caonillas Arriba	61
2	Block 2001	Caonillas Arriba	26
3	Block 2002	Caonillas Arriba	43
4	Block 2004	Caonillas Arriba	141
5	Block 2005	Caonillas Arriba	16
6	Block 2006	Caonillas Arriba	25
7	Block 2007	Caonillas Arriba	102
8	Block 2008	Caonillas Arriba	25
9	Block 2009	Caonillas Arriba	171

Este query nos devuelve los bloques censales que tienen población cero o más y que a la vez pertenecen al barrio Caonillas Arriba. En total son 30 filas.

Estos resultados pueden ser mostrados en orden ascendente o descendente usando la cláusula **ORDER BY**, ya sea usando la modalidad ascendente **ASC** o descendente **DESC**. Esto lo veremos más adelante.

Pregunta: ¿Cuál sería el resultado del query anterior si cambiamos el operador AND por el operador OR?

Averigüemos:

- En la pestaña **SQL Editor <gisdb_lab> Script-1**, MODIFIQUE el query anterior

```

select name as bloque, barrio, pop100
from census_block_data_2020_villalba
where pop100 >= 0
OR barrio = 'Caonillas Arriba';

```

Grid	bloque	barrio	pop100
1	Block 1000	Hato Puerco Arriba	34
2	Block 1001	Hato Puerco Arriba	101
3	Block 1002	Hato Puerco Arriba	12
4	Block 1003	Hato Puerco Arriba	97
5	Block 1004	Vacas	183
6	Block 1005	Vacas	34
7	Block 1006	Vacas	60
8	Block 1007	Vacas	0
9	Block 1008	Hato Puerco Arriba	88
10	Block 1009	Hato Puerco Arriba	23
11	Block 1010	Hato Puerco Arriba	109
12	Block 1011	Hato Puerco Arriba	74
13	Block 1012	Hato Puerco Arriba	14

¿Por qué este query nos devuelve todas las filas de la tabla? Esto pasa porque las condiciones:

Pop100 ≥ 0 va a dar TRUE en todas las filas.

Barrio = 'Caonillas Arriba' dará TRUE en algunos casos.

Ya que todos los récords tienen algún valor desde cero, el query va a devolver todas las filas. La condición OR barrio='Caonillas Arriba' no surte efecto alguno porque la condición de población ≥ 0 hace TRUE a todo el conjunto de datos de la tabla.

Podemos ponerle el nombre de cualquier barrio, incluso otro nombre que no está y devolverá todas las filas.

OPERADORES IN, BETWEEN Y LIKE

El operador IN se usa para buscar filas que tengan valores que pareen dentro de una lista definida en el query.

Hagamos un ejemplo:



Tutorial PostGIS, 3.x

- En la pestaña **SQL Editor <gisdb_lab> Script-1**, MODIFIQUE el query anterior

```
select name as bloque, barrio, pop100
from census_block_data_2020_villalba
where barrio in
('Caonillas Abajo', 'Caonillas Arriba', 'Villalba CUT');
```

	bloque	barrio	pop100
1	Block 2000	Caonillas Arriba	61
2	Block 2001	Caonillas Arriba	26
3	Block 2002	Caonillas Arriba	43
4	Block 2004	Caonillas Arriba	141
5	Block 2005	Caonillas Arriba	16
6	Block 2006	Caonillas Arriba	25
7	Block 2007	Caonillas Arriba	102
8	Block 2008	Caonillas Arriba	25
9	Block 2009	Caonillas Arriba	121
10	Block 2010	Caonillas Arriba	82
11	Block 2011	Caonillas Arriba	65
12	Block 3000	Caonillas Arriba	47

El operador **IN** está separado por un CRLF (enter) pero esto no es necesario. Lo escribí de esta manera para que se pueda ver en este espacio.

El resultado de este query nos devuelve **78 filas que corresponden a todos los bloques que cumplen la condiciones de pertenecer a los barrios Caonillas Arriba o Caonillas Abajo, o a Villalba CUT**. Es un atajo para evitar escribir `barrio = 'Caonillas Arriba' OR barrio = 'Caonillas Abajo' OR barrio = 'Villalba CUT'`.

El operador **BETWEEN** se usa para devolver filas que cumplan estar dentro de una gama de valores **numéricos, fechas o de texto**.

Tarea: Provea una lista de bloques censales con población entre 100 a 200 habitantes para el censo 2020 en el Municipio de Villalba.

- En la pestaña **SQL Editor <gisdb_lab> Script-1**, modifique y ejecute el query anterior

```
select name as bloque, barrio, pop100
from census_block_data_2020_villalba
where pop100 between 100 and 220;
```

	bloque	barrio	pop100
1	Block 1001	Hato Puerco Arriba	101
2	Block 1004	Vacas	183
3	Block 1010	Hato Puerco Arriba	109
4	Block 2004	Caonillas Arriba	141
5	Block 2007	Caonillas Arriba	102
6	Block 2009	Caonillas Arriba	121
7	Block 3003	Caonillas Arriba	112
8	Block 3023	Hato Puerco Abajo	133
9	Block 4000	Caonillas Abajo	130
10	Block 4001	Caonillas Abajo	170
11	Block 4008	Caonillas Abajo	182
12	Block 4010	Hato Puerco Abajo	217
13	Block 4020	Caonillas Abajo	122

Note que **no usamos OR** dentro del operador **BETWEEN**.

En este query el resultado devuelto son los bloques censales que tienen población dentro de las cantidades 100 a 220 habitantes. Se devolverán 53 filas.

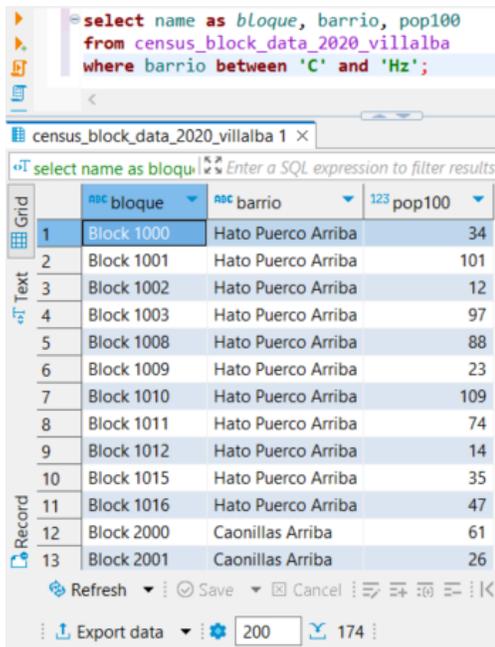
El operador **BETWEEN** puede usarse **con datos alfanuméricos**, por ejemplo:

Tarea: Hacer una lista de barrios cuyos nombres comiencen con la letra C hasta la H.

- En la pestaña **SQL Editor <gisdb_lab> Script-1**, modifique y ejecute el query anterior



Tutorial PostGIS, 3.x

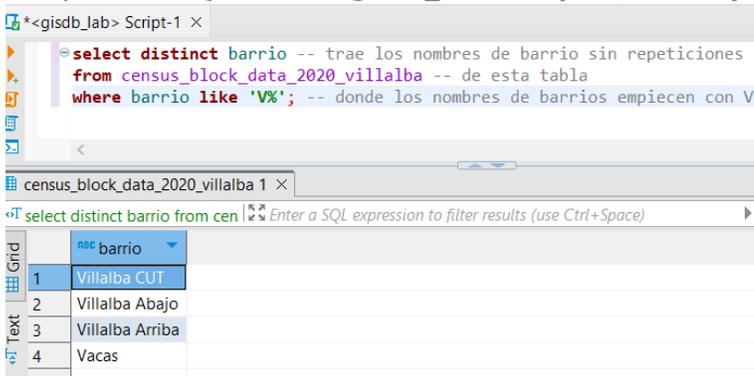


Este query nos devuelve las filas que tengan **nombres de barrios que comiencen con la letra C** hasta los barrios cuyos nombres comiencen con **H**.

¿Por qué **H**? Esto hará que se incluyan todos los barrios con nombres comiencen con la letra H, seguido del resto de letras, hasta la combinación Hz (que no está presente, pero es para asegurar incluir las demás letras después de la H).

El operador **LIKE** se usa **para buscar un patrón específico de caracteres**. Por ejemplo, hacer una lista de barrios cuyo nombre comience con V. Use la cláusula **DISTINCT** para traer solamente los nombres de los barrios.

- En la pestaña **SQL Editor <gisdb_lab> Script-1**, modifique y ejecute el query anterior



Este query nos devuelve una **lista de barrios cuyo nombre comienza con la letra V**. Podemos modificar el patrón y escribir LIKE 'Villa%' para seleccionar solamente los barrios con nombres que comiencen con el patrón 'Villa%' en su nombre.

Es necesario incluir el signo de porcentaje % para incluir el resto de los caracteres después de 'Villa...' de lo contrario, el query no traerá filas porque ningún barrio tiene como nombre V.

Note el uso de la cláusula **DISTINCT** para traer solo una instancia del nombre del barrio. Esta tabla es de bloques censales, pero contiene barrios también.

ILIKE: El operador **ILIKE** hace lo mismo que LIKE, pero a diferencia de éste, **ILIKE no es case sensitive**.

ORDER BY, LIMIT

La cláusula **ORDER BY** tiene como propósito **ordenar las filas de forma ascendente o descendente en una o más columnas**. Veamos un ejemplo.

- En la pestaña **<gisdb_lab> Script 1**, escriba y ejecute el siguiente query:



Tutorial PostGIS, 3.x

```
*<gisdb_lab> Script-1 ×
select name as bloque, barrio, pop100 -- trae los nombres de bloque, barrio y población
from census_block_data_2020_villalba -- de la tabla census_block_data_2020_villalba
where pop100 between 100 and 200 -- donde la población sea entre 100 y 200
order by pop100 asc , barrio asc; -- ordena primero por pop100 asc, luego por barrio asc
```

Se puede especificar el uso de **ASC** para que esté claro que **el orden es ascendente**. Sin embargo, la opción por defecto es **ASC** y no es necesario escribirla. Por el contrario, si deseamos que el orden sea **descendente**, debemos escribir **DESC**. Además, podemos usar **más de una columna para ordenar**. En este caso las dos columnas que estamos usando se ordenarán de manera ascendente.

Grid	ABC bloque	ABC barrio	123 pop100
1	Block 3004	Villalba Abajo	100
2	Block 1001	Hato Puerco Arriba	101
3	Block 1001	Hato Puerco Arriba	101
4	Block 2007	Caonillas Arriba	102
5	Block 1010	Villalba Arriba	102
6	Block 1001	Villalba Arriba	103
7	Block 3020	Villalba Abajo	104
8	Block 2002	Vacas	105
9	Block 2004	Hato Puerco Arriba	106

Refresh Save Cancel Export data 47 row(s) fetched - 0.001s, on 2024-02-14 at 10:38:47

Por su parte, la cláusula **LIMIT** se usa para limitar el número de filas que el query va a devolver. Podemos usarla para ordenar y presentar los primeros o últimos casos de una serie de datos. Veamos un ejemplo:

Tarea: Haga una **lista de bloques censales con mayor población; incluya el nombre del bloque, barrio y población. Ordene la lista usando la columna pop100 de manera descendente. Limite las filas solo a 5.**

- En la pestaña **SQL Editor <gisdb_lab> Script-1 modifique y ejecute** este query de la siguiente manera:

```
*<gisdb_lab> Script-1 ×
select name as bloque, barrio, pop100 -- trae los nombres de bloque, barrio y población
from census_block_data_2020_villalba -- de la tabla census_block_data_2020_villalba
where pop100 between 100 and 200 -- donde la población sea entre 100 y 200
order by pop100 desc , barrio asc -- ordena primero por pop100 desc, luego por barrio desc
limit 5; -- limita el número de filas devueltas a 5
```

Este query nos trae los primeros cinco bloques censales con mayor población en el Municipio de Villalba para el Censo de 2020.

Grid	ABC bloque	ABC barrio	123 pop100
1	Block 1007	Vacas	200
2	Block 1006	Villalba Arriba	199
3	Block 1019	Villalba Arriba	193
4	Block 1001	Vacas	186
5	Block 1015	Villalba Arriba	185

Refresh Save Cancel Export data 5 row(s) fetched - 0.001s, on 2024-02-14 at 10:41:23



Tutorial PostGIS, 3.x

OFFSET

Offset sirve para obviar filas y traer otras. Usualmente se usa en combinación con LIMIT. Sirve para **mostrar la cantidad de filas siguientes, especificada en OFFSET** en una **lista ordenada** ascendente o descendente. Por ejemplo:

Pregunta: **¿Cuáles son los siguientes bloques censales con mayor población, luego de los primeros 10?**

- En la pestaña **SQL Editor <gisdb_lab> Script-1** modifique y ejecute este query de la siguiente manera:

```

select name as bloque, barrio, pop100 -- trae los nombres de bloque, barrio y población
from census_block_data_2020_villalba -- de la tabla census_block_data_2020_villalba
where pop100 between 100 and 200 -- donde la población sea entre 100 y 200
order by pop100 desc , barrio asc -- ordena primero por pop100 desc, luego por barrio desc
limit 10 -- limita el número de filas devueltas a 10
offset 10; -- muestra las siguientes 10 filas después de la 10ma

```

	bloque	barrio	pop100
1	Block 4001	Caonillas Abajo	170
2	Block 1009	Vacas	169
3	Block 4004	Hato Puerco Arriba	163
4	Block 2012	Hato Puerco Arriba	153
5	Block 3003	Hato Puerco Arriba	149
6	Block 2008	Hato Puerco Arriba	147
7	Block 4022	Caonillas Abajo	143

Este query nos devuelve los siguientes 10 bloques censales con mayor población después de los primeros 10. Es como tener la lista de los primeros 20 pero solo mostrar los últimos 10.

ORDEN DE PRECEDENCIA AL EVALUAR CONDICIONALMENTE

Por defecto, los programas RDBMS evalúan las condiciones (expuestas en la cláusula WHERE) de **izquierda a derecha**. Cuando una condición de búsqueda contiene varios tipos de condiciones simples el RDBMS los evalúa en orden a base del operador utilizado en cada condición (Viescas, 2018).

Orden de evaluación	Tipo de operador
1ro	() evaluar lo que esté dentro del paréntesis
2do	Signo positivo (+), signo negativo (-)
3ro	Multiplicación (*), división (/)
4to	Suma (+), resta (-)
5to	=, <>, <, >, <=, >=, BETWEEN, IN, LIKE, IS NULL
6to	NOT
7mo	AND
8vo	OR

Fuente: Viescas, 2018. SQL For Mere Mortals.



GROUP BY: QUERIES CON DATOS AGREGADOS

Ya que hemos visto distintas maneras de filtrar filas y columnas en una tabla, ahora pasaremos a discutir cómo hacemos queries para **hacer cálculos agregados (conteos, sumas, promedios, y otros)**.

Por ejemplo, tenemos esta tabla de bloques censales, la cual tiene varios niveles geográficos (summary levels). Podemos **agregar o agrupar** datos en niveles geográficos más altos. Podemos **sumar la población de todos los bloques que componen cada barrio o cada unidad electoral** o una selección de barrios o unidades electorales. Para esto se usa **GROUP BY**.

Pregunta/Tarea: Use la tabla `census_block_data_2020_villalba` para calcular la **población por barrio**. La lista debe tener el **nombre del barrio** y la **población**. **Ordene** la lista **por población** de manera **descendente (mayor a menor)**.

- En la pestaña **SQL Editor <gisdb_lab> Script-1** modifique y ejecute este query de la siguiente manera:

```
*<gisdb_lab> Script-1 x
select barrio, -- trae los nombres de barrio, use coma , para separar columnas
sum(pop100) as población -- conteo bloques, alias "cantidad_bloques"
from census_block_data_2020_villalba -- de la tabla census_block_data_2020_villalba
group by barrio -- agrupa las sumas de población por barrio
order by población desc; -- ordena primero por sum(pop100) o alias "población" desc
```

Grd	barrio	población
1	Hato Puerco Arriba	6,304
2	Villalba Arriba	5,062
3	Vacas	3,473
4	Villalba Abajo	2,524
5	Hato Puerco Abajo	1,459
6	Caonillas Arriba	1,288
7	Caonillas Abajo	1,249
8	Villalba CUT	734

Este query nos devuelve una lista con los barrios y su población para el Censo 2020. La cláusula **GROUP BY** sirve para **agregar** (en este caso, **sumar**) la **población de bloques que componen cada barrio**. Al lado de **GROUP BY** se escribe cuál es (son) la(s) **columna(s)** que tiene(n) la **dimensión** o **nivel** en el cual **queremos agregar los datos**. En este caso, agregar población solo por barrio. Note que **SUM(pop100)** **no se añade al GROUP BY**. **SUM(pop100)** **no es un nivel de agregación** sino un grupo de **valores que vamos a computar**. No necesariamente tiene que ser de tipo de dato numérico. Puede ser conteos.

8 row(s) fetched - 0.001s, on 2024-02-14 at 11:44:21

Pregunta/Tarea: ¿Cuántos bloques censales hay por cada barrio?

Para esta tarea no usaremos sumas, sino **conteos de bloques** censales para cada barrio. La función por usar es **COUNT()** o **COUNT(*)**. Esto nos devolverá el número de filas con una más características que hemos de escoger. No hay bloques censales entre barrios.



Tutorial PostGIS, 3.x

- En la pestaña **<gisdb_lab> Script 1**, escriba y ejecute el siguiente query:

```

select barrio, -- trae los nombres de barrio, use coma , para separar columnas
count(geoids) as cantidad_bloques -- conteo bloques, alias "cantidad_bloques"
from census_block_data_2020_villalba -- de la tabla census_block_data_2020_villalba
group by barrio -- agrupa las sumas de población por barrio
order by cantidad_bloques desc; -- ordena primero por count(geoids) o alias desc

```

Grid	barrio	cantidad_bloques
1	Hato Puerco Arriba	87
2	Villalba Arriba	85
3	Vacas	49
4	Villalba Abajo	49
5	Hato Puerco Abajo	30
6	Caonillas Arriba	30
7	Caonillas Abajo	27
8	Villalba CUT	21

En este query estamos usando la función **COUNT()** usando la columna **geoids** para que nos devuelva el número de bloques censales que componen cada barrio en el Municipio de Villalba. El campo **geoids** es el campo que tiene los **identificadores de cada bloque censal**. Además necesitamos la cláusula **GROUP BY** para agrupar conteos por barrio y **ORDER BY** para ordenar por cantidad de bloques de manera descendente.

QUERIES SIMPLES CON GEODATOS

Las preguntas a continuación se contestan en diferentes queries que utilizarán la función **ST_Area(geometry)** en PostGIS.

Pregunta 1: ¿Cuáles son los 10 municipios de mayor área?

- En la pestaña **<gisdb_lab> Script 1**, escriba y ejecute el siguiente query:

```

select municipio, -- trae los nombres de municipios
st_area(geom)/1000000 as sqkm -- trae el área municipal/1,000,000 conv en kms cuadrados
from g03_legales_municipios_2015 -- de la tabla g03_legales_municipios_2015
order by sqkm desc -- ordena por sqkm descendente
limit 10; -- reduce la cantidad de filas devueltas a 10

```

Grid	municipio	sqkm
1	Arecibo	328.5324788907
2	Ponce	302.7296063593
3	Utua	297.7907433289
4	Coamo	202.0604296195
5	Mayagüez	201.6116075629
6	Cabo Rojo	186.7987971071
7	San Sebastián	184.5929428018
8	Salinas	181.3948914822

En este query lo único nuevo es el uso de la función **ST_Area(geom)**. Esta función devuelve el área superficial de una geometría plana en las unidades en que están expresadas en el sistema de coordenadas del geodato (**metros**, en este caso). El resultado de la función **ST_Area(geom)** está dividido por el factor de conversión **1000000** (un millón), para convertir **metros cuadrados** a **kilómetros cuadrados**.

El geodato **g03_legales_municipios_2015** contiene una **columna de geometría (geom)**. Estas columnas de geometría son las que usaremos para hacer las tareas de geoprocésamiento más adelante. Estamos usando **ORDER BY** para ordenar las superficies y **LIMIT 10** para reducir la lista.

Reto: Con las cláusulas ya aprendidas hasta ahora:

¿Cómo podemos modificar este query y mostrar los 10 municipios de menor superficie? Hint: order by ...

Pregunta 2: ¿Cuáles son las áreas ocupadas por cada zona de riesgo por deslizamiento en Villalba?



Tutorial PostGIS, 3.x

- En la pestaña **<gisdb_lab> Script 1**, escriba y ejecute el siguiente query:

```
select description as riesgo_deslizamiento, --trae los nombres de las zonas de riesgo
sum(st_area(geom))/1000000 as sqkm --trae el área/1,000,000 conv en kms cuadrados
from g23_riesgo_geol_landslides_villalba -- de la tabla g23_riesgo_geol_landslides_villalba
group by description -- agrupa datos usando el campo description
order by sqkm desc; -- ordena por nuevo campo ó alias "sqkm" descendente
```

	riesgo_deslizamiento	sqkm
1	Moderate susceptibility to landsliding	55.3720627432
2	High susceptibility to landsliding	32.133139331
3	Highest susceptibility to landsliding	5.7400948243
4	Low susceptibility to landsliding	2.6305040314

En este query **estamos aplicando y practicando las funciones y cláusulas aprendidas hasta ahora:**
SUM(ST_Area(geom)): para obtener las áreas de cada geometría (poligonal) y sumarlas/agregarlas **GROUP BY** por **tipo de riesgo "description"** dentro de Villalba.

Estamos usando **ORDER BY** para ordenar por *sqkm*.

Reto:

El área total del municipio es de **95.876** kms cuadrados.

¿Cómo definirías otra columna que calcule el porcentaje de ocupación para cada zona de riesgo?

USAR ALIAS PARA IDENTIFICAR TABLAS TEMPORALMENTE

Como habrá notado, puede ser tedioso nombrar tablas cada vez se va a invocar su uso. Cuando se añaden más tablas entonces puede ser necesario escribir el nombre de la columna y el nombre de la tabla, seguido de un punto. Por ejemplo: gid.census_block_data2020_villalba, sum(pop100.census_block_data_2020_villalba). Como vio en la sección de uso de alias para columnas, además de columnas también se puede usar alias para identificar **temporalmente** tablas. Esto es útil cuando utilizamos más de una tabla en un query.

Pregunta 3: ¿Cuántas edificaciones hay en cada barrio del Municipio de Villalba?

Esta pregunta se debe contestar combinando **dos geodatos**: el de **barrios** y el geodato de **centroides de edificios**.



Tutorial PostGIS, 3.x

- En la pestaña **<gisdb_lab> Script 1**, escriba y ejecute el siguiente query:

```

select b.barrio, count(e.gid) as num_edifs -- trae los barrios, conteo de edificios
from g03_legales_barrios_2015_villalba as b, -- de la tabla, alias "b"
g33_mapa_base_building_centroids_1998_villalba as e -- de la tabla, alias "e"
where st_intersects(b.geom,e.geom) -- predicado topológico: donde intersecan geoms
group by barrio -- agrupa por el campo barrio, tabla "b"
order by num_edifs desc; -- ordena por num_edificios, descendente

```

barrio	num_edifs
Hato Puercio Arriba	3,705
Villalba Arriba	2,598
Vacas	1,748
Villalba Abajo	1,510
Caonillas Arriba	1,055
Caonillas Abajo	833

Alias en tablas:
 Note el alias "e" usado para nombrar este geodato. Al igual que los campos, **podemos usar alias** para nombrar temporalmente las tablas. De esa manera nombramos la tabla de **barrios** como **b** y la tabla de **edificios** como **e**.

Predicado Topológico: **st_intersects(b.geom, e.geom)** puesto en cláusula **WHERE** como **condición: intersección geométrica** entre el geodato/tabla de barrios y la de centroides de edificios. El predicado **st_intersects()** es **conmutativo**, así que el orden de las tablas no importa.

** Para **contar** los **edificios** usaremos la función **count()** y el campo para hacer el conteo es **gid** de la tabla/geodato **g33_mapa_base_building_centroids**.

** Si escribe **count(e.gid)** o **count(*)**, dan el mismo resultado. Inténtelo. **Count(*)** hace **conteo de filas** y eso es lo que buscamos.

¿Por qué no usamos la función SUM()? Vea también que **estamos agregando conteos por barrio**, así que usamos la cláusula **GROUP BY** y el campo **barrio**.

¿Por qué no añadimos el campo num_edifs en la cláusula GROUP BY?

ORDEN DE OPERACIONES SQL

Para un mejor desempeño en la búsqueda de datos, es importante conocer sobre el orden de las operaciones en SQL. Lo primero que se evalúa es la fuente de datos. Por lo tanto, lo primero que se evalúa en una sentencia (query) SQL es la cláusula FROM y JOIN porque son las que se usan para hacer referencia a las tablas/objetos de la base de datos. El orden se recogerá en esta tabla.

ORDEN	CLÁUSULA	FUNCIÓN
1	FROM/JOIN	Escoger y/o unir tablas para obtener los datos.
2	WHERE	Filtros que se van a aplicar a los datos que están en las tablas de interés.
3	GROUP BY	Agregar los datos si fuere necesario.
4	HAVING	Filtro adicional a los datos agregados por GROUP BY.
5	SELECT	Devuelve los datos finales.
6	ORDER BY	Ordena los datos finales.
7	LIMIT	Limita o reduce el número de filas al número determinado en esta cláusula.

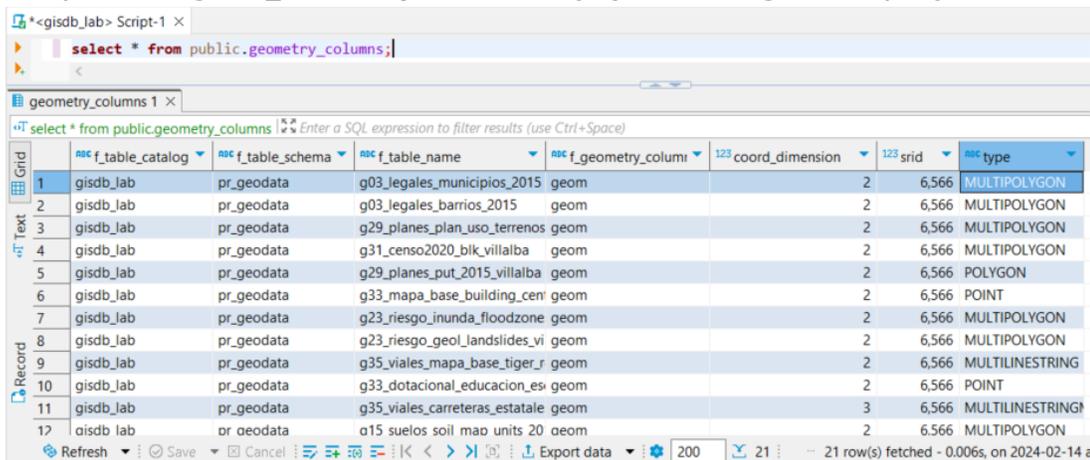


Tutorial PostGIS, 3.x

METADATOS DE LAS TABLAS CON GEOMETRÍAS

Información importante (metadatos) sobre los geodatos en tablas y las columnas (campos) PostGIS guarda un tipo de **tabla virtual “View”**, la cual es actualizada con frecuencia. Esta tabla virtual llamada **“geometry_columns”** contiene información sobre los geodatos. Para ver esta tabla virtual (view):

- En la pestaña **<gisdb_lab> Script 1**, escriba y ejecute el siguiente query:



Estas son las tablas que tienen una columna que contenga algún tipo de geometría.

Columnas:

Columna/Campo	Descripción
type	tipos de geometría
srid	código numérico del sistema de referencia espacial . En esta base de datos, los geodatos están registrados en el sistema de coordenadas planas estatales con proyección cartográfica Cónica Conforme de Lambert, Zona 5200 , unidades en metros y datum NAD83 versión/realización de 2011 . Ese sistema tiene el código EPSG:6566 .
f_geometry_column	nombre de la columna que guarda las geometrías. Se le asignó el nombre arbitrario “geom” a cada una de ellas.
f_table_schema	es nombre del schema (especie de folder) donde están los geodatos. Su creación es arbitraria. Nombre: pr_geodata .
f_table_catalog	contiene el nombre de la base de datos .



HACER UN LISTADO DE COLUMNAS DE UNA TABLA

En ocasiones, es conveniente saber cuáles son las columnas que componen una tabla:

- En la pestaña **<gisdb_lab> Script 1**, escriba y ejecute el siguiente query:

The screenshot shows a SQL query window with the following text:

```
select ordinal_position as colnumber, --trae columna ordinal_position, alias colnumber
column_name, data_type --trae columnas column_name y data type. No use coma al final
from information_schema.columns --del 'view' llamado 'information_shema.columns'
where table_schema = 'pr_geodata' --condiciones: valor del campo table_schema = 'pr_geodata'
and table_name = 'census_block_data_2020_villalba'; --and la tabla= census_block_data_2020_villalba
```

Below the query, a table view displays the results:

Grid	colnumber	column_name	data_type
1	1	gid	integer
2	2	logreco	integer
3	3	sumlev	character varying
4	4	geoids	character varying
5	5	cousub	character varying
6	6	barrio	character varying
7	7	tract	character varying
8	8	blkgrp	character varying
9	9	block	character varying
10	10	name	character varying
11	11	vtd	character varying
12	12	pop100	integer
13	13	hu100	integer
14	14	h0010002	integer
15	15	h0010003	integer

A text box on the right side of the screenshot contains the following explanation:

En este query solamente escogimos las columnas **ordinal_position**, **column_name** y **data_type** del **view** **information_schema.columns**. Un **view** es un tipo de tabla virtual. La **tabla de interés** es **census_block_data_2020_villalba** y el **schema** es **pr_geodata**.

Modificar datos: DROP, CREATE, INSERT, ALTER, UPDATE

En esta parte veremos otros comandos SQL para borrar tablas, crear tablas, insertar filas en una tabla, cambiar nombres de columnas y actualizar filas de una tabla.

DROP

Se utiliza para borrar bases de datos, tablas, views, columnas de tablas y otros objetos en una base de datos.

Ejemplos:

DROP DATABASE nombre_base_de_datos;	Elimina una base de datos. Solo es posible por medio de usuarios con privilegios administrativos.
DROP TABLE nombre_tabla;	Borrar tabla.
DROP VIEW nombre_view;	Borrar el view o tabla virtual.
ALTER TABLE tabla DROP COLUMN columna;	Escoger la tabla primero, luego DROP COLUMN columna para eliminarla.



Tutorial PostGIS, 3.x

CREATE

Comandos para generar bases de datos, tablas, tablas copiando otra tabla, índices, views, procedures (segmento o script SQL reusable).

Ejemplos:

CREATE DATABASE nombre_base_de_datos;	Crear una base de datos. Solo es posible por medio de usuarios con privilegios administrativos.
CREATE TABLE nombre_tabla (gid SERIAL, columna 1, columna 2); *	Generar una tabla con un campo "gid" numérico entero, secuencial y dos columnas.
CREATE TABLE tabla_nueva AS TABLE tabla_existente;	Crear una tabla idéntica a otra tabla existente.
CREATE TABLE tabla_copia AS SELECT columna1, columna2 FROM tabla fuente;	Crear una tabla de dos columnas sin filtro usando una tabla existente.

* <https://www.postgresqltutorial.com/postgresql-tutorial/postgresql-serial/>

INSERT

Se usa para insertar filas en una tabla, siguiendo el orden de las columnas. Se usa con la palabra INTO. Ejemplo

INSERT INTO tabla(columna1, columna2, etc.) VALUES (valor1, valor2, etc.)	Insertar una fila poniendo valor1 en la columna1 y valor2 en la columna2. Las columnas que no estén mencionadas tendrán valor NULL (vacío).
---	---

Procedimiento recomendado para insertar filas desde un query. Fuente: Martínez Llario, 2020.*

CREATE TABLE pr_geodata.g29_planes_put_usos_urbanos_villalba (gid serial primary key, geom geometry ('MULTIPOLYGON', 6566), tipo character varying, municipio character varying, clasiput character varying, descripput character varying, r_put character varying, v_put character varying);	Crear una tabla con el nombre g29_planes_put_usos_urbanos_villalba con toda su definición de columnas y columna geométrica con sistema de referencia geoespacial SRID=6566 (PR, NAD83-2011)
INSERT INTO pr_geodata.g29_planes_put_usos_urbanos_villalba (geom, tipo, municipio, clasiput, descripput, r_put, v_put)	Insertar filas en la tabla anteriormente creada. Las filas se generarán a partir de un query donde el municipio = Villalba y tenga usos urbanos.
SELECT geom, tipo, municipio, clasiput, descripput, r_put, v_put FROM g29_planes_plan_uso_terrenos_2015 WHERE municipio = 'Villalba' and clasiput like 'SU%';	

* Martínez Llario, 2020, p. 114.

ALTER

Se usa para hacer cambios en una tabla existente: cambiar el nombre de la tabla, añadir o modificar columnas (cambiar nombre, cambiar el tipo de datos).

ALTER TABLE tabla RENAME TO nuevo nombre;	Cambiar el nombre de una tabla.
ALTER TABLE tabla RENAME COLUMN nombre_viejo TO nombre nuevo ;	Cambiar el nombre de una columna de una tabla.
ALTER TABLE tabla ADD nueva columna tipo de dato;	Añadir una columna y su definición de tipo de dato en una tabla existente.



ALTER TABLE tabla DROP COLUMN columna;	Quitar una columna de la tabla.
ALTER TABLE tabla ALTER COLUMN col1 TYPE varchar(80) ALTER COLUMN col2 TYPE varchar(100);	Cambiar el tipo de datos de dos columnas de una tabla en una sola sentencia SQL.

UPDATE

El enunciado UPDATE permite hacer cambios en los datos de una o varias columnas en una tabla.

UPDATE tabla SET columna1 = valor1, SET columna2 = valor2, SET columna3 = valor3 WHERE condición o filtro;	Modificar los valores una o más columnas. Aplicar filtro WHERE para cambiar solamente un subconjunto de datos. De lo contrario, asignará el mismo valor a todas las filas de las columnas nombradas.
---	--

Ya que contamos con las herramientas para conocer el contenido de la base de datos, y de qué columnas se componen las tablas, podemos pasar a la sección de preguntas de práctica para este módulo de SQL Básico. Además, incluiremos algunas preguntas que conlleven usar geodatos.

Prácticas con SQL Básico

En esta sección proponemos una serie de preguntas para contestar mediante queries a la base de datos.

- Dé una lista de **TODAS** las filas y **TODAS** las columnas de la tabla **census_block_data_2020_villalba**. La cantidad de filas devuelta debe ser 378. Recuerde que por defecto, DBeaver solo da las primeras 200 filas. Use el botón  **Calculate Total Row Count** para obtener el conteo de filas.
- Use la cláusula **DISTINCT** para hacer una **lista de los barrios** que aparecen en la tabla **census_block_data_2020_villalba**. Solo incluya la columna **barrio**. En la lista debe haber **solo 8 barrios**.
- Use la cláusula **DISTINCT** para hacer una **lista de los sectores censales** (tract) que aparecen en la tabla **census_block_data_2020_villalba**. Solo incluya la columna **tract**. En la lista **debe haber 6 filas**.
- ¿Cuántas unidades electorales hay en Villalba (según el Censo de 2020)? Haga primero una lista de las unidades electorales.** Las unidades electorales están en la columna **vtd** de la tabla **census_block_data_2020_villalba**. Debe usar la cláusula **DISTINCT** seguido del campo **vtd**: `Count(Distinct(vtd))`. ¿Cuántas filas devuelve si no usamos **DISTINCT**? ¿Por qué?
- ¿Cuál fue la población total del municipio de Villalba para el censo 2020?** Use la función `sum()` en la columna **pop100**. Debe devolver solo una fila.
- ¿Cuántos habitantes, viviendas, viviendas ocupadas y viviendas vacantes había en Villalba para el Censo de 2020?** Use la función `SUM(column_name)` para sumar cada campo: **pop100**, **hu100**, **h0010002**, **h0010003**. **Provea un alias AS a cada columna** para este query: **pop100 as población2020**, **hu100 as viviendas2020**, **h0010002 as ocupadas**, **h0010003 as vacantes**. Use la tabla **census_block_data_2020_villalba**.



Tutorial PostGIS, 3.x

Debe producir **una sola fila** con las sumas para cada columna.

Resultado:

Grid	123 población2020	123 vivienda2020	123 ocupadas	123 vacantes
1	22,093	8,875	7,989	886

Queries que usan GROUP BY

- Haga una **lista de barrios con la suma de población y viviendas, ocupadas y desocupadas**. Solo modifique el query anterior. Añada primero el barrio y luego las demás columnas. Debe usar la cláusula **GROUP BY** y luego añadir el campo barrio. Use la tabla **census_block_data_2020_villalba**. Recuerde que **GROUP BY** se escribe debajo o luego del nombre de la tabla en la cláusula **FROM**. Debe producir **8 filas**.

barrio	124 población2020	124 vivienda2020	124 ocupadas	124 vacantes
1 Villalba Abajo	2,524	1,062	954	108
2 Villalba Arriba	5,062	2,009	1,839	170
3 Caonillas Abajo	1,249	552	462	90
4 Caonillas Arriba	1,288	527	457	70
5 Hato Puerco Arriba	6,304	2,525	2,298	227
6 Villalba CUT	734	301	257	44
7 Hato Puerco Abajo	1,459	578	506	72
8 Vacas	3,473	1,321	1,216	105

Resultado:

- Haga una **lista de unidades electorales (vtd)** y añada la **población, viviendas, ocupadas y vacantes**. Solo modifique el query anterior cambiando la columna barrio por la columna vtd. Use la cláusula **GROUP BY** y añada la columna **vtd**. Ordene los resultados con **ORDER BY** usando la columna **población2020** de manera **descendente DESC**. Use la tabla **census_block_data_2020_villalba**. ¿Cuál es la unidad electoral con menor población? El número de filas debe ser 17.

vtd	124 población2020	124 vivienda2020	124 ocupadas	124 vacantes
1 065.17	227	67	52	15
2 0065.2	265	128	124	4
3 0065.8	521	216	181	35
4 0065.5	554	233	205	28
5 065.10	577	273	218	55
6 0065.9	603	220	204	16
7 0065.6	877	338	301	37
8 0065.4	1,025	415	367	48
9 065.16	1,128	456	402	54
10 065.14	1,141	499	450	49
11 065.11	1,333	570	506	64
12 0065.1	1,447	619	538	81
13 065.13	1,774	727	649	78
14 0065.7	1,936	836	762	74
15 065.15	2,422	906	865	41
16 0065.3	3,088	1,188	1,098	90
17 065.12	3,175	1,184	1,067	117

Resultados:

- Desglose por unidades electorales y barrios:** Use el mismo query anterior. En la cláusula **SELECT** añada la columna **barrio** después de la columna **vtd**. Incluya la coma, ejemplo: **vtd, barrio**, y deje las demás columnas nombradas. En la cláusula **GROUP BY** inserte la columna **vtd** antes de **barrio**, añada una coma. Por ejemplo: **GROUP BY vtd, barrio**. En la cláusula **ORDER BY** inserte primero el campo **vtd ASC**, añada coma y **barrio asc**. Mantenga **población2020 DESC**. ¿Qué pasó con las sumas de población



Tutorial PostGIS, 3.x

y viviendas por unidad electoral (vtd) y por qué? El número de filas debe ser 34.

Grid	asc vtd	asc barrio	123 población2020	124 vivienda2020	125 ocupadas	126 vacantes
1	0065.1	Villalba Arriba	1,447	619	538	81
2	0065.2	Villalba Arriba	265	128	124	4
3	0065.3	Hato Puerco Arriba	1,283	538	468	70
4	0065.3	Vacas	1,796	649	629	20
5	0065.3	Villalba CUT	9	1	1	0
6	0065.4	Vacas	1,025	415	367	48
7	0065.5	Vacas	554	233	205	28
8	0065.6	Hato Puerco Abajo	262	119	90	29
9	0065.6	Hato Puerco Arriba	615	219	211	8
10	0065.7	Caonillas Arriba	17	2	2	0
11	0065.7	Hato Puerco Arriba	1,858	820	755	65
12	0065.7	Vacas	61	14	5	9
13	0065.8	Caonillas Arriba	433	178	159	19
14	0065.8	Hato Puerco Arriba	88	38	22	16
15	0065.9	Caonillas Arriba	603	220	204	16
16	0065.10	Caonillas Abajo	419	174	148	26

Resultados:

10. Haga un **conteo de municipios por letra inicial**. Por ejemplo, ¿cuántos municipios empiezan con la letra A, cuántos con la letra B, hasta con la Y? Use la función **count(municipio)** para hacer los conteos.

** Para este query debe usar la función **left(municipio,1)** o la función **substr(municipio,1,1)** para obtener la letra inicial. Para hacer la lista de municipios hacia la derecha, debe usar la función **string_agg(municipio, ',')**. Recuerde dejar un espacio después de la coma. Haga GROUP BY por el campo de letra inicial. Haga orden alfabético ascendente por municipio y de forma descendente por conteo.

Geodato a usar: **g03_legales_municipios_2015**. Por ejemplo:

```

SELECT left(municipio,1) as letra_inicial,
count(municipio) as cuántos,
string_agg(municipio, ',') as municipios -- no escribir coma aquí.
FROM ESCRIBA EL NOMBRE DEL GEODATO MUNICIPAL
GROUP BY letra_inicial;
ORDER BY cuántos desc;

```

¿Cuál sería el campo por usar para agrupar?
 ¿Cuál es el campo para ordenar? ¿de cuál manera?
 ¿Cuáles son las letras iniciales que no se repiten en los nombres de municipios?

Grid	asc letra_inicial	123 cuántos	asc cuáles
1	C	14	Cabo Rojo, Caguas, Camuy, Canóvanas, Carolina, Cataño, Cayey, Ceiba, C...
2	A	8	Adjuntas, Aguada, Aguadilla, Aguas Buenas, Aibonito, Arecibo, Arroyo, A...
3	S	7	Sabana Grande, Salinas, San Germán, San Juan, San Lorenzo, San Sebasti...
4	M	6	Manatí, Maricao, Mayaguez, Mayagüez, Moca, Morovis

Resultados parciales:

11. **Crear una copia de una tabla existente:** Use el comando **CREATE TABLE tabla_nueva AS (SELECT * FROM Tabla_Existente)** para hacer una copia de una tabla partir de la tabla **census_block_data_2020_villalba**. El nombre debe ser **block_data2020_estudiante1**. Presencial: Sustituya el número de la tabla con el número que le corresponde en el taller.

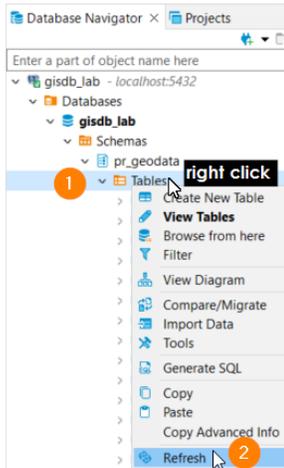
Name	Value
Updated Rows	378
Query	create table block_data2020_estudiante1 as (select * from census_block_data_2020_villalba)
Start time	Thu Feb 15 11:54:05 BOT 2024
Finish time	Thu Feb 15 11:54:05 BOT 2024

Resultado: Click en tab **Statistics 2**



Tutorial PostGIS, 3.x

Para ver la tabla nueva, vaya al **panel Database Navigator** Database Navigator x y haga **right click** en **Tables** y escoja **Refresh** o **F5**



Verifique que la tabla tiene el contenido deseado:
Escriba y ejecute el siguiente query:

```
SELECT * FROM block_data2020_estudiante1;
```

12. Supongamos que usted entiende que el nombre del barrio **“Villalba CUT”** está incorrecto o le indicaron que debe **cambiarlo** a **“Villalba Pueblo”**. Use el enunciado **UPDATE** para cambiar solamente el nombre de este barrio. **Tiene que filtrar los datos usando WHERE**. Si no lo hace, le pondrá el nombre **Villalba Pueblo** a todas las filas de la columna barrio en la tabla que usted creó. Por ejemplo:

```
UPDATE block_data2020_estudiante1
```

```
SET barrio = 'Villalba Pueblo'
```

```
WHERE barrio = _____;
```

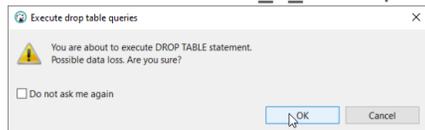
Resultado:

Name	Value
Updated Rows	21
Query	update block_data2020_estudiante1 bde set barrio = 'Villalba Pueblo' where barrio = 'Villalba CUT'
Start time	Thu Feb 15 12:11:54 BOT 2024
Finish time	Thu Feb 15 12:11:54 BOT 2024

Para **verificar**: escriba el query :
select DISTINCT ¿cuál campo? from
block_data2020_estudiante1;

13. **Elimine la tabla block_data2020_estudiante1** usando el comando **DROP**.

```
DROP TABLE tabla_a_borrar;
```



Click **OK** para confirmar.

Name	Value
Updated Rows	0
Query	drop table block_data2020_estudiante1
Start time	Thu Feb 15 12:22:26 BOT 2024
Finish time	Thu Feb 15 12:22:26 BOT 2024

Resultado:



Queries geográficos:

14. ¿Cuáles son los 20 municipios de mayor área superficial?

Use el geodato `g03_legales_municipios_2015`. Incluya el nombre del municipio, utilice la función `ST_Area(geom)`. Compute el área en **millas** y **kilómetros cuadrados**. Por ejemplo `ST_Area(geom)/2589988.1103` para **millas cuadradas AS sqmi** y `ST_Area(geom)/1000000` para **kilómetros cuadrados AS sqkm**. Use la cláusula **ORDER BY DESC** para ordenar el área de manera descendente por `sqkm` o `sqmi`. Use **LIMIT** para reducir la lista a solo 20 filas.

Grid	municipio	sqmi	sqkms
	Arecibo	126.84709925275642	328.53247889068314
	Ponce	116.88455447166889	302.72960635933515
	Utuado	114.97764879483009	297.79074332885904
	Coamo	78.01596803317535	202.060429619469
	Mayagüez	77.84267686833975	201.61160756292477
	Cabo Rojo	72.12341877717205	186.79879710706336
	San Sebastián	71.27173366845338	184.59294280176243
	Salinas	70.03695915082955	181.3948914822153
	Yauco	68.28242950458271	176.85068055926715
	Adjuntas	67.11879257749183	173.8368747533957
	Ciales	66.7350693378012	172.84303612495123
	Guayama	65.61200417029657	169.93431069402212
	Orocovis	63.76471098869199	165.14984331742798
	Lares	61.643874858119574	159.65690295535077
	Río Grande	60.80566858452703	157.48595867276723
	Juana Díaz	60.757060550029045	157.3600644413524
	Lajas	60.52811537892443	156.76709917028086
	Caguas	59.024424034101834	152.87255646562932
	Isabela	55.381066548245165	143.436303895688
	Yabucoa	55.23850684868433	143.06707596881753

Resultado:

15. ¿Cuáles la longitud total de carreteras y calles en el municipio de Villalba? Utilice el geodato `g35_viales_mapa_base_tiger_rds_2006se_villalba`. Debe utilizar las funciones `SUM()` y `ST_Length(geom)` para sumar las longitudes. La sintaxis debe ser:

SUM(ST_Length(geom)/1000) AS kms

SUM(ST_Length(geom)/1609.344) AS mi. Use el geodato

g35_viales_mapa_base_tiger_rds_2006se_villalba. Recuerde que el resultado solo va a tener una fila.

Grid	kms	mi
	349.4341702467361	217.1283269746775

Resultado:

16. La Comisión Estatal de Elecciones ha ubicado asentamientos existentes alrededor de toda la isla. Estos asentamientos pueden ser **sectores, urbanizaciones, condominios, centros de cuido** y están representados por un punto o dos en el caso de estar entre dos o más límites legales.

Dé una lista que muestre cada asentamiento y en cuál unidad electoral está ubicado. En la parte FROM use un alias para cada geodato:

g31_censo2020_blk_villalba AS e y el geodato **g25_asentamientos_sectores_2017 AS a**. **Separe las tablas con una coma. En la parte SELECT añada: a.tipo, a.nombre, e.vtd as unidad_electoral. En la cláusula WHERE debe ubicar la función**



ST_Intersects(a.geom, e.geom). Haga **ORDER BY** por **a.nombre** de forma **ascendente**. El resultado debe tener **81 filas**.

tipo	nombre	vtd
Sector	Aceituna	065.17
Urbanización	Alturas de Villalba	0065.3
Urbanización	Alturas del Alba	065.15
Sector	Apeadero	0065.4
Barriada	Borinquen	065.15
Sector	Camarones Abajo	065.12
Sector	Cerro Gordo	065.10

Resultado parcial:

17. Realice un **conteo de edificaciones ubicadas en cada tipo de zona inundable**.

En la parte **FROM** los geodatos a combinar son:

g23_riesgo_inunda_floodzones_2017_villalba AS f,

g33_mapa_base_building_centroids_1998_villalba AS e.

-- Borre los alias para las tablas que DBEaver asigna por defecto. --

En la parte **SELECT**, use las columnas **fld_zone**, **zone_subty** y añada la función **count(e.gid)** as **num_edifs** para contar filas.

En la cláusula **WHERE** debe ubicar la función **ST_Intersects(f.geom, e.geom)**.

En la cláusula **GROUP BY** use las columnas **fld_zone**, **zone_subty**.

En la cláusula **ORDER BY** ordene el conteo de puntos **num_edifs** de manera **descendente**.

fld_zone	zone_subty	num_edifs
X	0.2% Annual Chance Flood	241
A		227
A	FLOODWAY	47

Resultados:

Show query results as formatted plain text

Definición de **zona inundable "A"** según FEMA:

Areas with a 1% annual chance of flooding and a 26% chance of flooding over the life of a 30-year mortgage. Because detailed analyses are not performed for such areas; no depths or base flood elevations are shown within these zones. Last updated April 20, 2023.

18. Modifique el query anterior y haga otro **conteo de edificios** pero esta vez use el geodato **g29_planes_put_2015_villalba**.

En la cláusula **FROM** los geodatos son:

g33_mapa_base_building_centroids_1998_villalba AS e

g29_planes_put_2015_villalba AS p

En la cláusula **SELECT** añada el **p.clasiput** as **código_de_uso** y **p.descrippt** as **descripción**.

En la cláusula **GROUP BY** añada las columnas **p.clasiput** y **p.descrippt**.

	abc código_de_uso	abc descripción	123 num_edifs
1	SU	Suelo Urbano	5,746
2	SRC	Suelo Rústico Común	5,362
3	SREP-A	Suelo Rústico Especialmente Protegido Agrícola	561
4	SREP	Suelo Rústico Especialmente Protegido	443
5	SREP-H	Suelo Rústico Especialmente Protegido Hídrico	386
6	SREP-E	Suelo Rústico Especialmente Protegido Ecológico	144
7	VIAL	Vial	44
8	SURP	Suelo Urbanizable Programado	9
9	SREP-AH	Suelo Rústico Especialmente Protegido Agrícola e Hídrico	8
10	SURNP	Suelo Urbanizable No Programado	2
11	AGUA	Agua	2

Resultado:

19. ¿Cuáles son los **5 barrios con mayor área territorial** y a qué municipio pertenecen?

FROM: g03_legales_barrios_2015. Expanda los nodos para ver la lista de campos.



Tutorial PostGIS, 3.x

SELECT: municipio, barrio, SUM(ST_Area(geom)/1000000) AS sqkms para que le devuelva los valores **en kilómetros cuadrados**.

ORDER BY en la nueva columna **sqkms desc**.

Use **LIMIT** para devolver las primeras 5 filas

municipio	barrio	sqkm
Salinas	Lapa	64.15836650564829
Arecibo	Sabana Hoyos	59.604931017627514
Mayagüez	Isla de Mona e Islote Monito	56.85899241097673
Salinas	Aguirre	52.05658306768688
Cabo Rojo	Llanos Costa	51.978746695017534

Resultado:

20. Realice un conteo de edificios ubicados en suelos potencialmente agrícolas. (Recuerde que la cláusula SELECT se escribe antes de FROM)

FROM: g33_mapa_base_building_centroids_1998_villalba AS e,

g15_suelos_soil_map_units_2018_villalba AS s

SELECT: s.farm_class, s.muname, count(e.gid) as num_edifs

WHERE: st_intersects(¿geometrías?) AND s.farm_class <> 'Not prime farmland'

GROUP BY: s.farm_class, s.muname

ORDER BY: s.farm_class, num_edifs desc

farm_class	muname	num_edifs
All areas are prime farmland	Montegrande clay, 2 to 12 percent slopes	1,343
All areas are prime farmland	Lares clay, 5 to 12 percent slopes	498
All areas are prime farmland	Toa silty clay loam, 0 to 2 percent slopes, occasionally flooded	204
All areas are prime farmland	Dagüey clay, 12 to 20 percent slopes	41
Farmland of statewide importance	Quebrada silty clay loam, 12 to 20 percent slopes, eroded	1,447
Farmland of statewide importance	Los Guineos clay, 20 to 40 percent slopes	95
Farmland of statewide importance	Humatas clay, 20 to 40 percent slopes	76
Farmland of statewide importance	Alonso clay, 20 to 40 percent slopes	37
Farmland of statewide importance	Callabo silty clay loam, 12 to 20 percent slopes	32
Farmland of statewide importance	Múcara silty clay, 12 to 20 percent slopes, eroded	25
Farmland of statewide importance	Jácana clay, 5 to 12 percent slopes	7
Prime farmland if irrigated	Llanos clay, 2 to 5 percent slopes	7

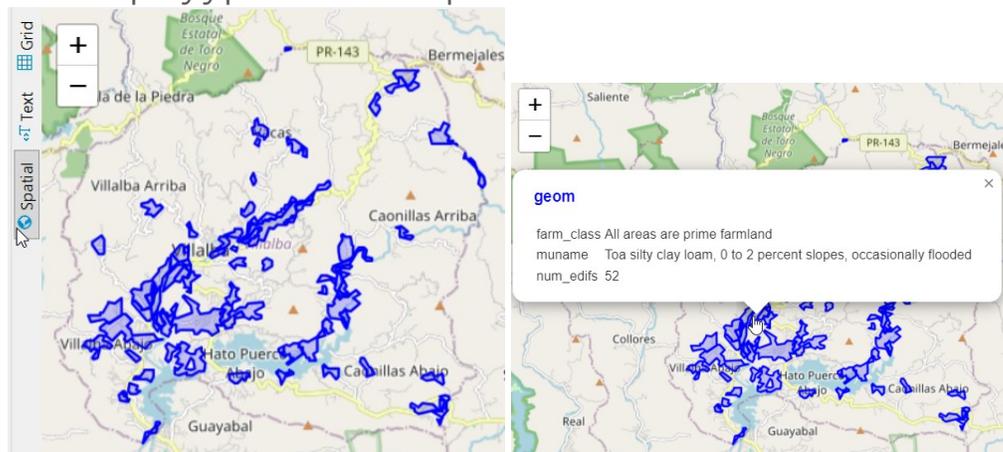
Resultado:

21. Si deseamos **ver este resultado en el contexto de un mapa**, solo debemos **añadir la geometría del geodato** de suelos en la cláusula SELECT después del count:

count(e.gid) num_edifs, s.geom

y en la cláusula GROUP BY después de s.muname: **s.muname, s.geom**

Corra el query y presione el tab Spatial





SQL espacial para geometrías vectoriales

Operaciones espaciales: Adyacencia, continencia, intersección, distancia

En esta parte vamos a practicar algunas funciones espaciales para búsqueda/selección de datos geográficos. En la sección anterior ya habíamos practicado algunas búsquedas geográficas, tanto en las prácticas de demostración, como en las prácticas asignadas.

