

Kapitel 5

PostGIS räumliche Funktionen und Operatoren

Stefan Keller

Dank an Dr. Andreas Neumann

Überblick räumliche Funktionen/Operatoren

- ◆ Data Management Functions (Datenmanagement)
 - ◆ Geometry Constructors (Geometrie erzeugen)
 - ◆ Geometry Accessors (Zugriffsfunktionen)
 - ◆ Geometry Editors (Editierfunktionen)
 - ◆ Geometry Relationship Functions (Beziehungen)
 - ◆ Geometry Processing Functions (Berechnungen)
 - ◆ Measurement Functions (Messfunktionen)
-
- ◆ Linear Referencing (Berechnungen entlang einer Linie)
 - ◆ SQL-MM spezifische Funktionen
 - ◆ ArcSDE spezifische Funktionen

Räumliche Operatoren

Ansicht in pgAdmin III (Schema public)

The screenshot shows the pgAdmin III interface. The left pane displays the 'Objektbrowser' (Object Browser) for the 'public' schema. The right pane shows the 'Eigenschaften' (Properties) tab for the 'abfall_umwelt' table.

Objektbrowser (public schema):

- Aggregate (19)
- Collations (0)
- Konversionen (0)
- Domänen (1)
- Volltextsuche - Konfigurationen (0)
- Volltextsuche - Wörterbücher (0)
- Volltextsuche - Parser (0)
- Volltextsuche - Vorlagen (0)
- Funktionen (846)
- Operatoren (75)
 - && (geometry, geometry)
 - && (geography, geography)
 - &&& (geometry, geometry)
 - &< (geometry, geometry)
 - &<| (geometry, geometry)
 - &> (geometry, geometry)
 - < (ltree, ltree)
 - < (geography, geography)
 - < (geometry, geometry)
 - <#> (geometry, geometry)
 - <-> (geometry, geometry)
 - << (geometry, geometry)
 - <<| (geometry, geometry)
 - <= (geography, geography)
 - <= (ltree, ltree)
 - <= (geometry, geometry)
 - <> (ltree, ltree)
 - <@ (ltree, _ltree)
 - <@ (_ltree, ltree)
 - <@ (ltree, ltree)
 - = (ltree, ltree)
 - = (geometry, geometry)
 - = (geography, geography)
 - > (geometry, geometry)

Eigenschaften (abfall_umwelt):

Eigenschaft	Wert
Name	abfall_umwelt
OID	678491
Eigentümer	an
ACL	
Standard Tabellen ACL	
Standard Sequence ACL	
Standard Funktions-ACL	
System-Schema?	Nein
Kommentar	

SQL-Feld:

```
-- Schema: abfall_umwelt
-- DROP SCHEMA abfall_umwelt;

CREATE SCHEMA abfall_umwelt
AUTHORIZATION an;
```

Retrieving details on schema public... Fertig. 0.45 Sek.

Räumliche Funktionen

Ansicht in pgAdmin4 (Schema public)

The screenshot shows the pgAdmin4 web interface. On the left, the 'public' schema is expanded, showing various objects including Functions (1225). The main panel displays the 'Query Editor' with the following SQL query:

```
1 SELECT * FROM public.geometry_columns
2
```

Below the query editor, the 'Data Output' tab is active, showing the results of the query. The results are displayed in a table with the following columns:

f_table_catalog	f_table_schema	f_table_name	f_geometry_column	coord_dimension	srid
character varying (256)	name	name	name	integer	integer

Beispiele für Geometry Constructors (Geometrien erzeugen)

- ◆ **ST_GeomFromText** – Geometrie aus WKT Format erzeugen
- ◆ **ST_GeographyFromText** – Geographie aus WKT Format erzeugen
- ◆ **ST_MakePoint** – Punkt erzeugen
- ◆ **ST_MakeLine** – Linie aus Einzelpunkten erzeugen
- ◆ **ST_MakeBox2D** – 2D Box erzeugen