La mayoría de los queries geográficos practicados han utilizado la función o “predicado” `ST_Intersects(geomA,geomB)` para seleccionar geometrías que tengan algún punto en común. Sin embargo, hay variedad de estos predicados que pueden ser utilizados, según el tipo de búsqueda que interesamos, así como también los tipos de geometría que estaremos comparando. Los apéndices A-1 y A-2 explican en detalle estos predicados. El apéndice A-1 hace un inventario de las posibles combinaciones entre relaciones topológicas por tipo de geometría (dimensión=0 para punto, 1 para líneas, 2 para polígonos). El apéndice A-2 hace otro inventario de los predicados espaciales que vimos en la sección anterior, tales como `Intersects`, `Disjoint`, `Touches`, `Overlaps`, etc. Todos estos se comparan con las distintas dimensiones geométricas. Todas las descripciones en los apartados A-1, y A-2 provienen de **JC Proteau, Matrices de Clementini et Prédicas Spatiaux de l'OGC, 2011**. En el apartado A-2 se añade la fuente de la documentación de PostGIS <https://postgis.net/docs/> para la versión “dev” del 23 de enero de 2024.

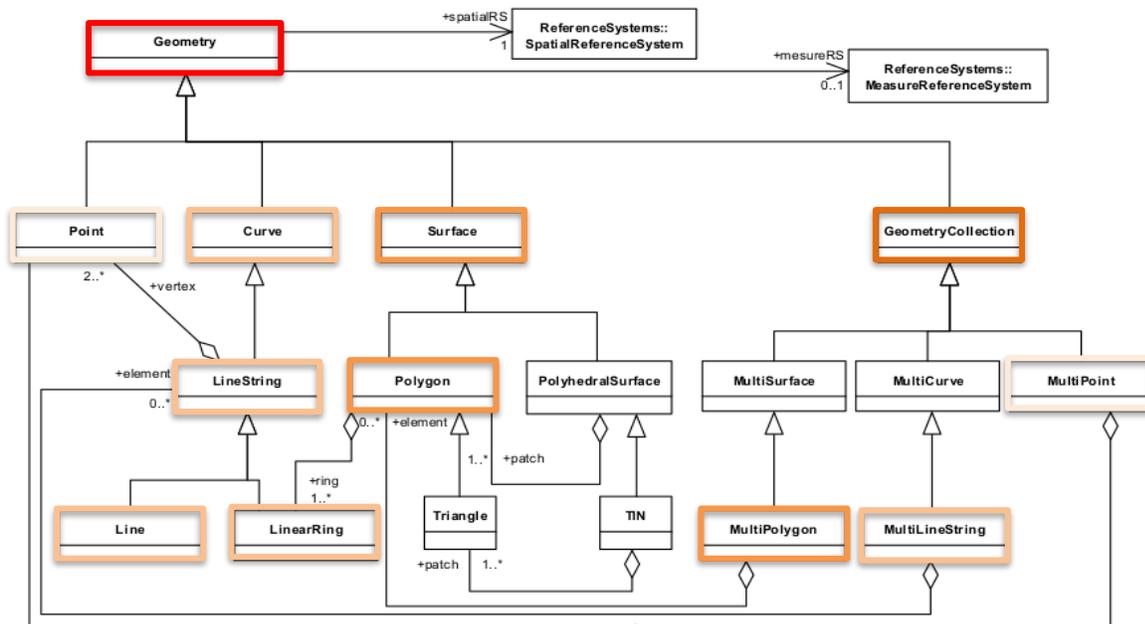
Comenzaremos por mostrar el modelo de elementos geográficos simples del [Open Geospatial Consortium](#).



El modelo *Simple Features* del Open Geospatial Consortium

Muchos programas *open source* de manejo de datos geográficos utilizan este estándar para codificar y registrar las geometrías que se usarán para representar elementos geográficos y eventos localizables. QGIS hace uso de este estándar y es buena idea describir algunos aspectos de importancia. De esta manera, podremos entender mejor el comportamiento de éste y otros programas que adoptan este estándar.

El siguiente diagrama, extraído del documento [OpenGIS® Implementation Standard for Geographic information - Simple feature access - Part 1: Common architecture](#), versión 1.2.1.145



En este podemos notar las jerarquías de las geometrías. Las de arriba son las geometrías abstractas, *Geometry*, *Point*, *Curve*, *Surface*, *GeometryCollection*, de las cuales se derivan *LineString*, *Polygon*, *MultiPolygon*, *MultiLineString* y otras. QGIS y otros programas tipo *Desktop*, no leen directamente el tipo *GeometryCollection*, pero sí *MultiPolygon*, *MultiPoint* y *MultiLineString*.

Dimensión de las geometrías

Como se ha mencionado antes, estas son las dimensiones numéricas otorgadas a los tipos básicos de geometría, según los estándares del OGC. Estas dimensiones no están relacionadas con el número de ejes x,y,z,m que pueden tener algunas geometrías.

Geometría	Dimensión
Punto, MultiPoint	0
Entidad lineal	1
Entidad superficial	2



Interior, contorno y exterior de las geometrías

Estas características de las geometrías nos ayudan a entender las relaciones espaciales y el uso de los operadores y predicados espaciales:

Point/MultiPoint		Interior	El mismo punto o puntos.
		Contorno/Límite	Vacío/No existe.
Line/MultiLine		Interior	Puntos que no estén en los puntos del contorno.
		Contorno/Límite	Puntos de inicio y final . MultiLine: puntos de contorno que estén en líneas componentes que sean impares .
Polygon/MultiPolygon		Interior	Puntos del interior de los anillos: MultiPolygon: Puntos del interior de los anillos.
		Contorno/Límite	Conjunto de anillos exteriores e interiores.

El **exterior** de estas geometrías estará compuesto por los puntos que no estén ni en el interior ni el contorno. Referencia: https://portal.ogc.org/files/?artifact_id=13227.

	Interior	Límite/Contorno/Boundary	Exterior
Punto (Dim=0)	Dim=0	Vacío / No existe	
Poli-línea (Dim=1)	Dim=1	Dim=0	
Polígono/área (Dim=2)	Dim=2	Dim=1	 Dim=2

Referencia: [Predicados OGC: 2.2: Conceptos de interior, límite y exterior de objetos](#), pág. 6. Ministerio de Ecología, desarrollo sustentable, transportación y vivienda, Gobierno de Francia.



Predicados para las relaciones topológicas

A continuación, una breve explicación de los predicados disponibles en PostGIS. Tabla 3.

Predicado	Geometría: P punto, L polilínea, S polígono	Condiciones
ST_Equals	Todas	geom(A) es igual a geom(B) si: * la <i>relación topológica</i> entre estos es idéntica ... aunque ... * el número de vértices y la dirección de las líneas pueden ser diferentes
ST_Disjoint	Todas	geom(A) es desjunto o separado de geom(B) si: * los objetos no tienen ningún punto en común (interior o límite). (Este es el inverso de <i>Intersects</i>)
ST_Touches	S/S, L/S, L/L, P/S, P/L	geom(A) toca a geom(B) si: * los límites (contornos) de los objetos tienen al menos un punto en común... y ... * si los interiores de ambos no tienen algún punto en común
ST_Crosses	P/S,P/L,L/S,L/L	geom(A) cruza a geom(B) si: * los interiores de los objetos tienen al menos un punto en común... pero ... * no todos en común... y ... * si la dimensión de la intersección de los interiores es <i>inferior</i> a la dimensión máxima de los objetos geom(A) y geom(B) (no aplica a P/P, S/S)
ST_Within	Todas	geom(A) está dentro de geom(B) si: * todo punto de geom(A) está dentro de geom(B) ... y ... * si los interiores tienen al menos algún punto en común ... * ningún punto de geom(A) está en el exterior de geom(B) * (inverso de Contains)
ST_Contains	Todas	geom(A) contiene a geom(B) si: * todo punto de geom(B) es un punto de geom(A) ... y ... * si los interiores tienen al menos algún punto en común * ningún punto de geom(B) está en el exterior de A * (inverso de Within)
ST_Overlaps	S/S,L/L,P/P	geom(A) solapa a geom(B) si a la vez: * geom(A) y geom(B) tienen la misma dimensión (no aplica a P/L, P/S, L/S) ... * geom(A) y geom(B) tienen puntos en común... pero no todos ... * La <i>intersección</i> de los interiores de geom(A) y geom(B) tiene la misma <i>dimensión</i> que geom(A) y geom(B)
ST_Intersects	Todas	geom(A) interseca a geom(B) si: * geom(A) y geom(B) tienen al menos un punto en común (interior o límite) * (Inverso de Disjoint)
ST_Covers	Todas	geom(A) cubre a geom(B) si: * ningún punto de geom(B) está en el exterior de A * todo punto de geom(B) es un punto de geom(A) Compárese con <i>Contains</i>
ST_CoveredBy	Todas	geom(A) está cubierto por geom(B) si: * ningún punto de geom(A) está en el exterior de geom(B) * todo punto de geom(A) es un punto de geom(B) Compárese con <i>Within</i>
ST_Relate (AB, DE-9IM Pattern Matrix)	Todas	* Explica la relación espacial entre geom(A) y geom(B) mediante la aplicación del modelo DE9IM. * Permite la generalización de los predicados espaciales para 98 relaciones topológicas. De las 98 relaciones, 92% son entre LL, LS y SS

***ST_Relate** solo se puede usar desde consultas SQL espaciales a la base de datos o a través del DB Manager de QGIS.



Tutorial PostGIS, 3.x

La función **ST_DWithin(geom_a, geom_b, distancia)** para **análisis de proximidad** es una implementación de **PostGIS**. En cuanto sea posible, **sustituya el uso de ST_Intersects con ST_DWithin**, ya que el parámetro de distancia en ST_DWithin sirve además como umbral de tolerancia mediante una distancia no significativa. (Martínez Llario, 2018, pp. 143-46.) Hsu y Obe (2015, p249-50) se refieren a esta función como un "ST_Intersects con tolerancia".

DESGLOSE DE RELACIONES TOPOLÓGICAS POR GEOMETRÍA

En la tabla anterior, en la sección ST_Relate se mencionó que existen 98 tipos diferentes de relaciones topológicas para geometrías puntuales, lineales y de área. Estas se dividen:

Desglose de Relaciones Topológicas por tipo de geometría					
Puntos		Líneas		Áreas(S)	
Relaciones	Cantidad	Relaciones	Cantidad	Relaciones	Cantidad
Punto/Punto	2	Línea/Línea	47	Área/Área(S)	12
Punto/Línea	3	Línea/Área(S)	31		
Punto/Área(S)	3				

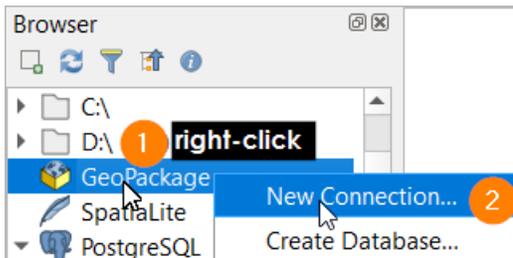
USAR QGIS PARA EJEMPLOS DE RELACIONES TOPOLÓGICAS

En esta parte vamos a practicar algunas estas funciones de relación, de manera que las podamos visualizar y poder tener una idea más clara de cómo funcionan. En lugar de escribir los predicados, usaremos la herramienta **Select by Location** de QGIS. Esta contiene la mayoría de los predicados, excepto ST_Relate, el cual es un poco más complejo de construir. **La ventaja de QGIS es que podemos ver todos los geodatos**, a diferencia de PostGIS, donde se ve uno a la vez.

- Abra una sesión de QGIS.



- Vaya al panel **Browser** y haga **right click** en **Geopackage** y escoja la opción **New Connection...**

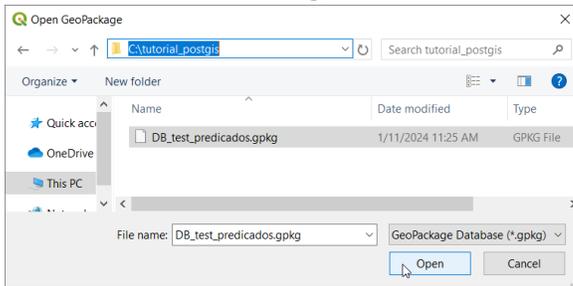


- Localice y escoja (**Open**) el archivo **DB_test_predicados.gpkg** en el directorio **c:\tutorial_postgis**. Esta localización puede variar, según donde se haya guardado

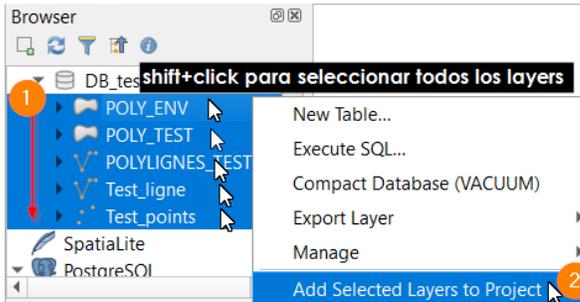


Tutorial PostGIS, 3.x

este archivo GeoPackage.

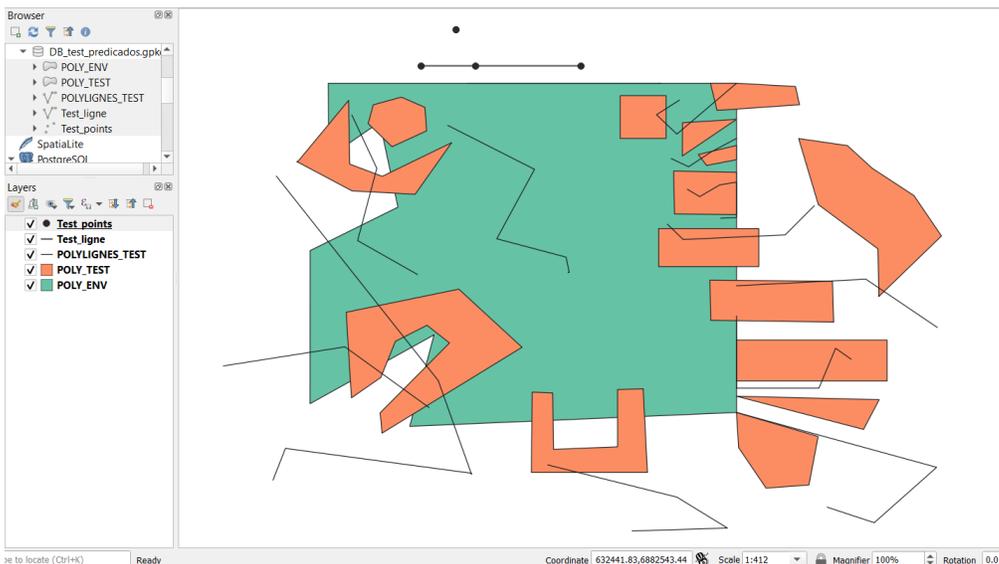


- Dentro de este GeoPackage, **escoja (shift+click) los 5 layers existentes.** Haga **right click** en la opción **Add Selected Layers to Project**



Este archivo GeoPackage fue copiado desde el [espacio interministerial de la información geográfica, del gobierno francés](#) (2011). El archivo original estaba en formato SpatialLite y fue exportado a GeoPackage, incluyendo una traducción de los textos originales de algunas filas en las tablas. La exportación y traducción fue hecha en nuestra agencia. Ninguna de sus coordenadas fue modificada ni el sistema de referencia geoespacial, ya que esto puede alterar las funciones que vamos a experimentar. Los geodatos tienen coordenadas reales en el sistema Lambert-93 ([EPSG:2154](#)) y están localizadas en la [Universidad CY Cergy-Paris](#). Por otro lado, los elementos geográficos de este banco de datos son ficticios.

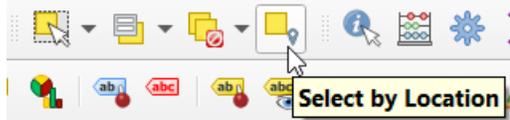
- En su visor de QGIS deberá ver los geodatos experimentales para las pruebas que vamos a realizar.





Tutorial PostGIS, 3.x

- Vaya a la **barra de botones** en el **lado superior de la interfaz gráfica de QGIS** y haga **click** en el botón **Select by Location**:



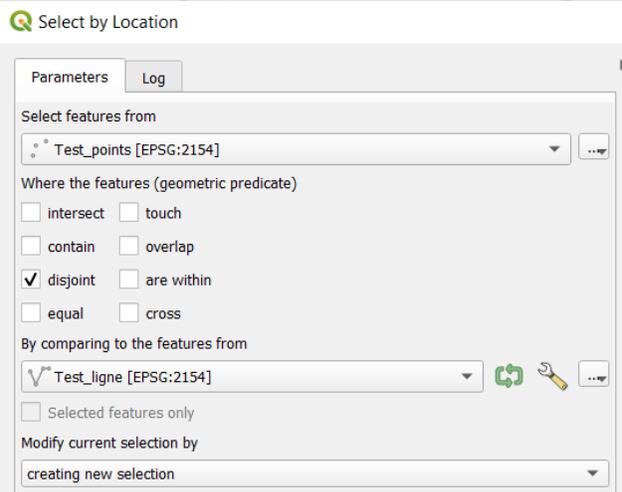
- Una vez aparezca la herramienta, comencemos a hacer las evaluaciones:

ST_Disjoint

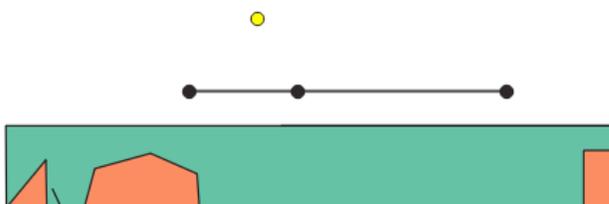
Predicado	Geometría: P punto, L polilínea, S polígono	Condiciones
ST_Disjoint	Todas	geom(A) es desajunto o separado de geom(B) si: * los objetos no tienen ningún punto en común (interior o límite). (Este es el inverso de <i>Intersects</i>)

Disjoint: Puntos con líneas

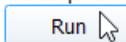
Disjoint encontrará elementos que no tengan ningún punto en común entre las geometrías A y B, entre los elementos del geodato de puntos y la línea de prueba:



Resultado (selección en amarillo):



- Seleccionar elementos del geodato de puntos **test_points...**
- Geometric predicate: **disjoint** Que estén fuera de las geometrías de...
- By comparing to the features from: Test_ligne**
- Siempre vamos a hacer una **nueva selección.**
- Haga **click** en el botón **Run** para hacer esta prueba.

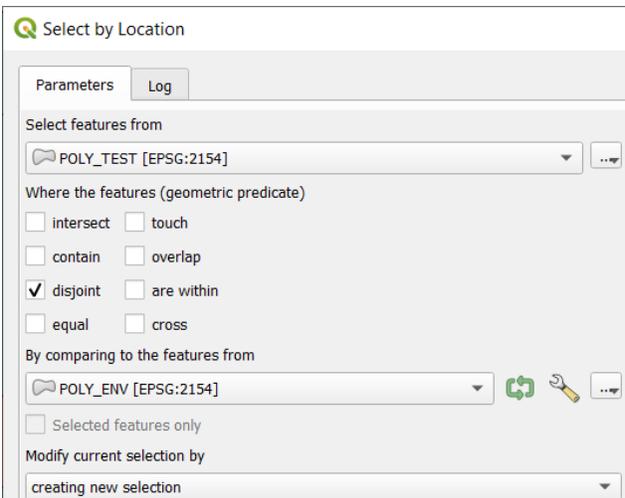


- Seleccionará el punto/elemento que está **fuera de la línea de prueba.** Hay que recordar que las líneas están compuestas de inicio y final (límite/borde) y su interior (todos los puntos entre los límites)

- Disjoint: Polígonos/Polígonos.** ¿Cuáles polígonos rojos (poly_test) están fuera del polígono verde (poly_env)? Hagamos la prueba.

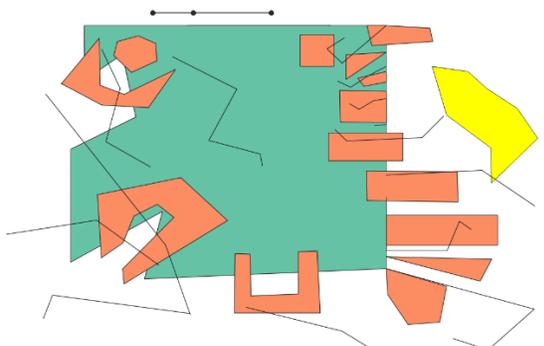


Tutorial PostGIS, 3.x



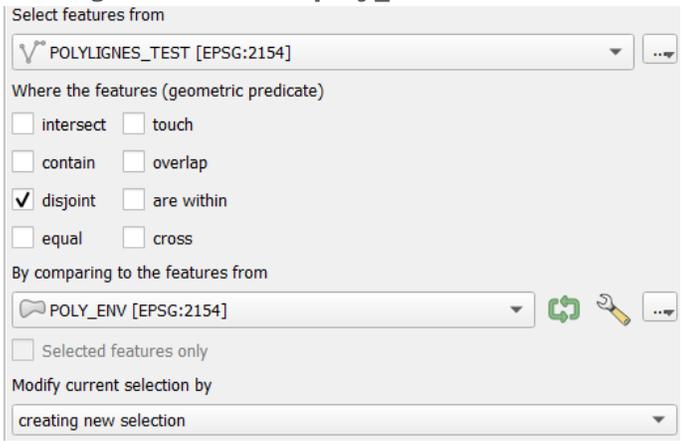
- Seleccionar elementos del geodato de puntos **poly_test...**
- Geometric predicate: Que estén fuera (disjoint) de la geometría de...
- By comparing to the features from: poly_env**
- Siempre vamos a hacer una **nueva selección.**
- Haga **click** en el botón **Run** para hacer esta prueba.

Resultado (selección en amarillo):



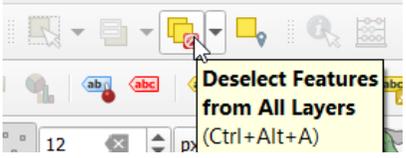
- Seleccionará (en amarillo) el área/elemento que está fuera del polígono (verde) de prueba.

- Realice la misma prueba con disjoint comparando el geodato de líneas **polylignes_test** con el geodato de área **poly_env**.



¿Cuántas geometrías de línea están disjuntas? _____

- Haga **click** en el botón **Deselect Features** para borrar esta selección.





ST_Touches

Predicado	Geometría: P punto, L polilínea, S polígono	Condiciones
ST_Touches	S/S, L/S, L/L, P/S, P/L	geom(A) toca a geom(B) si: * los límites (contornos) de los objetos tienen al menos un punto en común... y ... * si los interiores de ambos no tienen algún punto en común

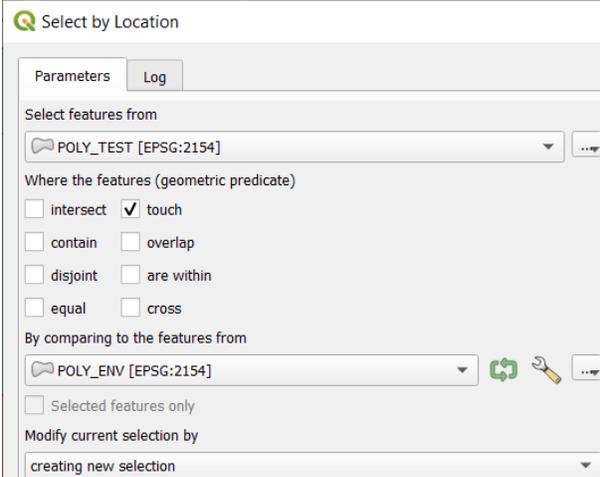
Como lee en la tabla,

- 1) los **contornos** de las geometrías **deben tener al menos un punto o vértice en común**;
- 2) los **interiores** de las geometrías **no pueden tener ningún punto o vértice en común**.

ST_Touches aplica para comparar/relacionar:
Área/Área, Línea/Área, Línea/Línea, Punto/Área, Punto/Línea

Pasemos a usar Select by Location para hacer las pruebas

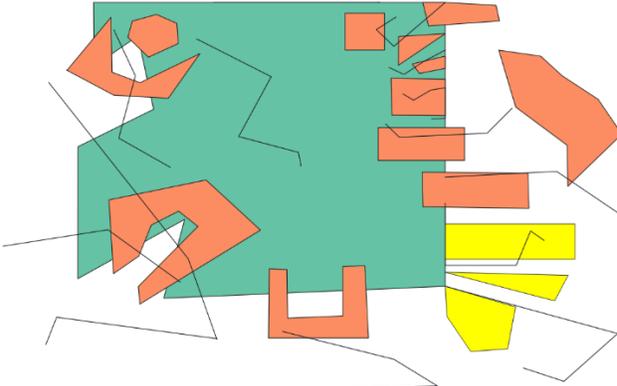
ST_Touches: Área/Área



- Seleccionar elementos del geodato de puntos **poly_test...**
- Geometric predicate: **touch**
- By comparing to the features from: poly_env**
- Siempre vamos a hacer una **nueva selección**.
- Haga **click** en el botón **Run** para hacer esta prueba.

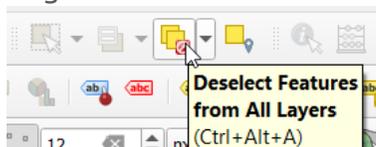
Run

Resultado (selección en amarillo):



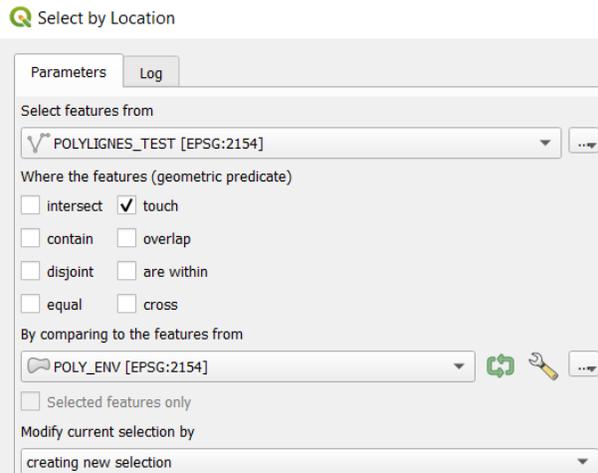
1. Según las reglas, los contornos (bordes) de las áreas deben tocarse o al menos tener un vértice en común y además los interiores no pueden tener puntos en común.
2. El resto de las áreas/polígonos están completamente dentro, o están parcialmente dentro y hay uno completamente fuera del área de poly_env.
3. Este predicado es útil para encontrar parcelas, municipios adyacentes por ejemplo.

- Haga **click** en el botón **Deselect Features** para borrar esta selección.





ST_Touches: Líneas/Áreas



Resultado (selección en amarillo):

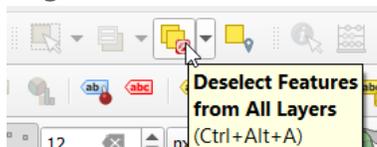


- Seleccionar elementos del geodato de puntos **polylignes_test...**
- Geometric predicate: **touch**
- By comparing to the features from: poly_env**
- Siempre vamos a hacer una **nueva selección.**
- Haga **click** en el botón **Run** para hacer esta prueba.

Run

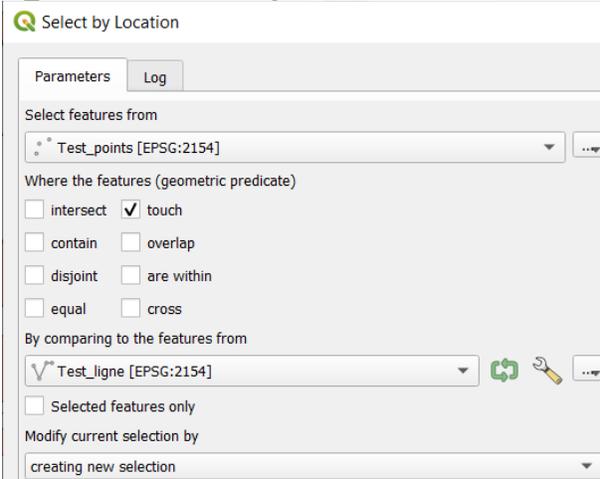
1. **Seleccionó 4 líneas.** Según las reglas, **los contornos (bordes) de las líneas deben tocar o al menos tener un vértice en común con poly_env.** Además **los interiores de las líneas no pueden tener puntos en común con poly_env.**
2. **Ninguna de las líneas seleccionadas tiene vértices en el interior ni están completamente fuera de poly_env.**
3. **Comparten uno más puntos en el borde de poly_env.**
4. Útil para cotejar conexión entre objetos de infraestructuras. Líneas con líneas, con puntos, con áreas.

- Haga **click** en el botón **Deselect Features** para borrar esta selección.

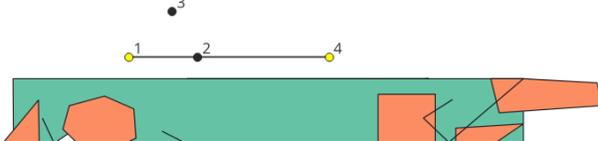




ST_Touches: Puntos/Líneas



Resultado (selección en amarillo):

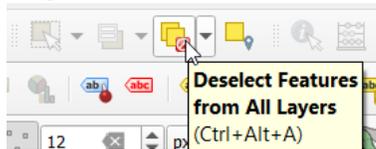


- Seleccionar elementos del geodato de puntos **test_points...**
- Geometric predicate: **touch**
- By comparing to the features from: test_ligne**
- Siempre vamos a hacer una **nueva selección.**
- Haga **click** en el botón **Run** para hacer esta prueba.



1. Seleccionó 2 puntos: 1 y 4. El punto 2 está en el interior de la línea. El punto 3 está fuera de la línea.
2. El predicado **Touch devolverá cierto si los puntos están en el contorno de la línea.** Los contornos de la línea son los puntos de inicio y final. En el caso de una MultiLine los puntos de contorno están en líneas componentes que sean impares.
3. Útil para cotejar conexión entre objetos de infraestructuras. Líneas con líneas, con puntos, con áreas.

- Haga **click** en el botón **Deselect Features** para borrar esta selección.



Ejemplos: Linestrings:

				<p>Key</p> <p>s start</p> <p>e end</p>
(a) línea simple	(b) línea no-simple porque se auto cruza	(c) línea simple y cerrada	(d) línea no-simple y cerrada	

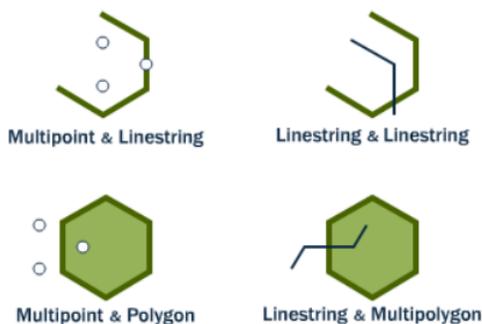
Start y end son los contornos de una línea.



ST_CROSSES

Predicado	Geometría: MP multipoint, L polilínea, S polígono	Condiciones
ST_Crosses	MP/S, MP/L, L/S, L/L	geom(A) crusa a geom(B) si: * los interiores de los objetos tienen al menos un punto en común... pero ... * no todos en común... y ... * si la dimensión de la intersección de los interiores es <i>inferior</i> a la dimensión máxima de los objetos geom(A) y geom(B) (no aplica a P/P, S/S, pero aplica a MultiPoint)

Cross

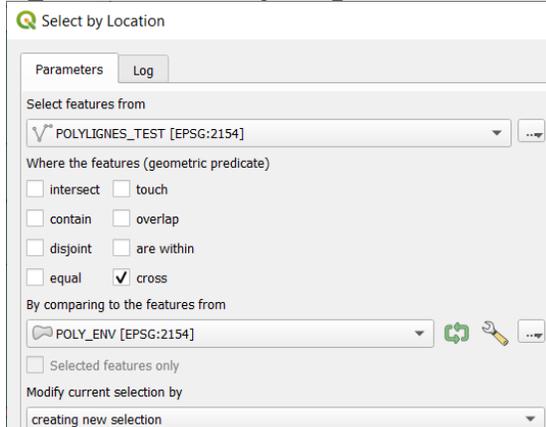


En todos estos casos, hay cruce entre las geometrías comparadas.

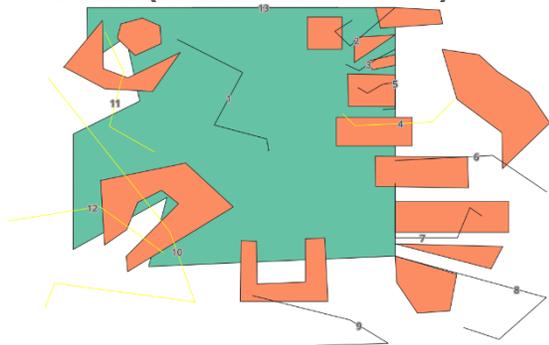
- Multipoint/Line
- Line/Line
- Multipoint/Polígono
- Línea/Multipolígono

Fuente: https://postgis.net/workshops/postgis-intro/spatial_relationships.html

ST_Crosses: Líneas/Polígonos



Resultado (selección en amarillo):



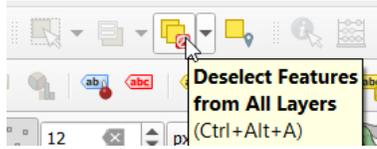
- Seleccionar elementos del geodato de puntos **polylignes_test...**
- Geometric predicate: **touch**
- By comparing to the features from: poly_env**
- Siempre vamos a hacer una **nueva selección.**
- Haga **click** en el botón **Run** para hacer esta prueba.

Run

1. Las líneas seleccionadas (en amarillo) son la 4, 10, 11 y la 12.
2. El predicado **cross** devolverá cierto si los interiores de las líneas tienen uno o más puntos (pero no todos) en el interior del polígono.



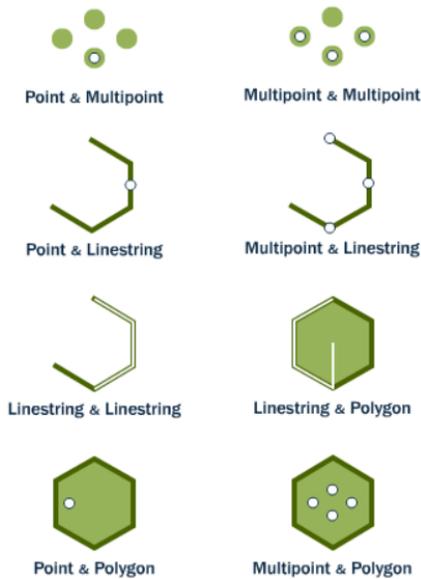
- Haga **click** en el botón **Deselect Features** para borrar esta selección.



ST_WITHIN/ST_CONTAINS

Predicado	Geometría: MP multipoint, L polilínea, S polígono	Condiciones
ST_Within	Todas	geom(A) está dentro de geom(B) si: * todo punto de geom(A) está dentro de geom(B) ... y ... * si los interiores tienen al menos algún punto en común ... * ningún punto de geom(A) está en el exterior de geom(B) * (inverso de Contains)
ST_Contains	Todas	geom(A) contiene a geom(B) si: * todo punto de geom(B) es un punto de geom(A) ... y ... * si los interiores tienen al menos algún punto en común * ningún punto de geom(B) está en el exterior de A * (inverso de <i>Within</i>)

Within/Contains



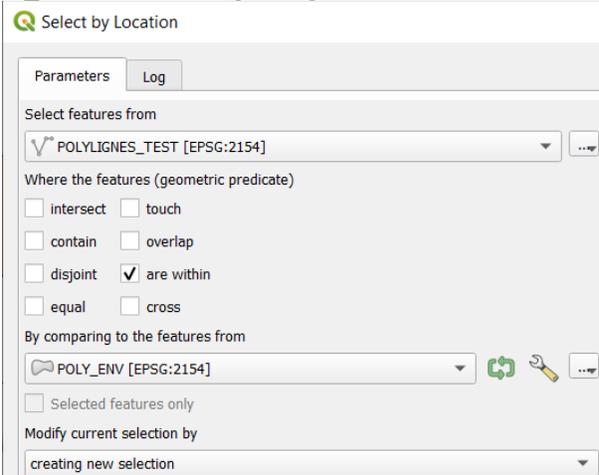
En todos estos casos, hay contigencia entre las geometrías comparadas.

- Punto/Multipunto
- Multipunto/Multipunto
- Punto/Línea
- Multipunto/Línea
- Línea/Línea
- Punto/Área
- Multipunto/Área

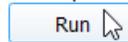
Fuente: https://postgis.net/workshops/postgis-intro/spatial_relationships.html



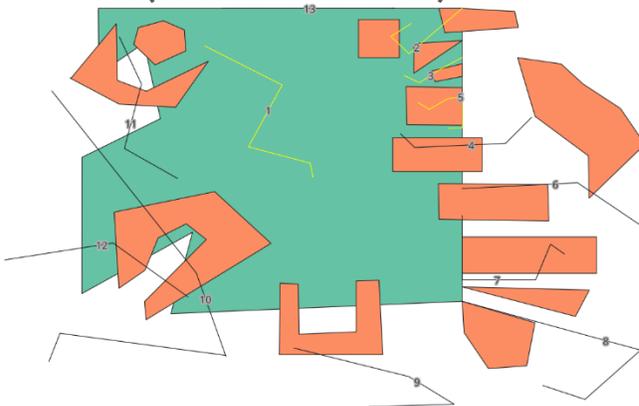
ST_Within: Líneas/Polígonos



- Seleccionar elementos del geodato de puntos **polylignes_test...**
- Geometric predicate: **are within**
- By comparing to the features from: poly_env**
- Siempre vamos a hacer una **nueva selección.**
- Haga **click** en el botón **Run** para hacer esta prueba.

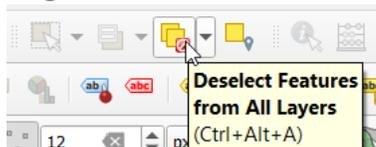


Resultado (selección en amarillo):



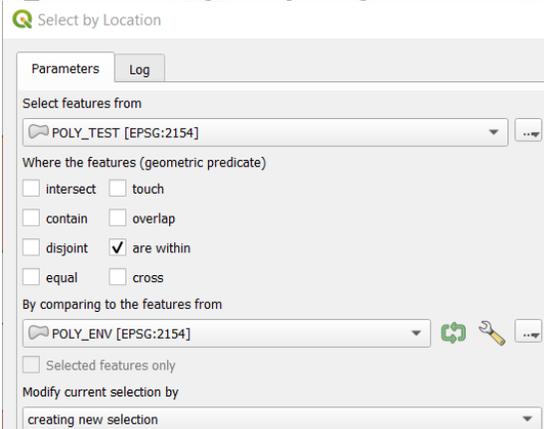
1. Las líneas seleccionadas (en amarillo) son la 1, 2, 3 y la 5.
2. El predicado **are within devolverá cierto y seleccionará cuando todos los puntos que componen los elementos están dentro de la otra geometría.**

- Haga **click** en el botón **Deselect Features** para borrar esta selección.

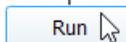




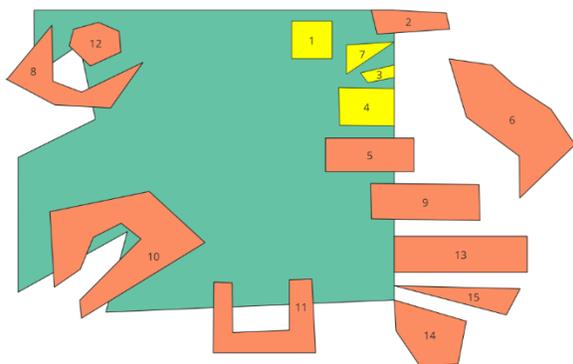
ST_Within: Polígonos/Polígonos



- Seleccionar elementos del geodato de puntos **poly_test...**
- Geometric predicate: **are within**
- By comparing to the features from: poly_env**
- Siempre vamos a hacer una **nueva selección.**
- Haga **click** en el botón **Run** para hacer esta prueba.

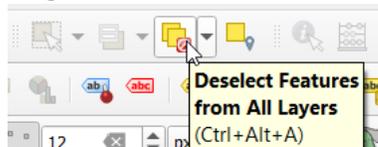


Resultado (selección en amarillo):



1. Los polígonos seleccionados (en amarillo) son el 1, 3, 4 y 7.
2. El predicado **are within** devolverá cierto y seleccionará cuando todos los puntos que componen los elementos están dentro de la otra geometría (incluyendo el borde).

- Haga **click** en el botón **Deselect Features** para borrar esta selección.



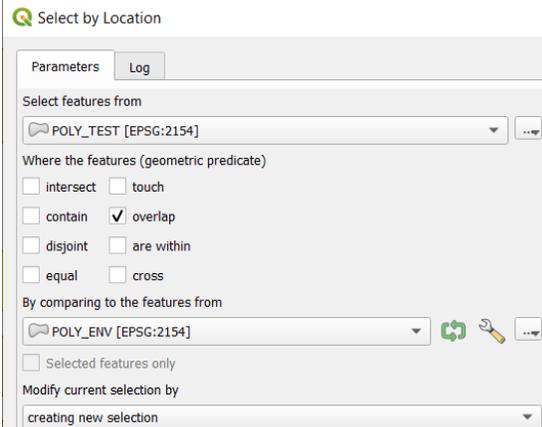
ST_OVERLAPS

Predicado	Geometría: MP multipoint, L polilínea, S polígono	Condiciones
ST_Overlaps	S/S,L/L,P/P	geom(A) solapa a geom(B) si a la vez: * geom(A) y geom(B) tienen la misma dimensión (no aplica a P/L, P/S, L/S) ... * geom(A) y geom(B) tienen puntos en común... pero no todos ... * La <u>intersección</u> de los interiores de geom(A) y geom(B) tiene la misma <u>dimensión</u> que geom(A) y geom(B)

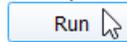




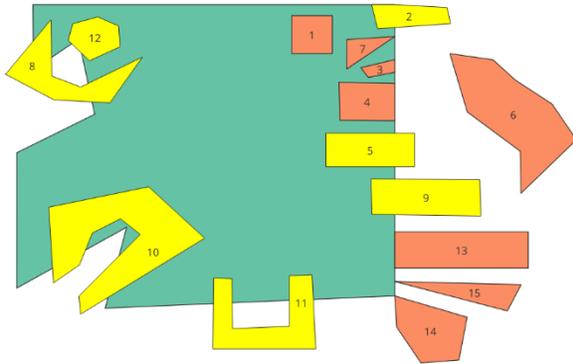
ST_Overlaps: Polígonos/Polígonos



- Seleccionar elementos del geodato de puntos **poly_test...**
- Geometric predicate: **overlap**
- By comparing to the features from: poly_env**
- Siempre vamos a hacer una **nueva selección.**
- Haga **click** en el botón **Run** para hacer esta prueba.

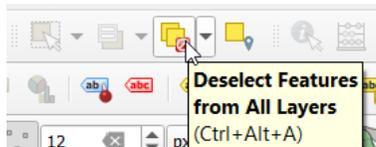


Resultado (selección en amarillo):



3. Los polígonos seleccionados (en amarillo) son el 2, 5, 8, 9 10, 11, y 12.
4. El predicado **overlap** devolverá **cierto** y seleccionará cuando los interiores de ambas geometrías tengan puntos en común pero que a la vez no estén completamente dentro de poly_env.

- Haga **click** en el botón **Deselect Features** para borrar esta selección.

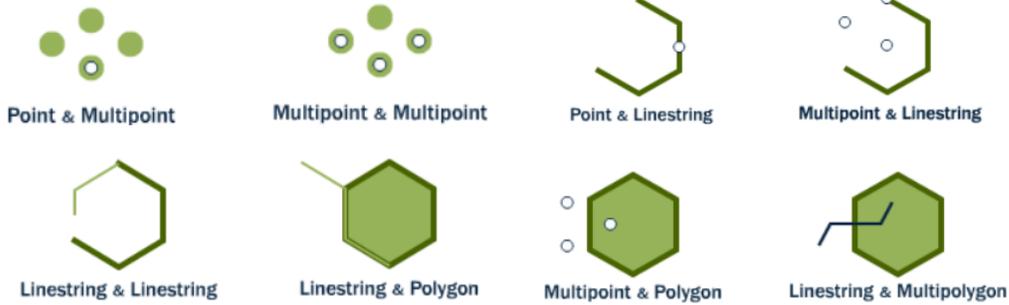




ST_INTERSECTS

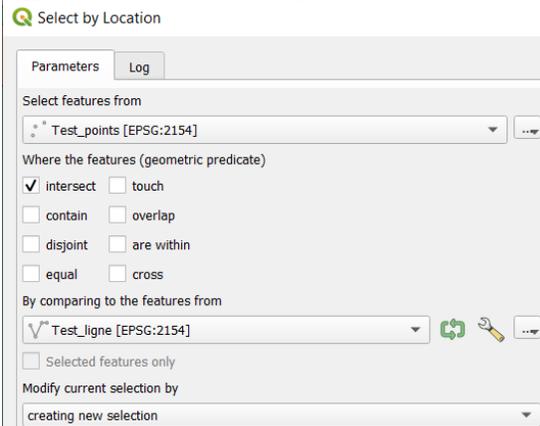
Predicado	Geometría: MP multipoint, L polilínea, S polígono	Condiciones
ST_Intersects	Todas	geom(A) interseca a geom(B) si: * geom(A) y geom(B) tienen al menos un punto en común en el interior o en el límite. * (Inverso de <i>Disjoint</i>)

ST_Intersects:

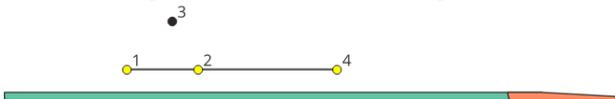


Fuente: https://postgis.net/workshops/postgis-intro/spatial_relationships.html

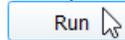
ST_Intersects: Puntos/Líneas



Resultado (selección en amarillo):

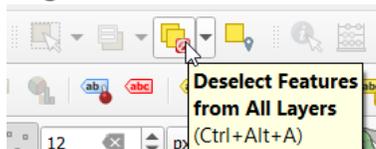


- Seleccionar elementos del geodato de puntos **test_points...**
- Geometric predicate: **intersect**
- By comparing to the features from: test_ligne**
- Siempre vamos a hacer una **nueva selección.**
- Haga **click** en el botón **Run** para hacer esta prueba.



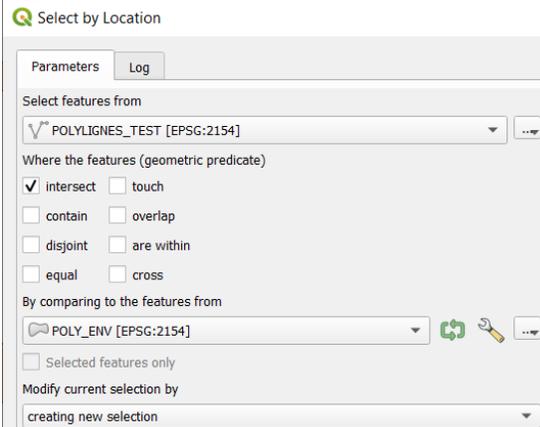
1. Los puntos seleccionados (en amarillo) son el 1, 2, y 4. El punto 3 está fuera de la línea.
2. El predicado **intersect devolverá cierto y seleccionará cuando las geometrías de ambos geodatos tengan al menos un punto en común tanto en el interior, como en el límite o borde.**

- Haga **click** en el botón **Deselect Features** para borrar esta selección.

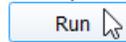




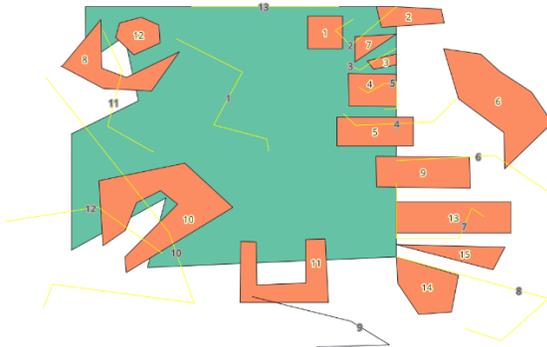
ST_Intersects: Líneas/Polígonos



- Seleccionar elementos del geodato de puntos **polylignes_test...**
- Geometric predicate: **intersect**
- By comparing to the features from: poly_env**
- Siempre vamos a hacer una **nueva selección.**
- Haga **click** en el botón **Run** para hacer esta prueba.

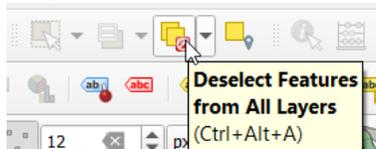


Resultado (selección en amarillo):

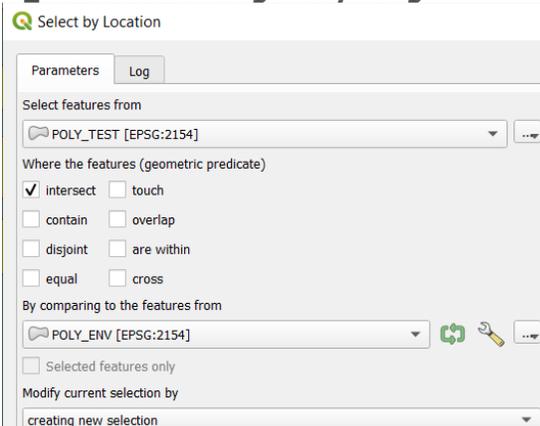


1. Las líneas seleccionadas (en amarillo) son 1, 2, 3, 4, 5, 6, 7, 8, 10, 11, 12 y 13. La línea 9 está fuera de poly_env.
2. El predicado **intersect devolverá cierto y seleccionará cuando las geometrías de ambos geodatos tengan al menos un punto en común tanto en el interior, como en el límite o borde.**

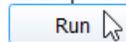
- Haga **click** en el botón **Deselect Features** para borrar esta selección.



ST_Intersects: Polígonos/Polígonos



- Seleccionar elementos del geodato de puntos **poly_test...**
- Geometric predicate: **intersect**
- By comparing to the features from: test_ligne**
- Siempre vamos a hacer una **nueva selección.**
- Haga **click** en el botón **Run** para hacer esta prueba.

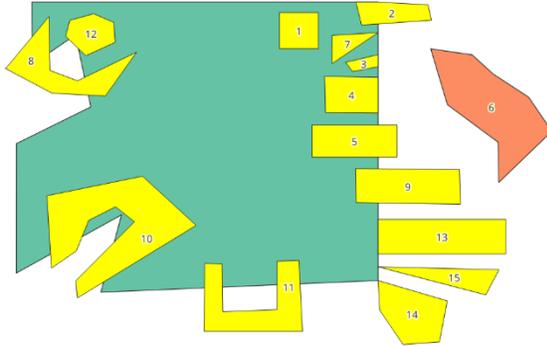


Resultado (selección en amarillo):

1. **Todos** los polígonos del geodato poly_test fueron seleccionados **excepto el número 6** porque está fuera de poly_env.

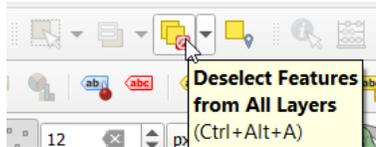


Tutorial PostGIS, 3.x



- El predicado **intersect** devolverá cierto y seleccionará cuando las geometrías de ambos geodatos tengan al menos un punto/vértice en común tanto en el interior, como en el límite o borde.

- Haga **click** en el botón **Deselect Features** para borrar esta selección.



- Cierre** esta sesión de QGIS. No lo usaremos por ahora.

ST_COVERS & ST_COVEREDBY

Estas son dos **funciones propias de PostGIS** entre otras como **ST_DWithin**.

ST_Covers	Todas	geom(A) cubre a geom(B) si: * ningún punto de geom(B) está en el exterior de A * todo punto de geom(B) es un punto de geom(A) Compárese con <i>Contains</i>
ST_CoveredBy	Todas	geom(A) está cubierto por geom(B) si: * ningún punto de geom(A) está en el exterior de geom(B) * todo punto de geom(A) es un punto de geom(B) Compárese con <i>Within</i>

Según la documentación de PostGIS, ellos recomiendan usar **ST_Covers** en lugar de ST_Contains. Su definición es más simple y no tiene la particularidad de que las geometrías no contienen a sus límites. **Estas funciones** al igual que ST_DWithin **no están disponibles en QGIS**.

Los predicados geométricos se explican en el apartado A de este tutorial. Como se mencionó, son 98 situaciones con los tres tipos básicos de geometrías.

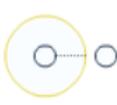
Fuente: https://www.geoinformations.developpement-durable.gouv.fr/fichier/pdf/Predicats_OGC_V3_cle119417.pdf?arg=177828548&cle=baf65cf1b040772ae1d97f164cf9ca67f6dbb698&file=pdf%2FPredicats_OGC_V3_cle119417.pdf

ST_DWITHIN

Esta función/predicado seleccionará elementos geométricos que compartan puntos/vértices en común entre geometrías basado en una distancia de búsqueda. Como se mencionó antes, **funciona como ST_Intersects, pero con un umbral de tolerancia** (distancia).



Point & Point (True)



Point & Point (False)



Polygon & Point (True)



Polygon & Point (False)

Fuente: https://postgis.net/workshops/postgis-intro/spatial_relationships.html

ST_DWithin no es un predicado del OGC. Es una implementación de PostGIS.



Ejemplo:

ST_DWithin para mostrar distancias entre escuelas y carreteras estatales.

Regrese a la interfaz de DBEaver.

- En la pestaña **<gisdb_lab> Script 1**, escriba y ejecute el siguiente query para seleccionar las **escuelas** en Villalba que estén ubicadas hasta **300 metros** de distancia de vías del **sistema estatal de carreteras**. Incluya la **distancia** mediante la función **ST_Distance**.

```

select e.escuela, c.route_id, --traer campos escuela, route_id
st_distance(e.geom,c.geom) as distancia --st_distance() para distancias
FROM g33_dotacional_educacion_escuelas_2021_villalba e, --tabla escuelas as e
g35_viales_carreteras_estatales_segmentadas_agosto_2021 c --tabla carreteras as c
where st_dwithin(c.geom,e.geom,300) --condición: dist entre geometrías <300m
order by c.route_id, distancia; --ordena por route_id asc, luego por distancia asc

```

Grid	escuela	route_id	distancia
1	DANIEL SERRANO RIVERA	PR-149	82.7400391886
2	LYSANDER BORRERO TERRY	PR-149	103.2738510828
3	CRISTINA (AMADA) MARTINEZ MARTINEZ	PR-149	257.3467495876
4	SU HATILLO SILVIA TORRES TORRES	PR-150	101.274528967
5	(CROEV)ESPECIALIZADA EN CIENCIAS Y MATEMATICAS	PR-150	229.2012484341

Le devolverá 19 filas. Note las funciones **ST_Distance**(geom, geom) y **ST_DWithin**(geom, geom). Las escuelas se pueden repetir.