- ◆ **ST_3DMakeBox** – 3D Box erzeugen
- ◆ **ST_GeomFromGML** – Geometrie von GML erzeugen
- ◆ **ST_GeomFromKML** – Geometrie von KML erzeugen

Beispiele für Geometry Accessors (Zugriffsfunktionen)

- ◆ **ST_GeometryType(geometry)** – z.B. POINT, LINESTRING, ...
- ◆ **ST_Npoints(geometry)** – Anzahl Stützpunkte
- ◆ **ST_StartPoint(geometry)** – erster Stützpunkt
- ◆ **ST_EndPoint(geometry)** – letzter Stützpunkt
- ◆ **ST_PointN(geometry)** – nter Stützpunkt
- ◆ **ST_NumGeometries(geometry)** – Anzahl Geometrien in Multi/GeomCollection
- ◆ **ST_Nrings(geometry)** – Anzahl Ringe (Löcher und Exklaven)
- ◆ **ST_Xmax/ST_Xmin/ST_Ymax/ST_Ymin** – Bounding box Werte
- ◆ **ST_SRID(geometry)** – gibt Nr Koordinatensystem zurück
- ◆ **ST_IsValidDetail(geometry)** – gibt zurück ob Geometrie Fehler enthält – mit Fehlerreport

Beispiele für Geometry Editors (Editierfunktionen)

- ◆ **ST_AddPoint** – Punkt hinzufügen
- ◆ **ST_RemovePoint** – Punkt löschen
- ◆ **ST_Reverse** – Laufrichtung von Linie/Polygon umdrehen
- ◆ **ST_Rotate** - rotieren
- ◆ **ST_Scale** – skalieren
- ◆ **ST_Translate** – verschieben
- ◆ **ST_Affine** – affine Transformation
- ◆ **ST_Multi** – Umwandlung in Multigeometrie
- ◆ **ST_Segmentize** – Linien segmentieren
- ◆ **ST_Snap** – Geometrie auf Grid snappen
- ◆ **ST_SetPoint** – xten Punkt ersetzen

Beispiele für räumliche Beziehungsfunktionen

- ◆ **ST_Distance(geometry, geometry)**
- ◆ **ST_DWithin(geometry, geometry, float)**
- ◆ **ST_Equals(geometry, geometry)**
- ◆ **ST_Disjoint(geometry, geometry)**
- ◆ **ST_Crosses(geometry, geometry)**
- ◆ **ST_Touches(geometry, geometry)**
- ◆ **ST_Intersects(geometry, geometry)**

- ◆ **ST_Within(geometry, geometry) != ST_Contains(...)**
- ◆ **ST_Covers(geometry, geometry) != ST_CoveredBy(...)**
- ◆ **ST_Overlaps(geometry, geometry)**

Beispiel „Crosses“

Prüfe ob der Fluss Emme der Gemeindegeometrie von Burgdorf durchläuft

```
SELECT ST_Crosses(  
  (SELECT geom FROM gemeinden WHERE name = 'Burgdorf'),  
  (SELECT geom FROM fluesse WHERE name = 'Emme'));
```

Ergebnis: „t“ (true)

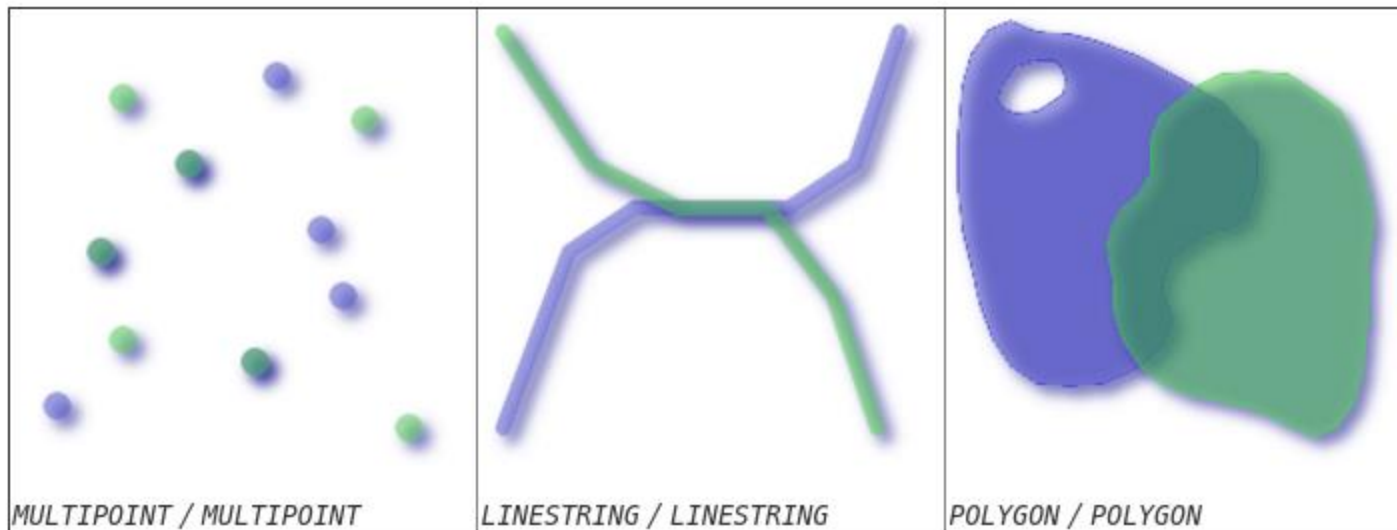
```
SELECT ST_Crosses(  
  (SELECT geom FROM gemeinden WHERE name = 'Basel'),  
  (SELECT geom FROM fluesse WHERE name = 'Emme'));
```

Ergebnis: „f“ (false)

Beispiel „Overlaps“

Prüfe ob 2 Geometrien sich überlappen

```
SELECT a.*, b.* FROM a, b  
WHERE ST_Overlaps(a.geom, b.geom) ;
```



Resultat bei allen obigen Beispielen: true

Beispiele für räumliche Berechnungsfunktionen (Processing Functions)

- ◆ **ST_Centroid(geometry)**
- ◆ **ST_Area(geometry)**
- ◆ **ST_Length(geometry)**
- ◆ **ST_PointOnSurface(geometry)**
- ◆ **ST_Boundary(geometry)**
- ◆ **ST_Buffer(geometry, double, [integer])**
- ◆ **ST_ConvexHull(geometry)**
- ◆ **ST_Intersection(geometry, geometry)**
- ◆ **ST_SymDifference(geometry A, geometry B)**
- ◆ **ST_Difference(geometry A, geometry B)**
- ◆ **ST_Union(geometry, geometry)**
- ◆ **ST_Union(geometry set)**
- ◆ **ST_MemUnion(geometry set)**

Beispiel „ST_Buffer“

Rechnet Buffer um Geometrie – mit Optionen

```
SELECT ST_Buffer(  
  ST_GeomFromText('POINT(100 90)'),  
  50, 'quad_segs=8');
```



```
SELECT ST_Buffer(  
  ST_GeomFromText(  
    'LINESTRING(50 50,150 150,150 50)'  
  ), 10, 'endcap=flat join=round');
```



Beispiele f. räumliche Managementfunktionen

- ◆ Abfragen von Versionen, Libraries, Build-Date, etc.
- ◆ AddGeometryColumn - Räumliche Spalten hinzufügen
- ◆ DropGeometryColumn – Räumliche Spalten löschen
- ◆ UpdateGeometrySRID - aktualisiert Koordinatensystem in Geometrien und Metadaten

Tipps

- ◆ Beachte den Unterschied von `st_intersects()` und `st_intersection()`
- ◆ «Contains/Covers/Intersects/Within?» by Michael Entin 2019
→ TL;DR: use `ST_Intersects` in most cases.
<https://mentin.medium.com/which-predicate-cb608b470471>
- ◆ Siehe Cheatsheet und GISWiki HSR/OST