QUERIES CON GEOMETRÍAS:

ST_Touches

Realice una sentencia SQL espacial en la cual se **haga una lista de municipios y cuántos municipios adyacentes tiene cada uno**. El predicado topológico a usar es **ST_Touches**, el cual está explicado en el [Apartado A-2](#). **ST_Touches** se debe usar, ya que seleccionará las **geometrías que comparten bordes, pero no tienen puntos en el interior de otra geometría**. **Agregue los datos (GROUP BY) por el campo de nombres de municipios**. Ordene los datos, **ORDER BY** usando la columna de conteos (posición #2) en el orden de columnas del query.

- En la pestaña **<gisdb_lab> Script 1**, escriba y ejecute el siguiente query:

```

select m1.municipio, --trae columna municipio del geodato m1
count(*) as cuantos_adyacentes --conteo de filas as cuantos_adyacentes
from g03_legales_municipios_2015 as m1, --de la tabla municipios_2015 as m1
g03_legales_municipios_2015 as m2 --hacer copia de municipios_2015 as m2
where st_touches(m1.geom,m2.geom) --predicado topol: que solo compartan bordes
group by m1.municipio --agrupa el conteo de ayacentes por m1.municipio
order by 2 desc --ordena por la columna #2 ó cuantos_adyacentes ó count(*)

```

- Para encontrar **municipios adyacentes copiamos el mismo geodato y le asignamos otro alias (m2)**.
- ST_Touches** es el predicado que necesitamos: **compartir solo bordes**.
- ORDER BY 2** indica que queremos hacer el **ordenamiento por la columna #2** y de forma descendente para que nos devuelva primero los municipios con **mayor cantidad de municipios adyacentes**.

El conteo de filas es 76. **¿Por qué no aparecen los municipios de Vieques ni de Culebra?**



Tutorial PostGIS, 3.x

¿Y si queremos saber cuáles son los nombres de cada municipio adyacente a cada municipio?

La función **string_agg(columna, 'patrón')** se usará para acomodar una lista de ítems de una columna o expresión dentro de una sola fila.

- Por ejemplo, si queremos acomodar una lista de municipios, separados por una coma: En la pestaña **<gisdb_lab> Script 1**, escriba y ejecute el siguiente query:

```

select m1.municipio, --trae columna municipio del geodato m1
count(*) as cuantos_adyacentes, --conteo de filas as cuantos_adyacentes. No olvidar poner coma aquí
string_agg(m2.municipio, ', ') as municipios_adyacentes |--string_agg(municipio y añade una coma ', ' y un espacio)
from g03_legales_municipios_2015 as m1, --de la tabla municipios_2015 as m1
g03_legales_municipios_2015 as m2 --hacer copia de municipios_2015 as m2
where st_touches(m1.geom,m2.geom) --predicado topol: que solo compartan bordes
group by m1.municipio --agrupa el conteo de ayacentes por m1.municipio
order by 2 desc --ordena por la columna #2 ó cuantos_adyacentes ó count(*)

```

municipio	cuantos_adyacentes	municipios_adyacentes
Lares	8	Yauco, Maricao, Las Marías, San Sebastián, Adjuntas, Utuado, Hatillo, Camuy
Orocovis	8	Ciales, Jayuya, Villalba, Juana Díaz, Coamo, Morovis, Corozal, Barranquitas
Las Piedras	7	Canóvanas, Juncos, San Lorenzo, Yabucoa, Humacao, Naguabo, Río Grande
San Sebastián	7	Añasco, Las Marías, Moca, Lares, Camuy, Quebradillas, Isabela

La diferencia es insertar la función **string_agg(m2.municipio, ', ')** luego de añadir una coma después de la nueva columna **cuantos_adyacentes**.

Al igual que el query anterior, aparecerán solo 76 filas.

¿Cuáles son los barrios con mayor número de barrios adyacentes?

Usar la función **ST_Touches(geom, geom)** y el mismo geodato de barrios con dos **alias b1 y b2** para compararlos.

- En la pestaña **<gisdb_lab> Script 1**, escriba y ejecute el siguiente query: Limite la lista a 10 filas. Esté atento a las comillas, espacios y el operador de concatenación: **||**.

```

select b1.municipio || ', Bo.' || b2.barrio as munic_barrio, --trae y concatena columnas municipio y barrio del geodato b1 as munic_barrio
count(*) as cuantos_adyacentes, --conteo de filas as cuantos_adyacentes. No olvidar poner coma aquí
string_agg('Bo.' || b2.barrio || ', ' || b2.municipio || ')', ', ') as barrios_adyacentes --concatenamos: paréntesis+Bo.+barrio+coma_con_espacio+municipio+paréntesis+coma+coma_con_espacio
from g03_legales_barrios_2015 as b1, --de la tabla municipios_2015 as m1
g03_legales_barrios_2015 as b2 --hacer copia de municipios_2015 as m2
where st_touches(b1.geom,b2.geom) --predicado topol: que solo compartan bordes
group by munic_barrio --agrupa el conteo de ayacentes por m1.municipio
order by 2 desc --ordena por la columna #2 ó cuantos_adyacentes ó count(*)
limit 10;

```

munic_barrio	cuantos_adyacentes	barrios_adyacentes
Maricao, Bo.Maricao Afuera	13	(Bo. Rosario Alto, San Germán), (Bo. Hoconuco Alto, San Germán), (Bo. Cain Alto, San Germán), (Bo. Guamá,
Ciales, Bo.Toro Negro	11	(Bo. Mameyes Arriba, Jayuya), (Bo. Pesas, Ciales), (Bo. Cialitos, Ciales), (Bo. Pozas, Ciales), (Bo. Dami
Guaynabo, Bo.Frailes	11	(Bo. Santa Rosa, Guaynabo), (Bo. Minillas, Bayamón), (Bo. Camarones, Guaynabo), (Bo. Pueblo Viejo, Guayna
Naguabo, Bo.Río Blanco	11	(Bo. Antón Ruiz, Humacao), (Bo. Mambiche, Humacao), (Bo. Peña Pobre, Naguabo), (Bo. El Río, Las Piedras),
Arroyo, Bo.Yaurel	11	(Bo. Mulás, Patillas), (Bo. Jagual, Patillas), (Bo. Quebrada Arriba, Patillas), (Bo. Muñoz Rivera, Patill
Arecibo, Bo.Río Arriba	10	(Bo. Caguana, Utuado), (Bo. Santa Rosa, Utuado), (Bo. Hato Viejo, Arecibo), (Bo. Carreras, Arecibo), (Bo.
Utuado, Bo.Caguana	10	(Bo. Ángeles, Utuado), (Bo. Santa Isabel, Utuado), (Bo. Tanamá, Adjuntas), (Bo. Guaónica, Utuado), (Bo. F
Caguas, Bo.Bairoa	10	(Bo. Tomás de Castro, Caguas), (Bo. Barrio Pueblo, Caguas), (Bo. Cañabón, Caguas), (Bo. Bairoa, Aguas Bue
Ciales, Bo.Frontón	10	(Bo. Sabana Hoyos, Arecibo), (Bo. Limón, Utuado), (Bo. Don Alonso, Utuado), (Bo. Mameyes Arriba, Jayuya),
Las Piedras, Bo.Montones	10	(Bo. Barrio Pueblo, Las Piedras), (Bo. Quebrada Arenas, Las Piedras), (Bo. Ceiba, Las Piedras), (Bo. Vale

- **||** Operador para **concatenar** cadenas de caracteres. En la función **string_agg()** *insertamos la cadena de caracteres, 'Bo. ' y concatenamos con || al campo **b2.barrio**, luego insertamos una coma con ||, '|| y luego concatenamos || el campo **b2.municipio**, para después concatenar el paréntesis de cierre || ')' y finalmente añadimos la coma para separar los valores.*
- Puede **sustituir LIMIT 10** con **HAVING count(b2.barrio) >=10**. **HAVING se coloca después de GROUP BY**



Tutorial PostGIS, 3.x

Pregunta: Si el total de barrios es 902, ¿por qué si quitamos la cláusula **LIMIT 10** o **HAVING count(b2.barrio)>=10** devuelve 901? ¿Cuál barrio no tiene barrio(s) adyacentes? Sabemos que Isla de Mona e islote Monito componen un barrio de Mayagüez, pero ¿cómo hacer el query para que devuelva este barrio?

- En la pestaña **<gisdb_lab> Script 1**, escriba y ejecute el siguiente query:

```
*<gisdb_lab> Script-1 X
select b1.municipio || ', Bo.' || b1.barrio as munic_barrio --trae y concatena columnas municipio y barrio del geodato b1 as munic_barrio
from g03_legales_barrios_2015 as b1, --de la tabla barrios_2015 as b1
g03_legales_barrios_2015 as b2 --copia la tabla barrios_2015 as b2
where not exists( --Devuelve fila(s) que quede(n) fuera de esta selección
select b1.municipio || ', Bo.' || b1.barrio as munic_barrio
from g03_legales_barrios_2015 as b2 --hacer copia de municipios_2015 as m2
where st_touches(b1.geom,b2.geom)) --predicado topol: que solo compartan bordes
group by munic_barrio --agrupa el conteo de ayacentes por munic_barrio o campo #1
```

Usar un **sub-select** o **query anidado** para que devuelva la fila que no cumple con el predicado ST_Touches: **where not exists(select expresión from tabla, where ST_Touches(b1.geom,b2.geom)**. Luego GROUP BY agrupa por la columna munic_barrio para que no devuelva 902 filas. Esta es la fila que cumple con la condición: barrio Isla de Mona e Islote Monito.

VISUALIZAR LAS GEOMETRÍAS EN POSTGIS

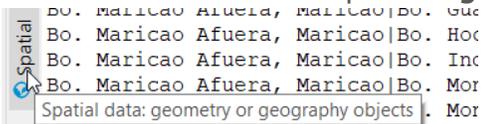
Usemos un query parecido al de búsqueda de barrios adyacentes. **Ya vimos que el barrio Maricao Afuera del Municipio de Maricao** comparte límites con 13 barrios. Para ver estos vecinos sin salir de PostGIS, haremos un query que incluya la geometría.

- En la pestaña **<gisdb_lab> Script 1**, escriba y ejecute el siguiente query.

```
*<gisdb_lab> Script-1 X
select 'Bo.' || b2.barrio || ', ' || b2.municipio as "barrio_de_referencia:", --traer barrio y munic de referencia, use "comillas" para incluir :
'Bo.' || b1.barrio || ', ' || b1.municipio as "barrios_adyacentes:", -- traer barrios y munic adyacentes, use "comillas" para incluir :
b1.geom -- trae la geometría del geodato de barrios, versión 1
from g03_legales_barrios_2015 b1, --tabla barrios_2015 as b1
g03_legales_barrios_2015 b2 --tabla barrios_2015 as b2
where st_touches(b1.geom,b2.geom) --predicado st_touches, solo barrios que comparten límites
and b2.barrio = 'Maricao Afuera' --además que sea el barrio 'Maricao Afuera'
order by 2; --ordena por la columna #2
```

Al traer la columna **b1.geom**, convertimos el resultado del query en un geodato temporal que podremos visualizar si usamos la pestaña **Spatial**.

- Luego que devuelva las 13 filas, haga **click** en la pestaña **Spatial** al margen izquierdo del listado devuelto en la pestaña **g03_legales_barrios_2015 1**





Tutorial PostGIS, 3.x

- Al activar la pestaña **Spatial**, podemos hacer **click** en cualquier elemento geométrico.

The screenshot shows the QGIS interface. In the top window, a SQL query is entered in the Script 1 editor. The query selects adjacent barrios from a table named 'g03_legales_barrios_2015'. The map view below shows a geographic area with a popup window for a selected geometry. The popup displays the geometry name 'geom' and its properties: 'barrio_de_referencia: Bo. Maricao Afuera, Maricao' and 'barrios_adyacentes: Bo. Palma Escrita, Las Marías'.

MOSTRAR ÁREA, PERÍMETRO Y TIPO DE GEOMETRÍA

Las propiedades de las geometrías son guardadas internamente. No son campos calculados como aparecería en los shapefiles, por ejemplo.

Muestre una lista con los nombres de barrios del Municipio de Villalba, el **área**, así como sus **perímetros** y **tipo de geometría**. Ordene los resultados por área de manera descendente. En la cláusula **ORDER BY** puede **usar un número de referencia de la columna** en vez del nombre de la columna.

- En la pestaña **gisdb_lab** **Script 1**, escriba y ejecute el siguiente query.

The screenshot shows the QGIS interface with a SQL query in the Script 1 editor. The query selects the name of the barrio, its area in square kilometers, its perimeter in kilometers, and its geometry type from the table 'g03_legales_barrios_2015_villalba', ordered by area in descending order. Below the editor, the results are displayed in a table view.

	barrio	area_sqkm	perim_km	tipo_geometría
1	Caonillas Arriba	20.9428427248	26.12686903	ST_MultiPolygon
2	Villalba Arriba	17.6469141407	19.6906081145	ST_MultiPolygon
3	Caonillas Abajo	16.4577126574	18.673760813	ST_MultiPolygon
4	Hato Puerto Arriba	13.3730023937	31.5976338459	ST_MultiPolygon
5	Vacas	12.1964044741	16.405680526	ST_MultiPolygon
6	Hato Puerto Abajo	7.8200882754	15.28077986	ST_MultiPolygon
7	Villalba Abajo	7.0668390792	12.7572214562	ST_MultiPolygon
8	Barrio Pueblo	0.3723566376	3.6007782863	ST_MultiPolygon

¿Puede haber distintos tipos de geometrías en una misma tabla geoespacial en PostGIS?

La respuesta es sí. Un campo tipo geometría o geografía puede estar compuesto por todas las variantes de POINT/POLYLINE/POLYGON y además **GeometryCollection**. Al momento, los SIG de interfaz gráfica tipo desktop, no dan soporte a las GeometryCollection y solo trabajan con un tipo de geometría a la vez.



Más adelante haremos overlay de geodatos. **¿Es posible que el cruce/overlay de dos geodatos nos devuelva distintos tipos de geometrías (GeometryCollection)? La respuesta es sí.** Eso puede ser un problema que atenderemos más adelante.

ST_LENGTH

¿Cuál es el kilometraje total de las **calles municipales** del Municipio de Villalba?

Antes, vamos a hacer una búsqueda de **cuáles son los códigos que usaremos como criterio de selección** de lo que son las **vías municipales, según los códigos censales** para vías (2006). Este archivo de viales se originó en 2006, pero se añadieron vías adicionales usando la imagen satelital de Google Maps en diciembre de 2023.

Debemos saber primero los códigos CFCC (Census Feature Class Codes, 2016).

- En la pestaña **<gisdb_lab> Script 1**, escriba y ejecute el siguiente query:

```
select distinct cfcc --traer valores únicos del campo cfcc
from g35_viales_mapa_base_tiger_rds_2006se_villalba --de esta tabla
order by 1; --ordena ascendente por la primera columna/campo
```

cfcc
A31
A35
A41
A51
A63

SELECT DISTINCT para devolver los códigos cfcc sin repeticiones.

Debe darnos 5 filas. [Según uno de los artículos de knowledge base de Esri](#), los códigos censales de vías que necesitamos son los tipo **A41** y, **A51**. Los demás **A31**, **A35** y **A63** están asociados con carreteras del sistema estatal.

**** Estamos simplificando este asunto****

Puede haber vías municipales que son mantenidas por la Autoridad de Carreteras y Transportación (ACT). En el caso de Villalba, en la versión de carreteras estatales 2021, no hay vías municipales mantenidas por la ACT.

Con esa información podemos redactar la sentencia (query) para contestar la pregunta: **Dar un estimado de kilometraje en vías municipales, mostrando el código censal cfcc y la suma de kilómetros para cada clase cfcc**, para el **Municipio de Villalba**. En este caso, vamos a sumar la longitud de las geometrías con la función **sum(ST_Length(geom))/1000** para que lo convierta en **kilómetros**, ya que la unidad de medida del geodato es el **metro**. Solo vamos a sumar las vías municipales, así que **en la cláusula WHERE** usaremos **cfcc IN ('A41','A51')**. En la cláusula **GROUP BY** vamos a **agrupar/sumar por** categoría **cfcc**.

- En la pestaña **<gisdb_lab> Script 1**, escriba y ejecute el siguiente query:



Tutorial PostGIS, 3.x

```

* <gisdb_lab> Script-1 x
select cfcc, --traer valores del campo cfcc
sum(st_length(geom)/1000) as kms --calcula sumatoria de longitudes geométricas en kms
from g35_viales_mapa_base_tiger_rds_2006se_villalba --de esta tabla
where cfcc in ('A41','A51') --condición: solo valores cfcc=A41,A51
group by cfcc --agrupar sumas de valores usando el campo cfcc
order by 1; --ordena ascendente por la primera columna/campo

```

cfcc	kms
A41	236.44279431139271
A51	16.9653976931371741

En este query utilizamos:

- **GROUP BY cfcc** para sumar por categoría.
- **WHERE cfcc IN ('A41', 'A51')** para seleccionar solamente las **vías municipales**.

Las vías **cfcc=A51** son por lo regular caminos privados sin asfaltar. Pueden variar en longitud. En fin, hay alrededor de **236 kilómetros en vías municipales (A41)** en este municipio.

ROUND()

Es una función de uso general para redondear números con decimales. Usaremos el query anterior para aplicar la función **round()** al cómputo de sumatoria de longitudes de las geometrías de vías en Villalba. El valor **ST_Length** devuelto es del [tipo de dato numérico double precision](#), dato numérico floating point, no exacto.

Según la [documentación de funciones matemáticas en PostgreSQL](#), la función round():

round (numeric) → numeric round (double precision) → double precision Rounds to nearest integer. For numeric, ties are broken by rounding away from zero. For double precision, the tie-breaking behavior is platform dependent, but "round to nearest even" is the most common rule. round(42.4) → 42	Puede recibir un dato tipo numeric y devuelve el valor entero más próximo. Puede recibir un tipo de dato double precision y devuelve el valor entero más próximo. En esta variante de round() no se usa el argumento <i>s</i> integer para añadir lugares decimales.
round (v numeric, s integer) → numeric Rounds v to s decimal places. Ties are broken by rounding away from zero. round(42.4382, 2) → 42.44 round(1234.56, -1) → 1230	Si usamos round(v numeric, s integer) el valor de entrada debe ser tipo numérico para poder aplicar el número entero que dará la cantidad de lugares decimales. Esta variante de round() debe recibir como entrada valores del tipo de dato <i>numeric</i> .

Por lo tanto, para aplicar round a la suma de longitudes extraídas de las geometrías, debemos cambiar el tipo de dato double precision a numeric. Esto se hace con la función **CAST campo_expresión AS NUMERIC** o usar su equivalente más corto **::**.

Usemos el query anterior para aplicar la función **round()** con dos lugares decimales.

- En la pestaña **<gisdb_lab> Script 1**, modifique y ejecute el pasado query:

```

* <gisdb_lab> Script-1 x
select cfcc, --traer valores del campo cfcc
round(sum(st_length(geom)/1000)::numeric,2) as kms --conv a numeric la sum de kms,usa round a 2 lug decimales
from g35_viales_mapa_base_tiger_rds_2006se_villalba --de esta tabla
where cfcc in ('A41','A51') --condición: solo valores cfcc=A41,A51
group by cfcc --agrupar sumas de valores usando el campo cfcc
order by 1; --ordena ascendente por la primera columna/campo

```

cfcc	kms
A41	236.44
A51	16.97

En la segunda línea de código, según el orden por paréntesis de adentro hacia afuera: **1:** obtener las longitudes del campo geom, **2:** luego sumarlas, **3:** dividir entre 1,000 las sumas para obtener kms, **4:** cambiar las sumas en tipo double precision **:: numeric**, **5:** redondear el dato numérico a 2 lugares decimales.



Tutorial PostGIS, 3.x

ST_DISTANCE

Mostrar las distancias entre cada plaza pública municipal. El geodato incluye otras plazas públicas. Este query va a devolver miles de filas, ya que estará comparando distancias de cada plaza pública “de recreo” con todas las demás plazas. **La función ST_Distance(geom, geom) puede abreviarse (geom<->geom).** En este caso, devolveremos distancias en kilómetros.

- En la pestaña **<gisdb_lab> Script 1**, escriba y ejecute el siguiente query:

```
*<gisdb_lab> Script-1 X
select plz1.nombre||', '||plz1.municipio as plaza_origen, --traer y concatenar nombre_plaza + , + municipio1 as plaza_origen
plz2.nombre||', '||plz2.municipio as plaza_destino, --traer y concatenar nombre_plaza2 + , + municipio2 as plaza_destino
round((plz1.geom<->plz2.geom)::numeric /1000,3) as dist_kms --obtener dists entre plz1 y plz2:::conv a num, div/1,000, round a 3 decimales, as dist_kms
from g33_dotacional_plazas_publicas_2010 plz1, --de esta tabla alias plz1
g33_dotacional_plazas_publicas_2010 plz2 -- copiar la tabla alias plz2
where plz1.gid<plz2.gid --condición: no devolver la misma plaza
order by plz1.municipio, dist_kms --ordena por municipio, y luego la plaza más cercana
```

Results 1 X

```
select plz1.nombre||', '||plz1.municipio as plaza_origen, --traer y concatenar nombre_plaza + , + municipio1 as plaza_origen
plz2.nombre||', '||plz2.municipio as plaza_destino, --traer y concatenar nombre_plaza2 + , + municipio2 as plaza_destino
round((plz1.geom<->plz2.geom)::numeric /1000,3) as dist_kms --obtener dists entre plz1 y plz2:::conv a num, div/1,000, round a 3 decimales, as dist_kms
from g33_dotacional_plazas_publicas_2010 plz1, --de esta tabla alias plz1
g33_dotacional_plazas_publicas_2010 plz2 -- copiar la tabla alias plz2
where plz1.gid<plz2.gid --condición: no devolver la misma plaza
order by plz1.municipio, dist_kms --ordena por municipio, y luego la plaza más cercana
```

plaza_origen	plaza_destino	dist_kms
Plaza de Recreo, Adjuntas	Plaza de Recreo, Utuado	11.637
Plaza de Recreo, Adjuntas	Plaza de Recreo, Peñuelas	11.644
Plaza de Recreo, Adjuntas	Plaza Castañer, Lares	11.941
Plaza de Recreo, Adjuntas	Plaza de Recreo, Jayuya	15.180
Plaza de Recreo, Adjuntas	Plaza de Recreo, Guayanilla	17.529
Plaza de Recreo, Adjuntas	Plaza de Recreo, Yauco	19.364
Plaza de Recreo, Adjuntas	Plaza de Recreo, Ponce	20.338
Plaza de Recreo, Adjuntas	Plaza de Recreo, Lares	21.943
Plaza de Recreo, Adjuntas	Plaza de Recreo, Villalba	24.701
Plaza de Recreo, Adjuntas	Plaza de Recreo, Juana Díaz	26.001
Plaza de Recreo, Adjuntas	Plaza de Recreo, Sabana Grande	26.688
Plaza de Recreo, Adjuntas	Plaza de Recreo, Maricao	27.237

Refresh Save Cancel Export data 200 6,972 200 row(s) fetched - 0.047s, on 2024-02-21 at 11:14:32

Calculate total row count BOT en

El query hará 6,972 comparaciones de distancia en cada una y el resto de las plazas. Si queremos limitar la comparación de distancias entre plazas, digamos, las primeras más cercanas, debemos hacer un subquery o query anidado, donde se limite la comparación al término que uno desee. Esto lo veremos más adelante en la sección de Subqueries.



ST_XMIN, ST_YMIN

Haga una lista de municipios y barrios que están al extremo sur de PR. Repita el procedimiento para encontrar los municipios con su barrio más al oeste de PR.

Las funciones que vamos a usar son `st_ymin(geom)`, `st_xmin(geom)`. Recuerde que estamos en un plano cartesiano donde las coordenadas Y aumentan de sur hacia el norte y que las coordenadas X aumentan de oeste hacia el este. **Ordene las filas usando order by st_ymin(geom)** y luego por `st_xmin(geom)`. **Limite los resultados solo a 10 filas.**

Municipios y barrios al extremo sur:

- En la pestaña `<gisdb_lab> Script 1`, escriba y ejecute el siguiente query:

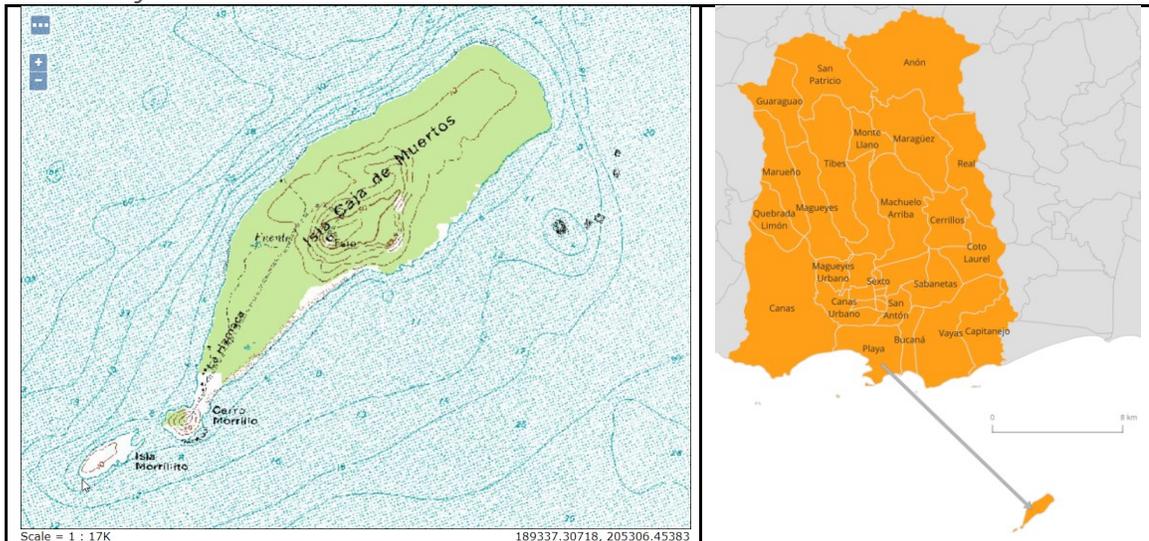
```

select barrio,municipio, st_ymin(geom) --trae campos barrio,municipio,y_mínimo
from g03_legales_barrios_2015 -- de la tabla de barrios_2015
order by 3 asc --ordena por campo #3: st_ymin(geom) ascendente
limit 10; --trae solo 10 las primeras 10 filas

```

barrio	municipio	st_ymin
Playa	Ponce	205305.3163
Aguirre	Salinas	208831.7663
Felicia I	Santa Isabel	209515.0259
Jobos	Guayama	209679.5433
Río Cañas Abajo	Juana Díaz	209962.391
Jauca I	Santa Isabel	210194.54
Montalva	Guánica	210426.4219
Llanos Costa	Cabo Rojo	210989.2501
Boquerón	Cabo Rojo	211138.6875
Machete	Guayama	211603.6226

El barrio más al extremo sur es el barrio Playa, en Ponce. Este barrio incluye la isla Caja de Muertos y la isla Morrillito



El punto más al sur, sin salir de la costa debe ser Punta Brea en el barrio Montalva en Guánica.

Si desea obtener los municipios más **al extremo norte**, sustituya la función por `st_ymax(geom)` con orden descendente.



Municipios y barrios al extremo oeste:

- En la pestaña <gisdb_lab> Script 1, escriba y ejecute el siguiente query:

```
*<gisdb_lab> Script-1 x
select barrio,municipio, st_xmin(geom) --trae campos barrio,municipio,x_mínimo
from g03_legales_barrios_2015 -- de la tabla de barrios_2015
order by 3 asc --ordena por campo #3: st_xmin(geom) ascendente
limit 10; --trae solo 10 las primeras 10 filas
```

barrio	municipio	st_xmin
Isla de Mona e Islote Monito	Mayagüez	39380.1016
Sabanetas	Mayagüez	88270.1459
Puntas	Rincón	111423.5469
Ensenada	Rincón	112165.3107
Pueblo	Rincón	113270.6046
Barrio Pueblo	Rincón	113299.0906
Calvache	Rincón	113997.2824
Río Grande	Rincón	114010.6636
Barrero	Rincón	114640.7176
Río Grande	Aguada	114805.8168

Observe que se trata del mismo query con la diferencia de sustituir por **st_xmin(geom)**. En este caso el orden es ascendente (por defecto, está implícito)

El municipio más al extremo oeste es Mayagüez, por el barrio Isla de Mona e islote Monito, seguido por el barrio Sabanetas, el cual incluye la isla Desecheo.

El punto más al oeste sin salir de la costa debe estar en Punta Higüero en el barrio Puntas del Municipio de Rincón.

ST_TRANSFORM

Transformar coordenadas:

¿Cómo expresamos las coordenadas planas del sistema que estamos usando ([EPSG:6566](#)) a coordenadas geográficas en sistema decimal con datum WGS84 ([EPSG:4326](#))? Para esto podemos usar la función **st_transform(geom, epsg_code)** para hacer la transformación de coordenadas sin tener que generar otro archivo geográfico. Por ejemplo:

- En la pestaña <gisdb_lab> Script 1, escriba y ejecute el siguiente query:

```
*<gisdb_lab> Script-1 x
select barrio,municipio, --traer barrio, municipio
st_ymax(
  st_transform(geom,4326) --primero st_transform geom desde epsg:6566 a 4326 (WGS84 lat long)
) as ymax -- luego ejecuta st_ymax para obtener coord más al norte, as ymax
from g03_legales_barrios_2015 --de la tabla de barrios_2015
order by ymax desc --ordena usando campo ymax, descendente
limit 10; --limita las filas a 10
```

barrio	municipio	ymax
Bajura	Isabela	18.515968727024617
Maleza Alta	Aguadilla	18.51335215921861
Aguacate	Aguadilla	18.51313300812173
Montaña	Aguadilla	18.512712201841573
Guayabos	Isabela	18.511676920269302
Maleza Baja	Aguadilla	18.510578613912248
Jobos	Isabela	18.510323027598414
Bejucos	Isabela	18.509521121061923
Barrio Pueblo	Isabela	18.50890802940048
Borinquen	Aguadilla	18.500098187279303

El municipio más norteño es Isabela, en el barrio Bajura.



GROUP BY (EJEMPLO TRIVIA)

Aunque este query no utiliza ninguna función geoespacial, podemos usar **GROUP BY** con las funciones **string_agg()** y **left()** para devolver una **lista de municipios por letra inicial**.
¿Cuántos comienzan con A, cuáles con B, etcétera?

- En la pestaña **<gisdb_lab> Script 1**, escriba y ejecute el siguiente query:

```
*<gisdb_lab> Script-1 X
select left(municipio,1) as letra_inicial, --trae primera letra del municipio
count(municipio) as cuántos, --conteo de municipios
string_agg(municipio, ' ') as municipios --trae lista municipios en solo una fila
from g03_legales_municipios_2015 --de la tabla municipios_2015
group by letra_inicial --agrupa por campo letra_inicial
order by 2 desc; --ordena por el campo #2, descendente
```

Results 1 X

```
select left(municipio,1) as letra_inicial, c
```

letra_inicial	cuántos	municipios
C	14	Cabo Rojo, Caguas, Camuy, Canóvanas, Carolina, Cataño,
A	8	Adjuntas, Aguada, Aguadilla, Aguas Buenas, Aibonito, Ar
S	7	Sabana Grande, Salinas, San Germán, San Juan, San Loren
M	6	Manatí, Maricao, Maunabo, Mayagüez, Moca, Morovis
L	6	Lajas, Lares, Las Marías, Las Piedras, Loíza, Luquillo
G	5	Guayama, Guayanilla, Guaynabo, Gurabo, Guánica
V	4	Vega Alta, Vega Baja, Vieques, Villalba
B	3	Barceloneta, Barranquitas, Bayamón
J	3	Jayuya, Juana Díaz, Juncos
P	3	Patillas, Peñuelas, Ponce

Refresh Save Cancel Export data 200

21 row(s) fetched - 0.001s, on 2024-02-22 at 09:34:20

Esta estructura del query puede adaptarse al uso de códigos identificadores y hacer resúmenes basados en partes de un código-ID.

SPATIAL JOINS

Este tipo de enlace (join) entre tablas geoespaciales es bastante común y no es necesario que haya un identificador común para unir las dos tablas. En muchos casos, podemos intercambiar el contenido de la cláusula WHERE e insertarlo en la parte de JOIN tabla ON *condición*.

Por ejemplo, devolver los sectores que están en cada barrio. El geodato de sectores tiene el barrio, pero podemos usar el predicado **ST_DWithin(geom, geom, distancia)** para mostrar cuáles son los sectores dentro de cada barrio del Municipio de Villalba. La distancia establecida en ST_DWithin no es significativa, solo para asignar cierto umbral de tolerancia. Igualmente se puede usar **ST_Intersects**, **ST_Covers** o **ST_CoveredBy** en la consulta.



Lista de sectores por barrio en Villalba.

- En la pestaña **Script-1**, escriba y ejecute las siguientes consultas separadamente.

Query usando WHERE:

```

*gisdb_lab> Script-1 x
select b.barrio, s.nombre as sectores --trae barrio-tablaBarrios, nombre de sectores
from g25_asentamientos_sectores_2017_villalba s, --de la tabla sectores, alias s
g03_legales_barrios_2015_villalba b --y de la tabla barrios_2015, alias b
where st_dwithin(s.geom,b.geom,0.001) --predicado st_dwithin() ambas geometrías deben
--intersecarse dentro de una distancia de 0.001 metros
order by b.barrio,s.nombre; -- ordena por barrio asc y por nombre de sector asc

```

g03_legales_barrios_2015_villalba(+)

```

select b.barrio, s.nombre as sectores fro

```

barrio	sectores
Barrio Pueblo	Borinquen
Barrio Pueblo	Cooperativa
Barrio Pueblo	Efraín Suárez
Barrio Pueblo	San Cristóbal

Query usando JOIN tabla ON condición:

```

*gisdb_lab> Script-1 x
select b.barrio, s.nombre as sectores --trae barrio-tablaBarrios, nombre de sectores
from g25_asentamientos_sectores_2017_villalba s --de la tabla sectores, alias s, -no usar coma
join g03_legales_barrios_2015_villalba b --JOIN de la tabla barrios_2015, alias b
on st_dwithin(s.geom,b.geom,0.001) --ON:usando el predicado st_dwithin() ambas geometrías deben
--intersecarse dentro de una distancia de 0.001 metros
order by b.barrio,s.nombre; --ordena por barrio asc y por nombre de sector asc

```

g03_legales_barrios_2015_villalba(+)

```

select b.barrio, s.nombre as sectores fro

```

barrio	sectores
Barrio Pueblo	Borinquen
Barrio Pueblo	Cooperativa
Barrio Pueblo	Efraín Suárez

En lugar de usar identificadores entre tablas, el JOIN se basa en la coincidencia espacial (si ambas geometrías intersecan dentro de 0.001 metros ST_DWithin(s.geom,b.geom, 0.001). El orden de geometrías no hace diferencia porque trabaja como st_intersects y es conmutativo. En ambas queries se devuelve el mismo número de filas y el tiempo de respuesta es prácticamente el mismo.

LEFT JOIN

Usar JOIN puede tener otra utilidad. El tipo de join por defecto es INNER JOIN y está implícito (no hay que escribirlo). Este tipo de JOIN devuelve la **intersección** de los conjuntos de filas que parean entre ambas tablas. Las que no pareen en la tabla izquierda no aparecerán en el resultado del query. Por otro lado, si usamos **LEFT JOIN**, preservamos todas las filas de la tabla izquierda (geodato de barrios).



Tutorial PostGIS, 3.x

Ejemplo usando INNER JOIN y LEFT JOIN: Enumerar barrios que tienen escuelas públicas.

- Escriba estos queries separadamente, en la pestaña **Script-1** de DBeaver:

INNER JOIN:

```
*gisdb_lab> Script-1 x
select b.barrio, e.escuela --trae barrio-tablaBarrios, nombre de escuelas
from g03_legales_barrios_2015_villalba b --de la tabla barrios, alias b, -no usar coma-
join g33_dotacional_educacion_escuelas_2021_villalba e --JOIN de la tabla escuelas, alias e
on st_dwithin(e.geom,b.geom,0.001) --ON:usando el predicado st_dwithin() ambas geometrias
--deben intersectarse dentro de una distancia de 0.001 metros
order by b.barrio,e.escuela; --ordena por barrio asc y por nombre de sector asc
```

```
g03_legales_barrios_2015_villalba(+ 1 x
select b.barrio, e.escuela from g03_lega
```

barrio	escuela
Barrio Pueblo	FRANCISCO ZAYAS SANTANA
Hato Puerco Arriba	CRISTINA (AMADA) MARTINEZ MARTINEZ
Hato Puerco Arriba	CROEV: ESPECIALIZADA EN CIENCIAS Y MATEMATICAS
Hato Puerco Arriba	ISABEL ALVARADO ALVARADO
Hato Puerco Arriba	NORMA I TORRES COLON
Hato Puerco Arriba	SU HATILLO SILVIA TORRES TORRES
Vacas	RAMON LOPEZ BERRIOS
Villalba Abajo	DANIEL SERRANO RIVERA
Villalba Arriba	LYSANDER BORRERO TERRY

Devuelve 9 filas, ya que solo hay 9 escuelas en Villalba (datos de 2021). No aparecen los barrios Hato Puerco Abajo, Caonillas Arriba, Caonillas Abajo.

LEFT JOIN:

```
*gisdb_lab> Script-1 x
select b.barrio, e.escuela --trae barrio-tablaBarrios, nombre de escuelas
from g03_legales_barrios_2015_villalba b --de la tabla barrios, alias b, -no usar coma-
left join g33_dotacional_educacion_escuelas_2021_villalba e --LEFT JOIN de la tabla escuelas, alias e
on st_dwithin(e.geom,b.geom,0.001) --ON:usando el predicado st_dwithin() ambas geometrias
--deben intersectarse dentro de una distancia de 0.001 metros
order by b.barrio,e.escuela; --ordena por barrio asc y por nombre de sector asc
```

```
g03_legales_barrios_2015_villalba(+ 1 x
select b.barrio, e.escuela from g03_lega
```

barrio	escuela
Barrio Pueblo	FRANCISCO ZAYAS SANTANA
Caonillas Abajo	
Caonillas Arriba	
Hato Puerco Abajo	
Hato Puerco Arriba	CRISTINA (AMADA) MARTINEZ MARTINEZ
Hato Puerco Arriba	CROEV: ESPECIALIZADA EN CIENCIAS Y MATEMATICAS
Hato Puerco Arriba	ISABEL ALVARADO ALVARADO
Hato Puerco Arriba	NORMA I TORRES COLON
Hato Puerco Arriba	SU HATILLO SILVIA TORRES TORRES
Vacas	RAMON LOPEZ BERRIOS
Villalba Abajo	DANIEL SERRANO RIVERA
Villalba Arriba	LYSANDER BORRERO TERRY

Devuelve 12 filas. Aparecen todos los barrios, independientemente tengan escuelas o no. Útil para saber cuáles barrios no tienen escuelas. Esto se puede hacer con **sectores censales** y añadir datos de población en edad escolar para saber dónde están concentradas estas edades.

En el query anterior puede hacerse una lista de los ocho barrios y concatenando los nombres de escuelas usando **LEFT JOIN, GROUP BY**, la función [string_agg\(\)](#) y el predicado [ST_Covers](#).



Por ejemplo:

- Escriba y ejecute el siguiente query en la pestaña **Script-1 de DBeaver**:

```

select b.barrio, --trae barrio-tabla barrios
count(e.gid) as cuantas_escuelas, --nm de escuelas as cuantas_escuelas
string_agg(e.escuela,', ') as escuelas --lista de escuelas, separadas por coma ', '
from g03_legales_barrios_2015_villalba b --de la tabla barrios, alias b, -no usar coma-
left join g33_dotacional_educacion_escuelas_2021_villalba e --JOIN de la tabla escuelas, alias e
on st_covers(b.geom,e.geom) --ON:usando el predicado st_covers(barrios cubren las escuelas)
group by b.barrio --agrupa conteos por barrio
order by 2 desc,1; --ordena por nmero de escuelas, desc, luego por barrio, asc

```

Grid	barrio	cuantas_escuelas	escuelas
1	Hato Puerco Arriba	5	NORMA I TORRES COLON, SU HATILLO SILVIA TORRES TORRES, ISABEL ALVARADO A
2	Barrio Pueblo	1	FRANCISCO ZAYAS SANTANA
3	Vacas	1	RAMON LOPEZ BERRIOS
4	Villalba Abajo	1	DANIEL SERRANO RIVERA
5	Villalba Arriba	1	LYSANDER BORRERO TERRY
6	Caonillas Abajo	0	[NULL]
7	Caonillas Arriba	0	[NULL]
8	Hato Puerco Abajo	0	[NULL]

- Note que podemos usar el predicado **st_covers(geom, geom)** para comparar polgonos con puntos. Barrios que ‘cubren’ escuelas.
- Usamos **GROUP BY** para hacer el conteo de escuelas por barrio.
- **LEFT JOIN**: Los ltimos tres barrios no tienen escuelas pblicas.

SUBQUERIES:

En esta parte, traemos la idea de un query anidado dentro de otro para poder resolver un problema de dos maneras.

ST_DISJOINT Y SUBQUERIES

Ejemplo: Queremos saber **cules son los municipios que NO tienen reas naturales protegidas terrestres**, segn el **geodato de reas naturales protegidas terrestres de 2019 (ANP)**.

En PostGIS, el predicado **st_disjoint(geom, geom)** es costoso, ya que compara todas las geometras de A con el exterior de todas las geometras de B. Por lo tanto, devolver mltiples filas comparadas.

Para que PostGIS nos devuelva solo los municipios sin ANP terrestres, tendremos que combinar/unir todas sus geometras en un subquery, y con esta geometra agregada o total, se hace la comparacin con el geodato de municipios. Esta manera es menos eficiente, pero es para demostrar el uso del predicado **st_disjoint** y cmo **hacer un subquery** dentro de otro.

- Escriba y ejecute el siguiente query en la pestaña **Script-1 de DBeaver**:



Tutorial PostGIS, 3.x

```

*gisdb_lab> Script-1 X
select municipio --trae nombres de municipios
from g03_legales_municipios_2015 m --de la tabla municipios_2015 as m
join (select st_union(geom) as geom --st_union para unir todas las
--geometrias entre uno o dos geodatos
from g11_conserv_areas_naturales_protegidas_terrestres_2019) as anp
/* en la parte entre paréntesis (subquery) se realiza la unión
* de todas las geometrias st_union(geom) geodato ANP. De lo contrario
* PostGIS hará el pareo de cada municipio con las distintas
* geometrias del geodato ANP
* el resultado de este subquery debe tener alias y se llama "anp".
*/
on st_disjoint(m.geom,anp.geom);--ahora que tendremos una sola geometria
--podemos compararlas

```

Grid	municipio
-----+	
Aguada	
Arroyo	
Rincón	

Como se mencionó, usaremos un subquery. Dentro de este subquery se realizará la unión `st_union(geom)` de las geometrías de las ANP. Los subqueries se encierran entre paréntesis. Si queremos ver las geometrías en DBEaver, inserte los campos `m.geom` y `anp.geom`.

`select municipio, m.geom,anp.geom`
Tendrá que esperar un rato (30 segundos o más) para que DBEaver produzca el resultado y que haga después el despliegue. Las áreas en azul son los municipios sin ANP terrestres, 2019.



Una alternativa al uso de `st_disjoint` y más eficiente puede ser la siguiente: Este query devuelve los mismos datos, aprovechando que `st_intersects` es el "inverso" del predicado `st_disjoint`.

□ Escriba y ejecute el siguiente query en la pestaña **Script-1 de DBEaver**:

```

*gisdb_lab> Script-1 X
select m.municipio --trae nombre municipios
from g03_legales_municipios_2015 m --tabla municipios_2015 alias m
where not exists --trae filas que NO devuelvan TRUE en el subquery()
(select --no es necesario escribir columnas aquí
from g11_conserv_areas_naturales_protegidas_terrestres_2019 anp
--de la tabla ANP2019 alias anp
where st_intersects(m.geom,anp.geom) --donde intersecan geoms
--st_intersects(municipio.geom,anp.geom)
); -- ) fin del subquery, ; fin del query

```

Grid	municipio
-----+	
Aguada	
Arroyo	
Rincón	

- En otras palabras, **el subquery** devuelve las filas donde no exista intersección (que no haya puntos en común entre ambas geometrías).
- **No fue necesario hacer la unión de geometrías** como en el query y subquery anterior. Por lo tanto, el query finaliza en 3 segundos.

Esto demuestra que puede haber varias maneras de resolver un problema o situación. Se recomienda que busquemos información de las propiedades de los predicados topológicos: para qué están diseñados (propósito), si usan algún índice para agilizar búsquedas, y si hay otras alternativas de predicados más eficientes. En este y muchos casos `ST_Intersects` se ha estado mejorando constantemente para hacerlo más eficiente y es uno de los predicados más utilizados.

CROSS JOIN LATERAL, ST_DISTANCE

En muchas ocasiones queremos saber las distancias entre geometrías para poder hacer análisis posteriores. Basado en el ejemplo que hicimos con [ST_Distance](#), en donde haremos un listado de plazas de recreo y las distancias de cada plaza con cada una de las otras plazas de recreo.

Antes de la llegada de los sistemas de información geográfica y los sistemas automatizados de rutas, las distancias de un pueblo a otro se medían desde una plaza de recreo a otra plaza



Tutorial PostGIS, 3.x

en un recorrido de distancia euclidiana (línea recta). Ahora con Google Maps es mucho más simple y accesible. PostGIS tiene la extensión [pgRouting](#) para estos fines, pero su explicación va más allá del objetivo de este tutorial.

De todos modos, para mostrar el uso de **ST_Distance**, hagamos este query. Debo mencionar que el [operador <->](#) puede sustituir el uso de **ST_Distance**. Para no hacer una tabla demasiado grande e ilegible, vamos a limitar el número de filas a devolver. Usaremos además [CROSS JOIN LATERAL](#) en un subquery para hacer la comparación de distancias entre cada plaza.

- Escriba y ejecute el siguiente query en la pestaña **Script-1 de DBeaver**:

```

select plz.nombre||', '||plz.municipio as plaza_origen,
--concatena cols nombre y municipio usando ||, || tabla alias plz
plz2.plaza as plaza_destino, --trae el campo plaza de la tabla alias plz2
round(plz2.dist_kms::numeric,2) as kms --trae las distancias en kms de la tabla plz2
--redondéalos a dos lugares decimales
from g33_dotacional_plazas_publicas_2010 plz --tabla plazas as plz
cross join lateral ( --cross join lateral compara una fila con todas las otras de plz2
select plz2.gid, --trae el campo plz2.gid
plz2.nombre||', '||plz2.municipio as plaza,
--concatena cols nombre y municipio usando ||, || tabla alias plz2
(plz.geom<->plz2.geom)/1000.0 as dist_kms
--calcular distancias, convertirlas a kms, alias kms
from g33_dotacional_plazas_publicas_2010 plz2 --copia la tabla plazas as plz2
order by dist_kms limit 2) plz2
--ordena por campo dist_kms de la tabla plz2 y
--limita solo a dos filas porque una será dist=0
--y la segunda fila será la más cercana <0
where plz.gid<plz2.gid --no deben tener el mismo identificador gid
order by plz.municipio, dist_kms; --ordena por campo "municipio" asc, tabla plz
--luego ordena por campo dist_kms asc, tabla plz2

```

Grid	plaza_origen	plaza_destino	kms
1	Plaza de Recreo, Adjuntas	Plaza de Recreo, Utuado	11.64
2	Plaza de Recreo, Aguada	Plaza de Recreo, Aguadilla	6.64
3	Plaza de Recreo, Aguadilla	Plaza de Recreo, Moca	5.78
4	Plaza de Recreo, Aguas Buenas	Plaza de Recreo, Caguas	7.58
5	Plaza de Recreo, Aibonito	Plaza de Recreo, Barranquitas	6.72

Aquí la lista devuelta está en **orden alfabético desde el Municipio de Adjuntas hasta el Municipio de Yauco**. En la cláusula ORDER BY puede añadir DESC para que muestre la lista desde el Municipio de Yauco

ORDENAR POR DISTANCIAS USANDO EL OPERADOR <-> EN ORDER BY

En el query pasado obtuvimos la plaza de recreo más próxima, por orden de municipios. Por otro lado, si deseamos obtener el listado anterior **ordenado por distancias**, podemos usar el [operador <->](#) en la cláusula **ORDER BY**. Por ejemplo:

- Escriba y ejecute el siguiente query en la pestaña **Script-1 de DBeaver**:

```

select plz.nombre||', '||plz.municipio as plaza_origen, --concatenar nombre+municipio, tabla1
plz2.nombre||', '||plz2.municipio as plaza_destino, --concatenar nombre+municipio, tabla2
round((plz.geom<->plz2.geom)::numeric/1000,3) as dist_kms -- operador geom<->geom para buscar
--distancias, luego cast a numeric,
--div /1000 para obtener kms
from g33_dotacional_plazas_publicas_2010 plz, --de la tabla plazas as pz
g33_dotacional_plazas_publicas_2010 plz2 --copia misma tabla as pz2
where plz.gid<plz2.gid --que no devuelva ids iguales
order by plz.geom<->plz2.geom asc; --ordena por <-> distancia, ascendente

```

Grid	plaza_origen	plaza_destino	dist_kms
1	Plaza de Recreo, Camuy	Plaza de Recreo, Hatillo	2.003
2	Plaza de Recreo, Hatillo	Plaza de Recreo, Camuy	2.003
3	Plaza de Recreo, Dorado	Plaza de Recreo, Toa Baja	2.036
4	Plaza de Recreo, Toa Baja	Plaza de Recreo, Dorado	2.036
5	Plaza de Recreo, Guánica	Plaza Ensenada, Guánica	2.59

Las plazas de recreo más cercanas son las de Hatillo y Camuy. Le siguen de cerca las plazas de Dorado y Toa Baja, Ensenada y Guánica (del mismo municipio), etc.

Fila 9: Cataño → Plaza de Armas, San Juan. Recuerde que son distancias lineales. La distancia por carreteras puede ser de 13 kilómetros aproximadamente.



Tutorial PostGIS, 3.x

Como en otras matrices de distancias, los lugares se repiten. En esas matrices, se separaban los lugares entre las filas y las columnas.

Por otro lado, si quiere saber cuáles son las plazas más lejanas, solo cambie ASC por DESC en la cláusula ORDER BY.

Si no desea contar a los municipios de Vieques y Culebra en la lista, añada dos condiciones en la cláusula WHERE:

```
*gisdb_lab> Script-1 x
select plz.nombre||', '||plz.municipio as plaza_origen, --concatenar nombre+municipio, tabla1
plz2.nombre||', '||plz2.municipio as plaza_destino, --concatenar nombre+municipio, tabla2
round((plz.geom<->plz2.geom)::numeric/1000,3) as dist_kms -- operador geom<->geom para buscar
--distancias, luego cast a numeric,
--div /1000 para obtener kms
from g33_dotacional_plazas_publicas_2010 plz, --de la tabla plazas as pz
g33_dotacional_plazas_publicas_2010 plz2 --copia misma tabla as pz2
where plz.gid<>plz2.gid --que no devuelva ids iguales
and plz2.municipio not in ('Vieques','Culebra') --no incluir Vieques, Culebra en tabla plz2
and plz.municipio not in ('Vieques','Culebra') --no incluir Vieques, Culebra en tabla plz
order by plz.geom<->plz2.geom desc; --ordena por <-> distancia, ascendente
```

Results 1 x

```
select plz.nombre||', '||plz.municipio as plaza_origen, '||plz2.nombre||', '||plz2.municipio as plaza_destino, round((plz.geom<->plz2.geom)::numeric/1000,3) as dist_kms
```

Grid	plaza_origen	plaza_destino	dist_kms
1	Plaza de Recreo, Rincón	Plaza de Recreo, Ceiba	169.615
2	Plaza de Recreo, Ceiba	Plaza de Recreo, Rincón	169.615
3	Plaza de Recreo, Rincón	Plaza de Recreo, Fajardo	168.872
4	Plaza de Recreo, Fajardo	Plaza de Recreo, Rincón	168.872
5	Plaza de Recreo, Ceiba	Plaza de Recreo, Aguada	163.355
6	Plaza de Recreo, Aguada	Plaza de Recreo, Ceiba	163.355
7	Plaza de Recreo, Fajardo	Plaza de Recreo, Aguada	162.427
8	Plaza de Recreo, Aguada	Plaza de Recreo, Fajardo	162.427

Refresh Save Cancel Export data 200 6.642

200 row(s) fetched - 0.027s, on 2024-02-23 at 13:39:10

En este query, decimos que no queremos comparar distancias con las islas de Vieques y Culebra con ellos mismos y los demás municipios.



PRÁCTICAS:

1. **¿Cuál es el barrio que está al extremo este de Puerto Rico sin contar a Vieques ni Culebra?** Si descontamos Vieques y Culebra, ¿cuál es el barrio y su municipio más al extremo este? Use el [query de ejemplo ST_XMin](#) anterior. Si usa la función `st_xmin(geom)`, en order by el orden por `st_xmin` debe ser ascendente. Si usa `ST_XMax`, el orden debe ser descendente En la cláusula Where, puede usar `NOT municipio IN ('Vieques', 'Culebra')` o usar `NOT (barrio = 'Vieques' OR barrio = 'Culebra')`. Limite los resultados a 10 filas. Resultado:

Grid	barrio	municipio	st_xmax
	Cabezas	Fajardo	295747.4422
	Machos	Ceiba	290707.0613
	Guayacán	Ceiba	285808.9294
	Quebrada Vueltas	Fajardo	285502.6251
	Barrio Pueblo	Fajardo	285268.2501
	Sardinera	Fajardo	284810.3437
	Demajagua	Fajardo	284764.9375
	Barrio Pueblo	Ceiba	284092.6221
	Quebrada Fajardo	Fajardo	283379.4742
	Chupacallos	Ceiba	282757.2186

2. **Haga una lista donde cada barrio muestre el número de sectores dentro del barrio y una lista de sectores hacia la derecha para cada barrio.**

Geodatos a usar: `barrios_2015_villalba` as b, `g25_asentamientos_sectores2017_villalba` as s.

Campos/columnas a usar: `b.barrio`, `count(escrIBA el campo) as num_sectores`, `string_agg(campo, 'patrón de separación') as sectores`

WHERE (necesario o no)

GROUP BY (necesario o no, ¿resumir por cuál campo?)

ORDER BY, según el resultado mostrado, ¿cuál fue el campo usado?

Resultado:

barrio	num_sectores	sectores
Barrio Pueblo	4	San Cristóbal, Borinquen, Cooperativa, Efraín Suárez
Caonillas Abajo	4	Los Fondos, La Cruz, El Rincón, Cerro Gordo

3. Use el geodato `g35_viales_mapa_base_tiger_rds_2006se_villalba` para **mostrar una lista de nombres de vías que conectan con la carretera PR-150.**

SELECT `v1.gid, v2.gid`, --para que vea cuales son las filas que se están comparando

`v2.fename` y `v2.geom` --para que traiga el nombre de la vía y la geometría del geodato copiado (v2)

FROM: Ya que va a comparar el mismo geodato, deberá usar dos alias `v1` y `v2`

WHERE: las condiciones serán:

`v1.fename = 'PR-150'`

Predicado: `ST_Touches(v1.geom, v2.geom)`. Puede usar también `st_intersects`.

`v1.gid <> v2.gid` --para que los id no sean iguales (no du

ORDER BY : `v1.gid` -- para ver cuáles son las filas comparadas

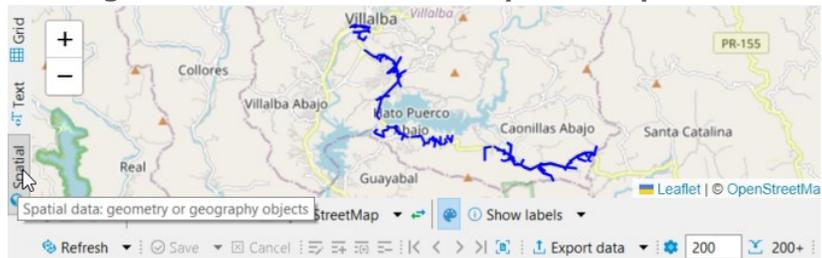


Tutorial PostGIS, 3.x

Debe devolver 340 filas (si los datos no han cambiado)

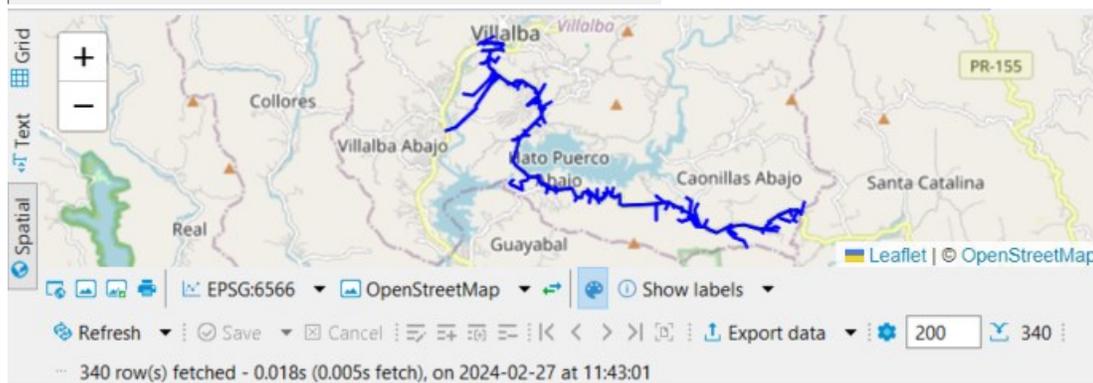
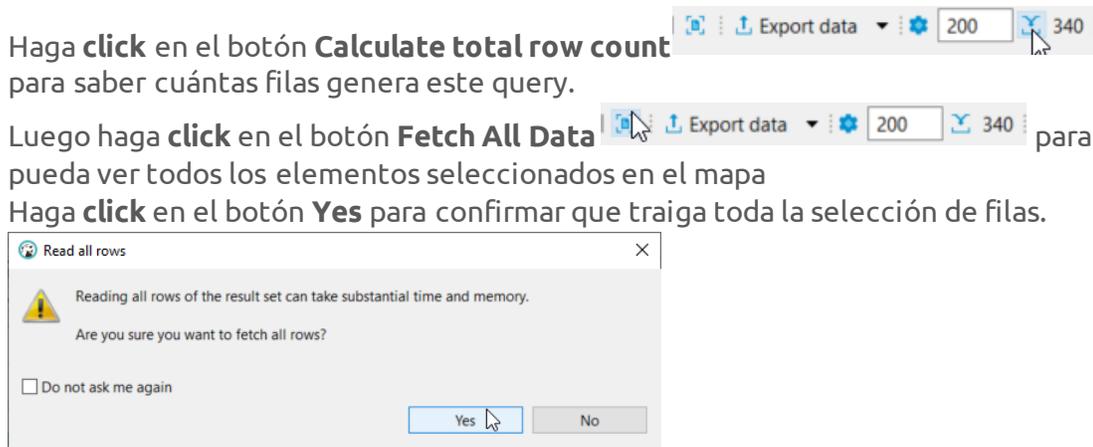
gid	lgid	fename	geom
6	474	PR-150	MULTILINESTRING ((200489.
164	987	PR-150	MULTILINESTRING ((193498.
164	287	Calle Borinquen	MULTILINESTRING ((193556.
164	2427	PR-150	MULTILINESTRING ((193600.
319	992		MULTILINESTRING ((193206.
319	3152	PR-150	MULTILINESTRING ((193645.
319	699	PR-150	MULTILINESTRING ((193396.
389	818	PR-150	MULTILINESTRING ((199570.

- Vea el geodato haciendo click en la pestaña Spatial.



¿Está esto correcto? No completamente. Por defecto solo muestra las primeras 200 filas.

- Haga click en el botón **Calculate total row count** para saber cuántas filas genera este query.
- Luego haga click en el botón **Fetch All Data** para que pueda ver todos los elementos seleccionados en el mapa
- Haga click en el botón **Yes** para confirmar que traiga toda la selección de filas.

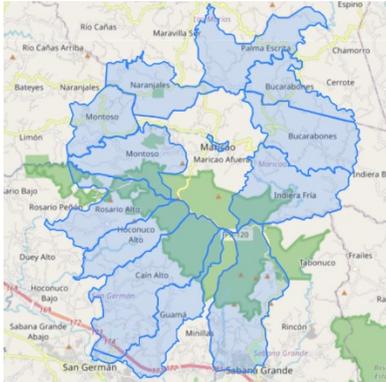


- ¿Cuáles son los barrios con más vecinos adyacentes? Use el [query para visualizar el número de barrios adyacentes con ST_Touches](#) como referencia. Esta vez, **sustituya** el predicado ST_Touches(b1.geom, b2.geom) con el predicado **ST_DWithin(b1.geom, b2.geom, 50)**, estableciendo una distancia de búsqueda de 50 m. Para que no repita el



Tutorial PostGIS, 3.x

barrio de referencia, incluya en el apartado WHERE, además de lo que aparece en el ejemplo de referencia, inserte la condición `and b1.gid<>b2.gid` para que no incluya en la lista el barrio de referencia. **¿Cuántos y cuáles son los barrios?**



Estos son los **13 barrios adyacentes** (comparten límites) con el barrio Maricao Afuera del Municipio de Maricao. Podrá ver que el barrio Minillas está bien próximo pero no comparten límites ni en un punto.

Si aplicamos **ST_DWithin con una distancia, digamos de 50 metros**, podríamos obtener una lista de los barrios que se tocan, además de los que están cerca de tocarse.

En la lista de salida de este nuevo query debe haber **14 filas**.

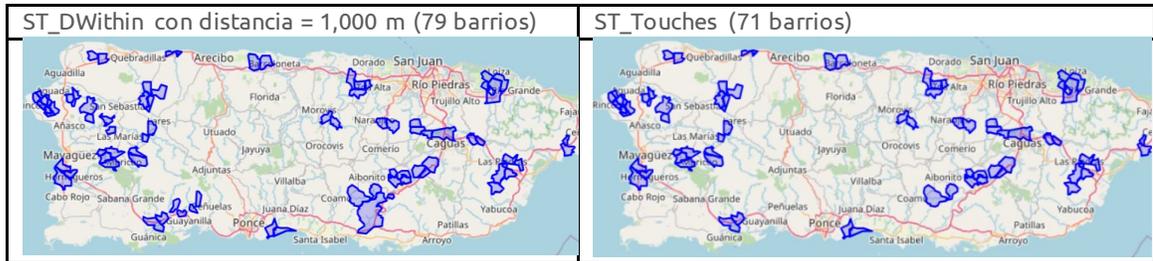
- Haga una lista de barrios de municipios adyacentes pero que tengan nombres iguales.** Por ejemplo, los municipios de Caguas, Cidra y Cayey son adyacentes y además tienen cada uno un barrio "Beatriz" que es adyacente o colindante con cada uno de estos municipios. Así hay cierto número de otros barrios adyacentes con el mismo nombre en municipios diferentes. Puede usar el predicado **ST_Touches** o usar el predicado **ST_DWithin(geom,geom,1000)** para establecer una distancia de búsqueda. **Escriba lo que falta en este query y ejecútelo en DBEaver.** Visualice el resultado en el tab **Spatial**

```

select b1.barrio || ', ' || b1.municipio as "barrio_municipio: ", -1:une barrio y municipio
count(b2.____) as cuántos_adyacentes, --2:cuenta cuántos barrios son adyacentes
string_agg(b2.____ || '- ' || b2.____, ', ') --3:agrega barrio-municipio a la lista
as "barrios_municipios_adyacentes:", --3:alias de la columna
round(sum((b1.____<->b2.____)):numeric/1000.0,3) as dist_kms, --4:redondea la dist_kms
b1.____ --añade las geometrías para verlas en el tab Spatial de DBEaver
from g03_legales_barrios_2015 as b1 --geodato de barrios
join _____ as b2 --copia del geodato de barrios
on ST_DWithin(b1.____,b2.____,1000) --dist de búsqueda
where __.barrio=__barrio --nombres de barrio idénticos entre bo. original y copia
and __.gid<>__.gid --el id no debe ser igual para no repetir el mismo barrio
group by b_.barrio,b_.municipio,b_.geom --agrupar por estos campos, geodato no copiado
order by 4, 2 desc, 1; --ordena por dist_kms, cuántos_adyacentes, barrio_municipio

```

Resultado en mapa. Barrios adyacentes con nombres idénticos en municipios adyacentes,



barrio_municipio:	cuántos_adyacentes	barrios_municipios_adyacentes:	dist_kms	geom
Beatriz, Caguas	2	Beatriz-Cayey, Beatriz-Cidra	0.00	MULT
Beatriz, Cayey	2	Beatriz-Cidra, Beatriz-Caguas	0.00	MULT



6. Haga una lista de escuelas y las distancias a hospitales en Villalba y del área.

Geodatos a usar: g33_dotacional_educacion_escuelas_2021_villalba as e, g33_dotacional_salud_hospitales_cerca_villalba as h

Función: ST_Distance o el operador <->.

Escriba lo que falta en este query y ejecútelo en DBeaver.

```

select e._____, --trae nombre de escuelas
_____.nombre, --trae nombre de hospitales
round((_____.geom<->h._____)::_____/_____,2) --redondea distancia kms entre geoms
as dist_kms --alias de la expresión: dist_kms
from g33_dotacional_educacion_escuelas_2021_villalba e, --tabla escuelas
g33_dotacional_salud_hospitales_cerca_villalba h --tabla hospitales
order by _____, --orden alfabético ascendente escuela
e._____<->_____.geom; --orden por distancia ascendente

```

Resultado:

escuela	nombre	dist_kms
CRISTINA (AMADA) MARTINEZ MARTINEZ	MedCentro Villalba	1.53
CRISTINA (AMADA) MARTINEZ MARTINEZ	Centro San Cristóbal Villalba	2.37
CRISTINA (AMADA) MARTINEZ MARTINEZ	CDT San Cristóbal Juana Díaz	6.94
CRISTINA (AMADA) MARTINEZ MARTINEZ	Centro Médico San Lucas	13.02
CRISTINA (AMADA) MARTINEZ MARTINEZ	Hospital Menonita Coamo	14.84
CRISTINA (AMADA) MARTINEZ MARTINEZ	Hospital Metropolitano Dr Pila	16.41
CRISTINA (AMADA) MARTINEZ MARTINEZ	Hospital Damas	17.46
CRISTINA (AMADA) MARTINEZ MARTINEZ	CDT Santa Isabel	18.52
CROEV: ESPECIALIZADA EN CIENCIAS Y MATEMATICAS	MedCentro Villalba	0.44
CROEV: ESPECIALIZADA EN CIENCIAS Y MATEMATICAS	Centro San Cristóbal Villalba	1.08

7. El query anterior nos devuelve las distancias de entre cada escuela con todos los hospitales registrados en el geodato de hospitales. Por eso tiene 72 filas: 9 escuelas y 8 hospitales.

Pero si nos piden: **Lista solamente los primeros 3 centros de salud (CDT y hospitales) más cercanos a cada escuela.** Para hacer esa lista, debemos hacer un **CROSS LATERAL JOIN** como el [query de ejemplo](#) de esa sección.

Escriba lo que falta en este query y ejecútelo en DBeaver.

```

select _____._____, --trae nombre escuela
_____._____, --trae nombre hospital
round(h.dist::_____/_____,2) as kms --calc y redondea distancia en m as kms
from g33_dotacional_educacion_escuelas_2021_villalba e --tabla escuelas
cross join lateral( --uso cross join lateral
  select h._____, --trae nombre hospital para devolverlo al outer query
  _____._____<->_____._____ as dist --calc distancias entre geoms escuelas y hospst
  from g33_dotacional_salud_hospitales_cerca_villalba h --tabla hospitales
  order by _____ --ordena por distancia, si no ordenas no sirve el limit 3
  limit _____) --trae solo ____ hospitales más cercanos
as _____ --necesario poner nombre o alias al subquery: h
order by _____, _____; --outer query: ordena resultados por cols 1 asc y 3 asc

```



Tutorial PostGIS, 3.x

escuela	nombre	kms
CRISTINA (AMADA) MARTINEZ MARTINEZ	MedCentro Villalba	1.53
CRISTINA (AMADA) MARTINEZ MARTINEZ	Centro San Cristóbal Villalba	2.37
CRISTINA (AMADA) MARTINEZ MARTINEZ	CDT San Cristóbal Juana Díaz	6.94
CROEV: ESPECIALIZADA EN CIENCIAS Y MATEMATICAS	MedCentro Villalba	0.44
CROEV: ESPECIALIZADA EN CIENCIAS Y MATEMATICAS	Centro San Cristóbal Villalba	1.08
CROEV: ESPECIALIZADA EN CIENCIAS Y MATEMATICAS	CDT San Cristóbal Juana Díaz	8.14
DANIEL SERRANO RIVERA	MedCentro Villalba	1.96
DANIEL SERRANO RIVERA	Centro San Cristóbal Villalba	2.83
DANIEL SERRANO RIVERA	CDT San Cristóbal Juana Díaz	6.54
FRANCISCO ZAVAS SANTANA	Centro San Cristóbal Villalba	0.35

Deben aparecer 27 filas ordenadas por escuela y distancia, ambos de forma ascendente.

8. Encuentre la cantidad de edificios que están en la subzona inundable zone_subty='FLOODWAY' en el Municipio de Villalba.

Geodatos a usar:

g23_riesgo_inunda_floodzones_2017_villalba as f

g33_mapa_base_building_centroids_1998_villalba as e,

Campos a usar: SELECT f.zone_subty, e.gid (¿cuál es la función para hacer conteo de edificios con el campo e.gid?)

Condición (WHERE):

Predicado a usar:

Intente cada uno de los predicados: **ST_Covers** o **ST_CoveredBy**,

ST_Intersects,

Condición adicional: **AND zone_subty = 'FLOODWAY'**

GROUP BY: use la columna #1.

Resultado: debe ser idéntico para los tres predicados

Grid	zone_subty	num_edificios
1	FLOODWAY	47

9. Ejecute el siguiente query en el tab <gisdb_lab> Script-1

```
select f.zone_subty,geom
from g23_riesgo_inunda_floodzones_2017_villalba as f
where f.zone_subty = 'FLOODWAY';
```

Verá que hay dos áreas 'FLOODWAY' en el municipio.

Grid	zone_subty	geom
1	FLOODWAY	MULTIPOLYGON (((196180.7
2	FLOODWAY	MULTIPOLYGON (((192611.1



¿Por qué cree que la otra zona Floodway no aparece en el query anterior #8?



10. Usando el query de la práctica anterior #8, haga los siguientes cambios:

SELECT: f.zone_subty, f.gid, count(e.gid)

FROM:

from g23_riesgo_inunda_floodzones_2017_villalba **as** f,
g33_mapa_base_building_centroids_1998_villalba **as** e

WHERE: cambie el predicado a **ST_DWithin(f.geom, e.geom, 1)**.

AND f.zone_subty = 'FLOODWAY'

GROUP BY: use los campos 1,2

¿Cuántos puntos aparecen si aplica una distancia de 1 metro, de 10, o de 20?. ¿Aparece más de una zona Floodway? _____

¿Por qué **aparece otra zona Floodway** cuando aplicamos **distancia mayor**? _____

11. Usando el query de la práctica anterior #10 Qué podemos hacer para que siempre aparezcan todas las zonas f.zone_subty = 'FLOODWAY' en el query? Hint: revise la sección [Spatial Joins](#). ¿Qué tipo de JOIN aplicaría?

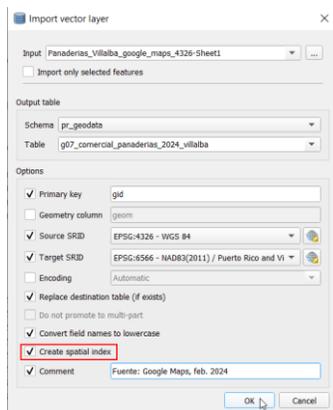


Consideraciones importantes antes del análisis geográfico y geoprocursos

ÍNDICES ESPACIALES

La búsqueda de datos se hace más eficiente al establecer índices. Por ejemplo, es más fácil buscar un tema en un libro usando el índice. El índice del libro tendrá una lista de temas y la localización (en qué página se encuentra). De la misma manera trabajará un índice espacial, reduciendo el número de comparaciones entre filas de una o más tablas.

En PostgreSQL el algoritmo de indexación espacial se le llama GiST (Generalized Search Tree), el cual en PostGIS se aplica a geometrías/geografías también. Cuando los datos geográficos comienzan a tener miles de filas, es altamente recomendable asegurarse de que tengan índice geoespacial.



Al importar datos a PostGIS desde la interfaz DB Manager de QGIS, podemos crear índices espaciales.

Ya hemos conocido el operador `geom_a<->geom_b` el cual devuelve la distancia entre las geometrías de dos geodatos. Este operador y otros no discutidos aquí, utilizan la indexación GiST. Para aprovechar el uso de índices, podemos usar el operador doble ampersand `&&` para comparar entre cajas de extensión territorial (bounding boxes) que encierran cada objeto geometry o geography. Este es el operador a utilizado cuando un predicado espacial no integra el uso de índices disponibles.

PREDICADOS ESPACIALES QUE USAN ÍNDICE GEOESPACIAL POR DEFECTO

Ya que los índices son tan importantes, deberíamos tener idea de cuáles predicados topológicos/espaciales integran el uso de índices. Esto es, comparan entre bounding boxes de geometrías.

- | | |
|---------------------|-------------------|
| ST_Intersects | ST_Equals |
| ST_Crosses | ST_OrderingEquals |
| ST_Within | ST_DWithin |
| ST_Contains | ST_3DDWithin |
| ST_Covers | ST_3DDFullyWithin |
| ST_CoveredBy | ST_3DDIntersects |
| ST_ContainsProperly | ST_Touches |
| ST_Overlaps | |

PostGIS nos da la opción de usar el predicado sin utilizar el índice. En esos casos, ponemos una barra baja `_` al inicio del predicado. Por ejemplo `_ST_Intersects` no usa índices.

CREACIÓN DE TABLAS A PARTIR DE UN QUERY

En ocasiones podemos aprovechar las filas y columnas devueltas por un query y convertirlos en una tabla. La forma recomendada por Martínez-Llario en su libro (pp. 114-117), es la siguiente:



Tutorial PostGIS, 3.x

1. Crear la tabla espacial receptora de los resultados de un query, comenzando con la sentencia CREATE TABLE, añadir las columnas/campos de interés, y la geometría.
2. Luego se insertan las filas y columnas mediante el comando INSERT INTO.

Ejemplo 1: Crear tabla definiendo los campos

```
CREATE TABLE g15_suelos_agricolas_prime_2018 (gid serial primary key, farm_class
varchar, geom geometry ('MULTIPOLYGON', 6566));
```

```
INSERT INTO g15_suelos_agricolas_prime_2018 (farm_class, geom)
SELECT farm_class geom FROM g15_suelos_soil_map_units_2018_villalba WHERE
farm_class <> 'Not prime farmland';
```

Explicación:

Creación de tabla:	
create table pr_geodata.g15_suelos_agricolas_prime_2018	create table Comando para crear la tabla con nombre <code>g15_suelos_agricolas_prime_2018</code> dentro del schema <code>pr_geodata</code> . *
(gid serial primary key , farm_class varchar , geom geometry ('MULTIPOLYGON', 6566));	Entre paréntesis van los nombres de los campos/columnas que van a crearse en la nueva tabla. gid serial primary key será el campo gid tipo serial y primary key. Serial es como el tipo <i>autonumber</i> en MS Access; números enteros que se incrementan automáticamente y no se repiten. farm_class será un campo tipo varchar , geom será el campo tipo geometry , que a su vez se definirá como 'MULTIPOLYGON' y con el sistema de referencia espacial EPSG:6566, que corresponde con STATE PLANE 5200, PR USVI, NAD83(2011), unidades en metros.

Insertar filas en la nueva tabla:	
insert into pr_geodata.g15_suelos_agricolas_prime_2018 (farm_class, geom)	Instrucción SQL para insertar valores en las filas de la nueva tabla <code>pr_geodata.g15_suelos_agricolas_prime_2018</code> en los campos <code>farm_class</code> y <code>geom</code> . No se incluye gid, ya que es tipo serial, como se explicó arriba.
select farm_class, geom from g15_suelos_soil_map_units_2018_villalba where farm_class <> 'Not prime farmland';	Luego se escribe el query que deseamos usar para insertar las filas. La condición en WHERE es que los suelos no estén clasificados como 'Not prime farmland'. Dicho de otro modo, todos los que sean tipo prime en todas sus variantes, y otros que están catalogados como suelos agrícolas de importancia estatal.

* Si no especificamos el **schema pr_geodata.tabla**, la tabla se va a crear dentro del schema "public".

¿Y si necesitamos todos los campos de la tabla original y pasarlos a la nueva tabla? En el próximo ejemplo, veremos cómo copiar la estructura de una tabla existente a una tabla vacía.

Ejemplo 2: Copiar la estructura de una tabla existente a una tabla nueva



Tutorial PostGIS, 3.x

El siguiente código sirve para copiar solamente la estructura de una tabla existente y generar una nueva tabla basada en esa estructura. Vamos a copiar la estructura de una tabla, incluyendo campos, y todas sus restricciones a la nueva tabla vacía.

```
CREATE TABLE pr_geodata.g15_suelos_agricolas_prime_2018 (like
g15_suelos_soil_map_units_2018_villalba including defaults including constraints
including indexes);
```

```
INSERT INTO pr_geodata.g15_suelos_agricolas_prime_2018
SELECT * FROM g15_suelos_soil_map_units_2018_villalba WHERE farm_class <> 'Not
prime farmland';
```

Explicación:

Creación de tabla:	
<code>create table pr_geodata.g15_suelos_agricolas_prime_2018</code>	create table: Comando para crear la tabla con nombre g15_suelos_agricolas_prime_2018 dentro del schema pr_geodata . *
<code>(like g15_suelos_soil_map_units_2018_villalba including defaults including constraints including indexes);</code>	"like" es la palabra clave para copiar estructura, incluyendo <i>defaults</i> o valores que adopta por defecto, <i>constraints</i> o restricciones y todos los índices que existan relacionados con la tabla fuente, que en este caso es la tabla g15_suelos_soil_map_units_2018_villalba

Insertar filas en la nueva tabla:	
<code>insert into pr_geodata.g15_suelos_agricolas_prime_2018</code>	Instrucción SQL para insertar valores en las filas de la tabla g15_suelos_agricolas_prime_2018 en todos los campos .
<code>select * from g15_suelos_soil_map_units_2018_villalba where farm_class <> 'Not prime farmland';</code>	Luego se escribe el query que deseamos usar para insertar las filas. La condición en WHERE es que los suelos no estén clasificados como <i>'Not prime farmland'</i> . Dicho de otro modo, todos los que sean tipo prime en todas sus variantes, y otros que están catalogados como suelos agrícolas de importancia estatal.

* Si no especificamos el **schema pr_geodata.tabla**, la tabla se va a crear dentro del schema "public".

Nota: Tener en cuenta que las secuencias del campo *gid serial* de la nueva tabla serán iguales a los de la tabla original.



Visualizar la nueva tabla:

En PostGIS	En QGIS

HOMOGENIZAR GEOMETRÍAS DEVUELTAS AL SOBREPONER GEODATOS

Al sobreponer geometrías en PostGIS es posible que el resultado del query esté compuesto de varios tipos de geometrías. Por ejemplo:

Sobreposiciones geométricas	Geometrías devueltas al sobreponer
P/P	Punto
P/L, L/P	Punto
P/S, S/P	Punto
L/L	Punto y/o línea
L/S, S/L	Punto y/o línea
S/S	Punto, línea, polígono(S)

* En algunos casos, PostGIS puede devolver filas con geometrías vacías.

Los SIG de interfaz gráfica, QGIS, ArcGIS y otros, **no** están diseñados por el momento para manejar una columna con *más de un tipo de geometría*. Por lo tanto, es preferible homogenizar los resultados geométricos devueltos por las funciones de sobreposición tales como [ST_Intersection](#)(geom, geom) entre otras.

Hagamos un ejemplo que involucre dos geodatos con geometría de polígonos.

□ Escriba y ejecute el siguiente query en la pestaña **Script-1 de DBEaver**:

```

/* Tipos de geometría que devuelve ST_Intersection
 * al combinar geodatos landslides y put2015
 * condición: where st_intersects(a.geom,b.geom)
 */
select geometrytype(geom) as tipo_geom, --trae tipo geometría
count(*) -- conteo de elementos
from (select st_intersection(a.geom,b.geom) as geom --trae la intersección geom de a y b as geom
      from g23_riesgo_geol_landslides_villalba a, --de las tablas landslides as a
      g29_planes_put_2015_villalba b --y put2015_villalba as b
      where st_intersects(a.geom,b.geom) --condición: geometrías que se intersequen entra a y b
      ) as tabla_subquery --guarda el resultado en un subquery, alias tabla_subquery
group by tipo_geom; --agrupar datos por columna tipo_geom

```

1ª parte: usar la función **GeometryType**(geom) as tipo_geom y el conteo de elementos.

2ª parte: Hacer **subquery**. Uso de **ST_Intersection**(a.geom, b.geom) as geom entre las tablas

g23_riesgo_geol_landslides_villalba as a y g29_planes_put_2015_villalba as b
Condición: (WHERE) **st_intersects**(a.geom, b.geom) alias "tabla_subquery"

3ª parte: GROUP BY, agrupa los resultados por la columna **tipo_geom**, que está en la primera parte o query exterior.

Como parte de este query de intersección geométrica ST_Intersection, PostGIS devuelve en la columna tipo_geom: GEOMETRYCOLLECTION, LINestring, MULTILINestring, MULTIPOLYGON, POINT y POLYGON.



Tutorial PostGIS, 3.x

Una de las formas para tratar de evitar que se produzcan distintos tipos de geometría devueltos en un query es utilizar el predicado [ST_Relate](#)(geom, geom, 'patrón'). 'Patrón' se refiere a uno de los múltiples patrones existentes al relacionar o sobre imponer geometrías, según la matriz de intersección Egenhofer/Clementini discutidas en los apéndices [A-1](#) y [A-2](#).

Si vamos a **intersecar áreas superficiales o polígonos (S)** y queremos obtener solo polígonos, podemos usar **ST_Relate(geom, geom, 'T*****')** para solo intersecar el interior de una geometría con el interior de la otra geometría. En el caso de intersecar los interiores de S/S, estos devolverán polígonos (S).

Ya que **ST_Relate no utiliza por defecto los índices**, antes será necesario forzar su utilización mediante el **operador de comparación de bounding boxes** doble ampersand **&&**. Por ejemplo, en la cláusula WHERE:

WHERE a.geom && b.geom AND ST_Relate(a.geom, b.geom, 'T***');**

Los resultados devueltos pueden combinarse en geometrías vacías, tipo MULTI o SIMPLE y GeometryCollection.

Hagamos un ejemplo que involucre dos geodatos con geometría de polígonos y usar el predicado **ST_Relate** en la cláusula WHERE y añada las geometrías usando el campo geom.

- Escriba y ejecute el siguiente query en la pestaña **Script-1 de DBEaver**:

```

1 /* Tipos de geometría que devuelve ST_Intersection
2 * al combinar geodatos landslides y put2015
3 * condición: where st_intersects(a.geom,b.geom)
4 */
5 select geometrytype(geom) as tipo_geom, --trae tipo geometría
6 geom --trae cada geometría
7 from (select st_intersection(a.geom,b.geom) as geom --trae la intersección geom de a y b as geom
8 from g23_riesgo_geol_landslides_villalba a, --de las tablas landslides as a
9 g29_planes_put_2015_villalba b --y put2015_villalba as b
10 where a.geom && b.geom --condición:forzar uso de índice:
11 --que los bounding boxes de geometrías a y intersequen
12 and st_relate(a.geom,b.geom,'T*****')--usar st_relate, solo inteseacar interior de geoms
13 ) as tabla_subquery --guarda el resultado en un subquery, alias tabla_subquery
14 group by tipo_geom,geom; --agrupar datos por columna tipo_geom

```

tipo_geom	geom
GEOMETRYCOLLECTION	GEOMETRYCOLLECTION (POLYGON ((201249.16002520797 229284.78789573976, 201249.16 229...
GEOMETRYCOLLECTION	GEOMETRYCOLLECTION (POLYGON ((197448.0687843631 238019.4732942823, 197448.0688 2380...
MULTIPOLYGON	MULTIPOLYGON (((196135.9419 238120.9331, 196165.9419 238119.7559, 196189.90748804293 23...
MULTIPOLYGON	MULTIPOLYGON (((189760.47182230128 230330.06427073063, 189774.2804 230313.3197, 189756...
MULTIPOLYGON	MULTIPOLYGON (((192199.3913 229803.3197, 192181.2737099135 229818.55362221098, 192182...
MULTIPOLYGON	MULTIPOLYGON (((192145.9419 229809.4418, 192129.24318203426 229819.43537041618, 192135...
MULTIPOLYGON	MULTIPOLYGON (((192147.55124717517 229923.25438549987, 192147.68869820915 229923.4114...
MULTIPOLYGON	MULTIPOLYGON (((192244.7811 229953.3197, 192235.9419 229961.4761, 192214.77535665088 22...
MULTIPOLYGON	MULTIPOLYGON (((191959.7644 230642.6718, 191941.619 230640.7033, 191923.5108 230638.418...

- 1ª parte: traer tipos de geometría **GeometryType(geom)** as **tipo_geom** y las geometrías.
- 2ª parte: Hacer **subquery**. Uso de **ST_Intersection(a.geom, b.geom)** as **geom** entre las tablas **g23_riesgo_geol_landslides_villalba** as **a** y **g29_planes_put_2015_villalba** as **b**
Condición: (WHERE) **a.geom && b.geom** para inducir uso de comparador de bounding boxes **ST_Relate(a.geom, b.geom, 'T*****')** alias "tabla_subquery"
- 3ª parte: **GROUP BY**, agrupa los resultados **por** las columnas **tipo_geom** y **geom**, que están en la primera parte o query exterior.

Con el uso de **ST_Relate** no devuelve las geometrías lineales. Sin embargo, el campo **tipo_geom** tiene tres tipos de geometría: **GEOMETRYCOLLECTION**, **MULTIPOLYGON** y **POLYGON**. **Vamos a usar la interfaz gráfica de DBEaver para indagar.**



- En la pestaña **Results 1**, vaya al campo **geom** y haga **click** en la primera celda **GEOMETRYCOLLECTION (POLYGON....**
- En la pestaña **Value** aparecerán los elementos que componen esta **GEOMETRYCOLLECTION**. Si deja el cursor encima de esta celda, le mostrará los elementos y la lista de coordenadas



- Podrá ver que **aun cuando se usó ST_Relate y el patrón 'T*****', esto no impidió que devolviera Linestrings en dos filas tipo GEOMETRYCOLLECTION**. Además el query devuelve tanto MULTIPOLYGON como POLYGON.

En PostGIS no hay inconvenientes con esto en cuanto a su almacenamiento y visualización en DBeaver. Sin embargo, este resultado no es compatible con un SIG de interfaz gráfica como QGIS o ArcGIS.

Por tal motivo, se hace necesario **homogenizar las geometrías devueltas por los queries**. Existe una función `ST_CollectionExtract`, sin embargo, esta función puede devolver geometrías tipo `GEOMETRYCOLLECTION empty` y sólo convierte a `MULTI` los elementos dentro de un `GEOMETRYCOLLECTION` (Martínez Llario, 2017, p. 132).

UTILIZACIÓN DE FUNCIÓN EXTRA: **STX_EXTRACT**

Para evitar las situaciones mencionadas anteriormente, el Prof. Martínez Llario desarrolló una función en lenguaje PL/PgSQL, la cual modifica el comportamiento de la función `ST_CollectionExtract`, permitiendo que este tipo de queries devuelvan geometrías de manera homogénea.

- Si la **geometría devuelta está vacía, devolverá un null** (la columna `geom` puede almacenar null).
- Solo devuelve entidades **MULTI**.
- Tiene un **segundo argumento para especificar la dimensión de las geometrías de salida: 0 para puntos, 1 para líneas y 2 para polígonos/superficies**.



Tutorial PostGIS, 3.x

```

DECLARE
geom alias for $1;
dimension alias for $2;
sencillo alias for $3;
out Geometry;
tipo Varchar;
dimensional integer;
BEGIN
IF (ST_IsEmpty (geom)) THEN
RETURN null;
END IF;

tipo:= geometrytype(geom);
dimensional:=st_dimension(geom);

--Una colección es: multipoint, multilinestring, multipolygon,
--geometrycollection, multisurface, multicurve, compoundcurve

--Si no es una colección, devuelve el elemento si coincide
--la dimensión solicitada con la dimensión del elemento
IF (st_iscollection(geom) = false) THEN
IF (dimension <> dimensional) THEN RETURN null; END IF;
IF (sencillo = false) THEN RETURN ST_Multi(geom); END IF;
RETURN geom;
END IF;

--Si Es una colección extrae los subelementos que tienen la
--dimensión solicitada. No extrae subelementos curvos.

out:= ST_collectionextract (geom, dimension + 1);
if (sencillo = false) THEN out:= ST_Multi(out);

--Devuelve un elemento sencillo
ELSE out:= ST_GeometryN (out, 1);
END IF;

IF (ST_IsEmpty (out)) THEN
RETURN null;
END IF;

RETURN out;
END;

```

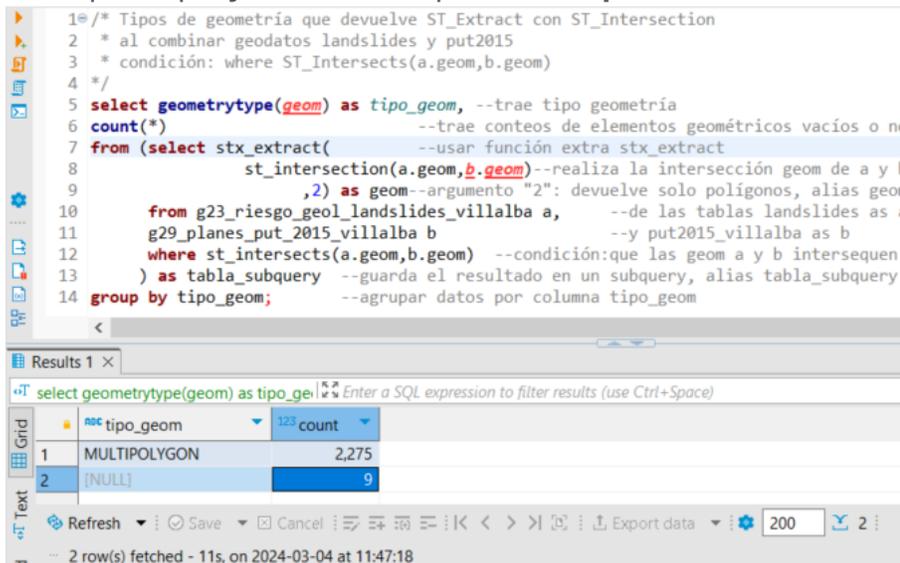
Este es el script de la función STX_Extract. Esta base de datos tiene esta función instalada. El código puede encontrarlo en <https://cartosig.web.upv.es> (Universidad Politécnica de Valencia, España).

La dirección contiene este y otros scripts. Las funciones SQL deberán ejecutarse en Windows desde el usuario Administrator o "run as Administrator".
 Consola> psql -U postgres -f ruta_al_archivo_funcionesextra.sql base_de_datos

psql se ejecuta desde el folder de instalación de Postgres/PostGIS o haber declarado el path en las variables de System en Windows. Puede ser: ...\\PostgreSQL\\release_number\\bin (Martínez Llario, 2017, pp. 31-32).

Vamos a usar el query anterior con la función extra **STX_Extract** para demostrar el resultado.

- Modifique el query anterior en la pestaña **Script-1 de DBEaver**:



Podrá notar que el query anterior le devuelve solamente MULTIPOLYGON y 9 elementos vacíos NULL. Recuerde que PostGIS permite guardar geometrías vacías (NULL).

Si no desea que le devuelva filas NULL, inserte lo siguiente en una cláusula WHERE, en el query exterior:

```

13 ) as tabla_subquery --guarda el resultado en un subquery, alias tabla_subquery
14 where geometrytype(geom) notnull --no devolver geometrías NULL
15 group by tipo_geom; --agrupar datos por columna tipo_geom

```



TOLERANCIA EN EL ANÁLISIS ESPACIAL

Ya hemos visto en el ejemplo anterior que la función de intersección **ST_Intersection(geomA,geomB)**. No debemos confundir **ST_Intersection** con el predicado **ST_Intersects**. El predicado **ST_Intersects** devuelve TRUE o FALSE al comparar dos geometrías. La función **ST_Intersection** devolverá geometría(s) o null.

¿Qué situación encontramos al usar **ST_Intersection**? Primero, al hacer la prueba CIERTO o FALSO el predicado **ST_Intersects** da paso a cualquier tipo de geometría que pase la prueba de intersección: ya sea de punto, línea o área. Vimos también cómo intentamos “cerrar el paso” a otras geometrías al usar **ST_Relate** con el patrón ‘T*****’. El patrón se usó para especificar que solo las geometrías que se intersecan interiores serán seleccionadas. Entonces, nos devolvió dos geometrías tipo GEOMETRYCOLLECTION que tenían polígonos y segmentos de línea de dos vértices, además del resto de polígonos que fueron MULTIPOLYGON y POLYGON.

Luego vimos cómo la función extra **STX_Extract** nos ayudó a poder ser más selectivos y solamente devolver polígonos tipo MULTI. Sin embargo, esto no “resuelve” el problema de fondo. Lo que hace es proveer un remedio a un problema generado desde PostGIS y todos los programas SIG que usan las bibliotecas [GEOS](#) y que operan con el estándar discutido anteriormente en la sección [Simple Features](#) de este manual. Las operaciones de intersección nos pueden devolver geometrías insignificantes. Por tal razón el modelo de tolerancias ha sido un buen recurso en los geoprocementos, tanto para ArcGIS, ArcInfo y para GRASS. Estos programas usan un modelo topológico persistente y ‘planarizan’ las geometrías, forzando la creación de nodos en cada intersección entre geometrías. En estos programas se establece un umbral de tolerancia, con el cual podemos eliminar datos espurios o innecesarios. Las dos GEOMETRYCOLLECTION que nos generó **ST_Intersection** eran geometrías insignificantes, al igual que otras geometrías minúsculas que pudieron haber sido generadas.

¿Qué criterio usamos para definir umbrales de tolerancia?

Dependerá principalmente de la **escala de compilación** en que se preparó el geodato. Originalmente los mapas eran preparados en un medio fijo, ya fuera papel, acetato (mylar) y otros. La escala de compilación era fija y se establecía una equivalencia entre distancias en el mapa y en el terreno.

En los [estándares de exactitud de mapas de los EE. UU.](#) Se establece que para mapas a escala 1:24,000 de la serie de mapas topográficos de 7.5 minutos la exactitud horizontal requerida para el 90% de los puntos escogidos para prueba, estos deben tener una exactitud alrededor de 1/50 de pulgada (0.05 cm) en el mapa. A una escala de 1:24,000, 1/50 de pulgada son 40 pies (12.2 m). Más adelante continúa este escrito: **para mapas con escala de publicación mayor de 1:20,000, no más del 10% de los puntos para prueba deberán tener un error de 1/30 de pulgada, medido en la escala de publicación.** Estos puntos aplican solamente a “puntos bien definidos”, tales como **benchmarks, intersecciones de carreteras y líneas de ferrocarril, y esquinas de edificios.**



Tutorial PostGIS, 3.x

Martínez Llario (2020), pág. 215 muestra una **relación de exactitud cartográfica** entre los estándares de **EEUU** y de **España**:

Escala	Exactitud cartográfica en metros, según el intervalo de confianza	
	EE UU (95%)	España (99%)
1:100	0.06	0.02
1:500	0.31	0.10
1:1,000	2.45	0.20
1:5,000	3.06	1.00
1:20,000	12.25	4.00

Tomado de Martínez Llario (2020), p. 215.

Tomando esto en consideración, **la exactitud varía en función a la escala**. En muchos casos, dependerá del nivel de detalle del dato fuente, que puede ser una ortofotografía de alrededor de 0.3 x 0.3 m de resolución. Debemos buscar en los metadatos de la fuente para saber cuál es la escala de salida de este tipo de imagen para tener una idea del nivel de exactitud. A ese error se le sumarán otros errores al derivar datos de este tipo de imágenes.

Como mencionamos antes, tanto PostGIS y todos los programas que utilizan las bibliotecas [GEOS](#) y el estándar simple features, no tienen procedimientos para establecer umbrales de tolerancia como en GRASS o ArcInfo (ahora ArcGIS con [cluster tolerance](#)).

Sin embargo, esto no es excusa para descartar estos productos. Las funcionalidades que nos ofrecen han sido de mucha utilidad. Por otro lado, PostGIS está desarrollando topología persistente y se espera que fortalezcan este tipo de modelo para representar geodatos en un futuro cercano.

Una opción que se ofrece es usar la función [ST_Snap](#)(geomA,geomB, dist). Esto es bien parecido al caso de usar cluster tolerance en ArcGIS, dando prioridad a la geometría de A. **Los vértices de la geometría de B serán “pegados” a los vértices de la geometría de A, siempre y cuando estén dentro de la distancia de tolerancia.**

Si el query anterior nos devolvió múltiples tipos de geometría:

```

/* Tipos de geometría que devuelve ST_Intersection
 * al combinar geodatos landslides y put2015
 * condición: where st_intersects(a.geom,b.geom)
 */
select geometrytype(geom) as tipo_geom, --trae tipo geometría
count(*) -- conteo de elementos
from (select st_intersection(a.geom,b.geom) as geom --trae la intersección geom de a y b as geom
      from g23_riesgo_geol_landslides_villalba a, --de las tablas landslides as a
      g29_planes_put_2015_villalba b --y put2015_villalba as b
      where st_intersects(a.geom,b.geom) --condición: geometrías que se intersequen entra a y b
      ) as tabla_subquery --guarda el resultado en un subquery, alias tabla_subquery
group by tipo_geom; --agrupar datos por columna tipo_geom

```

tipo_geom	count
GEOMETRYCOLLECTION	2
LINESTRING	3
MULTIPOLYGON	354
POINT	6
POLYGON	1,919

5 row(s) fetched - 20s on 2024-03-04 at 09:26:58



Tutorial PostGIS, 3.x

Al usar **ST_Snap** con un umbral de tolerancia escogida de 0.25 metros para una escala de 1:1,000, solo devuelve geometrías tipo POLYGON si lo insertamos en el siguiente query modificado del anterior.

- Modifique el query anterior en insertando la función ST_Snap en el **Script-1 de**

DBeaver

```
1 /* Tipos de geometría que devuelve ST_Extract con ST_Intersection
2  * al combinar geodatos landslides y put2015
3  * condición: where ST_Intersects(a.geom,b.geom)
4  */
5 select geometrytype(geom) as tipo_geom, --trae tipo geometría
6 count(*) --trae conteos de elementos geométricos vacíos o no
7 from (select --iniciar subquery
8        st_intersection( --intersección geométrica de a y el resultado b de st_snap
9          st_snap(a.geom,b.geom,0.25), --st_snap: enganchar vértices de b en a
10         a.geom) --realiza la intersección geom de a y b
11        as geom -- alias geom
12 from g23_riesgo_geol_landslides_villalba b, --de las tablas landslides as a
13      g29_planes_put_2015_villalba a --y put2015_villalba as b
14 where st_intersects(a.geom,b.geom) --condición:que las geom a y b intersequen
15 ) as tabla_subquery --fin subquery: guarda el resultado, alias tabla_subquery
16 group by tipo_geom; --agrupar datos por columna tipo_geom
```

Results 1 ×

select geometrytype(geom) as tipo_geom | Enter a SQL expression to filter results (use Ctrl+Space)

tipo_geom	count
POLYGON	2,284

Refresh Save Cancel Export data 200 1

1 row(s) fetched - 3m 33s, on 2024-03-05 at 14:21:59

El conteo de 2,284 elementos incluye los 2,275 anteriores más los 9 Null.

Problema: Al añadir este proceso, resulta costoso; el tiempo de ejecución se incrementa a 3m con 33ss.

Una solución intermedia es **utilizar la función STX_Extract** preparada por Martínez Llario (2020). La podemos integrar cuando utilizemos las funciones ST_Intersection, ST_Buffer, ST_Difference, etcétera. Como vimos en este ejemplo pasado, la función STX_Extract “extrae” del resultado de ST_Intersection solamente las geometrías tipo “#2” = polígonos. El query ejecutó en 11 segundos.



Tutorial PostGIS, 3.x

- Modifique el query anterior en insertando la función STX_Extract en el **Script-1 de DBEaver**

```

1 /* Tipos de geometría que devuelve ST_Extract con ST_Intersection
2 * al combinar geodatos landslides y put2015
3 * condición: where ST_Intersects(a.geom,b.geom)
4 */
5 select geometrytype(geom) as tipo_geom, --trae tipo geometría
6 count(*) --trae conteos de elementos geométricos vacíos o no
7 from (select stx_extract(
8         st_intersection(a.geom,b.geom)--realiza la intersección geom de a y b
9         ,2) as geom--argumento "2": devuelve solo polígonos, alias geom
10 from g23_riesgo_geol_landslides_villalba a, --de las tablas landslides as a
11 g29_planes_put_2015_villalba b --y put2015_villalba as b
12 where st_intersects(a.geom,b.geom) --condición:que las geom a y b intersequen
13 ) as tabla_subquery --guarda el resultado en un subquery, alias tabla_subquery
14 group by tipo_geom; --agrupar datos por columna tipo_geom

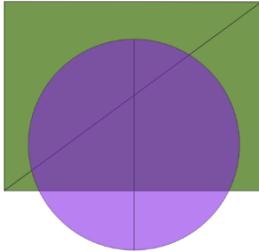
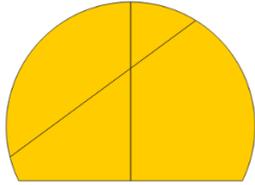
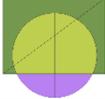
```

La solución ideal sería que se pudiese establecer tolerancias en estas operaciones y hacerlas más eficientes.

Análisis geoespacial

En adelante vamos a usar la función **STX_Extract** cuando sea pertinente en los queries de esta sección de análisis geoespacial, como vimos, para homogenizar las geometrías de los resultados.

INTERSECCIÓN

Layers de entrada		Layer resultante (A ∩ B)
 Layer A		
 Layer B		 Geometrías originales y resultado en amarillo
Dimensiones	Dimensiones posibles: 0, 1 o 2 Los inputs pueden tener geometrías diferentes	La dimensión de salida va a ser igual o menor que los layers de entrada
Atributos, campos, o columnas	--	Los campos de salida serán la combinación de los campos de A y B .

Ejemplo:

- Realice un **cruce de geodatos** (mediante intersección) entre el geodato de zonas inundables y el geodato de tipos de suelos.
Escriba y ejecute los siguientes queries en el **Script-1 de DBEaver**:



Tutorial PostGIS, 3.x

```

1 create table g00_overlays_inter_soil_zi --crear tabla inter_soil_zi
2 (
3 gid serial primary key, --gid para id interno pk
4 su_gid integer, --su_gid:id geodato suelos
5 musym varchar,muname varchar, farm_class varchar, --otros campos suelos
6 zi_gid integer, --zi_gid:id geodato zonas inundables
7 fld_zone varchar,zone_subty varchar, --campos zonas inundables
8 geom geometry(multipolygon,6566); --geom guardará geometrias resultantes
9 -- fin del enunciado CREATE TABLE
10 insert into g00_overlays_inter_soil_zi --inserta en tabla inter_soil_zi
11 (su_gid,musym,muname,farm_class, --campos de tabla suelos as su
12 zi_gid,fld_zone,zone_subty,geom) --campos de tabla zonas inundables as zi
13 --Usar select query para llenar las filas de la nueva tabla, no lleva ";"
14 select su.gid,musym,muname,farm_class, --query que inserta las filas
15 zi.gid,fld_zone,zone_subty, --con las columnas mencionadas,excepto gid
16 --no incluir campo gid. Se llenará automáticamente:datatype serial
17 stx_extract( --stx_extract para extraer de la ...
18 st_intersection(su.geom,zi.geom)--intersección de suelos y z_inundables y...
19 ,2)
20 as geom --geometría resultante alias geom
21 from g15_suelos_soil_map_units_2018_villalba as su, --tabla suelos alias su
22 g23_riesgo_inunda_floodzones_2017_villalba as zi --zonas inundables alias zi
23 where su.geom && zi.geom --comparación de bounding boxes (índice)
24 and st_relate(su.geom,zi.geom,'T*****');--cruzar solo interiores
25 --fin del select query
26 --fin del enunciado INSERT INTO
27 select * from g00_overlays_inter_soil_zi; --corroborar el resultado en grid y visor

```

La primera instrucción (que termina con ;) y la segunda, tardaron 3 segundos. Lo que tardó 0.169s fue el query de select * al final.

Lista de filas y columnas. Verá que los campos seleccionados de ambas tablas están integrados en la nueva tabla g00_overlays_inter_soil_zi

gid	su_gid	musym	muname	farm_class	zi_gid	fld_zone	zone_subty	geom
1	257	MuF2	Múcara silty clay, 40 to 60	Not prime farmland	1	A		MULTIPOLYGON (
2	175	MsC	Montegrande clay, 2 to 12	All areas are prime farmla	1	A		MULTIPOLYGON (
3	256	CaD2	Quebrada silty clay loam	Farmland of statewide im	1	A		MULTIPOLYGON (

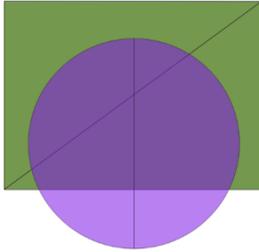
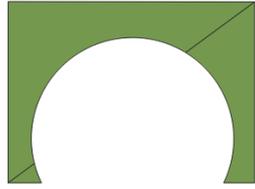
El proceso de escribir código al inicio es tedioso y susceptible a cometer errores. Por el lado positivo, hay mayor control sobre el resultado y además el procedimiento queda registrado en el código. Si comenta el código es mejor aún porque cuando lo vuelva a ver, entenderá qué hace cada parte, además que otros podrán entender la lógica del código escrito.



Tutorial PostGIS, 3.x

ERASE

[Erase](#) es un geoproceto que remueve el área de solape entre la geometría de un geodato A y un geodato B.

Layers de entrada		Layer resultante (A-B)
 Layer A		
 Layer B		
Dimensiones geométricas	Dimensiones posibles: 0, 1 o 2 La dimensión del layer B debe ser igual a, o superior a la dimensión del layer A.	El resultado será de la misma dimensión de la capa A.
Conmutativa	Es conmutativa. No importa el orden.	--
Atributos, campos, o columnas	--	Conserva los campos del layer A.

Este geoproceto se realiza en PostGIS mediante la combinación de dos funciones: [ST_Difference](#) y [ST_Union](#). No debe usarse solamente ST_Difference (como en el geoproceto Difference de QGIS), porque ST_Difference usada sola va a producir polígonos erróneos. Se recomienda usar ST_Union para crear solo una geometría en el layer B y luego sobre imponerla al layer A. Este proceso de unión puede ser ineficiente si generamos polígonos grandes con miles de vértices. Lo más recomendable es: **quitar a cada polígono del layer A el área de intersección de la unión de los polígonos del layer B.** En otras palabras, es **enfocar la búsqueda y geoproceto solo en el área de solape.**

Ejemplo: Usar geometrías de **suelos urbanos, urbano-programado, urbano-no-programado** del geodato del **plan de usos de terrenos, 2015** para **recortarlas (erase)** del geodato del **catastro de suelos de USDA-NRCS, 2018.**

- Escriba y ejecute los siguientes queries en el **Script-1 de DBEaver:**

```

1 create table g00_overlay_erase_suelos_menos_urbanos
2 (
3   gid serial primary key,
4   musym varchar, muname varchar, farm_class varchar,
5   geom geometry(multipolygon,6566)
6);
7 --comenzar enunciado insert into
8 insert into g00_overlay_erase_suelos_menos_urbanos --insertar filas en tabla
9 (musym,muname,farm_class,geom)
10 select musym,muname,farm_class, geom
11 from
12   (select s.gid,musym,muname,farm_class,
13     stx_extract(st_difference(s.geom,st_union(u.geom)),2) as geom
14   from g15_suelos_soil_map_units_2018_villalba s,
15     g29_planes_put_2015_villalba u
16   where clasiput like 'SU%' or clasiput = 'VIAL'
17   and s.geom && u.geom
18   and st_relate(s.geom,u.geom, 'T*****')
19   group by s.gid
20   ) as subq
21 where geom is not null;

```



Tutorial PostGIS, 3.x

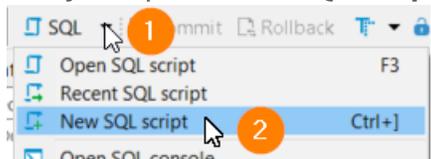
QUÉ ES DEPENDENCIA FUNCIONAL

Antes de seguir, aprovechamos para explicarlo. Desde la versión 9.1 de PostgreSQL se permite usar solo el campo de identificadores, en este caso, gid en la cláusula GROUP BY y evitar declarar todos los campos que están contenidos en la cláusula SELECT.

Continuando...

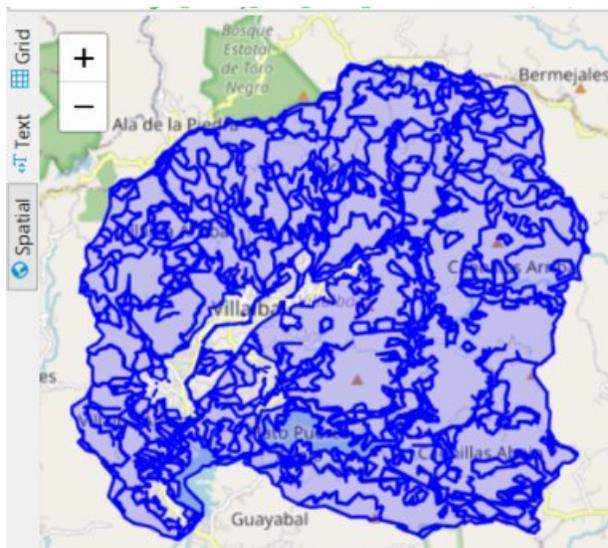
Para ver la tabla que acabamos de generar escriba lo siguiente, en la próxima línea debajo del último query:

- Abra otra pestaña (tab) para escribir un nuevo script. Haga **click** en el botón **SQL**, y escoja la opción **New SQL script** o teclee **ctrl+]**.



- Escriba: **select * from g00_overlay_erase_suelos_menos_urbanos;**

Grid	gid	musym	muname	farm_class	geom
1	1	LuE	Los Guineos clay, 20 to 40 percent slopes	Farmland of statewide importance	MULTIPOLYGON (((195429.542 236015.61...
2	2	LuF	Los Guineos clay, 40 to 60 percent slopes	Not prime farmland	MULTIPOLYGON (((190094.6931989159 23...
3	3	QeD2	Quebrada silty clay loam, 12 to 20 percent	Farmland of statewide importance	MULTIPOLYGON (((200514.9299 235987.0...
4	16	LeC	Lares clay, 5 to 12 percent slopes	All areas are prime farmland	MULTIPOLYGON (((191868.105 231244.64...
5	4	CbF2	Caguabo gravelly clay loam, 20 to 60 percent	Not prime farmland	MULTIPOLYGON (((197257.92 235870.308...
6	5	QeF2	Quebrada silty clay loam, 40 to 60 percent	Not prime farmland	MULTIPOLYGON (((191177.035 230308.63...



Note los espacios vacíos en el área urbana del Municipio. Estos corresponden a los suelos reglamentados para uso urbano actual, programado y no programado, si acerca más, verá que también se recortaron geometrías de suelos por el sistema viario:



- Comparamos área original...
- Borre el query anterior y escriba:

```
25 select sum(st_area(o.geom))/1000000 as area_orig_sqkm
26 from g15_suelos_soil_map_units_2018_villalba o;
```



Tutorial PostGIS, 3.x

Use `ctrl+\` para que el query aparezca en otro tab

```
Grid
-----+
95.87616038282688|
```

- Más abajo escriba este otro query para calcular el área del con áreas recortadas...

```
28 select sum(st_area(n.geom))/1000000 area_wo_urban_sqkm
29 from g00 overlay erase suelos menos urbanos n; **NO lo ejecute aún**
```

- Sombree solo este último query y teclee `ctrl+\` para ejecutar el script SQL en otra pestaña de resultados, para no borrar el resultado anterior.

```
26 --calcula área total del geodato recortado
27 select sum(st_area(n.geom))/10000 area_wo_urban_sqkm
28 from g00 overlay erase suelos menos urbanos n;
```

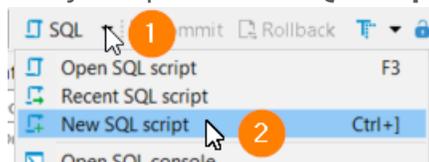
```
Grid
-----+
88.10095514603212|
```

¿Cuánta área de suelos potencialmente agrícolas se han perdido o se perderán para ser usados como suelo urbanizado?

Ya que hicimos el proceso **Erase**, podemos contestar la pregunta con dos queries:

Área en **prime farm land en todas sus variedades y farm land of statewide importance**:

- Abra otra pestaña (tab) para escribir un nuevo script. Haga **click** en el botón **SQL**, y escoja la opción **New SQL script** o teclee `ctrl+]`.



- Escriba el siguiente query combinado usando la cláusula UNION en la nueva pestaña.

```
1 --calcula cuántas hectáreas y cuerdas hay en prime farmland, dato original
2 select 'original' as geodato, sum(st_area(o.geom))/10000 as prime_farmland_ha, --suma área geometrías en hectáreas
3 sum(st_area(o.geom))/3930.395625 as prime_farmland_cuerdas --suma área geometrías en cuerdas
4 from g15 suelos soil_map_units_2018 villalba o --de la tabla original de suelos
5 where farm_class not like 'Not prime%' --condiciones: like '%importance' suelos de importancia agrícola...
6 union --ya que los queries tienen columnas iguales, podemos usar cláusula union y mostrar solo una tabla
7 --calcula cuántas hectáreas y cuerdas hay en prime farmland, dato erased
8 select 'erased', sum(st_area(n.geom))/10000 as prime_farmland_ha, --suma área geometrías en hectáreas
9 sum(st_area(n.geom))/3930.395625 as prime_farmland_cuerdas --suma área geometrías en cuerdas
10 from g00 overlay erase suelos menos urbanos n --de la tabla erased de suelos
11 where farm_class not like 'Not prime%'; --condiciones: like '%importance' suelos de importancia agrícola...
```

UNION: La cláusula UNION del lenguaje SQL usada aquí, sirve para unir dos conjuntos de filas que tengan igual estructura (las mismas columnas). Lo ejecutamos en una sola corrida.

Resultados, query: Área estimada en suelos potencialmente agrícolas:

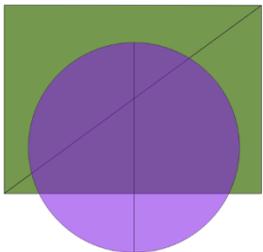
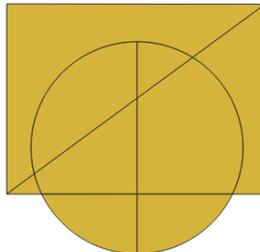
Grid	abc geodato	123 prime_farmland_ha	123 prime_farmland_cuerdas
1	erased	727.101715839	1,849.945362279
2	original	1,066.4514164667	2,713.3436890763

En vista de esto, **se ha perdido o se perderán hasta 863.4 cuerdas de suelo potencialmente agrícola** para ser usados como suelo urbanizado.



UNIÓN DE LAYERS

Este geoproceso se usa para unir las geometrías de dos geodatos, además de integrar los atributos que queramos extraer de ambas tablas de atributos asociados.

Layers de entrada		Layer resultante (A U B)
 Layer A  Layer B		
Dimensiones geométricas	Dimensiones posibles: 0, 1 o 2 Los layers deben tener igual dimensión geométrica. (p/p, l/l, s/s)	El resultado será de la misma dimensión de los layers A y B .
Conmutativa	Es conmutativa. No importa el orden.	--
Atributos, campos, o columnas	--	Los campos de salida serán la combinación de los campos de los layers A y B .

Este proceso no debe confundirse con la función [ST_Union](#). Ya usamos ST_Union en queries anteriores para integrar todas las geometrías de un geodato en una sola.

En cambio, el geoproceso 'Unión de layers' trabaja como un Intersect de layers, pero conservará además las geometrías que no intersequen ente ambos layers. Conserva todas las geometrías de A, más todas las geometrías de B, además de los atributos que queramos conservar de ambos layers.

El proceso **Unión de layers** sería descrito gráficamente de la siguiente forma:



Estos pasos se realizan en queries separados y corridos secuencialmente. **El geoproceso Unión de layers es conmutativo**, pero las funciones ST_Difference a usarse no lo serán, porque va a realizarse A – B y B – A, luego de la intersección $A \cap B$. La intersección $A \cap B$ sí es conmutativa.

Hagamos la unión de los geodatos de **riesgo a deslizamientos** con el geodato de **riesgo por inundaciones**. Las tablas tendrán **alias: deslizamientos as d y zonas inundables as i**.



Tutorial PostGIS, 3.x

Estos son los atributos de ambas tablas:

g23_riesgo_geol_landslides_1979_villalba	
Columns	
123	gid (serial4)
6566	geom (public.geometry(multipolygon, 6566))
1	risk_level (varchar(1))
80	description (varchar(80))

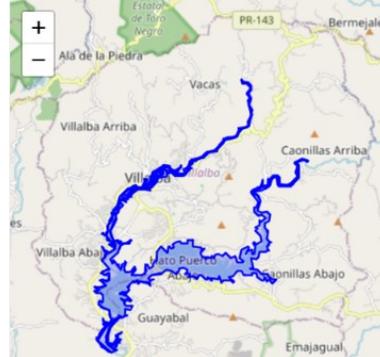
g23_riesgo_inunda_floodzones_2017_villalba	
Columns	
123	gid (serial4)
6566	geom (public.geometry(multipolygon, 6566))
	fld_zone (varchar)
8	st_bfe_m (float8)
	v_datum (varchar)
4	fzn_1pct_id (int4)
8	depth_m (float8)
	zone_subty (varchar)

Los **campos** que deseamos **conservar** serán **d.gid, d.description, d.risk_level, i.gid, i.fld_zone, i.zone_subty**. La geometría resultante de ambos campos geom de los layers de entrada se guardará en el nuevo geodato combinado.

Riesgo por deslizamientos de terreno



Riesgo por inundaciones



- Puede ver estos layers y explorar datos en DBEaver. Las sentencias SQL son:
select * from g23_riesgo_geol_landslides_villalba; --para layer deslizamientos
select * from g23_riesgo_inunda_floodzones_2017_villalba; --para layer zonas inundación
- Haga **click** en la pestaña **Spatial** para que pueda ver los geodatos en el mapa.

gid	geom
1	MULTIPOLYGON (((195913.3524999991
2	MULTIPOLYGON (((191240.120099999 2
3	MULTIPOLYGON (((197816.2562 227474
4	MULTIPOLYGON (((200309.4377 231089
5	MULTIPOLYGON (((198701.4361 237882
6	MULTIPOLYGON (((196225.7405999973-
7	MULTIPOLYGON (((199327.9286 237605
8	MULTIPOLYGON (((199034.6259000003:

Como mencionamos anteriormente, el geoprocso Unión de layers está compuesto de tres pasos:

- 1) **A - B**, usando **ST_Difference(geomA, ST_Union(geomB)) +**
- 2) **A ∩ B**, usando **ST_Intersection(geomA, geomB) +**
- 3) **B - A**, usando **ST_Difference(geomB, ST_Union(geomA))**

Hagamos el proceso por partes:

- Asegúrese que esté trabajando en el schema **pr_geodata**. De lo contrario, el resultado estará guardado en el schema *public*.



Primero: definir la nueva tabla que recibirá los datos desde queries que vamos a ejecutar.



Tutorial PostGIS, 3.x

```

1@create table g00_overlay_union_desliz_inunda --nueva tabla
2 (gid serial primary key, --primary key nueva tabla
3 d_gid integer, d_risk_level varchar, d_description varchar, --campos que recibirán datos de tabla deslizamientos
4 i_gid integer, i_fld_zone varchar, i_zone_subty varchar, --campos que recibirán datos de tabla zonas inundables
5 geom geometry (multipolygon, 6566)); --geometria multipoligonos, EPSG:6566 NAD83(2011)
6

```

Parte A – B consiste en dos subpartes; extraer las geometrías que intersecan ...

```

7@--Paso difference (A-B), (d-i) en este caso
8 --1a parte: poligonos tabla "d" que son borrados parcial o totalmente por poligonos tabla "i"
9 insert into g00_overlay_union_desliz_inunda --insertar filas en esta tabla
10 (d_gid, d_risk_level, d_description, geom) --para los campos d_gid,d_risk_level,d_descripcion,geom tabla nueva
11 select gid, risk_level, description, geom --la inserción se basa en el query a continuación
12 --los campos corresponden a campos de tablas originales
13 from
14 (select d.gid,d.risk_level,d.description,--campos tabla desliz
15 stx_extract(st_difference(d.geom, --función stx_extract para extraer multipol usando st_diff(d.geom y ...)
16 st_union(i.geom)),2) as geom --la unión de todas las geometrías de i.geom, #2= extrae multipolygon
17 from g23_riesgo_geol_landslides_1979_villalba as d, --tabla desliz
18 g23_riesgo_inunda_floodzones_2017_villalba as i --tabla zonas inundables
19 where d.geom && i.geom --comparación de bounding boxes
20 and st_relate(d.geom,i.geom,'T*****') --st_relate: que las geoms solo intersequen interiores
21 group by d.gid --agrupa por campos tabla d (dependencia funcional)
22 ) as subq --el nombre o alias del subquery:subq
23 where geom is not null; --condición: que las geometrías no estén vacías (null)

```

y luego extraer las geometrías que no intersecan.

```

25@--2a parte: poligonos tabla d que no son tocados por poligonos tabla i
26 insert into g00_overlay_union_desliz_inunda --insertar filas en esta tabla
27 (d_gid, d_risk_level, d_description, geom) --para los campos d_gid,d_risk_level,d_descripcion,geom tabla nueva
28 select gid, risk_level, description, geom --campos tabla desliz
29 from --iniciar subquery
30 (select d.gid,d.risk_level,d.description,i.gid as i_gid, --campos tabla desliz
31 st_multi(d.geom) as geom --convertir geometrías a tipo multipolygon
32 from g23_riesgo_geol_landslides_1979_villalba as d --tabla desliz
33 left join g23_riesgo_inunda_floodzones_2017_villalba as i --tabla zonas inundables
34 on d.geom && i.geom --comparador de bounding boxes
35 and st_relate(d.geom,i.geom,'T*****') --st_relate: que las geoms solo intersequen interiores
36 ) as subq --el nombre o alias del subquery:subq
37@where i_gid is null; --condición: que en el campo i_gid no haya filas vacías (null)
38 --Fin de paso Difference (A-B) ó (d-i)

```

Parte A ∩ B (intersección geométrica)

```

40@--Paso Intersection (A n B)
41 insert into g00_overlay_union_desliz_inunda --insertar filas en esta tabla
42 (d_gid,d_risk_level,d_description,--campos tabla d
43 i_gid,i_fld_zone,i_zone_subty,geom)--campos tabla i, más la de geometría (a n b)
44 select d.gid,d.risk_level,d.description,--campos tabla d
45 i.gid,i.fld_zone,i.zone_subty,--campos tabla i
46 stx_extract(st_intersection(d.geom,i.geom),2) as geom -- extraer multipolygons de geometría (a n b)
47 from g23_riesgo_geol_landslides_1979_villalba as d, --tabla desliz
48 g23_riesgo_inunda_floodzones_2017_villalba as i --tabla zonas inundables
49 where d.geom && i.geom --condición: comparador de bounding boxes and ...
50@ and st_relate(d.geom,i.geom,'T*****');--st_relate: que las geoms solo intersequen interiores
51 --fin paso Intersection

```

Parte B – A consiste en dos subpartes: extraer las geometrías que intersecan ...

```

53@--Paso difference (B-A)
54 --1a parte: poligonos tabla i que son borrados parcial o totalmente por poligonos tabla d
55 insert into g00_overlay_union_desliz_inunda --insertar filas en esta tabla
56 (i_gid, i_fld_zone, i_zone_subty, geom) --para los campos i_gid, i_fld_zone, i_zone_subty, geom en tabla nueva
57 select gid, fld_zone, zone_subty, geom --la inserción se basa en el query a continuación
58 --los campos corresponden a campos de tablas originales
59 from --de este subquery:
60 (select i.gid,i.fld_zone,i.zone_subty, --campos, tabla i
61 stx_extract(st_difference(i.geom,--extrae multipoligonos de la diff entre geoms de i - unión geoms de d
62 st_union(d.geom)),2) as geom --guárdalas con alias geom
63 from g23_riesgo_inunda_floodzones_2017_villalba as i,--tabla zonas inundables
64 g23_riesgo_geol_landslides_1979_villalba as d --tabla desliz
65 where i.geom && d.geom --condición: comparador de bounding boxes and ..
66 and st_relate(i.geom,d.geom,'T*****') --st_relate: que las geoms solo intersequen interiores
67 group by i.gid --agrupa por campos tabla i (dependencia funcional)
68 ) as subq --el nombre o alias del subquery:subq
69 where geom is not null; --condición: que las geometrías no estén vacías

```

y luego extraer las geometrías que no intersecan.



Tutorial PostGIS, 3.x

```

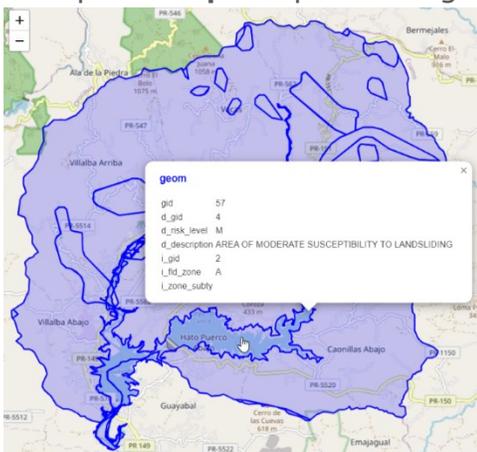
71 --2a parte: poligonos tabla i que no son tocados por poligonos tabla d
72 insert into g00_overlay_union_desliz_inunda --insertar filas en esta tabla
73 (i_gid, i_fld_zone, i_zone_subty, geom) --para los campos i_gid, i_fld_zone, i_zone_subty, geom en tabla nueva
74 select gid, fld_zone, zone_subty, geom --la inserción se basa en el query a continuación
75 from --de este subquery:
76 (select i_gid,i_fld_zone,i_zone_subty,d_gid as left_gid, --campos, tabla i y campo d_gid usado para contar filas
77 st_multi(i.geom) as geom --convierte geom de i en tipo multi
78 from g23_riesgo_inunda_floodzones_2017_villalba as i --tabla zonas inundables
79 left join --conservar todas las filas de la tabla i aunque no haya intersección
80 g23_riesgo_geol_landslides_1979_villalba as d --tabla desliz
81 on i.geom && d.geom --condición: comparador de bounding boxes and ..
82 and st_relate(i.geom,d.geom,'T*****') --st_relate: que las geoms solo intersequen interiores
83 ) as subq --el nombre o alias del subquery:subq
84 where left_gid is null; --condición: que el campo left_gid esté vacío.
85 --Fin de paso Difference (B-A)

```

Resultado:

- Escriba el siguiente query en otra pestaña de queries, ctrl+]


```
select * from g00_overlay_union_desliz_inunda;
```
- Use la pestaña Spatial para ver el geodato en el mapa:



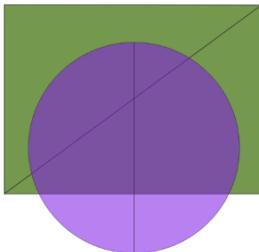
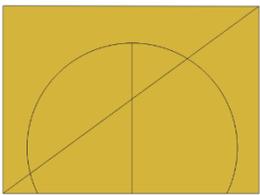
El geodato resultante puede mejorarse. Las áreas de lagos están catalogadas como de riesgo moderado. Esa era la clasificación que heredada del geodato original de deslizamientos. Al unirle el geodato de zonas inundables, podemos modificar o actualizar el resultado. Será apropiado entonces usar la cláusula UPDATE en esta tabla/geodato y modificar/actualizar los campos **d_risk_level**, cambiar el valor a **L**, igualmente, cambiar el valor **d_description** a **'AREA OF LOW SUSCEPTIBILITY TO LANDSLIDING'**. De igual manera pueden actualizarse las columnas de zonas inundables y añadir información a las áreas que no tienen datos.

Note que el **proceso final (B – A)** traerá las geometrías del geodato de zonas inundables **que están fuera** del geodato de deslizamientos de terreno.



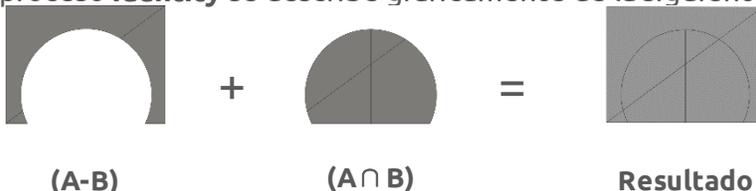
IDENTITY

Para el geoproceso Identity, lo que interesamos es preservar la geometría y atributos de uno de los layers de entrada: el input layer. Dicho de otro modo, es **obtener del layer Identity solo lo que corresponde geoméricamente con el layer Input, el cual mantendrá su extensión espacial.**

Layers de entrada		Layer resultante (A identity B)
 Layer A  Layer B		
Dimensiones geométricas	El layer A (input) puede ser dimensión: 0, 1 o 2 El layer B 'identity' será de igual dimensión que A.	El resultado será de la misma dimensión del layers A.
Conmutativa	No es conmutativa. El layer B es de <i>identidad</i> .	--
Atributos, campos, o columnas	--	Los campos de salida serán la combinación de los campos de los layers A y B.

El geoproceso 'Identity' se trata de una **diferencia $A - B$** , seguido por la **intersección $A \cap B$** . **Conserva todas las geometrías de A, más solo las geometrías de B que intersequen con A.**

El proceso **Identity** se describe gráficamente de la siguiente forma:

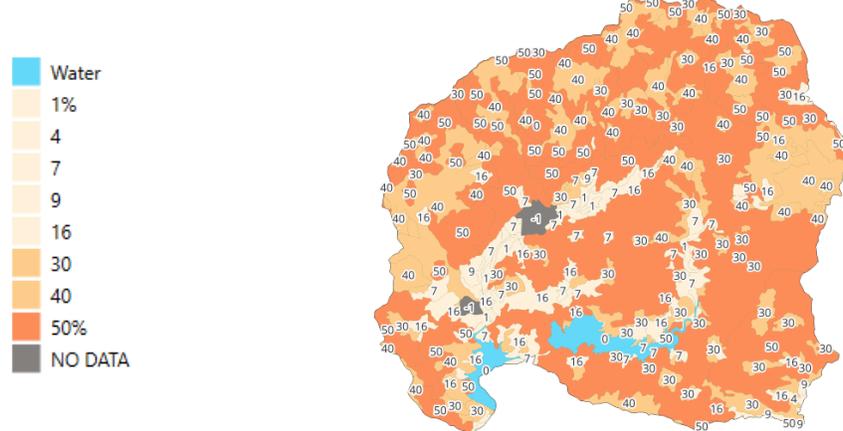


Estos pasos se realizan en queries separados y corridos secuencialmente.

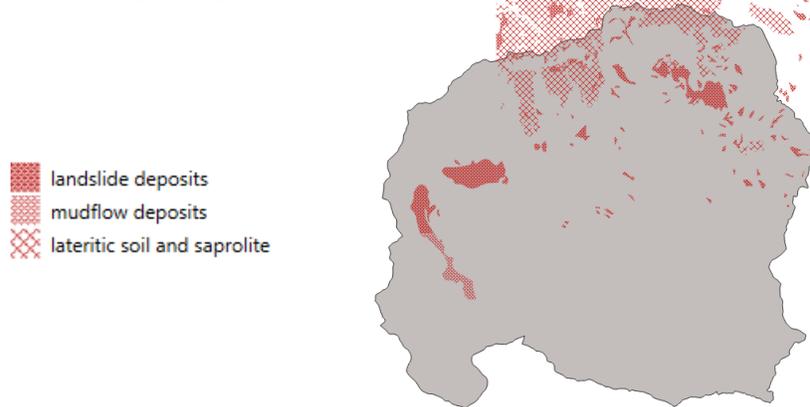
Para demostrar el geoproceso **Identity**, vamos a **producir un mapa de susceptibilidad a deslizamientos**, basado en el geodato del catastro de suelos del USDA-NRCS y el geodato de unidades geológicas de los cuadrángulos geológicos a escala 1:20,000.



Catastro de suelos: Slope gradient dominant component (%)



Unidades geológicas: Landslides, mudslides, saprolites, laterites debris



Ya que vamos a emular el geoproceso Identity, el geodato **input** será el de **suelos** y el geodato **identity** será el de **unidades geológicas**.

En la **pestaña gisdb_lab** para escribir queries, escriba los siguientes queries:

Sección para generar la tabla nueva:

```
1=drop table if exists g00_overlay_identity_landslides_2024_villalba cascade; --si la tabla existe, bórrala
2 create table g00_overlay_identity_landslides_2024_villalba --crear el nuevo geodato
3 (gid serial primary key, --gid campo interno para identificar
4 g_gid integer, --id del geodato geológico
5 g_quad_code varchar, --código de cuadrángulo geológico
6 g_unit_code varchar, --código de unidad geológica
7 g_unit_name varchar, --nombre de la unidad geológica
8 s_gid integer, --id del geodato de suelos
9 s_slope_gradient_dominant_component integer, --campo que tiene pendientes en por ciento del componente dominante
10 risk_level varchar, --nivel de riesgo: Luego de terminar se calculará basado en geología y pendientes-suelos
11 geom geometry (multipolygon, 6566)); --campo geometrias, multipolígono, SPCS NAD83(2011) EPSG:6566
```



Sección Difference (A-B)

```

13@--Geoproceso Identity
14 --consiste de (A-B) + (A n B)
15 --(A-B) consiste de dos partes
16 --areas que solapan entre geodatos + áreas que no solapan
17 --Paso difference (A-B), (d-l) en este caso
18 --1a parte: polígonos tabla "s" que son borrados parcial o totalmente por polígonos tabla "l"
19 insert into g00_overlay_identity Landslides 2024 villalBa --insertar filas en esta tabla
20 (s_gid, s_slope_gradient_dominant_component, geom) --para los campos d_gid,d_risk_level,d_descripcion,geom tabla nueva
21 select gid, slope_gradient_dominant_component, geom --la inserción se basa en el query a continuación
22 --los campos corresponden a campos de tablas originales
23 from
24 (select s.gid,s_slope_gradient_dominant_component,--campos tabla suelos
25 stx_extract(st_difference(s.geom, --función stx_extract para extraer multipol usando st_diff(d.geom y ...)
26 st_union(l.geom)),2) as geom --la unión de todas las geometrías de i.geom, #2= extrae multipolygon
27 from g15_riesgo_soil_slopes_2018_villalba as s, --tabla suelos
28 g15_riesgo_geologia_mapping_units_20k_landslides as l --tabla landslides
29 where s.geom && l.geom --comparación de bounding boxes
30 and st_relate(s.geom,l.geom,'T*****') --st_relate: que las geoms solo intersequen interiores
31 group by s.gid --agrupa por campos tabla d (dependencia funcional)
32 ) as subq --el nombre o alias del subquery:subq
33@where geom is not null; --condición: que las geometrías no estén vacías (null)
34 select * from g00_overlay_identity Landslides 2024 villalBa;
35
36@--2a parte: polígonos tabla s que no son tocados por polígonos tabla l
37 insert into g00_overlay_identity Landslides 2024 villalBa --insertar filas en esta tabla
38 (s_gid, s_slope_gradient_dominant_component, geom) --para los campos de la tabla nueva
39 select gid, slope_gradient_dominant_component, geom --campos tabla suelos
40 from --iniciar subquery
41 (select s.gid,s_slope_gradient_dominant_component,l.gid as l_gid, --campos tabla suelos y landslides
42 st_multi(s.geom) as geom --convertir geometrías a tipo multipolygon
43 from g15_riesgo_soil_slopes_2018_villalba as s --tabla deslíz
44 left join g15_riesgo_geologia_mapping_units_20k_landslides as l --tabla deslizamientos
45 on s.geom && l.geom --comparador de bounding boxes
46 and st_relate(s.geom,l.geom,'T*****') --st_relate: que las geoms solo intersequen interiores
47 ) as subq --el nombre o alias del subquery:subq
48@where l_gid is null; --condición: que en el campo l_gid no haya filas vacías (null)
49 --Fin de paso Difference (A-B) ó (d-i)
50 select * from g00_overlay_identity Landslides 2024 villalBa; --ver para comprobar
51
52@--Paso Intersection (A n B)
53 insert into g00_overlay_identity Landslides 2024 villalBa --insertar filas en esta tabla
54 (s_gid,s_slope_gradient_dominant_component,--campos tabla s
55 g_gid,g_quad_code,g_unit_code,g_unit_name, geom)--campos tabla g, más la de geometría (a n b)
56 select s.gid,s_slope_gradient_dominant_component,--campos tabla s
57 g.gid,quad_code,unit_code,unit_name,--campos tabla g
58 stx_extract(st_intersection(s.geom,g.geom),2) as geom -- extraer multipolygons de geometría (a n b)
59 from g15_riesgo_soil_slopes_2018_villalba as s, --tabla deslíz
60 g15_riesgo_geologia_mapping_units_20k_landslides as g --tabla deslizamientos
61 where s.geom && g.geom --condición: comparador de bounding boxes and ...
62@ and st_relate(s.geom,g.geom,'T*****'); --st_relate: que las geoms solo intersequen interiores
63 --fin paso Intersection
64 select * from g00_overlay_identity Landslides 2024 villalBa; --ver para comprobar

```

Sección intersection

```

52@--Paso Intersection (A n B)
53 insert into g00_overlay_identity Landslides 2024 villalBa --insertar filas en esta tabla
54 (s_gid,s_slope_gradient_dominant_component,--campos tabla s
55 g_gid,g_quad_code,g_unit_code,g_unit_name, geom)--campos tabla g, más la de geometría (a n b)
56 select s.gid,s_slope_gradient_dominant_component,--campos tabla s
57 g.gid,quad_code,unit_code,unit_name,--campos tabla g
58 stx_extract(st_intersection(s.geom,g.geom),2) as geom -- extraer multipolygons de geometría (a n b)
59 from g15_riesgo_soil_slopes_2018_villalba as s, --tabla deslíz
60 g15_riesgo_geologia_mapping_units_20k_landslides as g --tabla deslizamientos
61 where s.geom && g.geom --condición: comparador de bounding boxes and ...
62@ and st_relate(s.geom,g.geom,'T*****'); --st_relate: que las geoms solo intersequen interiores
63 --fin paso Intersection
64 select * from g00_overlay_identity Landslides 2024 villalBa; --ver para comprobar

```

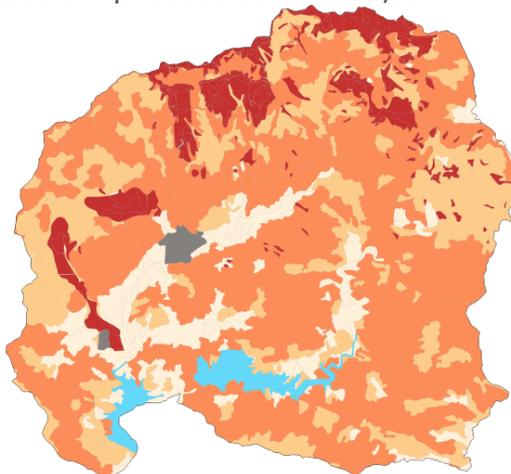


Tutorial PostGIS, 3.x

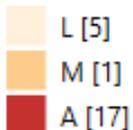
Si todo está bien, el próximo paso es **actualizar** el campo **risk_level** con los niveles que asignaremos mediante la cláusula **UPDATE** y la función **CASE-WHEN-END**.

```
67 --luego de comprobar que el geodato está correcto,
68 --hacemos update al campo risk_level y asignarle los valores de susceptibilidad
69 --basado en la geología y las pendientes.
70 update g00_overlay_identity_landslides_2024_villalba --actualiza esta tabla
71 set risk_level = --el campo risk_level
72 CASE WHEN g_gid is not NULL --cuando g_gid no esté vacío,
73 THEN 'Very High' --asigna 'Very High' a risk_level
74 WHEN s_slope_gradient_dominant_component >= 50 --cuando este campo sea >=50%,
75 THEN 'High' --asigna 'High' a risk_level
76 WHEN s_slope_gradient_dominant_component between 30 and 49 --cuando este campo esté entre 30 y 49%,
77 THEN 'Moderate' --asigna 'Moderate' a risk_level
78 WHEN s_slope_gradient_dominant_component between 1 and 29 --cuando este campo esté entre 1 y 29%,
79 THEN 'Low' --asigna 'Low' a risk_level
80 WHEN s_slope_gradient_dominant_component = 0 --cuando este campo = 0,%
81 THEN 'Water' --asigna 'Water' a risk_level
82 WHEN s_slope_gradient_dominant_component = -1 --cuando este campo = -1,
83 THEN 'Unknown' --asigna 'Unknown' a risk_level
84 end;
```

Ver el resultado en QGIS, con datos compilados a escala 1:20,000:



Comparar con el mapa original de 1979 compilado a escala 1:250,000:



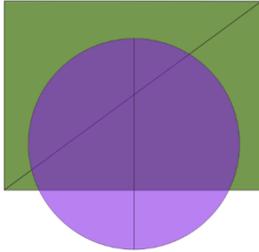
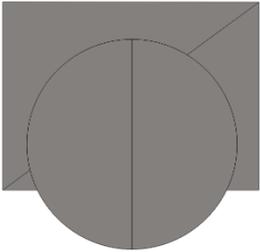
El geodato combinado de geología y suelos puede ser actualizado en términos de pendientes, si le hacemos un overlay con un ráster de pendientes y aplicamos [ST_SummaryStats](#).



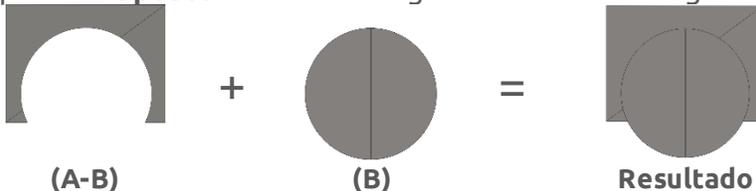
Tutorial PostGIS, 3.x

UPDATE

Este geoprocreso produce la intersección geométrica de las geometrías de entrada y añade las geometrías del geodato de actualización. Los atributos (que sean iguales) y geometrías del layer actualizado serán integrados en el layer de salida.

Layers de entrada		Layer resultante (A update B)
 Layer A		
 Layer B		
Dimensiones geométricas	El layer A 'input' y layer B 'update' deben ser de dimensión 2 .	El resultado será de la misma dimensión que el layer A .
Conmutativa	No es conmutativa. El layer B es de <i>actualización</i> .	--
Atributos, campos, o columnas	--	Los campos de salida serán los que contiene el layer A y los campos de B que sean iguales a los del layer A .

El proceso **Update** sería descrito gráficamente de la siguiente forma:

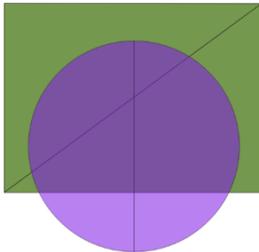
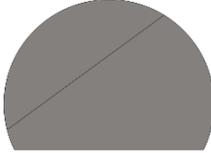


Estos pasos se realizan mediante queries corridos secuencialmente.



CLIP

Este es un proceso de **extracción** donde la forma o geometría de layer-clip (B) servirá para **recortar** las geometrías del layer input (A).

Layers de entrada		Layer resultante (A update B)
 Layer A  Layer B		
Dimensiones geométricas	El layer A 'input' puede ser de dimensión 0, 1 o 2 . El layer B debe ser de dimensión 2 .	El resultado será de la misma dimensión que el layer A .
Conmutativa	No es conmutativa . El layer B es el layer de <i>recorte</i> .	--
Atributos, campos, o columnas	--	Los campos de salida serán los que contiene el layer A .

Antes de realizar el corte, se aplicará **ST_Union** para consolidar las geometrías de B, si es que contiene más de un polígono. **Si el layer B es de un solo polígono**, podemos usar solamente **ST_Intersection(A,B)**.

Vamos a llevar a cabo dos queries emulando el geoproceso Clip sin hacer una tabla permanente como las anteriores. Solo para propósitos de demostración. Uno se hará sin la función **ST_Union** porque ya es un solo polígono (área municipal). El segundo query utilizará **ST_Union** (geodato de barrios, Villalba).

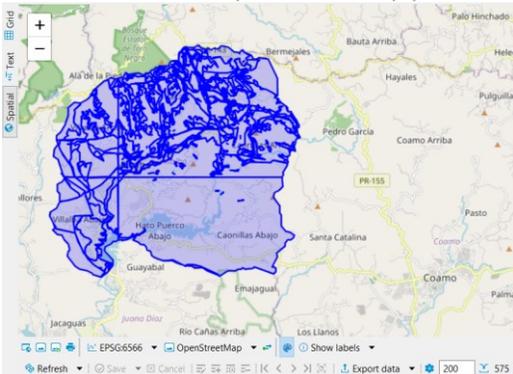
Query sin usar ST_Union:

En la pestaña para escribir queries **<gisdb_lab>** escriba el siguiente query

```

1 select g.gid as g_gid, g.quad_code, g.unit_code, g.unit_name, --traer columnas geodato geológico
2 g."period",g.epoch, g.unit_descript, g.quad_name,g.geol_id, --traer columnas geodato geológico
3 stx_extract(st_intersection(g.geom,v.geom),2) as geom --intersección geom sin st_union. v es un solo poligono,
4 --más usar stx_extract,2 para traer sólo polígonos
5 from g15_geologia_mapping_units_20k_2018 g, --geodato geologia 1:20k
6 g03_legales_municipios_2015_villalba v --geodato área municipal, Villalba
7 where g.geom && v.geom --comparador de bounding boxes para usar índice espacial
8 and st_relate(g.geom,v.geom, 'T*****'); --intersecar solo interiores de geometrías
9 --si no st_relate con patrón 'T*****', devolverá además otras filas con geometría null
10 --use ctrl+shift+= (fetch all data) para traer todos los datos

```



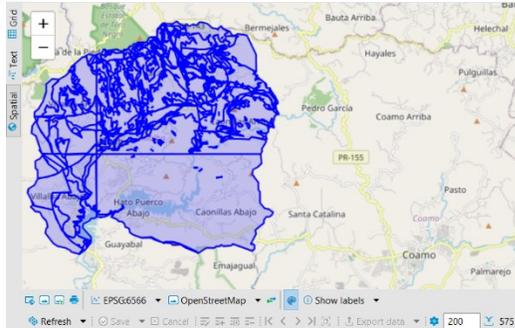
Si queremos hacer un geodato permanente, debemos usar **CREATE TABLE** como en los procesos anteriores. Luego use **INSERT INTO**, usando el query que está escrito arriba. Note que el nombre del campo "period" va encerrado en comillas dobles.



Tutorial PostGIS, 3.x

Query con ST_Union:

```
13 select g.gid as g_gid, g.quad_code, g.unit_code, g.unit_name, --traer columnas geodato geológico
14 g."period",g.epoch, g.unit_descript, g.quad_name,g.geol_id, --traer columnas geodato geológico
15 stx_extract(st_intersection(g.geom,(st_union(b.geom))),2) as geom --intersección geom con st_union
16 --porque b está compuesto de 8 polígonos (barrios), más usar stx_extract,2 para traer sólo polígonos
17 from q15_geologia_mapping_units_20k_2018 g, --geodato geología 1:20k
18 q03_legales_barrios_2015_villalba b --geodato barrios, Villalba
19 where g.geom && b.geom --comparador de bounding boxes para usar índice espacial
20 and st_relate(g.geom,b.geom, 'T*****') --intersecar solo interiores de geometrías
21 --para no devolver filas con geometría null
22 group by g.gid; --ya que usamos ST_Union, debemos agrupar por g.gid (dependencia funcional)
23 --ST_Union es una función de agregación de datos. Si no usamos GROUP BY, devolverá error
24 --use ctrl+shift+= (fetch all data) para traer todos los datos
```



El resultado debe ser el mismo, pero en el proceso se utilizó ST_Union para consolidar los polígonos de los barrios. Además, al usar ST_Union, por ser una función de agregación, debemos usar GROUP BY al final usando solo el campo g.gid ([ver explicación sobre Dependencia funcional](#)).

En ambos queries puede **sustituir WHERE g.geom && b.geom and ST_Relate(g.geom, b.geom, 'T*****')** por **WHERE ST_Intersects(g.geom, b.geom)** y le dará el mismo resultado en la mitad del tiempo porque ST_Intersects ya usa el índice espacial y este predicado lo están optimizando continuamente. Usar **ST_Relate** con el patrón **'T*****'** es lo más correcto, pero como este predicado no utiliza por defecto los índices, hay que insertar el comparador de bounding boxes. Sin embargo, recomendamos que siempre use la función extra **STX_Extract** para **homogenizar** las **geometrías** (todas serán multi) y para **devolver geometrías de la misma dimensión**.

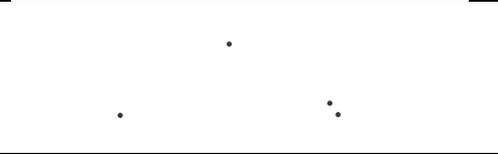
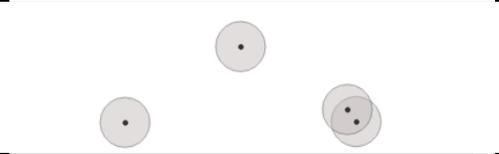
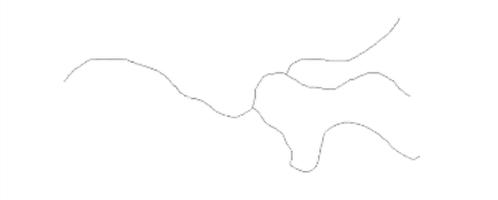
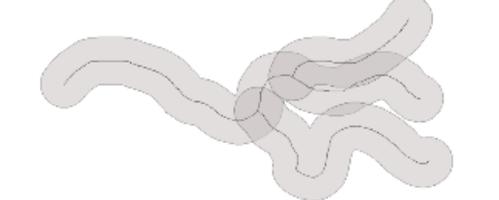
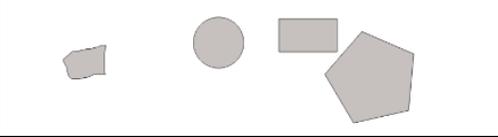
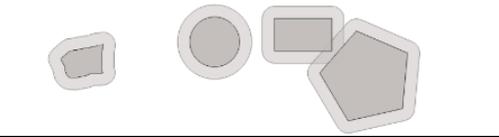
****Resista la tentación de disolver los polígonos adyacentes entre cuadrángulos geológicos. No necesariamente son lo mismo.****

Cada cuadrángulo es una interpretación independiente de cuadrángulos adyacentes. Las interpretaciones están basadas en estudios científicos, pero no siempre coinciden. En ocasiones hay formaciones geológicas que han cambiado de nombre y se han quedado así en los cuadrángulos. Sin duda, hace falta actualizar y completar estos mapas, ya que muchos son antiguos, otros son preliminares y otros no existen a escala 1:20,000.



BUFFER

Un buffer es un área de proximidad o influencia desde una geometría de dimensión 0, 1, o 2 y una distancia escogida. Su uso es muy común en SIG de interfaz gráfica. Se debe usar [ST_Buffer](#) por ejemplo, cuando queremos extraer un área de influencia a una geometría.

Geometrías	Input layers	Output
Punto		
Linestring		
Polygon		
Dimensiones requeridas	Un layer de una de tres dimensiones: 0, 1 o 2	Dimensión 2
Atributos	--	Solo atributos del input layer
Notas:	Luego de hacer el buffer, se puede aplicar la función ST_Union para consolidar los polígonos de salida, según se necesario.	

Sintaxis:

```
ST_Buffer(geometry, radio_del_buffer float, buffer_style_parameters = ' text);
```

```
ST_Buffer(geometry, radio_del_buffer float, num_segmentos_ct_círc integer);
```

El **tercer argumento** o parámetro sirve para controlar la exactitud y estilo del buffer. La exactitud del cuarto de círculo es dada por **el número de segmentos**, que **por defecto es 8**.

El estilo del buffer puede especificarse mediante **parámetros separados por un espacio**:

- **'quad_segs=16'**: número de segmentos usados para la aproximación del cuarto de círculo.
- **'endcap=round|flat|square'**: forma de terminación del buffer, por defecto, round.
- **'join=round|mitre|bevel'**: estilo del join, por defecto round. Se puede escribir 'miter'.
- **'mitre_limit=#.#'**: razón de límite del mitre "pico", el cual solo afecta al join mitre
- **'side=both|left|right'**: left o right realiza el buffer de un solo lado de la geometría, con el lado del buffer relativo a la dirección de la línea. Esto solo aplica a Linestrings y no afecta a geometrías de punto y de polígonos. Por defecto, los end caps son cuadrados.

El resultado de salida es un polígono o un multipolígono que representa todos los puntos en los cuales la distancia de una geometría/geografía es igual o menor a la distancia dada. Una distancia negativa puede reducir el polígono completamente y en tal caso, se devuelve un polígono vacío. Para puntos y linestrings las distancias negativas siempre devolverán geometrías/geografías vacías.



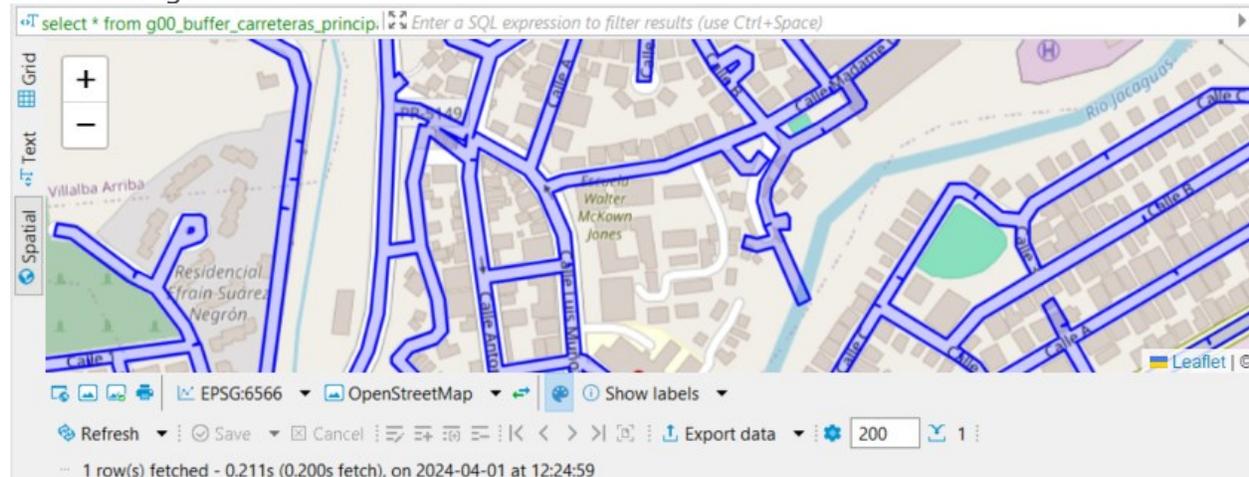
Tutorial PostGIS, 3.x

ST_Buffer siempre devuelve geometrías válidas. Por tal razón, es usado a veces como forma de validar geometrías, mediante `ST_Buffer(geom,0)`.

Ejemplo: Generar un geodato/tabla que contenga un buffer con **radio de 5 metros** de cada elemento del sistema vial del Municipio de Villalba. **Consolide el buffer** en una sola geometría. **Conviértala a geometría MultiPolygon.** Los **end caps** deben ser **flat/planos**

```
1 --script sql para generar una tabla que contenga un buffer zone alrededor del sistema vial en Villalba
2 drop table if exists g00_buffer_carreteras_principales_villalba cascade; --si la tabla existe, bórrala
3 create table g00_buffer_carreteras_principales_villalba --generar geodato/tabla
4 (gid serial primary key,geom geometry (multipolygon,6566)); --estos son los campos
5 --la geometría es multipolygon con CRS: EPSG:6566 o NAD83(2011)
6 insert into g00_buffer_carreteras_principales_villalba --insertar filas en la nueva tabla vacía
7 (geom) --solamente el campo geom. gid se genera automáticamente. Tipo serial
8 select st_union( --consolidar las geometrías
9 st_multi( --convertirlas a tipo multi
10 st_buffer(v.geom,5,'endcap=flat'))) --buffer de 5 metros, 12 segmentos, extremos flat
11 from g35_viales_mapa_base_tiger_rds_2006se_villalba v; --de esta tabla
12
13 select * from g00_buffer_carreteras_principales_villalba; --verificar en grid y tab Spatial
```

Detalle del geodato:



SELECCIÓN POR DISTANCIA

En programas SIG de interfaz gráfica estamos acostumbrados a utilizar un buffer para hacer una búsqueda por distancia, o utilizar una función de selección localización (por intersección geométrica) que conlleve una distancia. En el caso de PostGIS, hacer ese proceso como parte de un query es bastante costoso en recursos de memoria (tarda mucho). Para búsquedas por distancia es más eficiente utilizar [ST_DWithin](#). Esta función de PostGIS utiliza el índice espacial y es mucho más eficiente y rápida.

Ya hemos utilizado `ST_DWithin` anteriormente en queries espaciales en las secciones anteriores.



APPEND

El geoproceso Append se trata simplemente de usar la cláusula INSERT INTO como ha sido realizado en las secciones anteriores. Durante este proceso, se añaden nuevas filas a la tabla, ya sea nueva sin filas o con datos.

Se debe recordar que al hacer la inserción, no necesariamente se realiza una validación topológica y por ende, puede haber solape entre geometrías, con las consecuencias que esto pueda implicar.

DISSOLVE

Ya hemos visto indirectamente el proceso **Dissolve** en secciones anteriores cada vez que utilizamos la función [ST_Union](#). Por ejemplo, lo vimos en el proceso Buffer anterior cuando consolidamos los buffers de cada segmento de vía.

	Input Layer	Resultado
Unión de todas las geometrías		
Unión basada en un campo; generalizar/agregar geometrías. (Usar GROUP BY)		
Dimensiones requeridas	Solo un input layer de dimensión 0, 1 o 2.	Igual a la dimensión del input layer.

Hagamos un ejemplo de **Dissolve**, consolidando geometrías basado en un campo de la tabla.

En la pestaña **<gisdb_lab> Script**, escriba el siguiente script SQL:

```

1= drop table if exists g00_dissolve_prime_farmland_villalba cascade; --si la tabla existe, bórrala
2 create table g00_dissolve_prime_farmland_villalba --generar geodato/tabla
3 (gid serial primary key, farm_class varchar, --estos son los campos
4 geom geometry (multipolygon,6566)); --más el campo de geometrías:multipolygon, NAD83(2011)
5
6= insert into g00_dissolve_prime_farmland_villalba --insertar filas en la nueva tabla vacía
7 (farm_class, geom) --solo nos interesan los campos farm_class y geometría
8 select s.farm_class, --trae el campo farm_class de la tabla 's'
9 st_multi( --convertir a geometría multi
10 st_union(s.geom)) --consolidar geometrías de tabla 's'
11 from g15_suelos_soil_map_units_2018_villalba s --tabla suelos, alias 's'
12 group by s.farm_class; --agrega/consolida mediante el campo farm_class
13
14 select * from g00_dissolve_prime_farmland_villalba; --comprobar el resultado

```



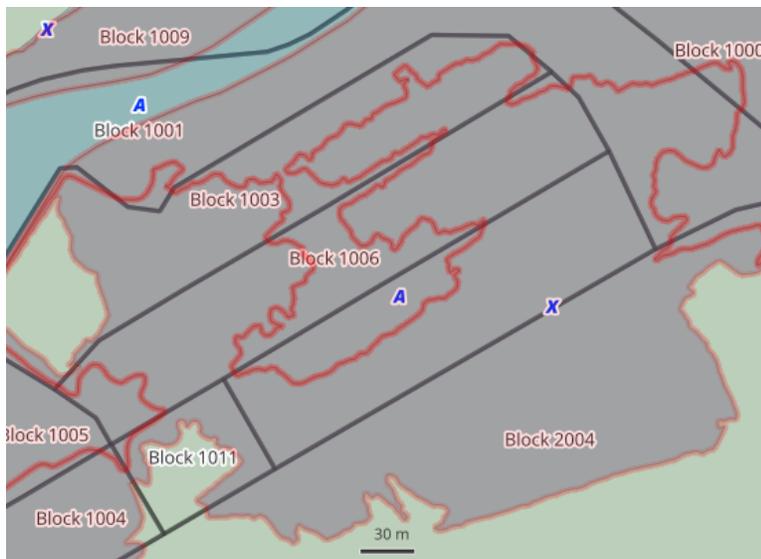
INTERPOLACIÓN AREAL

En esta parte veremos un ejemplo de uso del mecanismo de **interpolación areal** para **estimar población según el geodato de bloques censales, Censo-2020 que están en zonas inundables, FEMA-2017**. La manera más exacta de conocer la población afectada por ubicarse en zonas inundables es cuando las viviendas están identificadas por puntos. Solo haría falta seleccionar los puntos que intersequen o estén a cierta distancia de los bordes de zonas inundables y luego multiplicar por el factor de personas por viviendas para el Censo 2020 (2.65p/v).

Cuando no disponemos de puntos, podemos **usar la unidad cartográfica más pequeña y solaparla con el geodato de interés**. El geodato de **bloques censales 2020** tiene la unidad de área más pequeña con los conteos de 100% de los datos poblacionales.

Usaremos el geodato de **zonas inundables** como ejemplo para solapar este mapa con el mapa de bloques y utilizar el mecanismo de **interpolación areal** para **estimar la población afectada**. La estimación de población se basa en el principio de **proporción de área ocupada** del geodato de inundaciones en este caso dividido por el **área original del bloque censal**. Esta proporción de área (0 a 1) es multiplicada por el dato original, de población en este caso y nos dará indirectamente el estimado de población afectada.

Este método funciona mejor cuando la población está repartida de manera uniforme a través del área. Por otro lado, cabe la posibilidad de que la población **esté concentrada** en la zona inundable pero el bloque censal sea grande y el área de solape sea pequeño. En ese caso, la población estaría **subestimada**. **Otra desventaja** es que a medida que **nos alejamos del año censal decenal**, el conteo de población puede dejar de ser real.



Bloques censales 2020 y zonas inundables, 2017 en una sección del Municipio de Villalba.

El siguiente query mostrará áreas originales de bloques, zonas inundables, nombre de bloque, zona inundable, y las proporciones de área que serán multiplicadas por la población 2020 de cada bloque censal.



Tutorial PostGIS, 3.x

```

1 --Estimar población Censo 2020 afectada por zonas inundables
2 --usando interpolación areal.
3 --Resultados a nivel de bloque censal 2020, para Villalba
4 select b.geoid20, b.name20, b.pop100, --campos de tabla blocks: id,nombre,población 2020
5 round(st_area(b.geom)::numeric,2) area_blk, --área del bloque censal m²
6 i.fld_zone, --código de zona inundable
7 round(st_area(st_intersection(b.geom,i.geom))::numeric,4) inters_area, --área de intersección m²
8 --::numeric,4 se usa para cambiar el tipo de dato 'double' a numeric y 4 lugares decimales
9 round(st_area(st_intersection(b.geom,i.geom))::numeric,4)/round(st_area(b.geom)::numeric,4) proportion,
10 --dividir área de intersección / área original del bloque censal
11 --para obtener la proporción de área (rango: 0 a 1)
12 round(b.pop100*round(st_area(st_intersection(b.geom,i.geom))::numeric,4)/round(st_area(b.geom)::numeric,4)) pop_est,
13 --multiplicar la proporción de área de intersección * población del bloque
14 st_multi(st_intersection(b.geom,i.geom)) geom --homogenizar las geometrías a tipo multipolygon
15 from g31_censo2020_blk_villalba b, --geodato de bloques censales
16 g23_riesgo_inunda_floodzones_2017_villalba i --geodato de zonas inundables
17 where i.geom && b.geom --Condiciones: comparación de bounding boxes
18 and st_relate(b.geom,i.geom,'T*****') --y que los interiores de las geometrías intersequen
19 group by b.gid,i.gid --dependencia funcional: agrupar por gid de ambas tablas
20 order by geoid20 asc, pop_est desc --ordena por id-bloque asc y población estimada desc

```

Resultados:

geoid20	name20	pop100	area_blk	fld_zone	inters_area	proportion	pop_est	geom
721497201001004	Block 1004	183	3,590,759.25	A	13,985.2537	0.0038947901	1	MULTIPOLYGON (((196988.3678 2355
721497201001004	Block 1004	183	3,590,759.25	X	2,584.91	0.0007198784	0	MULTIPOLYGON (((196988.99626286
721497201001005	Block 1005	34	9,520.45	A	5,144.2921	0.540341554	18	MULTIPOLYGON (((196946.2164 2343
721497201001005	Block 1005	34	9,520.45	X	835.6402	0.0877732282	3	MULTIPOLYGON (((197017.5415 2343
721497201001008	Block 1008	88	64,460.81	A	15,573.4123	0.2415950537	21	MULTIPOLYGON (((197273.8158 2345
721497201001008	Block 1008	88	64,460.81	X	2,665.3495	0.0413483727	4	MULTIPOLYGON (((197288.72078127
721497201001009	Block 1009	23	53,235.17	A	10,079.5357	0.1893397911	4	MULTIPOLYGON (((196939.6535 2343
721497201001009	Block 1009	23	53,235.17	X	2,849.9637	0.0535353559	1	MULTIPOLYGON (((196935.92443290
721497201001013	Block 1013	0	182,304.61	A	8,165.9412	0.0447928398	0	MULTIPOLYGON (((196829.8874 2341





Tutorial PostGIS, 3.x

Si queremos el cómputo a nivel municipal agregado por zona inundable:

```

1 --Estimar población Censo 2020 afectada por zonas inundables
2 --usando interpolación areal.
3 --Obtener población estimada a nivel municipal usando subquery
4 select countyfp20, fld_zone, sum(pop_est) pop_est --insertar columna countyfp20:código municipal
5 from --SUBQUERY más o menos igual al query anterior
6 (select b.countyfp20, b.geoid20, b.name20, b.pop100, --campos de tabla blocks: id,nombre,población 2020
7 round(st_area(b.geom)::numeric,2) area_blk, --área del bloque censal m²
8 i.fld_zone, --código de zona inundable
9 round(st_area(st_intersection(b.geom,i.geom))::numeric,4) inters_area, --área de intersección m²
10 --::numeric,4 se usa para cambiar el tipo de dato 'double' a numeric y 4 lugares decimales
11 round(st_area(st_intersection(b.geom,i.geom))::numeric,4)/round(st_area(b.geom)::numeric,4) proportion,
12 --dividir área de intersección / área original del bloque censal
13 --para obtener la proporción de área (rango: 0 a 1)
14 round(b.pop100*round(st_area(st_intersection(b.geom,i.geom))::numeric,4)/round(st_area(b.geom)::numeric,4)) pop_est,
15 --multiplicar la proporción de área de intersección * población del bloque
16 st_multi(st_intersection(b.geom,i.geom)) geom --homogenizar las geometrías a tipo multipolygon
17 from g31_censo2020_blk_villalba b, --geodato de bloques censales
18 g23_riesgo_inunda_floodzones_2017_villalba i --geodato de zonas inundables
19 where i.geom && b.geom --Condiciones: comparación de bounding boxes
20 and st_relate(b.geom,i.geom,'T*****') --y que los interiores de las geometrías intersequen
21 group by b.gid,i.gid --dependencia funcional: agrupar por gid de ambas tablas
22 order by geoid20 asc, pop_est desc --ordena por población estimada
23 ) as subq --Fin del subquery
24 group by countyfp20, fld_zone --agrupar por código de municipio y zonas inundables
25 order by sum(pop_est) desc

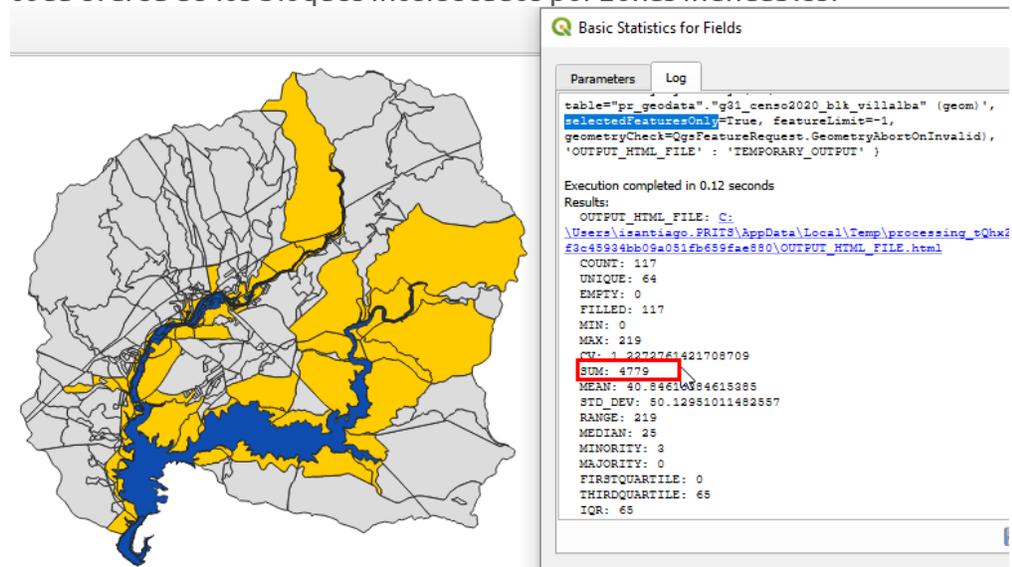
```

Resultado:

Grid	ABC countyfp20	ABC fld_zone	123 pop_est
1	149	A	755
2	149	X	373

Estas dos categorías sumadas nos darán un estimado de 1,128 personas afectadas a nivel municipal. Este estimado tiene las ventajas de basarse en el uso de la unidad cartográfica censal más pequeña y aplicar una proporción de área afectada por bloque.

Si usáramos un select by Location en QGIS el estimado es 4.2 veces mayor porque considerará toda el área de los bloques intersecados por zonas inundables.



El uso de puntos de estructuras sigue siendo la mejor opción pero deben distinguirse por uso comercial, residencial, etc. y diferenciarlos de estructuras menores auxiliares para no sobreestimar la población.



PRÁCTICAS

1. Utilice la función **ST_Intersection** para calcular el **kilometraje agregado de vías por tipo CFCC** (geodato de vías del Censo) **por barrio** en el municipio de Villalba. Este query se parece a otros hechos anteriormente usando **GROUP BY**. En este ejercicio debe usar **ST_Intersection** para obtener los **kilometrajes por barrio**.

Geodatos a usar: **g35_viales_mapa_base_tiger_rds_2006se_villalba** as v
g03_legales_barrios_2015_villalba as b

Query **sin usar ST_Intersection** y **sin usar** el campo **CFCC**.

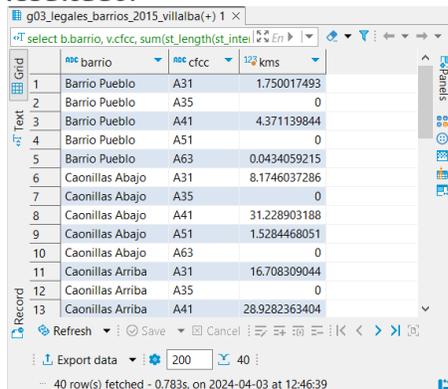
```
1 select b.barrio, sum(st_length(v.geom))/1000 as kms
2 from g35_viales_mapa_base_tiger_rds_2006se_villalba v,
3 g03_legales_barrios_2015_villalba b
4 group by b.barrio
5 order by b.barrio;
```

Resultado erróneo. Agrupa por barrio, pero el cómputo no es correcto.

Grid	barrio	kms
1	Barrio Pueblo	350.073673643
2	Caonillas Abajo	350.073673643
3	Caonillas Arriba	350.073673643
4	Hato Puerco Abajo	350.073673643
5	Hato Puerco Arriba	350.073673643
6	Vacas	350.073673643
7	Villalba Abajo	350.073673643
8	Villalba Arriba	350.073673643

¿Dónde va la función **ST_Intersection**? Esta **debe** colocarse en la parte declarativa, **dentro del paréntesis** de la función **ST_Length(geom)** para que devuelva el cómputo correcto: **sum(largo(intersección(v.geom,b.geom)))**. Recuerde que debe añadir el campo **cfcc** después del campo **b.barrio** y añadirlo en las cláusulas **GROUP BY** y en **ORDER BY**, ambos ascendentes.

Resultado:



Grid	barrio	cfcc	kms
1	Barrio Pueblo	A31	1.750017493
2	Barrio Pueblo	A35	0
3	Barrio Pueblo	A41	4.371139844
4	Barrio Pueblo	A51	0
5	Barrio Pueblo	A63	0.0434059215
6	Caonillas Abajo	A31	8.1746037286
7	Caonillas Abajo	A35	0
8	Caonillas Abajo	A41	31.228903188
9	Caonillas Abajo	A51	1.5284468051
10	Caonillas Abajo	A63	0
11	Caonillas Arriba	A31	16.708309044
12	Caonillas Arriba	A35	0
13	Caonillas Arriba	A41	28.9282363404

Si quiere ver el **cómputo por barrio sin categorizar por cfcc**, elimine ese campo de la cláusula **SELECT**, de **GROUP BY** y de **ORDER BY**.

Grid	barrio	kms
1	Barrio Pueblo	6.1645632584
2	Caonillas Abajo	40.9319537217
3	Caonillas Arriba	48.5927614639
4	Hato Puerco Abajo	25.3858483654
5	Hato Puerco Arriba	76.770263829
6	Vacas	42.6120686582
7	Villalba Abajo	42.9107030151
8	Villalba Arriba	66.7055113312

2. Use el [geoproceso Clip](#) para **generar una nueva tabla** que sea el producto de **extraer** del geodato de **catastro de suelos** el **área del Municipio de Barranquitas**. Utilice el geodato de municipios, donde la condición sea **where m.municipio = 'Barranquitas'**.

Primero, use el comando **DROP TABLE IF EXISTS g00_clip_soils_barranquitas CASCADE**; como preámbulo a generar la nueva tabla.

```
1 DROP TABLE IF EXISTS g00_clip_soils_barranquitas CASCADE;
```



Tutorial PostGIS, 3.x

El nuevo geodato de suelos **g00_clip_soils_barranquitas** solo debe tener los siguientes campos: **CREATE TABLE g00_clip_soils_barranquitas** (**gid** serial primary key, **musym** varchar, **muname** varchar, **slope_gradient_dominant_component** integer, **geom** geometry (multipolygon, 6566)).

```
1 DROP TABLE IF EXISTS g00_clip_soils_barranquitas CASCADE;
2 CREATE TABLE g00_clip_soils_barranquitas
3 (gid serial primary key,
4 musym varchar, muname varchar,
5 slope_gradient_dominant_component integer,
6 geom geometry (multipolygon, 6566));
```

Geodatos a utilizar: **g15_suelos_soil_map_units_2018** as **s**
g03_legales_municipios_2015 as **m**.

Use el comando **INSERT INTO** para insertar filas en la tabla nueva mediante un query que haga el CLIP, según discutido en [esa sección del tutorial](#).

```
8 insert into g00_clip_soils_barranquitas
9 --estos son los campos de la tabla que recibirán datos...
10 (musym, muname, slope_gradient_dominant_component, geom)
```

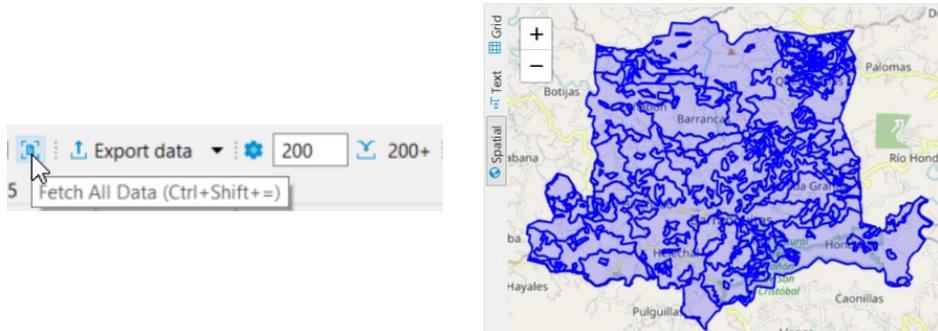
Y este será el query para insertar las filas a la nueva tabla. Recuerde usar la función extra **STX_Extract,2** para homogenizar las geometrías y devolver sólo multipolygons.

```
11 --usando el siguiente query:
12 select musym, muname, slope_gradient_dominant_component,
13 stx_extract(st_intersection(s.geom, m.geom), 2) as geom
14 --no hay que usar st_union porque m.geom tendrá
15 -- solo una geometría: el munic de Barranquitas
16 from g15_suelos_soil_map_units_2018 as s,
17 g03_legales_municipios_2015 as m
18 where m.municipio = 'Barranquitas'
19 and s.geom && m.geom
20 and st_relate(s.geom, m.geom, 'T*****');
21 --comprobar:
22 select * from g00_clip_soils_barranquitas;
```

Resultado:

Grid	gid	musym	muname	slope_gradient_dominant_component	geom
1	1	MuE2	Múcara silty clay, 20 to 40 percent slopes, erod	30	MULTIPOLYGON (((21
2	2	MuE2	Múcara silty clay, 20 to 40 percent slopes, erod	30	MULTIPOLYGON (((21
3	3	MkF2	Maricao clay, 20 to 60 percent slopes	40	MULTIPOLYGON (((21
4	4	MxD	Múcara clay, 12 to 20 percent slopes	15	MULTIPOLYGON (((21
5	5	CuE	Consumo clay, 20 to 40 percent slopes	30	MULTIPOLYGON (((21

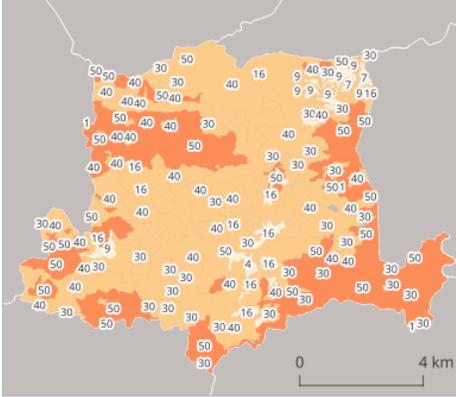
Cuando vaya a la pestaña **Spatial**, recuerde hacer **click** en el botón **Fetch All Data** porque la cantidad de filas es más de 200:





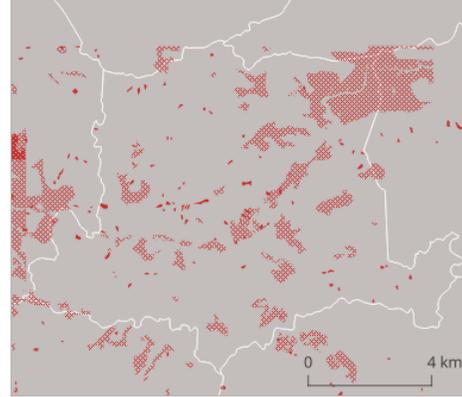
3. Use el [geoproceso Identity](#) para entrecruzar el **nuevo geodato derivado de suelos** para el **Municipio de Barranquitas** con el **geodato geológico** que contiene áreas de **muy alta susceptibilidad a deslizamientos de terreno**.

Pendientes en por ciento



Muy alta susceptibilidad

- debris
- landslide deposits
- mudflow deposits
- lateritic soil and saprolite



Haga una nueva tabla **g00_identity_suelos_landslides_barranquitas** que contenga los campos **g_gid**, **g_geol_id**, **g_unit_name** del **geodato geológico original**, **más** los campos del **geodato de suelos recortado** en el proceso **Clip** realizado en la práctica anterior: **s_gid**, **s_musym**, **s_muname** y **s_slope_gradient_dominant_component**. **Añada el campo risk_level**, que va a contener los niveles de riesgo. Esos niveles se asignarán después de este proceso. Añada al final el campo **geom** multipolygon con CRS código 6566.

Sección para generar la tabla nueva:

```

1 drop table if exists g00_identity_suelos_landslides_barranquitas cascade; --si existe, bórrala
2 create table g00_identity_suelos_landslides_barranquitas --crear tabla vacía
3 (gid serial primary key, --campo id serial primary key
4 g_gid integer, --id numérico original de cada fila del geodato geológico
5 g_geol_id varchar, --código id de cada mapping unit geológica
6 g_unit_name varchar, --nombre del mapping unit
7 s_gid integer, --id numérico original de cada fila del geodato suelos clip
8 s_musym varchar, --código id de unidad de suelo
9 s_muname varchar, --nombre de la unidad de suelo
10 s_slope_gradient_dominant_component integer, --slope percent
11 risk_level varchar, --nuevo campo: nivel de riesgo
12 geom geometry (multipolygon,6566)); --geometrías, multipolígono, NAD83(2011)

```

Resultado:

Name	Value
Updated Rows	0
Query	--si existe, bórrala create table g00_identity_suelos_landslides_barranquitas --crear tabla vacía (gid serial primary key, --campo id serial primary key g_gid integer, --id numérico original de cada fila del geodato geológico g_geol_id varchar, --código id de cada mapping unit geológica g_unit_name varchar, --nombre del mapping unit s_gid integer, --id numérico original de cada fila del geodato suelos clip s_musym varchar, --código id de unidad de suelo s_muname varchar, --nombre de la unidad de suelo s_slope_gradient_dominant_component integer, --slope percent risk_level varchar, --nuevo campo: nivel de riesgo geom geometry (multipolygon,6566))
Start time	Mon Apr 08 10:52:11 BOT 2024
Finish time	Mon Apr 08 10:52:11 BOT 2024

Creó una tabla vacía, solo con la estructura de campos. Updated rows = 0

Realice ahora el paso Difference: A-B



Sección Difference (A-B), primera parte: Insertar polígonos del geodato clip de suelos que no solapan las geometrías del geodato geológico de deslizamientos.

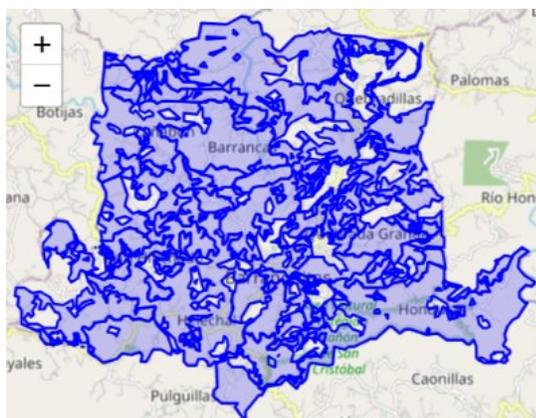
```

14= --Paso Difference (A-B) o (s-g)
15 --Primera parte: Devolver poligonos de la tabla "s"
16 --que son borrados parcial o totalmente por poligonos de tabla g
17 insert into g00_identity_suelos_landslides_barranquitas
18 (s_gid, s_musym, s_muname, s_slope_gradient_dominant_component, geom) --campos para llenar filas
19 select gid, musym, muname, slope_gradient_dominant_component, geom --campos, tabla suelos
20 from
21     (select s.gid, s.musym, s.muname, s.slope_gradient_dominant_component, --campos
22     stx_extract(
23         st_difference(s.geom,
24             st_union(g.geom)),2) as geom --(la unión geometrías de tabla g)
25     from g00_clip_soils_barranquitas s, --tabla suelos_clip
26     g15_geologia_mapping_units_20k_2018_landslides g --tabla geología landslides
27     where s.geom && g.geom --comparador de bounding boxes
28     and st_relate(s.geom,g.geom, 'T*****')--solo interiores geoms intersequen
29     group by s.gid --agrupar por s.gid (para no repetir el s.gid)
30     ) as subq --alias del subquery, como si fuera una tabla temporal
31= where geom is not null; --condición: que las geoms no estén vacías
32 --corroborar
33 select * from g00_identity_suelos_landslides_barranquitas;

```

Resultado parcial, paso 1:

Name	Value
Updated Rows	130
Query	<pre> --Paso Difference (A-B) o (s-g) --Primera parte: Devolver poligonos de la tabla "s" --que son borrados parcial o totalmente por poligonos de tabla g insert into g00_identity_suelos_landslides_barranquitas (s_gid, s_musym, s_muname, s_slope_gradient_dominant_component, geom) --campos para llenar filas select gid, musym, muname, slope_gradient_dominant_component, geom --campos, tabla suelos from (select s.gid,s.musym,s.muname,s.slope_gradient_dominant_component, --campos stx_extract(st_difference(s.geom, st_union(g.geom)),2) as geom --(la unión geometrías de tabla g) from g00_clip_soils_barranquitas s, --tabla suelos_clip g15_geologia_mapping_units_20k_2018_landslides g --tabla geología landslides where s.geom && g.geom --comparador de bounding boxes and st_relate(s.geom,g.geom, 'T*****')--solo interiores geoms intersequen group by s.gid --agrupar por s.gid (para no repetir el s.gid)) as subq --alias del subquery, como si fuera una tabla temporal where geom is not null </pre>
Start time	Mon Apr 08 10:58:28 BOT 2024
Finish time	Mon Apr 08 10:58:29 BOT 2024



Paso #2 Difference: Insertar polígonos del geodato clip de suelos que coinciden con las geometrías del geodato geológico de deslizamientos. En este paso #2, ST_Difference creará espacios vacíos donde coincidan las geometrías del geodato geológico. Esos espacios se llenarán más adelante en el paso #3 ST_Intersection.

```

34= --Paso Difference (A-B) o (s-g)
35 --Segunda parte: Devolver poligonos de la tabla "s"
36 --donde las geometrías de los interiores de "s"
37 --Sí intersecan geometrías de tabla "g"
38 insert into g00_identity_suelos_landslides_barranquitas
39 (s_gid,s_slope_gradient_dominant_component,geom) --campos para llenar filas
40 select gid, slope_gradient_dominant_component, geom --campos, tabla suelos
41 from
42     (select s.gid,s.slope_gradient_dominant_component, --campos tabla suelos
43     g.gid as g_gid, --id tabla geometría solo para comparar al final
44     st_multi(s.geom) as geom --homogenizar: convertir geometría en multipoligono
45     from g00_clip_soils_barranquitas s --tab suelos_clip left join guardar sus records
46     left join g15_geologia_mapping_units_20k_2018_landslides g --tabla geología
47     on s.geom && g.geom --condiciones sin usar where: comparar bounding boxes
48     and st_relate(s.geom,g.geom, 'T*****')--solo intersecar interior geometrías
49     ) as subq --alias del subquery, como si fuera una tabla temporal
50= where g_gid is null; --condición: que el campo g_gid no esté vacío
51 --corroborar
52 select * from g00_identity_suelos_landslides_barranquitas;

```

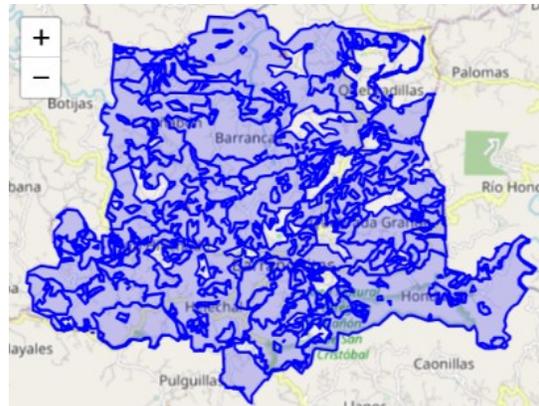


Tutorial PostGIS, 3.x

```

Statistics 1 X
Name Value
Updated Rows 101
Query
--Paso Difference (A-B) o (s-g)
--Segunda parte: Devolver polígonos de la tabla "s"
--que no solapan con geometrías de la tabla "g" y
--se conservan íntegros
insert into g00_identity_suelos_landslides_barranquitas
(s.gid, s.musym, s.muname, s.slope_gradient_dominant_component, geom) --campos para llenar filas
select gid, musym, muname, slope_gradient_dominant_component, geom --campos, tabla suelos
from
-- en lugar de una tabla física se obtienen de un subquery
(select s.gids.musym, s.muname, s.slope_gradient_dominant_component, --campos tabla suelos
g.gid as g.gid, --id tabla geometría solo para comparar al final
st_multi(s.geom) as geom --homogenizar: convertir geometría en multipolígono
from g00_clip_soils_barranquitas s --tab suelos, clip left join guardar sus records
left join g15_geologia_mapping_units_20k_2018_landslides g --tabla geología
on s.geom && g.geom --condiciones sin usar where: comparar bounding boxes
and st_relate(s.geom, g.geom, 'T*****') --solo intersectar interior geometrías
) as subq --alias del subquery, como si fuera una tabla temporal
where g.gid is null
Start time Mon Apr 08 11:00:03 BOT 2024
Finish time Mon Apr 08 11:00:03 BOT 2024

```



A este punto, nos queda **realizar la intersección geométrica** para completar los polígonos que faltan. Esos corresponden al geodato de geología que contiene solo deslizamientos, lateritas y saprolitas.

4. Use la función **ST_Intersection** para **insertar los polígonos geológicos** que faltan en el geodato `g00_overlay_identity_suelos_landslides_barranquitas`

Sección intersection:

```

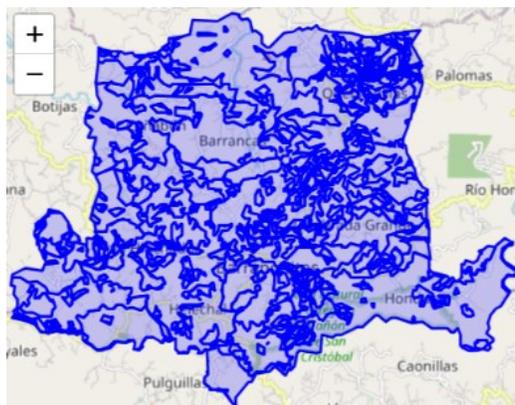
56 --Paso intersección A n B
57 insert into g00_identity_suelos_landslides_barranquitas
58 (g.gid, g.geol_id, g.unit_name, --campos de geología
59 s.gid, s.musym, s.muname, --datos de suelos
60 s.slope_gradient_dominant_component, --campos de suelos
61 geom) -- incluye geometría de los geoprocetos
62 select g.gid, g.geol_id, g.unit_name, --campos t geología
63 s.gid, s.musym, s.muname, --campos t suelos
64 s.slope_gradient_dominant_component, --campos t suelos
65 stx_extract( --extraer solo polígonos "opción #2"
66 st_intersection(g.geom, s.geom), 2) as geom --intersección gns as geom
67 from g15_geologia_mapping_units_20k_2018_landslides g, --tab geología
68 g00_clip_soils_barranquitas s --tab suelos
69 where g.geom && s.geom --comparador bounding boxes
70 and st_relate(g.geom, s.geom, 'T*****'); --solo las que intersequen interiores
71 --corroborar
72 select * from g00_identity_suelos_landslides_barranquitas;

```

```

Statistics 1 X
Name Value
Updated Rows 400
Query
--Paso intersección A n B
insert into g00_identity_suelos_landslides_barranquitas
(g.gid, g.geol_id, g.unit_name, --campos de geología
s.gid, s.musym, s.muname, --datos de suelos
s.slope_gradient_dominant_component, --campos de suelos
geom) -- incluye geometría de los geoprocetos
select g.gid, g.geol_id, g.unit_name, --campos t geología
s.gid, s.musym, s.muname, --campos t suelos
s.slope_gradient_dominant_component, --campos t suelos
stx_extract( --extraer solo polígonos "opción #2"
st_intersection(g.geom, s.geom), 2) as geom --intersección gns as geom
from g15_geologia_mapping_units_20k_2018_landslides g, --tab geología
g00_clip_soils_barranquitas s --tab suelos
where g.geom && s.geom --comparador bounding boxes
and st_relate(g.geom, s.geom, 'T*****')
Start time Mon Apr 08 11:08:48 BOT 2024
Finish time Mon Apr 08 11:08:49 BOT 2024

```





5. Actualice el contenido del campo **risk_level** basado en el siguiente script usando la cláusula CASE-WHEN-THEN. Usar **SET risk_level =**

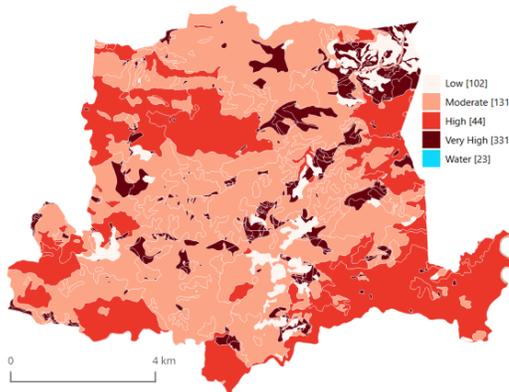
```

74--UPDATE risk_level
75 update g00_identity_suelos Landslides_barranquitas --actualiza esta tabla
76 set risk_level = --en el campo risk_level...
77 --cuando g_gid no esté vacío y slope >29%,
78 CASE WHEN g_gid is not NULL and s_slope_gradient_dominant_component > 29
79 THEN 'Very High' --asigna 'Very High' a risk_level
80 --cuando g_gid esté vacío y slope >=50%,
81 WHEN g_gid is null and s_slope_gradient_dominant_component >= 50
82 then 'High' --asigna 'High' a risk_level
83 --cuando g_gid esté vacío y slope esté entre 30 y 49%,
84 WHEN g_gid is null and s_slope_gradient_dominant_component between 30 and 49
85 then 'Moderate' --asigna 'Moderate' a risk_level
86 --cuando g_gid esté vacío y slope esté entre 1 y 29%,
87 WHEN g_gid is null and s_slope_gradient_dominant_component between 1 and 29
88 then 'Low' --asigna 'Low' a risk_level
89 --cuando g_gid no esté vacío y slope esté entre 1 y 29%,
90 WHEN g_gid is not null and s_slope_gradient_dominant_component between 1 and 29
91 then 'Low' --asigna 'Low' a risk_level
92 WHEN s_musym = 'W' --cuando este campo = W,
93 then 'Water' --asigna label 'Water' a risk_level
94 end;

```

Resultados:

Name	Value
Updated Rows	631
Query	<pre> --UPDATE risk_level update g00_identity_suelos Landslides_barranquitas --actualiza esta tabla set risk_level = --en el campo risk_level... CASE WHEN g_gid is not NULL and s_slope_gradient_dominant_component > 29 --cuando g_gid no esté vacío y slope >29%, THEN 'Very High' --asigna 'Very High' a risk_level WHEN g_gid is null and s_slope_gradient_dominant_component >= 50 --cuando g_gid esté vacío y slope >=50%, then 'High' --asigna 'High' a risk_level WHEN g_gid is null and s_slope_gradient_dominant_component between 30 and 49 --cuando g_gid esté vacío y slope esté entre 30 y 49%, then 'Moderate' --asigna 'Moderate' a risk_level WHEN g_gid is null and s_slope_gradient_dominant_component between 1 and 29 --cuando g_gid esté vacío y slope esté entre 1 y 29%, then 'Low' --asigna 'Low' a risk_level WHEN g_gid is not null and s_slope_gradient_dominant_component between 1 and 29 --cuando g_gid no esté vacío y slope esté entre 1 y 29%, then 'Low' --asigna 'Low' a risk_level WHEN s_musym = 'W' --cuando este campo = W, then 'Water' --asigna label 'Water' a risk_level end </pre>
Start time	Mon Apr 08 12:05:24 BOT 2024
Finish time	Mon Apr 08 12:05:24 BOT 2024



Visualizar en QGIS

Felicidades. Usted ha mejorado el mapa de susceptibilidad a deslizamientos de terreno en relación con el mapa de 1979.

6. Evalúe cuántas personas (Censo 2020) están dentro de zonas de riesgo por deslizamientos en Municipio de Villalba. Haga el desglose por nivel de riesgo. Para calcular el número de personas deberá hacer **interpolación por proporción de área ocupada** contra el dato de **población del censo** en el **geodato de bloques censales 2020**.

Geodatos a usar: **g00_overlay_suelos Landslides_2024_villalba**
g31_censo2020_blk_villalba

Realice el cómputo a nivel municipal y por nivel de riesgo.

Refiérase a la [sección interpolación areal](#) como guía para realizar esta práctica.



GEODATOS RÁSTER EN POSTGIS

Los geodatos en formato ráster también pueden ser guardados en PostgreSQL. Hay dos maneras para registrar datos: 1) Guardar los datos **en la base de datos**, y 2) Guardar **referencias a un archivo ráster externo**. En ocasiones, cuando la imagen o ráster es muy voluminoso, en Giga bytes, por ejemplo, se prefiere la opción de registrar el archivo como referencia externa.

Vamos a realizar un ejemplo con un ráster relativamente pequeño y previamente guardado en la base de datos.

Demo: Uso de geodato ráster para calcular una columna de geodato vectorial.

En este ejemplo haremos una **comparación entre los valores de pendientes en por ciento en los geodatos** de:

- suelos-NRCS
- ráster-DEM2015 (5x5 m por píxel)

Como se vio anteriormente en las prácticas el geodato de suelos-NRCS contiene una columna en la cual se registra la pendiente promedio en por ciento del componente dominante de una unidad cartográfica (mapping unit). Por otro lado, el geodato ráster de pendientes nos dará el cálculo directo de pendientes en por ciento en cada píxel.

Al final presentaremos un mapa para comparar la pendiente estimada en el geodato de suelos con las pendientes directas del geodato ráster.

Crear geodato vectorial que va a contener el campo **slope_dominant_component** de la tabla de suelos y el campo **mean_slope_pct** que se extraerá del geodato ráster de pendientes.

```
1 DROP TABLE IF EXISTS g00_slope_comparar_villalba CASCADE; -- si existe la tabla, bórrala
2 create table g00_slope_comparar_villalba --crear tabla/geodato solo para comparar
3 (gid serial primary key, mean_slope_pct double precision, -- definir campos de la tabla
4 g_unit_code varchar, slope_dominant_component integer, -- definir campos de la tabla
5 difference double precision, -- definir campos de la tabla
6 geom geometry (multipolygon, 6566)); -- defini campo geometry multipolygon, NAD83(2011)
```

Resultado:

Name	Value
Updated Rows	0
Query	-- si existe la tabla, bórrala create table g00_slope_comparar_villalba --crear tabla/geodato solo para comparar (gid serial primary key, mean_slope_pct double precision, -- definir campos de la tabla g_unit_code varchar, slope_dominant_component integer, -- definir campos de la tabla difference double precision, -- definir campos de la tabla geom geometry (multipolygon, 6566))
Start time	Tue Apr 16 09:32:51 BOT 2024
Finish time	Tue Apr 16 09:32:51 BOT 2024



Tutorial PostGIS, 3.x

Insertar filas en la tabla nueva. Este query puede tardar más que otros realizados.

```

8 insert into g00_slope_comparar_villalba --insertar filas en el geodato nuevo
9 (mean_slope_pct, slope_dominant_component,g_unit_code, difference, geom) --campos
10 -- A base de este query:
11 select (
12     st_summarystats(                --summary statistics...
13     st_union(                        --unir los tiles de píxeles
14     st_clip(r.rast,d.geom,true)),1)).mean --traer el promedio de pendientes%
15     as mean_slope_pct,              --as mean_slope_pct
16     s_slope_gradient_dominant_component, --campo slope % NRCS
17     g_unit_code,                    --código unidad geológica,
18     d.s_slope_gradient_dominant_component - --calc diferencia entre
19     (st_summarystats(                --el campo slope%-NRCS
20     st_union(st_clip(r.rast,d.geom,true)),1)).mean --menos la pendiente DEM
21     as difference,                  --as difference
22     d.geom as geom                 --traer geometrías, geodato nuevo landslides
23 from g15_dem2015_5m_slope_percent_villalba r --raster pendientes %
24 join g00_overlay_identity_landslides_2024_villalba d --landslides
25 on st_intersects(r.rast,d.geom) --Condición: que intersequen
26 group by d.gid;                   --agrupar por g.id

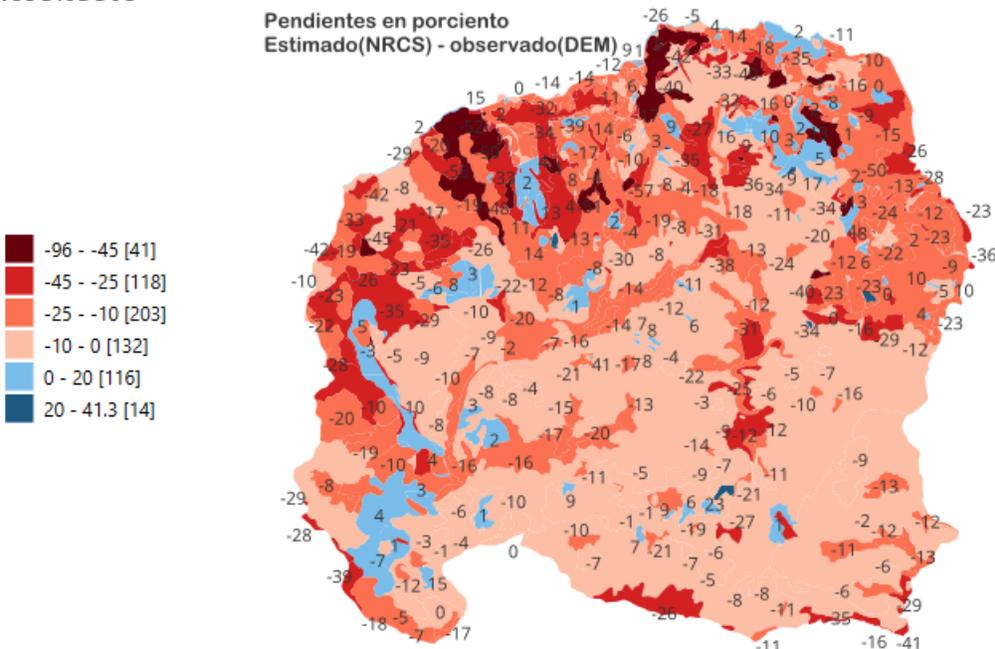
```

Resultado. El query demoró poco más de 3 minutos.

Name	Value
Updated Rows	625
Query	<pre> insert into g00_slope_comparar_villalba --insertar filas en el geodato nuevo (mean_slope_pct, slope_dominant_component,g_unit_code, difference, geom) --campos -- A base de este query: select (st_summarystats(--summary statistics... st_union(--unir los tiles de píxeles st_clip(r.rast,d.geom,true)),1)).mean --traer el promedio de pendientes% as mean_slope_pct, --as mean_slope_pct s_slope_gradient_dominant_component, --campo slope % NRCS g_unit_code, --código unidad geológica, d.s_slope_gradient_dominant_component - --calc diferencia entre (st_summarystats(--el campo slope%-NRCS st_union(st_clip(r.rast,d.geom,true)),1)).mean --menos la pendiente DEM as difference, --as difference d.geom as geom --traer geometrías, geodato nuevo landslides from g15_dem2015_5m_slope_percent_villalba r --raster pendientes % join g00_overlay_identity_landslides_2024_villalba d --landslides on st_intersects(r.rast,d.geom) --Condición: que intersequen group by d.gid </pre>
Start time	Tue Apr 16 09:35:03 BOT 2024
Finish time	Tue Apr 16 09:38:10 BOT 2024



Resultados



Las áreas en **tonos rojizos** corresponden con lugares donde la **estimación** de la pendiente en el **geodato de suelos-NRCS es menor al promedio de pendientes** observadas calculado mediante el DEM. Esto corresponde con las áreas más escarpadas del norte del Municipio. **Las áreas en azul** son lugares donde la estimación de pendientes en el geodato de suelos-NRCS fue mayor que el **promedio de pendientes, derivado del DEM. En su mayoría, las pendientes en el norte están subestimadas por el geodato de suelos-NRCS.** En el área central y sureste, la estimación no es muy diferente al promedio de pendientes calculado del DEM.

Si tomamos las áreas ocupadas por estimación, obtenemos el siguiente resultado:

```

29 select 'Subestimada en NRCS:' as área_comparada,
30 round((sum(st_area(geom))/1000000)::numeric,2) as sq_kms,
31 round((sum(st_area(geom))/1000000)/95.88*100)::numeric,2) as pct
32 from g00_slope_comparar_villalba
33 where difference <0
34 union
35 select 'Sobreestimada en NRCS:' as área_comparada,
36 round((sum(st_area(geom))/1000000)::numeric,2) as sq_kms,
37 round((sum(st_area(geom))/1000000)/95.88*100)::numeric,2) as pct
38 from g00_slope_comparar_villalba
39 where difference >0
40 order by pct desc;

```

área_comparada	sq_kms	pct
Subestimada en NRCS:	89.75	93.61
Sobreestimada en NRCS:	6.13	6.39

Donde el área total municipal es de 95.88 kilómetros cuadrados.



Descarga de datos para el ejercicio:



Apéndice A-1: Inventario de relaciones topológicas

Este es un inventario con una lista ilustrada de las 98 relaciones topológicas posibles entre objetos geográficos *simples* de tipo puntual, lineal y de área/superficie.

Ya que cada intersección de la matriz 9-IM puede tomar 4 valores diferentes (F, 0, 1, 2) existen potencialmente 4^9 combinaciones posibles o 262,144 matrices diferentes. Sin embargo, no todas se pueden realizar por causa de restricciones relacionadas a las características topológicas de estos objetos. Por ejemplo, un punto no tiene contorno/límite, por lo tanto los valores de intersecciones con objetos geométricos con el contorno/límite de un punto siempre será igual a F. Vea la primera línea horizontal de las matrices en la tabla A-2. Esto aplica a las relaciones de intersección entre los interiores, contornos/límites y los exteriores de los objetos.

Por ejemplo, la intersección de los exteriores de los objetos siempre será de dimensión = 2 (Área) y ningún polígono cubrirá totalmente el plano. Por tal razón, la novena celda de la matriz de intersección será siempre = 2. Otro ejemplo de restricción: si A y B son 2 objetos no puntuales, el contorno/límite de A interseca al menos una parte de B, ya sea el interior, límite o exterior y viceversa. De este modo, las matrices donde las líneas horizontales del medio horizontales o verticales solo tengan valores F no podrían existir. La lista de relaciones punto/punto (P/P), punto/polilínea (P/L), y punto/polígono (P/S) es evidente. Las relaciones que realmente existen entre polilínea/polilínea (L/L), polilínea y polígono (L/S) y polígono/polígono (S/S) han sido deducidas del conjunto de relaciones posibles entre los objetos mediante la aplicación de las restricciones topológicas descritas por Max J. Egenhofer y considerando la dimensión de las estas intersecciones (modelo DE-9IM)

Repasando:

Tabla A-1: Matriz de intersección 9-IM

	Interior	Límite/Contorno/Boundary	Exterior
Punto (Dim=0)	Dim=0	Vacío / No existe	
Polilínea (Dim=1)	Dim=1	Dim=0	
Polígono/área (Dim=2)	Dim=2	Dim=1	

Referencia: [Predicados OGC: 2.2: Conceptos de interior, límite y exterior de objetos](#), pág. 6. Ministerio de Ecología, desarrollo sustentable, transportación y vivienda, Gobierno de Francia.

La expresión de las restricciones relacionadas a la topología de los objetos se ilustra mediante las matrices de intersección de la tabla A-2.

Tabla A-2: Expresión de las limitaciones relacionados a la topología de los objetos.

P/P	P/L	P/S																											
<table border="1"><tr><td>F,0</td><td>F</td><td>F,0</td></tr><tr><td>F</td><td>F</td><td>F</td></tr><tr><td>F,0</td><td>F</td><td>2</td></tr></table>	F,0	F	F,0	F	F	F	F,0	F	2	<table border="1"><tr><td>F,0</td><td>F,0</td><td>F,0</td></tr><tr><td>F</td><td>F</td><td>F</td></tr><tr><td>1</td><td>0</td><td>2</td></tr></table>	F,0	F,0	F,0	F	F	F	1	0	2	<table border="1"><tr><td>F,0</td><td>F,0</td><td>F,0</td></tr><tr><td>F</td><td>F</td><td>F</td></tr><tr><td>2</td><td>1</td><td>2</td></tr></table>	F,0	F,0	F,0	F	F	F	2	1	2
F,0	F	F,0																											
F	F	F																											
F,0	F	2																											
F,0	F,0	F,0																											
F	F	F																											
1	0	2																											
F,0	F,0	F,0																											
F	F	F																											
2	1	2																											
L/P	L/L	L/S																											
<table border="1"><tr><td>F,0</td><td>F</td><td>1</td></tr><tr><td>F,0</td><td>F</td><td>0</td></tr><tr><td>F,0</td><td>F</td><td>2</td></tr></table>	F,0	F	1	F,0	F	0	F,0	F	2	<table border="1"><tr><td>F,0,1</td><td>F,0</td><td>F,1</td></tr><tr><td>F,0</td><td>F,0</td><td>F,0</td></tr><tr><td>F,1</td><td>F,0</td><td>2</td></tr></table>	F,0,1	F,0	F,1	F,0	F,0	F,0	F,1	F,0	2	<table border="1"><tr><td>F,1</td><td>F,0,1</td><td>F,1</td></tr><tr><td>F,0</td><td>F,0</td><td>F,0</td></tr><tr><td>2</td><td>1</td><td>2</td></tr></table>	F,1	F,0,1	F,1	F,0	F,0	F,0	2	1	2
F,0	F	1																											
F,0	F	0																											
F,0	F	2																											
F,0,1	F,0	F,1																											
F,0	F,0	F,0																											
F,1	F,0	2																											
F,1	F,0,1	F,1																											
F,0	F,0	F,0																											
2	1	2																											
S/P	S/L	S/S																											
<table border="1"><tr><td>F,0</td><td>F</td><td>2</td></tr><tr><td>F,0</td><td>F</td><td>1</td></tr><tr><td>F,0</td><td>F</td><td>2</td></tr></table>	F,0	F	2	F,0	F	1	F,0	F	2	<table border="1"><tr><td>F,1</td><td>F,0</td><td>2</td></tr><tr><td>F,0,1</td><td>F,0</td><td>1</td></tr><tr><td>F,1</td><td>F,0</td><td>2</td></tr></table>	F,1	F,0	2	F,0,1	F,0	1	F,1	F,0	2	<table border="1"><tr><td>F,2</td><td>F,1</td><td>F,2</td></tr><tr><td>F,1</td><td>F,0,1</td><td>F,1</td></tr><tr><td>F,2</td><td>F,1</td><td>2</td></tr></table>	F,2	F,1	F,2	F,1	F,0,1	F,1	F,2	F,1	2
F,0	F	2																											
F,0	F	1																											
F,0	F	2																											
F,1	F,0	2																											
F,0,1	F,0	1																											
F,1	F,0	2																											
F,2	F,1	F,2																											
F,1	F,0,1	F,1																											
F,2	F,1	2																											



Tutorial PostGIS, 3.x

Para cada matriz, las combinaciones de valores de intersección (F,0,1,2) no corresponden a relaciones topológicas reales.

Por cada relación se presentarán el esquema o dibujo geométrico, la matriz de intersección, si es simétrico; es decir si la relación topológica es conmutativa y el valor devuelto por los predicados.

Punto/Punto: 2 relaciones

Representación	Matriz	Conmutativa	Predicados																																	
	<p>Intersection Matrix</p> <p>AB FF0FFF0F2</p> <p>BA FF0FFF0F2</p> <p>B Int Bdy Ext</p> <p>Int F F 0</p> <p>A Bdy F F F</p> <p>Ext 0 F 2</p>	Sí	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>T</td><td>T</td></tr> <tr><td>Intersects</td><td>F</td><td>F</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	T	T	Intersects	F	F	Touches	F	F	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	T	T																																		
Intersects	F	F																																		
Touches	F	F																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		
	<p>Intersection Matrix</p> <p>AB OFFFFFFFF2</p> <p>BA OFFFFFFFF2</p> <p>B Int Bdy Ext</p> <p>Int 0 F F</p> <p>A Bdy F F F</p> <p>Ext F F 2</p>	Sí	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>T</td><td>T</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>T</td><td>T</td></tr> <tr><td>Contains</td><td>T</td><td>T</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>T</td><td>T</td></tr> <tr><td>CoveredBy</td><td>T</td><td>T</td></tr> </tbody> </table>		AB	BA	Equals	T	T	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	T	T	Contains	T	T	Overlaps	F	F	Covers	T	T	CoveredBy	T	T
	AB	BA																																		
Equals	T	T																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	F	F																																		
Crosses	F	F																																		
Within	T	T																																		
Contains	T	T																																		
Overlaps	F	F																																		
Covers	T	T																																		
CoveredBy	T	T																																		



Tutorial PostGIS, 3.x

Punto/Polilínea: 3 relaciones

Representación	Matriz	Conmutativa	Predicados																																	
	<p>Intersection Matrix</p> <p>AB FF0FFF102</p> <p>BA FF1FF00F2</p> <p>B <i>Int Bdy Ext</i></p> <p><i>Int</i> F F 0</p> <p>A <i>Bdy Ext</i></p> <p><i>Bdy</i> F F F</p> <p><i>Ext</i> 1 0 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>T</td><td>T</td></tr> <tr><td>Intersects</td><td>F</td><td>F</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	T	T	Intersects	F	F	Touches	F	F	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	T	T																																		
Intersects	F	F																																		
Touches	F	F																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		
	<p>Intersection Matrix</p> <p>AB F0FFFF102</p> <p>BA FF10F0FF2</p> <p>B <i>Int Bdy Ext</i></p> <p><i>Int</i> F 0 F</p> <p>A <i>Bdy Ext</i></p> <p><i>Bdy</i> F F F</p> <p><i>Ext</i> 1 0 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>T</td><td>T</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>T</td></tr> <tr><td>CoveredBy</td><td>T</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	T	CoveredBy	T	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	T	T																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	T																																		
CoveredBy	T	F																																		
	<p>Intersection Matrix</p> <p>AB 0FFFFFF102</p> <p>BA 0F1FF0FF2</p> <p>B <i>Int Bdy Ext</i></p> <p><i>Int</i> 0 F F</p> <p>A <i>Bdy Ext</i></p> <p><i>Bdy</i> F F F</p> <p><i>Ext</i> 1 0 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>T</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>T</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>T</td></tr> <tr><td>CoveredBy</td><td>T</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	T	F	Contains	F	T	Overlaps	F	F	Covers	F	T	CoveredBy	T	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	F	F																																		
Crosses	F	F																																		
Within	T	F																																		
Contains	F	T																																		
Overlaps	F	F																																		
Covers	F	T																																		
CoveredBy	T	F																																		



Tutorial PostGIS, 3.x

Polilínea/Polilínea: 47 relaciones

Representación	Matriz	Conmutativa	Predicados																																	
	<p>Intersection Matrix</p> <p>AB FF1FF0102</p> <p>BA FF1FF0102</p> <p>B Int Bdy Ext</p> <p>Int F F 1</p> <p>A Bdy F F 0</p> <p>Ext 1 0 2</p>	SÍ	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>T</td><td>T</td></tr> <tr><td>Intersects</td><td>F</td><td>F</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	T	T	Intersects	F	F	Touches	F	F	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	T	T																																		
Intersects	F	F																																		
Touches	F	F																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		
	<p>Intersection Matrix</p> <p>AB FF1F0F1F2</p> <p>BA FF1F0F1F2</p> <p>B Int Bdy Ext</p> <p>Int F F 1</p> <p>A Bdy F 0 F</p> <p>Ext 1 F 2</p>	SÍ	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>T</td><td>T</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	T	T																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		
	<p>Intersection Matrix</p> <p>AB FF1F00102</p> <p>BA FF1F00102</p> <p>B Int Bdy Ext</p> <p>Int F F 1</p> <p>A Bdy F 0 0</p> <p>Ext 1 0 2</p>	SÍ	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>T</td><td>T</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	T	T																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		
	<p>Intersection Matrix</p> <p>AB FF10FF102</p> <p>BA F01FF01F2</p> <p>B Int Bdy Ext</p> <p>Int F F 1</p> <p>A Bdy 0 F F</p> <p>Ext 1 0 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>T</td><td>T</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	T	T																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		
	<p>Intersection Matrix</p> <p>AB FF10F0102</p> <p>BA F01FF0102</p> <p>B Int Bdy Ext</p> <p>Int F F 1</p> <p>A Bdy 0 F 0</p> <p>Ext 1 0 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>T</td><td>T</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	T	T																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		



Tutorial PostGIS, 3.x

Representación	Matriz	Conmutativa	Predicados																																	
	<p>Intersection Matrix</p> <p>AB FF100F102</p> <p>BA F01F001F2</p> <p>B Int Bdy Ext Int F F 1 A Bdy 0 0 F Ext 1 0 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>T</td><td>T</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	T	T																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		
	<p>Intersection Matrix</p> <p>AB F01FF01F2</p> <p>BA FF10FF102</p> <p>B Int Bdy Ext Int F 0 1 A Bdy F F 0 Ext 1 F 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>T</td><td>T</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	T	T																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		
	<p>Intersection Matrix</p> <p>AB F01FF0102</p> <p>BA FF10F0102</p> <p>B Int Bdy Ext Int F 0 1 A Bdy F F 0 Ext 1 0 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>T</td><td>T</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	T	T																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		
	<p>Intersection Matrix</p> <p>AB F01F001F2</p> <p>BA FF100F102</p> <p>B Int Bdy Ext Int F 0 1 A Bdy F 0 0 Ext 1 F 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>T</td><td>T</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	T	T																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		
	<p>Intersection Matrix</p> <p>AB F010FF1F2</p> <p>BA F010FF1F2</p> <p>B Int Bdy Ext Int F 0 1 A Bdy 0 F F Ext 1 F 2</p>	SÍ	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>T</td><td>T</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	T	T																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		



Tutorial PostGIS, 3.x

Representación	Matriz	Conmutativa	Predicados																																																													
	<p>Intersection Matrix</p> <p>AB F010FF102</p> <p>BA F010F01F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td></td> <td><i>Int</i></td> <td>F</td> <td>0</td> </tr> <tr> <td></td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Ext</i></td> <td>1</td> <td>0</td> </tr> <tr> <td></td> <td><i>Int</i></td> <td>F</td> <td>0</td> </tr> <tr> <td></td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td></td> <td><i>Ext</i></td> <td>1</td> <td>0</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>		<i>Int</i>	F	0		<i>Bdy</i>	0	F	A	<i>Ext</i>	1	0		<i>Int</i>	F	0		<i>Bdy</i>	0	F		<i>Ext</i>	1	0	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>T</td> <td>T</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																													
	<i>Int</i>	F	0																																																													
	<i>Bdy</i>	0	F																																																													
A	<i>Ext</i>	1	0																																																													
	<i>Int</i>	F	0																																																													
	<i>Bdy</i>	0	F																																																													
	<i>Ext</i>	1	0																																																													
	<i>AB</i>	<i>BA</i>																																																														
Equals	F	F																																																														
Disjoint	F	F																																																														
Intersects	T	T																																																														
Touches	T	T																																																														
Crosses	F	F																																																														
Within	F	F																																																														
Contains	F	F																																																														
Overlaps	F	F																																																														
Covers	F	F																																																														
CoveredBy	F	F																																																														
	<p>Intersection Matrix</p> <p>AB F010F01F2</p> <p>BA F010FF102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td></td> <td><i>Int</i></td> <td>F</td> <td>0</td> </tr> <tr> <td></td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Ext</i></td> <td>1</td> <td>F</td> </tr> <tr> <td></td> <td><i>Int</i></td> <td>F</td> <td>0</td> </tr> <tr> <td></td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td></td> <td><i>Ext</i></td> <td>1</td> <td>F</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>		<i>Int</i>	F	0		<i>Bdy</i>	0	F	A	<i>Ext</i>	1	F		<i>Int</i>	F	0		<i>Bdy</i>	0	F		<i>Ext</i>	1	F	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>T</td> <td>T</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																													
	<i>Int</i>	F	0																																																													
	<i>Bdy</i>	0	F																																																													
A	<i>Ext</i>	1	F																																																													
	<i>Int</i>	F	0																																																													
	<i>Bdy</i>	0	F																																																													
	<i>Ext</i>	1	F																																																													
	<i>AB</i>	<i>BA</i>																																																														
Equals	F	F																																																														
Disjoint	F	F																																																														
Intersects	T	T																																																														
Touches	T	T																																																														
Crosses	F	F																																																														
Within	F	F																																																														
Contains	F	F																																																														
Overlaps	F	F																																																														
Covers	F	F																																																														
CoveredBy	F	F																																																														
	<p>Intersection Matrix</p> <p>AB F010F0102</p> <p>BA F010F0102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td></td> <td><i>Int</i></td> <td>F</td> <td>0</td> </tr> <tr> <td></td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Ext</i></td> <td>1</td> <td>0</td> </tr> <tr> <td></td> <td><i>Int</i></td> <td>F</td> <td>0</td> </tr> <tr> <td></td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td></td> <td><i>Ext</i></td> <td>1</td> <td>0</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>		<i>Int</i>	F	0		<i>Bdy</i>	0	F	A	<i>Ext</i>	1	0		<i>Int</i>	F	0		<i>Bdy</i>	0	F		<i>Ext</i>	1	0	SÍ	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>T</td> <td>T</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																													
	<i>Int</i>	F	0																																																													
	<i>Bdy</i>	0	F																																																													
A	<i>Ext</i>	1	0																																																													
	<i>Int</i>	F	0																																																													
	<i>Bdy</i>	0	F																																																													
	<i>Ext</i>	1	0																																																													
	<i>AB</i>	<i>BA</i>																																																														
Equals	F	F																																																														
Disjoint	F	F																																																														
Intersects	T	T																																																														
Touches	T	T																																																														
Crosses	F	F																																																														
Within	F	F																																																														
Contains	F	F																																																														
Overlaps	F	F																																																														
Covers	F	F																																																														
CoveredBy	F	F																																																														
	<p>Intersection Matrix</p> <p>AB F0100F1F2</p> <p>BA F0100F1F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td></td> <td><i>Int</i></td> <td>F</td> <td>0</td> </tr> <tr> <td></td> <td><i>Bdy</i></td> <td>0</td> <td>0</td> </tr> <tr> <td>A</td> <td><i>Ext</i></td> <td>1</td> <td>F</td> </tr> <tr> <td></td> <td><i>Int</i></td> <td>F</td> <td>0</td> </tr> <tr> <td></td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td></td> <td><i>Ext</i></td> <td>1</td> <td>F</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>		<i>Int</i>	F	0		<i>Bdy</i>	0	0	A	<i>Ext</i>	1	F		<i>Int</i>	F	0		<i>Bdy</i>	0	F		<i>Ext</i>	1	F	SÍ	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>T</td> <td>T</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																													
	<i>Int</i>	F	0																																																													
	<i>Bdy</i>	0	0																																																													
A	<i>Ext</i>	1	F																																																													
	<i>Int</i>	F	0																																																													
	<i>Bdy</i>	0	F																																																													
	<i>Ext</i>	1	F																																																													
	<i>AB</i>	<i>BA</i>																																																														
Equals	F	F																																																														
Disjoint	F	F																																																														
Intersects	T	T																																																														
Touches	T	T																																																														
Crosses	F	F																																																														
Within	F	F																																																														
Contains	F	F																																																														
Overlaps	F	F																																																														
Covers	F	F																																																														
CoveredBy	F	F																																																														
	<p>Intersection Matrix</p> <p>AB 1FFF0FFF2</p> <p>BA 1FFF0FFF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td></td> <td><i>Int</i></td> <td>1</td> <td>F</td> </tr> <tr> <td></td> <td><i>Bdy</i></td> <td>F</td> <td>0</td> </tr> <tr> <td>A</td> <td><i>Ext</i></td> <td>F</td> <td>F</td> </tr> <tr> <td></td> <td><i>Int</i></td> <td>1</td> <td>F</td> </tr> <tr> <td></td> <td><i>Bdy</i></td> <td>F</td> <td>0</td> </tr> <tr> <td></td> <td><i>Ext</i></td> <td>F</td> <td>F</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>		<i>Int</i>	1	F		<i>Bdy</i>	F	0	A	<i>Ext</i>	F	F		<i>Int</i>	1	F		<i>Bdy</i>	F	0		<i>Ext</i>	F	F	SÍ	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>T</td> <td>T</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>T</td> <td>T</td> </tr> <tr> <td>Contains</td> <td>T</td> <td>T</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>T</td> <td>T</td> </tr> <tr> <td>CoveredBy</td> <td>T</td> <td>T</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	T	T	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	T	T	Contains	T	T	Overlaps	F	F	Covers	T	T	CoveredBy	T	T
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																													
	<i>Int</i>	1	F																																																													
	<i>Bdy</i>	F	0																																																													
A	<i>Ext</i>	F	F																																																													
	<i>Int</i>	1	F																																																													
	<i>Bdy</i>	F	0																																																													
	<i>Ext</i>	F	F																																																													
	<i>AB</i>	<i>BA</i>																																																														
Equals	T	T																																																														
Disjoint	F	F																																																														
Intersects	T	T																																																														
Touches	F	F																																																														
Crosses	F	F																																																														
Within	T	T																																																														
Contains	T	T																																																														
Overlaps	F	F																																																														
Covers	T	T																																																														
CoveredBy	T	T																																																														



Tutorial PostGIS, 3.x

Representación	Matriz	Conmutativa	Predicados																																																	
	<p>Intersection Matrix</p> <p>AB 1FF0FF102</p> <p>BA 101FF0FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	F	F	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	0	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>T</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>T</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>T</td> </tr> <tr> <td>CoveredBy</td> <td>T</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	T	F	Contains	F	T	Overlaps	F	F	Covers	F	T	CoveredBy	T	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	1	F	F																																																	
A	<i>Bdy</i>	0	F																																																	
<i>Ext</i>	1	0	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	T	F																																																		
Contains	F	T																																																		
Overlaps	F	F																																																		
Covers	F	T																																																		
CoveredBy	T	F																																																		
	<p>Intersection Matrix</p> <p>AB 1FF00F102</p> <p>BA 101F00FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	F	F	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	0	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>T</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>T</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>T</td> </tr> <tr> <td>CoveredBy</td> <td>T</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	T	F	Contains	F	T	Overlaps	F	F	Covers	F	T	CoveredBy	T	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	1	F	F																																																	
A	<i>Bdy</i>	0	F																																																	
<i>Ext</i>	1	0	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	T	F																																																		
Contains	F	T																																																		
Overlaps	F	F																																																		
Covers	F	T																																																		
CoveredBy	T	F																																																		
	<p>Intersection Matrix</p> <p>AB 1F1FF0102</p> <p>BA 1F1FF0102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>F</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	F	1	A	<i>Bdy</i>	F	F	<i>Ext</i>	1	0	2	SÍ	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>T</td> <td>T</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	T	T	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	1	F	1																																																	
A	<i>Bdy</i>	F	F																																																	
<i>Ext</i>	1	0	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	T	T																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 1F1F0F1F2</p> <p>BA 1F1F0F1F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>F</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	F	1	A	<i>Bdy</i>	F	F	<i>Ext</i>	1	F	2	SÍ	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>T</td> <td>T</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	T	T	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	1	F	1																																																	
A	<i>Bdy</i>	F	F																																																	
<i>Ext</i>	1	F	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	T	T																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 1F1F00102</p> <p>BA 1F1F00102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>F</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	F	1	A	<i>Bdy</i>	F	F	<i>Ext</i>	1	0	2	SÍ	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>T</td> <td>T</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	T	T	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	1	F	1																																																	
A	<i>Bdy</i>	F	F																																																	
<i>Ext</i>	1	0	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	T	T																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		



Tutorial PostGIS, 3.x

Representación	Matriz	Conmutativa	Predicados																																	
	<p>Intersection Matrix</p> <p>AB 1F100F102</p> <p>BA 101F001F2</p> <p>B Int Bdy Ext</p> <p>Int 1 F 1</p> <p>A Bdy 0 0 F</p> <p>Ext 1 0 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>T</td><td>T</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	T	T	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	F	F																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	T	T																																		
Covers	F	F																																		
CoveredBy	F	F																																		
	<p>Intersection Matrix</p> <p>AB 1F10F0102</p> <p>BA 101FF0102</p> <p>B Int Bdy Ext</p> <p>Int 1 F 1</p> <p>A Bdy 0 F 0</p> <p>Ext 1 0 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>T</td><td>T</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	T	T	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	F	F																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	T	T																																		
Covers	F	F																																		
CoveredBy	F	F																																		
	<p>Intersection Matrix</p> <p>AB 1F10FF102</p> <p>BA 101FF01F2</p> <p>B Int Bdy Ext</p> <p>Int 1 F 1</p> <p>A Bdy 0 F F</p> <p>Ext 1 0 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>T</td><td>T</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	T	T	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	F	F																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	T	T																																		
Covers	F	F																																		
CoveredBy	F	F																																		
	<p>Intersection Matrix</p> <p>AB 101FF0FF2</p> <p>BA 1FF0FF102</p> <p>B Int Bdy Ext</p> <p>Int 1 0 1</p> <p>A Bdy F F 0</p> <p>Ext F F 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>T</td></tr> <tr><td>Contains</td><td>T</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>T</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>T</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	F	T	Contains	T	F	Overlaps	F	F	Covers	T	F	CoveredBy	F	T
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	F	F																																		
Crosses	F	F																																		
Within	F	T																																		
Contains	T	F																																		
Overlaps	F	F																																		
Covers	T	F																																		
CoveredBy	F	T																																		
	<p>Intersection Matrix</p> <p>AB 101FF01F2</p> <p>BA 1F10FF102</p> <p>B Int Bdy Ext</p> <p>Int 1 0 1</p> <p>A Bdy F F 0</p> <p>Ext 1 F 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>T</td><td>T</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	T	T	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	F	F																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	T	T																																		
Covers	F	F																																		
CoveredBy	F	F																																		



Tutorial PostGIS, 3.x

Representación	Matriz	Conmutativa	Predicados																																																	
	<p>Intersection Matrix</p> <p>AB 101FF0102</p> <p>BA 1F10F0102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	0	1	A	<i>Bdy</i>	F	F	<i>Ext</i>	1	0	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>T</td> <td>T</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	T	T	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	1	0	1																																																	
A	<i>Bdy</i>	F	F																																																	
<i>Ext</i>	1	0	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	T	T																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 101F00FF2</p> <p>BA 1FF00F102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	0	1	A	<i>Bdy</i>	F	0	<i>Ext</i>	F	F	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>T</td> </tr> <tr> <td>Contains</td> <td>T</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>T</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>T</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	F	T	Contains	T	F	Overlaps	F	F	Covers	T	F	CoveredBy	F	T
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	1	0	1																																																	
A	<i>Bdy</i>	F	0																																																	
<i>Ext</i>	F	F	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	F	T																																																		
Contains	T	F																																																		
Overlaps	F	F																																																		
Covers	T	F																																																		
CoveredBy	F	T																																																		
	<p>Intersection Matrix</p> <p>AB 101F001F2</p> <p>BA 1F100F102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	0	1	A	<i>Bdy</i>	F	0	<i>Ext</i>	1	F	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>T</td> <td>T</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	T	T	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	1	0	1																																																	
A	<i>Bdy</i>	F	0																																																	
<i>Ext</i>	1	F	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	T	T																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 1010FF1F2</p> <p>BA 1010FF1F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	0	1	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	F	2	SÍ	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>T</td> <td>T</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	T	T	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	1	0	1																																																	
A	<i>Bdy</i>	0	F																																																	
<i>Ext</i>	1	F	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	T	T																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 1010FF102</p> <p>BA 1010F01F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	0	1	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	0	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>T</td> <td>T</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	T	T	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	1	0	1																																																	
A	<i>Bdy</i>	0	F																																																	
<i>Ext</i>	1	0	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	T	T																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		



Tutorial PostGIS, 3.x

Representación	Matriz	Conmutativa	Predicados																																																	
	<p>Intersection Matrix</p> <p>AB 1010F01F2</p> <p>BA 1010FF102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	0	1	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	F	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>T</td> <td>T</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	T	T	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	1	0	1																																																	
A	<i>Bdy</i>	0	F																																																	
<i>Ext</i>	1	F	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	T	T																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 1010F0102</p> <p>BA 1010F0102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	0	1	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	0	2	SÍ	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>T</td> <td>T</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	T	T	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	1	0	1																																																	
A	<i>Bdy</i>	0	F																																																	
<i>Ext</i>	1	0	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	T	T																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 10100F1F2</p> <p>BA 10100F1F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	0	1	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	F	2	SÍ	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>T</td> <td>T</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	T	T	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	1	0	1																																																	
A	<i>Bdy</i>	0	F																																																	
<i>Ext</i>	1	F	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	T	T																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 0F1FF0102</p> <p>BA 0F1FF0102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>F</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	0	F	1	A	<i>Bdy</i>	F	F	<i>Ext</i>	1	0	2	SÍ	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>T</td> <td>T</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	0	F	1																																																	
A	<i>Bdy</i>	F	F																																																	
<i>Ext</i>	1	0	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 0F1F0F1F2</p> <p>BA 0F1F0F1F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>F</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	0	F	1	A	<i>Bdy</i>	F	F	<i>Ext</i>	1	F	2	SÍ	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>T</td> <td>T</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	0	F	1																																																	
A	<i>Bdy</i>	F	F																																																	
<i>Ext</i>	1	F	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		



Tutorial PostGIS, 3.x

Representación	Matriz	Conmutativa	Predicados																																																	
	<p>Intersection Matrix</p> <p>AB 0F1F00102</p> <p>BA 0F1F00102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>F</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	0	F	1	A	<i>Bdy</i>	F	0	<i>Ext</i>	1	0	2	SÍ	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>T</td> <td>T</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	0	F	1																																																	
A	<i>Bdy</i>	F	0																																																	
<i>Ext</i>	1	0	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 0F10FF102</p> <p>BA 001FF01F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>F</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	0	F	1	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	0	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>T</td> <td>T</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	0	F	1																																																	
A	<i>Bdy</i>	0	F																																																	
<i>Ext</i>	1	0	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 0F10F0102</p> <p>BA 001FF0102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>F</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	0	F	1	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	0	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>T</td> <td>T</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	0	F	1																																																	
A	<i>Bdy</i>	0	F																																																	
<i>Ext</i>	1	0	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 0F100F102</p> <p>BA 001F001F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>F</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	0	F	1	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	0	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>T</td> <td>T</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	0	F	1																																																	
A	<i>Bdy</i>	0	F																																																	
<i>Ext</i>	1	0	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 001FF01F2</p> <p>BA 0F10FF102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	0	0	1	A	<i>Bdy</i>	F	F	<i>Ext</i>	1	F	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>T</td> <td>T</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	0	0	1																																																	
A	<i>Bdy</i>	F	F																																																	
<i>Ext</i>	1	F	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		



Tutorial PostGIS, 3.x

Representación	Matriz	Conmutativa	Predicados																																																	
	<p>Intersection Matrix</p> <p>AB 001FF0102</p> <p>BA 0F10F0102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	0	0	1	A	<i>Bdy</i>	F	F	<i>Ext</i>	1	0	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>T</td> <td>T</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	0	0	1																																																	
A	<i>Bdy</i>	F	F																																																	
<i>Ext</i>	1	0	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 001F001F2</p> <p>BA 0F100F102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	0	0	1	A	<i>Bdy</i>	F	0	<i>Ext</i>	1	F	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>T</td> <td>T</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	0	0	1																																																	
A	<i>Bdy</i>	F	0																																																	
<i>Ext</i>	1	F	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 0010FF1F2</p> <p>BA 0010FF1F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	0	0	1	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	F	2	SÍ	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>T</td> <td>T</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	0	0	1																																																	
A	<i>Bdy</i>	0	F																																																	
<i>Ext</i>	1	F	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 0010FF102</p> <p>BA 0010F01F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	0	0	1	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	0	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>T</td> <td>T</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	0	0	1																																																	
A	<i>Bdy</i>	0	F																																																	
<i>Ext</i>	1	0	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 0010F01F2</p> <p>BA 0010FF102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	0	0	1	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	F	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>T</td> <td>T</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	0	0	1																																																	
A	<i>Bdy</i>	0	F																																																	
<i>Ext</i>	1	F	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		



Tutorial PostGIS, 3.x

Representación	Matriz	Conmutativa	Predicados																																																	
	<p>Intersection Matrix</p> <p>AB 0010F0102</p> <p>BA 0010F0102</p> <table border="1"> <thead> <tr> <th>B</th> <th>Int</th> <th>Bdy</th> <th>Ext</th> </tr> </thead> <tbody> <tr> <td>Int</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>A Bdy</td> <td>0</td> <td>F</td> <td>0</td> </tr> <tr> <td>Ext</td> <td>1</td> <td>0</td> <td>2</td> </tr> </tbody> </table>	B	Int	Bdy	Ext	Int	0	0	1	A Bdy	0	F	0	Ext	1	0	2	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>T</td><td>T</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	Int	Bdy	Ext																																																	
Int	0	0	1																																																	
A Bdy	0	F	0																																																	
Ext	1	0	2																																																	
	AB	BA																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 0010F0102</p> <p>BA 0010F0102</p> <table border="1"> <thead> <tr> <th>B</th> <th>Int</th> <th>Bdy</th> <th>Ext</th> </tr> </thead> <tbody> <tr> <td>Int</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>A Bdy</td> <td>0</td> <td>F</td> <td>0</td> </tr> <tr> <td>Ext</td> <td>1</td> <td>0</td> <td>2</td> </tr> </tbody> </table>	B	Int	Bdy	Ext	Int	0	0	1	A Bdy	0	F	0	Ext	1	0	2	SÍ	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>T</td><td>T</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	Int	Bdy	Ext																																																	
Int	0	0	1																																																	
A Bdy	0	F	0																																																	
Ext	1	0	2																																																	
	AB	BA																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 00100F1F2</p> <p>BA 00100F1F2</p> <table border="1"> <thead> <tr> <th>B</th> <th>Int</th> <th>Bdy</th> <th>Ext</th> </tr> </thead> <tbody> <tr> <td>Int</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>A Bdy</td> <td>0</td> <td>0</td> <td>F</td> </tr> <tr> <td>Ext</td> <td>1</td> <td>F</td> <td>2</td> </tr> </tbody> </table>	B	Int	Bdy	Ext	Int	0	0	1	A Bdy	0	0	F	Ext	1	F	2	SÍ	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>T</td><td>T</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	Int	Bdy	Ext																																																	
Int	0	0	1																																																	
A Bdy	0	0	F																																																	
Ext	1	F	2																																																	
	AB	BA																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		



Tutorial PostGIS, 3.x

Polígono/Polígono: 31 relaciones

Representación	Matriz	Conmutativa	Predicados																																																	
	<p>Intersection Matrix</p> <p>AB FF1FF0212</p> <p>BA FF2FF1102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>F</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	F	1	A	<i>Bdy</i>	F	F	<i>Ext</i>	2	1	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>T</td> <td>T</td> </tr> <tr> <td>Intersects</td> <td>F</td> <td>F</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	T	T	Intersects	F	F	Touches	F	F	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	F	F	1																																																	
A	<i>Bdy</i>	F	F																																																	
<i>Ext</i>	2	1	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	T	T																																																		
Intersects	F	F																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 1FF0FF212</p> <p>BA 102FF1FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	F	F	A	<i>Bdy</i>	0	F	<i>Ext</i>	2	1	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>T</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>T</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>T</td> </tr> <tr> <td>CoveredBy</td> <td>T</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	T	F	Contains	F	T	Overlaps	F	F	Covers	F	T	CoveredBy	T	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	1	F	F																																																	
A	<i>Bdy</i>	0	F																																																	
<i>Ext</i>	2	1	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	T	F																																																		
Contains	F	T																																																		
Overlaps	F	F																																																		
Covers	F	T																																																		
CoveredBy	T	F																																																		
	<p>Intersection Matrix</p> <p>AB F01FF0212</p> <p>BA FF20F1102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	0	1	A	<i>Bdy</i>	F	F	<i>Ext</i>	2	1	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>T</td> <td>T</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	F	0	1																																																	
A	<i>Bdy</i>	F	F																																																	
<i>Ext</i>	2	1	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	T	T																																																		
Crosses	F	F																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 101FF0212</p> <p>BA 1F20F1102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	0	1	A	<i>Bdy</i>	F	F	<i>Ext</i>	2	1	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>T</td> <td>T</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	1	0	1																																																	
A	<i>Bdy</i>	F	F																																																	
<i>Ext</i>	2	1	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 1010F0212</p> <p>BA 1020F1102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	0	1	A	<i>Bdy</i>	0	F	<i>Ext</i>	2	1	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>T</td> <td>T</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	1	0	1																																																	
A	<i>Bdy</i>	0	F																																																	
<i>Ext</i>	2	1	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		



Tutorial PostGIS, 3.x

Representación	Matriz	Conmutativa	Predicados																																																	
	<p>Intersection Matrix</p> <p>AB 10F0FF212</p> <p>BA 1020F1FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	0	F	A	<i>Bdy</i>	0	F	<i>Ext</i>	2	1	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>T</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>T</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>T</td> </tr> <tr> <td>CoveredBy</td> <td>T</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	T	F	Contains	F	T	Overlaps	F	F	Covers	F	T	CoveredBy	T	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	1	0	F																																																	
A	<i>Bdy</i>	0	F																																																	
<i>Ext</i>	2	1	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	T	F																																																		
Contains	F	T																																																		
Overlaps	F	F																																																		
Covers	F	T																																																		
CoveredBy	T	F																																																		
	<p>Intersection Matrix</p> <p>AB 1010FF212</p> <p>BA 1020F11F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	0	1	A	<i>Bdy</i>	0	F	<i>Ext</i>	2	1	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>T</td> <td>T</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	1	0	1																																																	
A	<i>Bdy</i>	0	F																																																	
<i>Ext</i>	2	1	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB F11FF0212</p> <p>BA FF21F1102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>1</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	1	1	A	<i>Bdy</i>	F	F	<i>Ext</i>	2	1	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>T</td> <td>T</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	F	1	1																																																	
A	<i>Bdy</i>	F	F																																																	
<i>Ext</i>	2	1	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	T	T																																																		
Crosses	F	F																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 111FF0212</p> <p>BA 1F21F1102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	1	1	A	<i>Bdy</i>	F	F	<i>Ext</i>	2	1	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>T</td> <td>T</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	1	1	1																																																	
A	<i>Bdy</i>	F	F																																																	
<i>Ext</i>	2	1	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 11F0FF212</p> <p>BA 1021F1FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>1</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	1	F	A	<i>Bdy</i>	0	F	<i>Ext</i>	2	1	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>T</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>T</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>T</td> </tr> <tr> <td>CoveredBy</td> <td>T</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	T	F	Contains	F	T	Overlaps	F	F	Covers	F	T	CoveredBy	T	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	1	1	F																																																	
A	<i>Bdy</i>	0	F																																																	
<i>Ext</i>	2	1	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	T	F																																																		
Contains	F	T																																																		
Overlaps	F	F																																																		
Covers	F	T																																																		
CoveredBy	T	F																																																		



Tutorial PostGIS, 3.x

Representación	Matriz	Conmutativa	Predicados																																	
	<p>Intersection Matrix</p> <p>AB 1110FF212</p> <p>BA 1021F11F2</p> <p>B <i>Int Bdy Ext</i></p> <p><i>Int</i> 1 1 1</p> <p>A <i>Bdy</i> 0 F F</p> <p><i>Ext</i> 2 1 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>T</td><td>T</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	F	F																																		
Crosses	T	T																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		
	<p>Intersection Matrix</p> <p>AB 1110F0212</p> <p>BA 1021F1102</p> <p>B <i>Int Bdy Ext</i></p> <p><i>Int</i> 1 1 1</p> <p>A <i>Bdy</i> 0 F 0</p> <p><i>Ext</i> 2 1 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>T</td><td>T</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	F	F																																		
Crosses	T	T																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		
	<p>Intersection Matrix</p> <p>AB FF1F0F212</p> <p>BA FF2F011F2</p> <p>B <i>Int Bdy Ext</i></p> <p><i>Int</i> F F 1</p> <p>A <i>Bdy</i> F 0 F</p> <p><i>Ext</i> 2 1 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>T</td><td>T</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	T	T																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		
	<p>Intersection Matrix</p> <p>AB FF1F00212</p> <p>BA FF2F01102</p> <p>B <i>Int Bdy Ext</i></p> <p><i>Int</i> F F 1</p> <p>A <i>Bdy</i> F 0 0</p> <p><i>Ext</i> 2 1 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>T</td><td>T</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	T	T																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		
	<p>Intersection Matrix</p> <p>AB 1FFF0F212</p> <p>BA 1F2F01FF2</p> <p>B <i>Int Bdy Ext</i></p> <p><i>Int</i> 1 F F</p> <p>A <i>Bdy</i> F 0 F</p> <p><i>Ext</i> 2 1 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>T</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>T</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>T</td></tr> <tr><td>CoveredBy</td><td>T</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	T	F	Contains	F	T	Overlaps	F	F	Covers	F	T	CoveredBy	T	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	F	F																																		
Crosses	F	F																																		
Within	T	F																																		
Contains	F	T																																		
Overlaps	F	F																																		
Covers	F	T																																		
CoveredBy	T	F																																		



Tutorial PostGIS, 3.x

Representación	Matriz	Conmutativa	Predicados																																	
	<p>Intersection Matrix</p> <p>AB 1FF00F212</p> <p>BA 102F01FF2</p> <p>B Int Bdy Ext</p> <p>Int 1 F F</p> <p>A Bdy 0 0 F</p> <p>Ext 2 1 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>T</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>T</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>T</td></tr> <tr><td>CoveredBy</td><td>T</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	T	F	Contains	F	T	Overlaps	F	F	Covers	F	T	CoveredBy	T	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	F	F																																		
Crosses	F	F																																		
Within	T	F																																		
Contains	F	T																																		
Overlaps	F	F																																		
Covers	F	T																																		
CoveredBy	T	F																																		
	<p>Intersection Matrix</p> <p>AB F01F0F212</p> <p>BA FF20011F2</p> <p>B Int Bdy Ext</p> <p>Int F 0 1</p> <p>A Bdy F 0 F</p> <p>Ext 2 1 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>T</td><td>T</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	T	T																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		
	<p>Intersection Matrix</p> <p>AB F01F00212</p> <p>BA FF2001102</p> <p>B Int Bdy Ext</p> <p>Int F 0 1</p> <p>A Bdy F 0 0</p> <p>Ext 2 1 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>T</td><td>T</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	T	T																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		
	<p>Intersection Matrix</p> <p>AB 10FF0F212</p> <p>BA 1F2001FF2</p> <p>B Int Bdy Ext</p> <p>Int 1 0 F</p> <p>A Bdy F 0 F</p> <p>Ext 2 1 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>T</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>T</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>T</td></tr> <tr><td>CoveredBy</td><td>T</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	T	F	Contains	F	T	Overlaps	F	F	Covers	F	T	CoveredBy	T	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	F	F																																		
Crosses	F	F																																		
Within	T	F																																		
Contains	F	T																																		
Overlaps	F	F																																		
Covers	F	T																																		
CoveredBy	T	F																																		
	<p>Intersection Matrix</p> <p>AB 101F0F212</p> <p>BA 1F20011F2</p> <p>B Int Bdy Ext</p> <p>Int 1 0 1</p> <p>A Bdy F 0 F</p> <p>Ext 2 1 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>T</td><td>T</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	F	F																																		
Crosses	T	T																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		



Tutorial PostGIS, 3.x

Representación	Matriz	Conmutativa	Predicados																																	
	<p>Intersection Matrix</p> <p>AB 101F00212</p> <p>BA 1F2001102</p> <p>B Int Bdy Ext Int 1 0 1 A Bdy F 0 0 Ext 2 1 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>T</td><td>T</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	F	F																																		
Crosses	T	T																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		
	<p>Intersection Matrix</p> <p>AB 10F00F212</p> <p>BA 102001FF2</p> <p>B Int Bdy Ext Int 1 0 F A Bdy 0 0 F Ext 2 1 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>T</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>T</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>T</td></tr> <tr><td>CoveredBy</td><td>T</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	T	F	Contains	F	T	Overlaps	F	F	Covers	F	T	CoveredBy	T	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	F	F																																		
Crosses	F	F																																		
Within	T	F																																		
Contains	F	T																																		
Overlaps	F	F																																		
Covers	F	T																																		
CoveredBy	T	F																																		
	<p>Intersection Matrix</p> <p>AB 10100F212</p> <p>BA 1020011F2</p> <p>B Int Bdy Ext Int 1 0 1 A Bdy 0 0 F Ext 2 1 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>T</td><td>T</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	F	F																																		
Crosses	T	T																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		
	<p>Intersection Matrix</p> <p>AB F1FF0F212</p> <p>BA FF2101FF2</p> <p>B Int Bdy Ext Int F 1 F A Bdy F 0 F Ext 2 1 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>T</td><td>T</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>T</td></tr> <tr><td>CoveredBy</td><td>T</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	T	CoveredBy	T	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	T	T																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	T																																		
CoveredBy	T	F																																		
	<p>Intersection Matrix</p> <p>AB F11F0F212</p> <p>BA FF21011F2</p> <p>B Int Bdy Ext Int F 1 1 A Bdy F 0 F Ext 2 1 2</p>	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>T</td><td>T</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
	AB	BA																																		
Equals	F	F																																		
Disjoint	F	F																																		
Intersects	T	T																																		
Touches	T	T																																		
Crosses	F	F																																		
Within	F	F																																		
Contains	F	F																																		
Overlaps	F	F																																		
Covers	F	F																																		
CoveredBy	F	F																																		



Tutorial PostGIS, 3.x

Representación	Matriz	Conmutativa	Predicados																																																	
	<p>Intersection Matrix</p> <p>AB F11F00212</p> <p>BA FF2101102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td></td> <td>F</td> <td>1</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>0 0</td> </tr> <tr> <td></td> <td>Ext</td> <td>2</td> <td>1 2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>		F	1	1	A	<i>Bdy</i>	F	0 0		Ext	2	1 2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td>AB</td> <td>BA</td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>T</td> <td>T</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
	F	1	1																																																	
A	<i>Bdy</i>	F	0 0																																																	
	Ext	2	1 2																																																	
	AB	BA																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	T	T																																																		
Crosses	F	F																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 11FF0F212</p> <p>BA 1F2101FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td></td> <td>1</td> <td>1</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>0 F</td> </tr> <tr> <td></td> <td>Ext</td> <td>2</td> <td>1 2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>		1	1	F	A	<i>Bdy</i>	F	0 F		Ext	2	1 2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td>AB</td> <td>BA</td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>T</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>T</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>T</td> </tr> <tr> <td>CoveredBy</td> <td>T</td> <td>F</td> </tr> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	T	F	Contains	F	T	Overlaps	F	F	Covers	F	T	CoveredBy	T	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
	1	1	F																																																	
A	<i>Bdy</i>	F	0 F																																																	
	Ext	2	1 2																																																	
	AB	BA																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	T	F																																																		
Contains	F	T																																																		
Overlaps	F	F																																																		
Covers	F	T																																																		
CoveredBy	T	F																																																		
	<p>Intersection Matrix</p> <p>AB 111F0F212</p> <p>BA 1F21011F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td></td> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>0 F</td> </tr> <tr> <td></td> <td>Ext</td> <td>2</td> <td>1 2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>		1	1	1	A	<i>Bdy</i>	F	0 F		Ext	2	1 2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td>AB</td> <td>BA</td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>T</td> <td>T</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
	1	1	1																																																	
A	<i>Bdy</i>	F	0 F																																																	
	Ext	2	1 2																																																	
	AB	BA																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 111F00212</p> <p>BA 1F2101102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td></td> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>0 0</td> </tr> <tr> <td></td> <td>Ext</td> <td>2</td> <td>1 2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>		1	1	1	A	<i>Bdy</i>	F	0 0		Ext	2	1 2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td>AB</td> <td>BA</td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>T</td> <td>T</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
	1	1	1																																																	
A	<i>Bdy</i>	F	0 0																																																	
	Ext	2	1 2																																																	
	AB	BA																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 11F00F212</p> <p>BA 102101FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td></td> <td>1</td> <td>1</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>0 F</td> </tr> <tr> <td></td> <td>Ext</td> <td>2</td> <td>1 2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>		1	1	F	A	<i>Bdy</i>	0	0 F		Ext	2	1 2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td>AB</td> <td>BA</td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>T</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>T</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>T</td> </tr> <tr> <td>CoveredBy</td> <td>T</td> <td>F</td> </tr> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	T	F	Contains	F	T	Overlaps	F	F	Covers	F	T	CoveredBy	T	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
	1	1	F																																																	
A	<i>Bdy</i>	0	0 F																																																	
	Ext	2	1 2																																																	
	AB	BA																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	T	F																																																		
Contains	F	T																																																		
Overlaps	F	F																																																		
Covers	F	T																																																		
CoveredBy	T	F																																																		
	<p>Intersection Matrix</p> <p>AB 11100F212</p> <p>BA 1021011F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td></td> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>0 F</td> </tr> <tr> <td></td> <td>Ext</td> <td>2</td> <td>1 2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>		1	1	1	A	<i>Bdy</i>	0	0 F		Ext	2	1 2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td>AB</td> <td>BA</td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>T</td> <td>T</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	T	T	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
	1	1	1																																																	
A	<i>Bdy</i>	0	0 F																																																	
	Ext	2	1 2																																																	
	AB	BA																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	T	T																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		



Tutorial PostGIS, 3.x

Polígono/Polígono: 12 relaciones

Representación	Matriz	Conmutativa	Predicados																																																	
	<p>Intersection Matrix</p> <p>AB 2FFF1FFF2</p> <p>BA 2FFF1FFF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>2</td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>1</td> </tr> <tr> <td><i>Ext</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	2	F	F	A	<i>Bdy</i>	F	1	<i>Ext</i>	F	F	2	SÍ	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>T</td> <td>T</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>T</td> <td>T</td> </tr> <tr> <td>Contains</td> <td>T</td> <td>T</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>T</td> <td>T</td> </tr> <tr> <td>CoveredBy</td> <td>T</td> <td>T</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	T	T	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	T	T	Contains	T	T	Overlaps	F	F	Covers	T	T	CoveredBy	T	T
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	2	F	F																																																	
A	<i>Bdy</i>	F	1																																																	
<i>Ext</i>	F	F	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	T	T																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	T	T																																																		
Contains	T	T																																																		
Overlaps	F	F																																																		
Covers	T	T																																																		
CoveredBy	T	T																																																		
	<p>Intersection Matrix</p> <p>AB FF2FF1212</p> <p>BA FF2FF1212</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	F	2	A	<i>Bdy</i>	F	F	<i>Ext</i>	2	1	2	SÍ	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>T</td> <td>T</td> </tr> <tr> <td>Intersects</td> <td>F</td> <td>F</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	T	T	Intersects	F	F	Touches	F	F	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	F	F	2																																																	
A	<i>Bdy</i>	F	F																																																	
<i>Ext</i>	2	1	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	T	T																																																		
Intersects	F	F																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB FF2F01212</p> <p>BA FF2F01212</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	F	2	A	<i>Bdy</i>	F	0	<i>Ext</i>	2	1	2	SÍ	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>T</td> <td>T</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	F	F	2																																																	
A	<i>Bdy</i>	F	0																																																	
<i>Ext</i>	2	1	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	T	T																																																		
Crosses	F	F																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB FF2F11212</p> <p>BA FF2F11212</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>1</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	F	2	A	<i>Bdy</i>	F	1	<i>Ext</i>	2	1	2	SÍ	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>T</td> <td>T</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>F</td> </tr> <tr> <td>Contains</td> <td>F</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>F</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>F</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	F	F	2																																																	
A	<i>Bdy</i>	F	1																																																	
<i>Ext</i>	2	1	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	T	T																																																		
Crosses	F	F																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 212FF1FF2</p> <p>BA 2FF1FF212</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	2	1	2	A	<i>Bdy</i>	F	F	<i>Ext</i>	F	F	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>T</td> </tr> <tr> <td>Contains</td> <td>T</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>T</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>T</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	F	T	Contains	T	F	Overlaps	F	F	Covers	T	F	CoveredBy	F	T
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	2	1	2																																																	
A	<i>Bdy</i>	F	F																																																	
<i>Ext</i>	F	F	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	F	T																																																		
Contains	T	F																																																		
Overlaps	F	F																																																		
Covers	T	F																																																		
CoveredBy	F	T																																																		
	<p>Intersection Matrix</p> <p>AB 212F01FF2</p> <p>BA 2FF10F212</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	2	1	2	A	<i>Bdy</i>	F	0	<i>Ext</i>	F	F	2	NO	<p>Binary Predicates</p> <table border="1"> <tr> <td></td> <td><i>AB</i></td> <td><i>BA</i></td> </tr> <tr> <td>Equals</td> <td>F</td> <td>F</td> </tr> <tr> <td>Disjoint</td> <td>F</td> <td>F</td> </tr> <tr> <td>Intersects</td> <td>T</td> <td>T</td> </tr> <tr> <td>Touches</td> <td>F</td> <td>F</td> </tr> <tr> <td>Crosses</td> <td>F</td> <td>F</td> </tr> <tr> <td>Within</td> <td>F</td> <td>T</td> </tr> <tr> <td>Contains</td> <td>T</td> <td>F</td> </tr> <tr> <td>Overlaps</td> <td>F</td> <td>F</td> </tr> <tr> <td>Covers</td> <td>T</td> <td>F</td> </tr> <tr> <td>CoveredBy</td> <td>F</td> <td>T</td> </tr> </table>		<i>AB</i>	<i>BA</i>	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	F	T	Contains	T	F	Overlaps	F	F	Covers	T	F	CoveredBy	F	T
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>																																																	
<i>Int</i>	2	1	2																																																	
A	<i>Bdy</i>	F	0																																																	
<i>Ext</i>	F	F	2																																																	
	<i>AB</i>	<i>BA</i>																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	F	T																																																		
Contains	T	F																																																		
Overlaps	F	F																																																		
Covers	T	F																																																		
CoveredBy	F	T																																																		



Tutorial PostGIS, 3.x

Representación	Matriz	Conmutativa	Predicados																																																	
	<p>Intersection Matrix</p> <p>AB 212111212</p> <p>BA 212111212</p> <table border="1"> <thead> <tr> <th>B</th> <th>Int</th> <th>Bdy</th> <th>Ext</th> </tr> </thead> <tbody> <tr> <td>Int</td> <td>2</td> <td>1</td> <td>2</td> </tr> <tr> <td>A Bdy</td> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>Ext</td> <td>2</td> <td>1</td> <td>2</td> </tr> </tbody> </table>	B	Int	Bdy	Ext	Int	2	1	2	A Bdy	1	1	1	Ext	2	1	2	SÍ	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>T</td><td>T</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	T	T	Covers	F	F	CoveredBy	F	F
B	Int	Bdy	Ext																																																	
Int	2	1	2																																																	
A Bdy	1	1	1																																																	
Ext	2	1	2																																																	
	AB	BA																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	T	T																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 212101212</p> <p>BA 212101212</p> <table border="1"> <thead> <tr> <th>B</th> <th>Int</th> <th>Bdy</th> <th>Ext</th> </tr> </thead> <tbody> <tr> <td>Int</td> <td>2</td> <td>1</td> <td>2</td> </tr> <tr> <td>A Bdy</td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>Ext</td> <td>2</td> <td>1</td> <td>2</td> </tr> </tbody> </table>	B	Int	Bdy	Ext	Int	2	1	2	A Bdy	1	0	1	Ext	2	1	2	SÍ	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>T</td><td>T</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	T	T	Covers	F	F	CoveredBy	F	F
B	Int	Bdy	Ext																																																	
Int	2	1	2																																																	
A Bdy	1	0	1																																																	
Ext	2	1	2																																																	
	AB	BA																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	T	T																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 2FF1FF212</p> <p>BA 212FF1FF2</p> <table border="1"> <thead> <tr> <th>B</th> <th>Int</th> <th>Bdy</th> <th>Ext</th> </tr> </thead> <tbody> <tr> <td>Int</td> <td>2</td> <td>F</td> <td>F</td> </tr> <tr> <td>A Bdy</td> <td>1</td> <td>F</td> <td>F</td> </tr> <tr> <td>Ext</td> <td>2</td> <td>1</td> <td>2</td> </tr> </tbody> </table>	B	Int	Bdy	Ext	Int	2	F	F	A Bdy	1	F	F	Ext	2	1	2	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>T</td><td>T</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>F</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>F</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>F</td></tr> <tr><td>CoveredBy</td><td>F</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	T	T	Crosses	F	F	Within	F	F	Contains	F	F	Overlaps	F	F	Covers	F	F	CoveredBy	F	F
B	Int	Bdy	Ext																																																	
Int	2	F	F																																																	
A Bdy	1	F	F																																																	
Ext	2	1	2																																																	
	AB	BA																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	T	T																																																		
Crosses	F	F																																																		
Within	F	F																																																		
Contains	F	F																																																		
Overlaps	F	F																																																		
Covers	F	F																																																		
CoveredBy	F	F																																																		
	<p>Intersection Matrix</p> <p>AB 2FF11F212</p> <p>BA 212F11FF2</p> <table border="1"> <thead> <tr> <th>B</th> <th>Int</th> <th>Bdy</th> <th>Ext</th> </tr> </thead> <tbody> <tr> <td>Int</td> <td>2</td> <td>F</td> <td>F</td> </tr> <tr> <td>A Bdy</td> <td>1</td> <td>1</td> <td>F</td> </tr> <tr> <td>Ext</td> <td>2</td> <td>1</td> <td>2</td> </tr> </tbody> </table>	B	Int	Bdy	Ext	Int	2	F	F	A Bdy	1	1	F	Ext	2	1	2	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>T</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>T</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>T</td></tr> <tr><td>CoveredBy</td><td>T</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	T	F	Contains	F	T	Overlaps	F	F	Covers	F	T	CoveredBy	T	F
B	Int	Bdy	Ext																																																	
Int	2	F	F																																																	
A Bdy	1	1	F																																																	
Ext	2	1	2																																																	
	AB	BA																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	T	F																																																		
Contains	F	T																																																		
Overlaps	F	F																																																		
Covers	F	T																																																		
CoveredBy	T	F																																																		
	<p>Intersection Matrix</p> <p>AB 2FF10F212</p> <p>BA 212F01FF2</p> <table border="1"> <thead> <tr> <th>B</th> <th>Int</th> <th>Bdy</th> <th>Ext</th> </tr> </thead> <tbody> <tr> <td>Int</td> <td>2</td> <td>F</td> <td>F</td> </tr> <tr> <td>A Bdy</td> <td>1</td> <td>0</td> <td>F</td> </tr> <tr> <td>Ext</td> <td>2</td> <td>1</td> <td>2</td> </tr> </tbody> </table>	B	Int	Bdy	Ext	Int	2	F	F	A Bdy	1	0	F	Ext	2	1	2	NO	<p>Binary Predicates</p> <table border="1"> <thead> <tr> <th></th> <th>AB</th> <th>BA</th> </tr> </thead> <tbody> <tr><td>Equals</td><td>F</td><td>F</td></tr> <tr><td>Disjoint</td><td>F</td><td>F</td></tr> <tr><td>Intersects</td><td>T</td><td>T</td></tr> <tr><td>Touches</td><td>F</td><td>F</td></tr> <tr><td>Crosses</td><td>F</td><td>F</td></tr> <tr><td>Within</td><td>T</td><td>F</td></tr> <tr><td>Contains</td><td>F</td><td>T</td></tr> <tr><td>Overlaps</td><td>F</td><td>F</td></tr> <tr><td>Covers</td><td>F</td><td>T</td></tr> <tr><td>CoveredBy</td><td>T</td><td>F</td></tr> </tbody> </table>		AB	BA	Equals	F	F	Disjoint	F	F	Intersects	T	T	Touches	F	F	Crosses	F	F	Within	T	F	Contains	F	T	Overlaps	F	F	Covers	F	T	CoveredBy	T	F
B	Int	Bdy	Ext																																																	
Int	2	F	F																																																	
A Bdy	1	0	F																																																	
Ext	2	1	2																																																	
	AB	BA																																																		
Equals	F	F																																																		
Disjoint	F	F																																																		
Intersects	T	T																																																		
Touches	F	F																																																		
Crosses	F	F																																																		
Within	T	F																																																		
Contains	F	T																																																		
Overlaps	F	F																																																		
Covers	F	T																																																		
CoveredBy	T	F																																																		



Apéndice A-2: Relaciones topológicas clasificadas por predicado

Servirá para ilustrar de manera concreta los efectos de cada predicado.

PATRÓN DE 9 CARACTERES PARA MATRIZ DE-9IM

Las variantes de la matriz DE-9IM se pueden representar por un patrón de 9 caracteres de texto mediante los símbolos F, 0, 1, 2, por ejemplo 'FF1FF0102'. Un tipo específico de relación espacial puede expresarse al parrear la matriz de intersección con el patrón escogido. Los patrones pueden tener además los símbolos:

Símbolo	Significado
T	"la intersección no es vacía/nula"
*	"cualquier valor"

Por ejemplo, la relación "Intersección-de-interiores" tendrá el patrón 'T*****' el cual no está evaluado en ninguno de los predicados nombrados.

Repasando:

Tabla A-1: Matriz de intersección 9-IM

	Interior	Límite/Contorno/Boundary	Exterior
Punto (Dim=0) 	Dim=0 	Vacío / No existe	
Poli-línea (Dim=1) 	Dim=1 	Dim=0 	
Polígono/área (Dim=2) 	Dim=2 	Dim=1 	

Referencia: [Predicados OGC: 2.2: Conceptos de interior, límite y exterior de objetos](#), pág. 6. Ministerio de Ecología, desarrollo sustentable, transportación y vivienda, Gobierno de Francia.



ST_EQUALS

Equivalente para usar en ST_Relate: [T * F ** F F F *].

Citado de https://postgis.net/docs/ST_Equals.html

Devuelve TRUE cuando las geometrías evaluadas son “topológicamente iguales”. Use esto como una mejor respuesta que decir que son iguales (=). Igualdad topológica significa que las geometrías tienen:

- La misma dimensión
- Los puntos ocupan el mismo espacio

Esto quiere decir que el orden de los vértices puede ser diferente entre geometrías topológicamente iguales.

La siguiente relación es cierta: $ST_Equals(a,b) \leftrightarrow ST_Within(a,b) \wedge ST_Within(b,a)$.

Relación	Diagrama	Matriz
P/P		<p>Intersection Matrix</p> <p>AB 0FFFFFFF2</p> <p>BA 0FFFFFFF2</p> <p>B <i>Int Bdy Ext</i></p> <p><i>Int</i> 0 F F</p> <p>A <i>Bdy</i> F F F</p> <p><i>Ext</i> F F 2</p>
L/L		<p>Intersection Matrix</p> <p>AB 1FFF0FFF2</p> <p>BA 1FFF0FFF2</p> <p>B <i>Int Bdy Ext</i></p> <p><i>Int</i> 1 F F</p> <p>A <i>Bdy</i> F 0 F</p> <p><i>Ext</i> F F 2</p>
S/S		<p>Intersection Matrix</p> <p>AB 2FFF1FFF2</p> <p>BA 2FFF1FFF2</p> <p>B <i>Int Bdy Ext</i></p> <p><i>Int</i> 2 F F</p> <p>A <i>Bdy</i> F 1 F</p> <p><i>Ext</i> F F 2</p>



Tutorial PostGIS, 3.x

ST_DisJOINT

Equivalente para usar en ST_Relate: [FF*FF****].

Citado de https://postgis.net/docs/ST_Disjoint.html

Devuelve cierto si las dos geometrías evaluadas están desjuntas. Las geometrías están desjuntas si no tienen ningún punto en común. Si alguna otra relación espacial es cierta para un par de geometrías, entonces no están desjuntas. Disjoint implica que ST_Intersects devolverá falso.

Relación	Diagrama	Matriz
P/P		Intersection Matrix AB FF0FFF0F2 BA FF0FFF0F2 B Int Bdy Ext Int F F 0 A Bdy F F F Ext 0 F 2
P/L		Intersection Matrix AB FF0FFF102 BA FF1FF00F2 B Int Bdy Ext Int F F 0 A Bdy F F F Ext 1 0 2
P/S		Intersection Matrix AB FF0FFF212 BA FF2FF10F2 B Int Bdy Ext Int F F 0 A Bdy F F F Ext 2 1 2
L/P		Intersection Matrix AB FF1FF00F2 BA FF0FFF102 B Int Bdy Ext Int F F 1 A Bdy F F 0 Ext 0 F 2
L/L		Intersection Matrix AB FF1FF0102 BA FF1FF0102 B Int Bdy Ext Int F F 1 A Bdy F F 0 Ext 1 0 2
L/S		Intersection Matrix AB FF1FF0212 BA FF2FF1102 B Int Bdy Ext Int F F 1 A Bdy F F 0 Ext 2 1 2



Tutorial PostGIS, 3.x

ST_Disjoint (cont.)

Relación	Diagrama	Matriz																
S/P		<p>Intersection Matrix</p> <p>AB FF2FF10F2</p> <p>BA FF0FFF212</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td></td> <td><i>Int</i></td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td></td> <td><i>Ext</i></td> <td>0</td> <td>F</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>		<i>Int</i>	F	F	A	<i>Bdy</i>	F	F		<i>Ext</i>	0	F
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
	<i>Int</i>	F	F															
A	<i>Bdy</i>	F	F															
	<i>Ext</i>	0	F															
S/L		<p>Intersection Matrix</p> <p>AB FF2FF1102</p> <p>BA FF1FF0212</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td></td> <td><i>Int</i></td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td></td> <td><i>Ext</i></td> <td>1</td> <td>0</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>		<i>Int</i>	F	F	A	<i>Bdy</i>	F	F		<i>Ext</i>	1	0
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
	<i>Int</i>	F	F															
A	<i>Bdy</i>	F	F															
	<i>Ext</i>	1	0															
S/S		<p>Intersection Matrix</p> <p>AB FF2FF1212</p> <p>BA FF2FF1212</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td></td> <td><i>Int</i></td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td></td> <td><i>Ext</i></td> <td>2</td> <td>1</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>		<i>Int</i>	F	F	A	<i>Bdy</i>	F	F		<i>Ext</i>	2	1
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
	<i>Int</i>	F	F															
A	<i>Bdy</i>	F	F															
	<i>Ext</i>	2	1															



Tutorial PostGIS, 3.x

ST_TOUCHES

Citado de https://postgis.net/docs/ST_Touches.html

Devuelve TRUE si las geometrías a y b se intersectan, pero sus interiores no intersectan. De igual manera, a y b deben tener al menos un punto en común y estos puntos en común deben estar al menos en un límite/contorno. Al evaluar esta función con geometrías Punto/Punto, la relación es siempre FALSE, ya que los puntos no tienen contorno.

Equivalentes para usar en ST_Relate: [FT***], [F***T****], [F**T*****].**

Esta función utiliza los comparadores de cajas de extensión (bounding boxes) que hacen uso de cualquier índice espacial disponible para las geometrías. Si no desea usar el índice disponible, sustituya por la función `_ST_Touches`.

Relación	Diagrama	Matriz
P/L		<pre> Intersection Matrix: AB F0FFFF102 BA FF10F0FF2 B Int Body Ext Int F 0 F A Body F F F Ext 1 0 2 </pre>
P/S		<pre> Intersection Matrix: AB FF0FFF212 BA FF2FF10F2 B Int Body Ext Int F F 0 A Body F F F Ext 2 1 2 </pre>
L/P		<pre> Intersection Matrix: AB FF10F0FF2 BA F0FFFF102 B Int Body Ext Int F F 1 A Body 0 F 0 Ext F F 2 </pre>
L/L		<pre> Intersection Matrix: AB FF1F0F1F2 BA FF1F0F1F2 B Int Body Ext Int F F 1 A Body F 0 F Ext 1 F 2 </pre>
L/L		<pre> Intersection Matrix: AB FF1F00102 BA FF1F00102 B Int Body Ext Int F F 1 A Body F 0 0 Ext 1 0 2 </pre>
L/L		<pre> Intersection Matrix: AB FF10FF102 BA F01FF01F2 B Int Body Ext Int F F 1 A Body 0 F F Ext 1 0 2 </pre>



Tutorial PostGIS, 3.x

ST_Touches (cont.)

Relación	Diagrama	Matriz																
L/L		<p>Intersection Matrix</p> <p>AB FF10F0102</p> <p>BA F01FF0102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>F</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	F	1	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	0	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	F	F	1															
A	<i>Bdy</i>	0	F															
<i>Ext</i>	1	0	2															
L/L		<p>Intersection Matrix</p> <p>AB FF100F102</p> <p>BA F01F001F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>F</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	F	1	A	<i>Bdy</i>	0	0	<i>Ext</i>	1	0	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	F	F	1															
A	<i>Bdy</i>	0	0															
<i>Ext</i>	1	0	2															
L/L		<p>Intersection Matrix</p> <p>AB F01FF01F2</p> <p>BA FF10FF102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	0	1	A	<i>Bdy</i>	F	F	<i>Ext</i>	1	F	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	F	0	1															
A	<i>Bdy</i>	F	F															
<i>Ext</i>	1	F	2															
L/L		<p>Intersection Matrix</p> <p>AB F01FF0102</p> <p>BA FF10F0102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	0	1	A	<i>Bdy</i>	F	F	<i>Ext</i>	1	0	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	F	0	1															
A	<i>Bdy</i>	F	F															
<i>Ext</i>	1	0	2															
L/L		<p>Intersection Matrix</p> <p>AB F01F001F2</p> <p>BA FF100F102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	0	1	A	<i>Bdy</i>	F	0	<i>Ext</i>	1	F	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	F	0	1															
A	<i>Bdy</i>	F	0															
<i>Ext</i>	1	F	2															
L/L		<p>Intersection Matrix</p> <p>AB F010FF1F2</p> <p>BA F010FF1F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	0	1	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	F	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	F	0	1															
A	<i>Bdy</i>	0	F															
<i>Ext</i>	1	F	2															
L/L		<p>Intersection Matrix</p> <p>AB F010FF102</p> <p>BA F010F01F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	0	1	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	0	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	F	0	1															
A	<i>Bdy</i>	0	F															
<i>Ext</i>	1	0	2															



Tutorial PostGIS, 3.x

ST_Touches (cont.)

Relación	Diagrama	Matriz																
L/L		<p>Intersection Matrix</p> <p>AB F010F01F2</p> <p>BA F010FF102</p> <table border="1"> <thead> <tr> <th>B</th> <th>Int</th> <th>Bdy</th> <th>Ext</th> </tr> </thead> <tbody> <tr> <td>Int</td> <td>F</td> <td>0</td> <td>1</td> </tr> <tr> <td>A Bdy</td> <td>0</td> <td>F</td> <td>0</td> </tr> <tr> <td>Ext</td> <td>1</td> <td>F</td> <td>2</td> </tr> </tbody> </table>	B	Int	Bdy	Ext	Int	F	0	1	A Bdy	0	F	0	Ext	1	F	2
B	Int	Bdy	Ext															
Int	F	0	1															
A Bdy	0	F	0															
Ext	1	F	2															
L/L		<p>Intersection Matrix</p> <p>AB F010F01F2</p> <p>BA F010FF102</p> <table border="1"> <thead> <tr> <th>B</th> <th>Int</th> <th>Bdy</th> <th>Ext</th> </tr> </thead> <tbody> <tr> <td>Int</td> <td>F</td> <td>0</td> <td>1</td> </tr> <tr> <td>A Bdy</td> <td>0</td> <td>F</td> <td>0</td> </tr> <tr> <td>Ext</td> <td>1</td> <td>F</td> <td>2</td> </tr> </tbody> </table>	B	Int	Bdy	Ext	Int	F	0	1	A Bdy	0	F	0	Ext	1	F	2
B	Int	Bdy	Ext															
Int	F	0	1															
A Bdy	0	F	0															
Ext	1	F	2															
L/L		<p>Intersection Matrix</p> <p>AB F010F0102</p> <p>BA F010F0102</p> <table border="1"> <thead> <tr> <th>B</th> <th>Int</th> <th>Bdy</th> <th>Ext</th> </tr> </thead> <tbody> <tr> <td>Int</td> <td>F</td> <td>0</td> <td>1</td> </tr> <tr> <td>A Bdy</td> <td>0</td> <td>F</td> <td>0</td> </tr> <tr> <td>Ext</td> <td>1</td> <td>0</td> <td>2</td> </tr> </tbody> </table>	B	Int	Bdy	Ext	Int	F	0	1	A Bdy	0	F	0	Ext	1	0	2
B	Int	Bdy	Ext															
Int	F	0	1															
A Bdy	0	F	0															
Ext	1	0	2															
L/L		<p>Intersection Matrix</p> <p>AB F0100F1F2</p> <p>BA F0100F1F2</p> <table border="1"> <thead> <tr> <th>B</th> <th>Int</th> <th>Bdy</th> <th>Ext</th> </tr> </thead> <tbody> <tr> <td>Int</td> <td>F</td> <td>0</td> <td>1</td> </tr> <tr> <td>A Bdy</td> <td>0</td> <td>0</td> <td>F</td> </tr> <tr> <td>Ext</td> <td>1</td> <td>F</td> <td>2</td> </tr> </tbody> </table>	B	Int	Bdy	Ext	Int	F	0	1	A Bdy	0	0	F	Ext	1	F	2
B	Int	Bdy	Ext															
Int	F	0	1															
A Bdy	0	0	F															
Ext	1	F	2															
L/S		<p>Intersection Matrix</p> <p>AB F01FF0212</p> <p>BA FF20F1102</p> <table border="1"> <thead> <tr> <th>B</th> <th>Int</th> <th>Bdy</th> <th>Ext</th> </tr> </thead> <tbody> <tr> <td>Int</td> <td>F</td> <td>0</td> <td>1</td> </tr> <tr> <td>A Bdy</td> <td>F</td> <td>F</td> <td>0</td> </tr> <tr> <td>Ext</td> <td>2</td> <td>1</td> <td>2</td> </tr> </tbody> </table>	B	Int	Bdy	Ext	Int	F	0	1	A Bdy	F	F	0	Ext	2	1	2
B	Int	Bdy	Ext															
Int	F	0	1															
A Bdy	F	F	0															
Ext	2	1	2															
L/S		<p>Intersection Matrix</p> <p>AB F11FF0212</p> <p>BA FF21F1102</p> <table border="1"> <thead> <tr> <th>B</th> <th>Int</th> <th>Bdy</th> <th>Ext</th> </tr> </thead> <tbody> <tr> <td>Int</td> <td>F</td> <td>1</td> <td>1</td> </tr> <tr> <td>A Bdy</td> <td>F</td> <td>F</td> <td>0</td> </tr> <tr> <td>Ext</td> <td>2</td> <td>1</td> <td>2</td> </tr> </tbody> </table>	B	Int	Bdy	Ext	Int	F	1	1	A Bdy	F	F	0	Ext	2	1	2
B	Int	Bdy	Ext															
Int	F	1	1															
A Bdy	F	F	0															
Ext	2	1	2															
L/S		<p>Intersection Matrix</p> <p>AB FF1F0F212</p> <p>BA FF2F011F2</p> <table border="1"> <thead> <tr> <th>B</th> <th>Int</th> <th>Bdy</th> <th>Ext</th> </tr> </thead> <tbody> <tr> <td>Int</td> <td>F</td> <td>F</td> <td>1</td> </tr> <tr> <td>A Bdy</td> <td>F</td> <td>0</td> <td>F</td> </tr> <tr> <td>Ext</td> <td>2</td> <td>1</td> <td>2</td> </tr> </tbody> </table>	B	Int	Bdy	Ext	Int	F	F	1	A Bdy	F	0	F	Ext	2	1	2
B	Int	Bdy	Ext															
Int	F	F	1															
A Bdy	F	0	F															
Ext	2	1	2															



Tutorial PostGIS, 3.x

ST_Touches (cont.)

Relación	Diagrama	Matriz																
L/S		<p>Intersection Matrix</p> <p>AB FF1F00212</p> <p>BA FF2F01102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>F</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	F	1	A	<i>Bdy</i>	F	0	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	F	F	1															
A	<i>Bdy</i>	F	0															
<i>Ext</i>	2	1	2															
L/S		<p>Intersection Matrix</p> <p>AB F01F0F212</p> <p>BA FF20011F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	0	1	A	<i>Bdy</i>	F	0	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	F	0	1															
A	<i>Bdy</i>	F	0															
<i>Ext</i>	2	1	2															
L/S		<p>Intersection Matrix</p> <p>AB F01F00212</p> <p>BA FF2001102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	0	1	A	<i>Bdy</i>	F	0	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	F	0	1															
A	<i>Bdy</i>	F	0															
<i>Ext</i>	2	1	2															
L/S		<p>Intersection Matrix</p> <p>AB F1FF0F212</p> <p>BA FF2101FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>1</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	1	F	A	<i>Bdy</i>	F	0	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	F	1	F															
A	<i>Bdy</i>	F	0															
<i>Ext</i>	2	1	2															
L/S		<p>Intersection Matrix</p> <p>AB F11F0F212</p> <p>BA FF21011F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>1</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	1	1	A	<i>Bdy</i>	F	0	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	F	1	1															
A	<i>Bdy</i>	F	0															
<i>Ext</i>	2	1	2															
L/S		<p>Intersection Matrix</p> <p>AB F11F00212</p> <p>BA FF2101102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>1</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	1	1	A	<i>Bdy</i>	F	0	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	F	1	1															
A	<i>Bdy</i>	F	0															
<i>Ext</i>	2	1	2															
S/P		<p>Intersection Matrix</p> <p>AB FF20F1FF2</p> <p>BA F0FFFF212</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	F	2	A	<i>Bdy</i>	0	F	<i>Ext</i>	F	F	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	F	F	2															
A	<i>Bdy</i>	0	F															
<i>Ext</i>	F	F	2															



Tutorial PostGIS, 3.x

ST_Touches (cont.)

Relación	Diagrama	Matriz
S/L		Intersection Matrix AB FF20F1102 BA F01FF0212 B Int Bdy Ext Int F F 2 A Bdy 0 F 1 Ext 1 0 2
S/L		Intersection Matrix AB FF21F1102 BA F11FF0212 B Int Bdy Ext Int F F 2 A Bdy 1 F 1 Ext 1 0 2
S/L		Intersection Matrix AB FF2F011F2 BA FF1F0F212 B Int Bdy Ext Int F F 2 A Bdy F 0 1 Ext 1 F 2
S/L		Intersection Matrix AB FF2F01102 BA FF1F00212 B Int Bdy Ext Int F F 2 A Bdy F 0 1 Ext 1 0 2
S/L		Intersection Matrix AB FF20011F2 BA F01F0F212 B Int Bdy Ext Int F F 2 A Bdy 0 0 1 Ext 1 F 2
S/L		Intersection Matrix AB FF2001102 BA F01F00212 B Int Bdy Ext Int F F 2 A Bdy 0 0 1 Ext 1 0 2
S/L		Intersection Matrix AB FF2101FF2 BA F1FF0F212 B Int Bdy Ext Int F F 2 A Bdy 1 0 1 Ext F F 2



Tutorial PostGIS, 3.x

ST_Touches (cont.)

Relación	Diagrama	Matriz																
S/L		<p>Intersection Matrix</p> <p>AB FF21011F2</p> <p>BA F11F0F212</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>1</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	F	2	A	<i>Bdy</i>	1	0	<i>Ext</i>	1	F	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	F	F	2															
A	<i>Bdy</i>	1	0															
<i>Ext</i>	1	F	2															
S/L		<p>Intersection Matrix</p> <p>AB FF2101102</p> <p>BA F11F00212</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>1</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	F	2	A	<i>Bdy</i>	1	0	<i>Ext</i>	1	0	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	F	F	2															
A	<i>Bdy</i>	1	0															
<i>Ext</i>	1	0	2															
S/S		<p>Intersection Matrix</p> <p>AB FF2F01212</p> <p>BA FF2F01212</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	F	2	A	<i>Bdy</i>	F	0	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	F	F	2															
A	<i>Bdy</i>	F	0															
<i>Ext</i>	2	1	2															
S/S		<p>Intersection Matrix</p> <p>AB FF2F11212</p> <p>BA FF2F11212</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>1</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	F	F	2	A	<i>Bdy</i>	F	1	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	F	F	2															
A	<i>Bdy</i>	F	1															
<i>Ext</i>	2	1	2															



Tutorial PostGIS, 3.x

ST_CROSSES

Citado de https://postgis.net/docs/ST_Crosses.html

Compara dos objetos geométricos y devuelve cierto si su intersección “cruza espacialmente”; esto es:

- que las geometrías tienen uno o más pero no todos sus puntos interiores en común.
- La intersección de los interiores debe ser no-vacío y su dimensión debe ser menor que la dimensión máxima entre las dos geometrías,
- Además, la intersección de las dos geometrías no puede ser igual que cualquiera de las geometrías. Dicho de otro modo, devolverá falso.

La relación “crosses” es simétrica e irreflexiva.

Las geometrías cruzan si la matriz de intersección DE-9IM tiene estos pareos en ST_Relate:

- T*T***** al evaluar P/L, P/S y L/S
- T*****T** al evaluar L/P, S/P y S/L
- 0***** al evaluar L/L

El resultado será falso al evaluar geometrías P/P y S/S.

Esta función utiliza los comparadores de cajas de extensión (bounding boxes) que hacen uso de cualquier índice espacial disponible para las geometrías. Si no desea usar el índice disponible, sustituya por la función `_ST_Crosses`.

Relación	Diagrama	Matriz
L/L		<pre> Intersection Matrix AB 0F1FF0102 BA 0F1FF0102 B Int Bdy Ext Int 0 F 1 A Bdy F F 0 Ext 1 0 2 </pre>
L/L		<pre> Intersection Matrix AB 0F1F0F1F2 BA 0F1F0F1F2 B Int Bdy Ext Int 0 F 1 A Bdy F 0 F Ext 1 F 2 </pre>
L/L		<pre> Intersection Matrix AB 0F1F00102 BA 0F1F00102 B Int Bdy Ext Int 0 F 1 A Bdy F 0 0 Ext 1 0 2 </pre>
L/L		<pre> Intersection Matrix AB 0F10FF102 BA 001FF01F2 B Int Bdy Ext Int 0 F 1 A Bdy 0 F F Ext 1 0 2 </pre>



Tutorial PostGIS, 3.x

ST_Crosses (cont.)

Relación	Diagrama	Matriz																
L/L		<p>Intersection Matrix</p> <p>AB 0F10F0102</p> <p>BA 001FF0102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>F</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	0	F	1	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	0	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	0	F	1															
A	<i>Bdy</i>	0	F															
<i>Ext</i>	1	0	2															
L/L		<p>Intersection Matrix</p> <p>AB 0F10F0102</p> <p>BA 001FF0102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>F</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	0	F	1	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	0	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	0	F	1															
A	<i>Bdy</i>	0	F															
<i>Ext</i>	1	0	2															
L/L		<p>Intersection Matrix</p> <p>AB 0F100F102</p> <p>BA 001F001F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>F</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	0	F	1	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	0	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	0	F	1															
A	<i>Bdy</i>	0	F															
<i>Ext</i>	1	0	2															
L/L		<p>Intersection Matrix</p> <p>AB 001FF01F2</p> <p>BA 0F10FF102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	0	0	1	A	<i>Bdy</i>	F	F	<i>Ext</i>	1	F	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	0	0	1															
A	<i>Bdy</i>	F	F															
<i>Ext</i>	1	F	2															
L/L		<p>Intersection Matrix</p> <p>AB 001FF0102</p> <p>BA 0F10F0102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	0	0	1	A	<i>Bdy</i>	F	F	<i>Ext</i>	1	0	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	0	0	1															
A	<i>Bdy</i>	F	F															
<i>Ext</i>	1	0	2															
L/L		<p>Intersection Matrix</p> <p>AB 001F001F2</p> <p>BA 0F100F102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	0	0	1	A	<i>Bdy</i>	F	0	<i>Ext</i>	1	F	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	0	0	1															
A	<i>Bdy</i>	F	0															
<i>Ext</i>	1	F	2															
L/L		<p>Intersection Matrix</p> <p>AB 0010FF1F2</p> <p>BA 0010FF1F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	0	0	1	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	F	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	0	0	1															
A	<i>Bdy</i>	0	F															
<i>Ext</i>	1	F	2															



Tutorial PostGIS, 3.x

ST_Crosses (cont.)

Relación	Diagrama	Matriz																
L/L		<p>Intersection Matrix</p> <p>AB 0010FF102</p> <p>BA 0010F01F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td></td> <td><i>Int</i></td> <td>0</td> <td>0</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>0</td> <td>F</td> </tr> <tr> <td></td> <td><i>Ext</i></td> <td>1</td> <td>0</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>		<i>Int</i>	0	0	A	<i>Body</i>	0	F		<i>Ext</i>	1	0
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
	<i>Int</i>	0	0															
A	<i>Body</i>	0	F															
	<i>Ext</i>	1	0															
L/L		<p>Intersection Matrix</p> <p>AB 0010F01F2</p> <p>BA 0010FF102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td></td> <td><i>Int</i></td> <td>0</td> <td>0</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>0</td> <td>F</td> </tr> <tr> <td></td> <td><i>Ext</i></td> <td>1</td> <td>F</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>		<i>Int</i>	0	0	A	<i>Body</i>	0	F		<i>Ext</i>	1	F
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
	<i>Int</i>	0	0															
A	<i>Body</i>	0	F															
	<i>Ext</i>	1	F															
L/L		<p>Intersection Matrix</p> <p>AB 0010F0102</p> <p>BA 0010F0102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td></td> <td><i>Int</i></td> <td>0</td> <td>0</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>0</td> <td>F</td> </tr> <tr> <td></td> <td><i>Ext</i></td> <td>1</td> <td>0</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>		<i>Int</i>	0	0	A	<i>Body</i>	0	F		<i>Ext</i>	1	0
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
	<i>Int</i>	0	0															
A	<i>Body</i>	0	F															
	<i>Ext</i>	1	0															
L/L		<p>Intersection Matrix</p> <p>AB 00100F1F2</p> <p>BA 00100F1F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td></td> <td><i>Int</i></td> <td>0</td> <td>0</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>0</td> <td>F</td> </tr> <tr> <td></td> <td><i>Ext</i></td> <td>1</td> <td>F</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>		<i>Int</i>	0	0	A	<i>Body</i>	0	F		<i>Ext</i>	1	F
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
	<i>Int</i>	0	0															
A	<i>Body</i>	0	F															
	<i>Ext</i>	1	F															
L/S		<p>Intersection Matrix</p> <p>AB 101FF0212</p> <p>BA 1F20F1102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td></td> <td><i>Int</i></td> <td>1</td> <td>0</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>F</td> <td>F</td> </tr> <tr> <td></td> <td><i>Ext</i></td> <td>2</td> <td>1</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>		<i>Int</i>	1	0	A	<i>Body</i>	F	F		<i>Ext</i>	2	1
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
	<i>Int</i>	1	0															
A	<i>Body</i>	F	F															
	<i>Ext</i>	2	1															
L/S		<p>Intersection Matrix</p> <p>AB 1010F0212</p> <p>BA 1020F1102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td></td> <td><i>Int</i></td> <td>1</td> <td>0</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>0</td> <td>F</td> </tr> <tr> <td></td> <td><i>Ext</i></td> <td>2</td> <td>1</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>		<i>Int</i>	1	0	A	<i>Body</i>	0	F		<i>Ext</i>	2	1
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
	<i>Int</i>	1	0															
A	<i>Body</i>	0	F															
	<i>Ext</i>	2	1															
L/S		<p>Intersection Matrix</p> <p>AB 1010FF212</p> <p>BA 1020F11F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td></td> <td><i>Int</i></td> <td>1</td> <td>0</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>0</td> <td>F</td> </tr> <tr> <td></td> <td><i>Ext</i></td> <td>2</td> <td>1</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>		<i>Int</i>	1	0	A	<i>Body</i>	0	F		<i>Ext</i>	2	1
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
	<i>Int</i>	1	0															
A	<i>Body</i>	0	F															
	<i>Ext</i>	2	1															



Tutorial PostGIS, 3.x

ST_Crosses (cont.)

Relación	Diagrama	Matriz
L/S		Intersection Matrix AB 111FF0212 BA 1F21F1102 B Int Bdy Ext Int 1 1 1 A Bdy F F 0 Ext 2 1 2
L/S		Intersection Matrix AB 1110FF212 BA 1021F11F2 B Int Bdy Ext Int 1 1 1 A Bdy 0 F F Ext 2 1 2
L/L		Intersection Matrix AB 1110F0212 BA 1021F1102 B Int Bdy Ext Int 1 1 1 A Bdy 0 F 0 Ext 2 1 2
L/S		Intersection Matrix AB 101F0F212 BA 1F20011F2 B Int Bdy Ext Int 1 0 1 A Bdy F 0 F Ext 2 1 2
L/S		Intersection Matrix AB 101F00212 BA 1F2001102 B Int Bdy Ext Int 1 0 1 A Bdy F 0 0 Ext 2 1 2
L/S		Intersection Matrix AB 10100F212 BA 1020011F2 B Int Bdy Ext Int 1 0 1 A Bdy 0 0 F Ext 2 1 2
L/S		Intersection Matrix AB 1010FF212 BA 1020F11F2 B Int Bdy Ext Int 1 0 1 A Bdy 0 F F Ext 2 1 2



Tutorial PostGIS, 3.x

ST_Crosses (cont.)

Relación	Diagrama	Matriz																
L/S		<p>Intersection Matrix</p> <p>AB 111F0F212</p> <p>BA 1F21011F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	1	1	A	<i>Bdy</i>	F	0	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	1	1	1															
A	<i>Bdy</i>	F	0															
<i>Ext</i>	2	1	2															
L/S		<p>Intersection Matrix</p> <p>AB 111F00212</p> <p>BA 1F2101102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	1	1	A	<i>Bdy</i>	F	0	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	1	1	1															
A	<i>Bdy</i>	F	0															
<i>Ext</i>	2	1	2															
L/S		<p>Intersection Matrix</p> <p>AB 11100F212</p> <p>BA 1021011F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	1	1	A	<i>Bdy</i>	0	0	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	1	1	1															
A	<i>Bdy</i>	0	0															
<i>Ext</i>	2	1	2															
S/L		<p>Intersection Matrix</p> <p>AB 1F20F1102</p> <p>BA 101FF0212</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	F	2	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	0	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	1	F	2															
A	<i>Bdy</i>	0	F															
<i>Ext</i>	1	0	2															
S/L		<p>Intersection Matrix</p> <p>AB 1020F1102</p> <p>BA 1010F0212</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	0	2	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	0	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	1	0	2															
A	<i>Bdy</i>	0	F															
<i>Ext</i>	1	0	2															
S/L		<p>Intersection Matrix</p> <p>AB 1020F11F2</p> <p>BA 1010FF212</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	0	2	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	F	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	1	0	2															
A	<i>Bdy</i>	0	F															
<i>Ext</i>	1	F	2															
S/L		<p>Intersection Matrix</p> <p>AB 1020F11F2</p> <p>BA 1010FF212</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Bdy</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> <tr> <td>A</td> <td><i>Bdy</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>	<i>Int</i>	1	0	2	A	<i>Bdy</i>	0	F	<i>Ext</i>	1	F	2
B	<i>Int</i>	<i>Bdy</i>	<i>Ext</i>															
<i>Int</i>	1	0	2															
A	<i>Bdy</i>	0	F															
<i>Ext</i>	1	F	2															



Tutorial PostGIS, 3.x

ST_Crosses (cont.)

Relación	Diagrama	Matriz
S/L		Intersection Matrix AB 1F21F1102 BA 111FF0212 B Int Bdy Ext Int 1 F 2 A Bdy 1 F 1 Ext 1 0 2
S/L		Intersection Matrix AB 1021F11F2 BA 1110FF212 B Int Bdy Ext Int 1 0 2 A Bdy 1 F 1 Ext 1 F 2
S/L		Intersection Matrix AB 1021F1102 BA 1110F0212 B Int Bdy Ext Int 1 0 2 A Bdy 1 F 1 Ext 1 0 2
S/L		Intersection Matrix AB 1F20011F2 BA 101F0F212 B Int Bdy Ext Int 1 F 2 A Bdy 0 0 1 Ext 1 F 2
S/L		Intersection Matrix AB 1F2001102 BA 101F00212 B Int Bdy Ext Int 1 F 2 A Bdy 0 0 1 Ext 1 0 2
S/L		Intersection Matrix AB 1020011F2 BA 10100F212 B Int Bdy Ext Int 1 0 2 A Bdy 0 0 1 Ext 1 F 2
S/L		Intersection Matrix AB 1F21011F2 BA 111F0F212 B Int Bdy Ext Int 1 F 2 A Bdy 1 0 1 Ext 1 F 2



Tutorial PostGIS, 3.x

ST_Crosses (cont.)

Relación	Diagrama	Matriz
S/L		<pre> -Intersection Matrix- AB 1F2101102 BA 111F00212 B Int Bdy Ext Int 1 F 2 A Bdy 1 0 1 Ext 1 0 2 </pre>
S/L		<pre> -Intersection Matrix- AB 1021011F2 BA 11100F212 B Int Bdy Ext Int 1 0 2 A Bdy 1 0 1 Ext 1 F 2 </pre>

ST_WITHIN

Citado de https://postgis.net/docs/ST_Within.html

Devolverá TRUE si la geometría A está dentro de la geometría B. La geometría A está dentro de la geometría B si y solo si:

- Todos los puntos que componen A están dentro (ya sea en el interior o límite/contorno) de B
- Dicho de otro modo, ningún punto de A está en el exterior de B
- Además, los interiores de A y B al menos tienen un punto en común.

Para que esta función tenga sentido, las geometrías que se van a comparar deben estar en el mismo sistema de coordenadas; deben tener el mismo SRID.

La relación within es reflexiva: cada geometría está dentro de ella misma. La relación es antisimétrica: si $ST_Within(A,B) = true$ and $ST_Within(B,A) = true$, entonces las dos geometrías deben ser topológicamente iguales ($ST_Equals(A,B) = true$).

ST_Within es el reverso de $ST_Contains$, por lo tanto, $ST_Within(A,B) = ST_Contains(B,A)$.

Expresión equivalente en matriz DE-9IM para uso con $ST_Relate(geomA,geomB,'patrón')$ [T*F**F***].

Nota:

Ya que los interiores deben tener al menos un punto en común, un detalle de la definición de ST_Within es que las líneas y puntos que estén completamente en el contorno de polígonos o líneas NO están dentro de la geometría de estos polígonos. Para más detalles, refiérase a las [definiciones/sutilezas de Covers, Contains y de Within](#). El predicado [ST_CoveredBy](#) provee una relación más inclusiva.

Esta función utiliza los comparadores de cajas de extensión (bounding boxes) que hacen uso de cualquier índice espacial disponible para las geometrías. Si no desea usar el índice disponible, sustituya por la función $_ST_Within$.



Tutorial PostGIS, 3.x

ST_Within (cont.)

Relación	Diagrama	Matriz																
P/P		<p>Intersection Matrix</p> <p>AB 0FFFFFF2</p> <p>BA 0FFFFFF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	0	F	F	A	<i>Body</i>	F	F	<i>Ext</i>	F	F	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	0	F	F															
A	<i>Body</i>	F	F															
<i>Ext</i>	F	F	2															
P/L		<p>Intersection Matrix</p> <p>AB 0FFFFFF102</p> <p>BA 0F1FF0FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	0	F	F	A	<i>Body</i>	F	F	<i>Ext</i>	1	0	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	0	F	F															
A	<i>Body</i>	F	F															
<i>Ext</i>	1	0	2															
P/S		<p>Intersection Matrix</p> <p>AB 0FFFFFF212</p> <p>BA 0F2FF1FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	0	F	F	A	<i>Body</i>	F	F	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	0	F	F															
A	<i>Body</i>	F	F															
<i>Ext</i>	2	1	2															
L/L		<p>Intersection Matrix</p> <p>AB 1FFF0FFF2</p> <p>BA 1FFF0FFF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	1	F	F	A	<i>Body</i>	F	0	<i>Ext</i>	F	F	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	1	F	F															
A	<i>Body</i>	F	0															
<i>Ext</i>	F	F	2															
L/L		<p>Intersection Matrix</p> <p>AB 1FF0FF102</p> <p>BA 101FF0FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	1	F	F	A	<i>Body</i>	0	F	<i>Ext</i>	1	0	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	1	F	F															
A	<i>Body</i>	0	F															
<i>Ext</i>	1	0	2															
L/L		<p>Intersection Matrix</p> <p>AB 1FF00F102</p> <p>BA 101F00FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>0</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	1	F	F	A	<i>Body</i>	0	0	<i>Ext</i>	1	0	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	1	F	F															
A	<i>Body</i>	0	0															
<i>Ext</i>	1	0	2															
L/S		<p>Intersection Matrix</p> <p>AB 1FF0FF212</p> <p>BA 102FF1FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	1	F	F	A	<i>Body</i>	0	F	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	1	F	F															
A	<i>Body</i>	0	F															
<i>Ext</i>	2	1	2															



Tutorial PostGIS, 3.x

ST_Within (cont.)

Relación	Diagrama	Matriz
L/S		<p>Intersection Matrix</p> <p>AB 10F0FF212</p> <p>BA 1020F1FF2</p> <p>B Int Bdy Ext</p> <p>Int 1 0 F</p> <p>A Bdy 0 F F</p> <p>Ext 2 1 2</p>
L/S		<p>Intersection Matrix</p> <p>AB 11F0FF212</p> <p>BA 1021F1FF2</p> <p>B Int Bdy Ext</p> <p>Int 1 1 F</p> <p>A Bdy 0 F F</p> <p>Ext 2 1 2</p>
L/S		<p>Intersection Matrix</p> <p>AB 1FFF0F212</p> <p>BA 1F2F01FF2</p> <p>B Int Bdy Ext</p> <p>Int 1 F F</p> <p>A Bdy F 0 F</p> <p>Ext 2 1 2</p>
L/S		<p>Intersection Matrix</p> <p>AB 1FF00F212</p> <p>BA 102F01FF2</p> <p>B Int Bdy Ext</p> <p>Int 1 F F</p> <p>A Bdy 0 0 F</p> <p>Ext 2 1 2</p>
L/S		<p>Intersection Matrix</p> <p>AB 1FF00F212</p> <p>BA 102F01FF2</p> <p>B Int Bdy Ext</p> <p>Int 1 F F</p> <p>A Bdy 0 0 F</p> <p>Ext 2 1 2</p>
L/L		<p>Intersection Matrix</p> <p>AB 10FF0F212</p> <p>BA 1F2001FF2</p> <p>B Int Bdy Ext</p> <p>Int 1 0 F</p> <p>A Bdy F 0 F</p> <p>Ext 2 1 2</p>
L/S		<p>Intersection Matrix</p> <p>AB 10F00F212</p> <p>BA 102001FF2</p> <p>B Int Bdy Ext</p> <p>Int 1 0 F</p> <p>A Bdy 0 0 F</p> <p>Ext 2 1 2</p>



Tutorial PostGIS, 3.x

ST_Within (cont.)

Relación	Diagrama	Matriz																
L/S		<p>Intersection Matrix</p> <p>AB 11FF0F212</p> <p>BA 1F2101FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>1</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	1	1	F	A	<i>Body</i>	F	0	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	1	1	F															
A	<i>Body</i>	F	0															
<i>Ext</i>	2	1	2															
L/S		<p>Intersection Matrix</p> <p>AB 11F00F212</p> <p>BA 102101FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>1</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>0</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	1	1	F	A	<i>Body</i>	0	0	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	1	1	F															
A	<i>Body</i>	0	0															
<i>Ext</i>	2	1	2															
S/S		<p>Intersection Matrix</p> <p>AB 2FFF1FFF2</p> <p>BA 2FFF1FFF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>2</td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>F</td> <td>1</td> </tr> <tr> <td><i>Ext</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	2	F	F	A	<i>Body</i>	F	1	<i>Ext</i>	F	F	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	2	F	F															
A	<i>Body</i>	F	1															
<i>Ext</i>	F	F	2															
S/S		<p>Intersection Matrix</p> <p>AB 2FF1FF212</p> <p>BA 212FF1FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>2</td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>1</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	2	F	F	A	<i>Body</i>	1	F	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	2	F	F															
A	<i>Body</i>	1	F															
<i>Ext</i>	2	1	2															
S/S		<p>Intersection Matrix</p> <p>AB 2FF11F212</p> <p>BA 212F11FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>2</td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>1</td> <td>1</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	2	F	F	A	<i>Body</i>	1	1	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	2	F	F															
A	<i>Body</i>	1	1															
<i>Ext</i>	2	1	2															
S/S		<p>Intersection Matrix</p> <p>AB 2FF10F212</p> <p>BA 212F01FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>2</td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>1</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	2	F	F	A	<i>Body</i>	1	0	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	2	F	F															
A	<i>Body</i>	1	0															
<i>Ext</i>	2	1	2															



Tutorial PostGIS, 3.x

ST_CONTAINS

Citado de https://postgis.net/docs/ST_Contains.html

Devuelve TRUE si la geometría A contiene a la geometría B. La geometría A contiene a B si y solo si:

- Todos los puntos de B están dentro (en el interior o límite de) A o igualmente, ningún punto de B está en el exterior de A)
- Y que los interiores de A y B tengan al menos un punto en común.

La relación Contains es reflexiva: cada geometría se contiene a sí misma. En contraste, en el predicado ST_ContainsProperly la geometría no está contenida propiamente en ella. Esta relación es antisimétrica: si ST_Contains(A,B) = true y ST_Contains(B,A) = true, entonces las dos geometrías deben ser topológicamente iguales (ST_Equals(A,B) = true).

ST_Contains es el inverso de ST_Within. Entonces, ST_Contains(A,B) = ST_Within(B,A).

Expresión equivalente en matriz DE-9IM para uso con ST_Relate(geomA,geomB,'patrón') [T*T***T**].

Nota:

Debido a que los interiores deben tener un punto en común, una de las sutilezas de esta definición es que tanto polígonos como líneas no contienen líneas y puntos completamente en su contorno. Para más detalles, refiérase a las [definiciones/sutilezas de Covers, Contains y de Within](#). El predicado [ST_Covers](#) provee una relación más inclusiva.

Esta función utiliza los comparadores de cajas de extensión (bounding boxes) que hacen uso de cualquier índice espacial disponible para las geometrías. Si no desea usar el índice disponible, sustituya por la función `_ST_Contains`.

Relación	Diagrama	Matriz
P/P		<pre> Intersection Matrix: AB 0FFFFFF2 BA 0FFFFFF2 B Int Body Ext Int 0 F F A Body F F F Ext F F 2 </pre>
L/P		<pre> Intersection Matrix: AB 0F1FF0FF2 BA 0FFFFFF102 B Int Body Ext Int 0 F 1 A Body F F 0 Ext F F 2 </pre>
L/L		<pre> Intersection Matrix: AB 1FFF0FFF2 BA 1FFF0FFF2 B Int Body Ext Int 1 F F A Body F 0 F Ext F F 2 </pre>



Tutorial PostGIS, 3.x

ST_Contains (cont.)

Relación	Diagrama	Matriz																
L/L		<p>Intersection Matrix</p> <p>AB 101FF0FF2</p> <p>BA 1FF0FF102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	1	0	1	A	<i>Body</i>	F	F	<i>Ext</i>	F	F	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	1	0	1															
A	<i>Body</i>	F	F															
<i>Ext</i>	F	F	2															
L/L		<p>Intersection Matrix</p> <p>AB 101F00FF2</p> <p>BA 1FF00F102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	1	0	1	A	<i>Body</i>	F	0	<i>Ext</i>	F	F	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	1	0	1															
A	<i>Body</i>	F	0															
<i>Ext</i>	F	F	2															
S/P		<p>Intersection Matrix</p> <p>AB 0F2FF1FF2</p> <p>BA 0FFFFFF212</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>F</td> <td>2</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	0	F	2	A	<i>Body</i>	F	F	<i>Ext</i>	F	F	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	0	F	2															
A	<i>Body</i>	F	F															
<i>Ext</i>	F	F	2															
S/S		<p>Intersection Matrix</p> <p>AB 2FFF1FFF2</p> <p>BA 2FFF1FFF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>2</td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>F</td> <td>1</td> </tr> <tr> <td><i>Ext</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	2	F	F	A	<i>Body</i>	F	1	<i>Ext</i>	F	F	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	2	F	F															
A	<i>Body</i>	F	1															
<i>Ext</i>	F	F	2															
S/S		<p>Intersection Matrix</p> <p>AB 212FF1FF2</p> <p>BA 2FF1FF212</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	2	1	2	A	<i>Body</i>	F	F	<i>Ext</i>	F	F	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	2	1	2															
A	<i>Body</i>	F	F															
<i>Ext</i>	F	F	2															
S/S		<p>Intersection Matrix</p> <p>AB 212F01FF2</p> <p>BA 2FF10F212</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	2	1	2	A	<i>Body</i>	F	0	<i>Ext</i>	F	F	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	2	1	2															
A	<i>Body</i>	F	0															
<i>Ext</i>	F	F	2															
S/S		<p>Intersection Matrix</p> <p>AB 212F11FF2</p> <p>BA 2FF11F212</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>F</td> <td>1</td> </tr> <tr> <td><i>Ext</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	2	1	2	A	<i>Body</i>	F	1	<i>Ext</i>	F	F	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	2	1	2															
A	<i>Body</i>	F	1															
<i>Ext</i>	F	F	2															



Tutorial PostGIS, 3.x

ST_OVERLAPS

Citado de https://postgis.net/docs/ST_Overlaps.html

Devuelve TRUE si la geometría A y la B se solapan espacialmente. Dos geometrías se solapan si:

- Tienen la misma dimensión (P/P, L/L, S/S)
- Sus interiores se deben intersectar en esa dimensión.
- Cada geometría tiene al menos un punto dentro de la otra geometría, dicho de otra manera, ninguna debe cubrir (Covers) la otra.

La relación Overlaps es simétrica e irreflexiva.

Esta función incluye de manera automática el uso de cajas de extensión (bounding boxes), lo cual hará uso de cualquier índice espacial existente. Si no desea usar índices, use la función `_ST_Overlaps`.

Expresión en matrices DE-9IM para uso con `ST_Relate(geomA,geomB,'patrón')`

[T*T***T**] (P/P, S/S)

[1*T***T**] (L/L)

Relación	Diagrama	Matriz
P/P		<pre> Intersection Matrix AB 0FFFFFFF2 BA 0FFFFFFF2 B Int Bdy Ext Int 0 F F A Bdy F F F Ext F F 2 </pre>
L/L		<pre> Intersection Matrix AB 1F1FF0102 BA 1F1FF0102 B Int Bdy Ext Int 1 F 1 A Bdy F F 0 Ext 1 0 2 </pre>
L/L		<pre> Intersection Matrix AB 1F1F0F1F2 BA 1F1F0F1F2 B Int Bdy Ext Int 1 F 1 A Bdy F 0 F Ext 1 F 2 </pre>
L/L		<pre> Intersection Matrix AB 1F1F00102 BA 1F1F00102 B Int Bdy Ext Int 1 F 1 A Bdy F 0 0 Ext 1 0 2 </pre>



Tutorial PostGIS, 3.x

ST_Overlaps (cont.)

Relación	Diagrama	Matriz																
L/L		<p>Intersection Matrix</p> <p>AB 1F100F102</p> <p>BA 101F001F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>F</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>0</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	1	F	1	A	<i>Body</i>	0	0	<i>Ext</i>	1	0	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	1	F	1															
A	<i>Body</i>	0	0															
<i>Ext</i>	1	0	2															
L/L		<p>Intersection Matrix</p> <p>AB 1F10F0102</p> <p>BA 101FF0102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>F</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	1	F	1	A	<i>Body</i>	0	F	<i>Ext</i>	1	0	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	1	F	1															
A	<i>Body</i>	0	F															
<i>Ext</i>	1	0	2															
L/L		<p>Intersection Matrix</p> <p>AB 101FF01F2</p> <p>BA 1F10FF102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	1	0	1	A	<i>Body</i>	F	F	<i>Ext</i>	1	F	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	1	0	1															
A	<i>Body</i>	F	F															
<i>Ext</i>	1	F	2															
L/L		<p>Intersection Matrix</p> <p>AB 101FF0102</p> <p>BA 1F10F0102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	1	0	1	A	<i>Body</i>	F	F	<i>Ext</i>	1	0	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	1	0	1															
A	<i>Body</i>	F	F															
<i>Ext</i>	1	0	2															
L/L		<p>Intersection Matrix</p> <p>AB 101F001F2</p> <p>BA 1F100F102</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	1	0	1	A	<i>Body</i>	F	0	<i>Ext</i>	1	F	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	1	0	1															
A	<i>Body</i>	F	0															
<i>Ext</i>	1	F	2															
L/L		<p>Intersection Matrix</p> <p>AB 1010FF1F2</p> <p>BA 1010FF1F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	1	0	1	A	<i>Body</i>	0	F	<i>Ext</i>	1	F	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	1	0	1															
A	<i>Body</i>	0	F															
<i>Ext</i>	1	F	2															
L/L		<p>Intersection Matrix</p> <p>AB 1010FF102</p> <p>BA 1010F01F2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	1	0	1	A	<i>Body</i>	0	F	<i>Ext</i>	1	0	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	1	0	1															
A	<i>Body</i>	0	F															
<i>Ext</i>	1	0	2															



Tutorial PostGIS, 3.x

ST_Overlaps (cont.)

Relación	Diagrama	Matriz
L/L		<pre> -Intersection Matrix- AB 1010F01F2 BA 1010FF102 B Int Bdy Ext Int 1 0 1 A Bdy 0 F 0 Ext 1 F 2 </pre>
L/L		<pre> -Intersection Matrix- AB 1010F0102 BA 1010F0102 B Int Bdy Ext Int 1 0 1 A Bdy 0 F 0 Ext 1 0 2 </pre>
L/L		<pre> -Intersection Matrix- AB 10100F1F2 BA 10100F1F2 B Int Bdy Ext Int 1 0 1 A Bdy 0 0 F Ext 1 F 2 </pre>
S/S		<pre> -Intersection Matrix- AB 212111212 BA 212111212 B Int Bdy Ext Int 2 1 2 A Bdy 1 1 1 Ext 2 1 2 </pre>
S/S		<pre> -Intersection Matrix- AB 212101212 BA 212101212 B Int Bdy Ext Int 2 1 2 A Bdy 1 0 1 Ext 2 1 2 </pre>



ST_INTERSECTS

El predicado ST_Intersects aplica a todas las relaciones, con excepción de ST_Disjoint, ya que que las geometrías tendrían al menos un punto en común, ya sea en el interior o el límite/contorno.

Ejemplos de patrones en matrices DE-9IM para uso con ST_Relate(geomA,geomB,'patrón')

InteriorA/InteriorB	[T*****]
InteriorA/ContornoB	*T*****]
InteriorB/InteriorA	***T*****]
InteriorB/ContornoA	****T*****]

Esta función incluye de manera automática el uso de cajas de extensión (bounding boxes), lo cual hará uso de cualquier índice espacial existente.

OPERADOR &&

Citado de https://postgis.net/docs/geometry_overlaps.html

El operador de **doble ampersand &&** ([en español literalmente: "y por sí mismo"](#)), devuelve TRUE si la caja de extensión (bounding box) 2D de la geometría A interseca con la caja de extensión 2D de la geometría B.

Este operador hace uso de índices existentes en las geometrías. Úselo para combinarlo con ST_Relate(geomA,geomB,'patrón'), ya que ST_Relate no usa índices automáticamente.

Es preferible usar relaciones con nombre, en vez de ST_Relate debido al hecho que esos utilizan el comparador de cajas de extensión o índices de geometrías, además de mejoras en desempeño que no están en ST_Relate. Solo debe usarse ST_Relate para casos especiales

ST_COVERS

Citado de https://postgis.net/docs/ST_Covers.html

**** NO es un estándar del OGC ****

Devuelve TRUE si cada punto en la geometría/geografía B está dentro del interior o el contorno de la geometría/geografía de A. De forma equivalente, prueba que ningún punto de B está fuera o en el exterior de A.

ST_Covers es el reverso de ST_CoveredBy, por lo tanto, **ST_Covers(A,B) = ST_CoveredBy(B,A)**.

Por lo general, la función ST_Covers debe ser usada en lugar de ST_Contains, ya que su definición no tiene la restricción "las geometrías no contienen su contorno".

Esta función incluye de manera automática el uso de cajas de extensión (bounding boxes), lo cual hará uso de cualquier índice espacial existente. Si no desea usar índices, use la función ST_Covers.

Ejemplos de patrones en matrices DE-9IM para uso con ST_Relate(geomA,geomB,'patrón')

InteriorA/InteriorB	[T*****FF*]
InteriorA/ContornoB	*T*****FF*]
InteriorB/InteriorA	***T*****FF*]
InteriorB/ContornoA	****T*****FF*]



Tutorial PostGIS, 3.x

ST_Covers (cont.)

Relación	Diagrama	Matriz
P/P		<p>Intersection Matrix</p> <p>AB 0FFFFFF2</p> <p>BA 0FFFFFF2</p> <p>B Int Body Ext</p> <p>Int 0 F F</p> <p>A Body F F F</p> <p>Ext F F 2</p>
L/P		<p>Intersection Matrix</p> <p>AB FF10FOFF2</p> <p>BA F0FFFF102</p> <p>B Int Body Ext</p> <p>Int F F 1</p> <p>A Body 0 F 0</p> <p>Ext F F 2</p>
L/P		<p>Intersection Matrix</p> <p>AB 0F1FF0FF2</p> <p>BA 0FFFFFF102</p> <p>B Int Body Ext</p> <p>Int 0 F 1</p> <p>A Body F F 0</p> <p>Ext F F 2</p>
L/L		<p>Intersection Matrix</p> <p>AB 1FFF0FFF2</p> <p>BA 1FFF0FFF2</p> <p>B Int Body Ext</p> <p>Int 1 F F</p> <p>A Body F 0 F</p> <p>Ext F F 2</p>
L/L		<p>Intersection Matrix</p> <p>AB 101FF0FF2</p> <p>BA 1FF0FF102</p> <p>B Int Body Ext</p> <p>Int 1 0 1</p> <p>A Body F F 0</p> <p>Ext F F 2</p>
L/L		<p>Intersection Matrix</p> <p>AB 101F00FF2</p> <p>BA 1FF00F102</p> <p>B Int Body Ext</p> <p>Int 1 0 1</p> <p>A Body F 0 0</p> <p>Ext F F 2</p>
S/P		<p>Intersection Matrix</p> <p>AB FF20F1FF2</p> <p>BA F0FFFF212</p> <p>B Int Body Ext</p> <p>Int F F 2</p> <p>A Body 0 F 1</p> <p>Ext F F 2</p>



Tutorial PostGIS, 3.x

ST_Covers (cont.)

Relación	Diagrama	Matriz
S/P		<p>Intersection Matrix</p> <p>AB 0F2FF1FF2</p> <p>BA 0FFFFFF212</p> <p>B Int Body Ext</p> <p>Int 0 F 2</p> <p>A Body F F 1</p> <p>Ext F F 2</p>
S/L		<p>Intersection Matrix</p> <p>AB 102FF1FF2</p> <p>BA 1FF0FF212</p> <p>B Int Body Ext</p> <p>Int 1 0 2</p> <p>A Body F F 1</p> <p>Ext F F 2</p>
S/L		<p>Intersection Matrix</p> <p>AB 1020F1FF2</p> <p>BA 10F0FF212</p> <p>B Int Body Ext</p> <p>Int 1 0 2</p> <p>A Body 0 F 1</p> <p>Ext F F 2</p>
S/L		<p>Intersection Matrix</p> <p>AB 1021F1FF2</p> <p>BA 11F0FF212</p> <p>B Int Body Ext</p> <p>Int 1 0 2</p> <p>A Body 1 F 1</p> <p>Ext F F 2</p>
S/L		<p>Intersection Matrix</p> <p>AB 1F2F01FF2</p> <p>BA 1FFF0F212</p> <p>B Int Body Ext</p> <p>Int 1 F 2</p> <p>A Body F 0 1</p> <p>Ext F F 2</p>
S/L		<p>Intersection Matrix</p> <p>AB 102F01FF2</p> <p>BA 1FF00F212</p> <p>B Int Body Ext</p> <p>Int 1 0 2</p> <p>A Body F 0 1</p> <p>Ext F F 2</p>
S/L		<p>Intersection Matrix</p> <p>AB 1F2001FF2</p> <p>BA 10FF0F212</p> <p>B Int Body Ext</p> <p>Int 1 F 2</p> <p>A Body 0 0 1</p> <p>Ext F F 2</p>



Tutorial PostGIS, 3.x

ST_Covers (cont.)

Relación	Diagrama	Matriz
S/L		<p>Intersection Matrix</p> <p>AB 102001FF2</p> <p>BA 10F00F212</p> <p>B Int Body Ext</p> <p>Int 1 0 2</p> <p>A Body 0 0 1</p> <p>Ext F F 2</p>
S/L		<p>Intersection Matrix</p> <p>AB FF2101FF2</p> <p>BA F1FF0F212</p> <p>B Int Body Ext</p> <p>Int F F 2</p> <p>A Body 1 0 1</p> <p>Ext F F 2</p>
S/L		<p>Intersection Matrix</p> <p>AB 1F2101FF2</p> <p>BA 11FF0F212</p> <p>B Int Body Ext</p> <p>Int 1 F 2</p> <p>A Body 1 0 1</p> <p>Ext F F 2</p>
S/S		<p>Intersection Matrix</p> <p>AB 2FFF1FFF2</p> <p>BA 2FFF1FFF2</p> <p>B Int Body Ext</p> <p>Int 2 F F</p> <p>A Body F 1 F</p> <p>Ext F F 2</p>
S/S		<p>Intersection Matrix</p> <p>AB 212FF1FF2</p> <p>BA 2FF1FF212</p> <p>B Int Body Ext</p> <p>Int 2 1 2</p> <p>A Body F F 1</p> <p>Ext F F 2</p>
S/S		<p>Intersection Matrix</p> <p>AB 212F01FF2</p> <p>BA 2FF10F212</p> <p>B Int Body Ext</p> <p>Int 2 1 2</p> <p>A Body F 0 1</p> <p>Ext F F 2</p>
S/S		<p>Intersection Matrix</p> <p>AB 212F11FF2</p> <p>BA 2FF11F212</p> <p>B Int Body Ext</p> <p>Int 2 1 2</p> <p>A Body F 1 1</p> <p>Ext F F 2</p>



Tutorial PostGIS, 3.x

ST_COVEREDBY

Citado de https://postgis.net/docs/ST_CoveredBy.html

** NO es un estándar del OGC **

Devuelve TRUE si cada punto en la geometría/geografía A está dentro del interior o el contorno de la geometría/geografía de B. De forma equivalente, prueba que ningún punto de A está fuera o en el exterior de B.

ST_CoveredBy es el reverso de ST_Covers, por lo tanto, **ST_CoveredBy(A,B) = ST_Covers(B,A).**

Por lo general, la función ST_CoveredBy debe ser usada en lugar de ST_Within, ya que su definición no tiene la restricción "las geometrías no contienen su contorno".

Esta función incluye de manera automática el uso de cajas de extensión (bounding boxes), lo cual hará uso de cualquier índice espacial existente. Si no desea usar índices, use la función ST_CoveredBy.

Ejemplos de patrones en matrices DE-9IM para uso con ST_Relate(geomA,geomB,'patrón')

InteriorA/InteriorB	[T*F**F***]
InteriorA/ContornoB	[*TF**F***]
InteriorB/InteriorA	[**FT*F***]
InteriorB/ContornoA	[**F*TF***]

Relación	Diagrama	Matriz
P/P		<pre> Intersection Matrix AB OFFFFFFF2 BA OFFFFFFF2 B Int Body Ext Int 0 F F A Body F F F Ext F F 2 </pre>
P/L		<pre> Intersection Matrix AB FF10FOFF2 BA FOFFFF102 B Int Body Ext Int F F 1 A Body 0 F 0 Ext F F 2 </pre>
P/L		<pre> Intersection Matrix AB OFFFFF102 BA OF1FF0FF2 B Int Body Ext Int 0 F F A Body F F F Ext 1 0 2 </pre>



Tutorial PostGIS, 3.x

ST_CoveredBy (cont.)

Relación	Diagrama	Matriz																
P/S		<p>Intersection Matrix</p> <p>AB F0FFFF212</p> <p>BA FF20F1FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>F</td> <td>0</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	F	0	F	A	<i>Body</i>	F	F	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	F	0	F															
A	<i>Body</i>	F	F															
<i>Ext</i>	2	1	2															
P/S		<p>Intersection Matrix</p> <p>AB 0FFFFFF212</p> <p>BA 0F2FF1FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>0</td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>F</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	0	F	F	A	<i>Body</i>	F	F	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	0	F	F															
A	<i>Body</i>	F	F															
<i>Ext</i>	2	1	2															
L/L		<p>Intersection Matrix</p> <p>AB 1FFF0FFF2</p> <p>BA 1FFF0FFF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>F</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>F</td> <td>F</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	1	F	F	A	<i>Body</i>	F	0	<i>Ext</i>	F	F	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	1	F	F															
A	<i>Body</i>	F	0															
<i>Ext</i>	F	F	2															
L/L		<p>Intersection Matrix</p> <p>AB 1FF0FF102</p> <p>BA 101FF0FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	1	F	F	A	<i>Body</i>	0	F	<i>Ext</i>	1	0	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	1	F	F															
A	<i>Body</i>	0	F															
<i>Ext</i>	1	0	2															
L/L		<p>Intersection Matrix</p> <p>AB 1FF00F102</p> <p>BA 101F00FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>0</td> <td>0</td> </tr> <tr> <td><i>Ext</i></td> <td>1</td> <td>0</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	1	F	F	A	<i>Body</i>	0	0	<i>Ext</i>	1	0	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	1	F	F															
A	<i>Body</i>	0	0															
<i>Ext</i>	1	0	2															
L/S		<p>Intersection Matrix</p> <p>AB 1FF0FF212</p> <p>BA 102FF1FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>F</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	1	F	F	A	<i>Body</i>	0	F	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	1	F	F															
A	<i>Body</i>	0	F															
<i>Ext</i>	2	1	2															
L/S		<p>Intersection Matrix</p> <p>AB 10F0FF212</p> <p>BA 1020F1FF2</p> <table border="1"> <tr> <td>B</td> <td><i>Int</i></td> <td><i>Body</i></td> <td><i>Ext</i></td> </tr> <tr> <td><i>Int</i></td> <td>1</td> <td>0</td> <td>F</td> </tr> <tr> <td>A</td> <td><i>Body</i></td> <td>0</td> <td>F</td> </tr> <tr> <td><i>Ext</i></td> <td>2</td> <td>1</td> <td>2</td> </tr> </table>	B	<i>Int</i>	<i>Body</i>	<i>Ext</i>	<i>Int</i>	1	0	F	A	<i>Body</i>	0	F	<i>Ext</i>	2	1	2
B	<i>Int</i>	<i>Body</i>	<i>Ext</i>															
<i>Int</i>	1	0	F															
A	<i>Body</i>	0	F															
<i>Ext</i>	2	1	2															



Tutorial PostGIS, 3.x

ST_CoveredBy (cont.)

Relación	Diagrama	Matriz
L/S		<p>Intersection Matrix</p> <p>AB 11F0FF212</p> <p>BA 1021F1FF2</p> <p>B Int Body Ext</p> <p>Int 1 1 F</p> <p>A Body 0 F F</p> <p>Ext 2 1 2</p>
L/S		<p>Intersection Matrix</p> <p>AB 1FFF0F212</p> <p>BA 1F2F01FF2</p> <p>B Int Body Ext</p> <p>Int 1 F F</p> <p>A Body F 0 F</p> <p>Ext 2 1 2</p>
L/S		<p>Intersection Matrix</p> <p>AB 1FF00F212</p> <p>BA 102F01FF2</p> <p>B Int Body Ext</p> <p>Int 1 F F</p> <p>A Body 0 0 F</p> <p>Ext 2 1 2</p>
L/S		<p>Intersection Matrix</p> <p>AB 10FF0F212</p> <p>BA 1F2001FF2</p> <p>B Int Body Ext</p> <p>Int 1 0 F</p> <p>A Body F 0 F</p> <p>Ext 2 1 2</p>
L/S		<p>Intersection Matrix</p> <p>AB 10F00F212</p> <p>BA 102001FF2</p> <p>B Int Body Ext</p> <p>Int 1 0 F</p> <p>A Body 0 0 F</p> <p>Ext 2 1 2</p>
L/S		<p>Intersection Matrix</p> <p>AB 1FF0FF212</p> <p>BA 102FF1FF2</p> <p>B Int Body Ext</p> <p>Int 1 F F</p> <p>A Body 0 F F</p> <p>Ext 2 1 2</p>
L/S		<p>Intersection Matrix</p> <p>AB 10F0FF212</p> <p>BA 1020F1FF2</p> <p>B Int Body Ext</p> <p>Int 1 0 F</p> <p>A Body 0 F F</p> <p>Ext 2 1 2</p>



Tutorial PostGIS, 3.x

ST_CoveredBy (cont.)

Relación	Diagrama	Matriz
L/S		<p>Intersection Matrix</p> <p>AB 11F00F212</p> <p>BA 102101FF2</p> <p>B Int Body Ext</p> <p>Int 1 1 F</p> <p>A Body 0 0 F</p> <p>Ext 2 1 2</p>
S/S		<p>Intersection Matrix</p> <p>AB 2FFF1FFF2</p> <p>BA 2FFF1FFF2</p> <p>B Int Body Ext</p> <p>Int 2 F F</p> <p>A Body F 1 F</p> <p>Ext F F 2</p>
S/S		<p>Intersection Matrix</p> <p>AB 2FF1FF212</p> <p>BA 212FF1FF2</p> <p>B Int Body Ext</p> <p>Int 2 F F</p> <p>A Body 1 F F</p> <p>Ext 2 1 2</p>
S/S		<p>Intersection Matrix</p> <p>AB 2FF11F212</p> <p>BA 212F11FF2</p> <p>B Int Body Ext</p> <p>Int 2 F F</p> <p>A Body 1 1 F</p> <p>Ext 2 1 2</p>
S/S		<p>Intersection Matrix</p> <p>AB 2FF10F212</p> <p>BA 212F01FF2</p> <p>B Int Body Ext</p> <p>Int 2 F F</p> <p>A Body 1 0 F</p> <p>Ext 2 1 2</p>



Referencias

Interpolación areal

Flowerdew, R., & Green, M. (1994). Areal interpolation and types of data. *Spatial analysis and GIS*, 121, 145.

Documentación de PostGIS

"PostGIS Documentation", *PostGIS*, fecha de consulta 10 abril 2024, en <https://postgis.net/>.

PROTEAU, J. C., *Matrices de Clementini et Prédicats spatiaux de l'OGC*.

Llario, J. C. M. (2020). PostGIS: análisis espacial avanzado. Juan Carlos Martínez Llario.