



# BANCO DE DADOS ESPACIAL I

Marinêz da Silva

# Operações com Arquivos Shape

---

RS.shp

- Tabela: rs
- Tabela: santa\_maria

# CRIAÇÃO DA TABELA santa\_maria

---

```
CREATE TABLE santa_maria as  
SELECT *  
FROM rs  
WHERE nome = 'Santa Maria';
```

# CRIAÇÃO DA TABELA sm\_mancha\_urbana

```
CREATE TABLE sm_mancha_urbana as
SELECT st_intersection(r.geom, m.geom) AS
       intersection_geom, r.*
FROM rs_manchas_urbanas AS r, rs AS m
WHERE r.geom && m.geom
AND st_intersects(r.geom, m.geom)
AND m.nome = 'Santa Maria';
```

**&&** - Retorna TRUE se a caixa delimitadora intersecta a caixa delimitadora de B.

# CRIAÇÃO DA TABELA `sm_mancha_urbana`

---

## **CREATE TABLE `sm_mancha_urbana` as**

- Cria uma nova tabela chamada `sm_mancha_urbana`
- A tabela será preenchida com os dados retornados pela consulta SQL – `SELECT ...`

## **SELECT ...**

- Define os campos que serão incluídos na nova tabela

# CRIAÇÃO DA TABELA sm\_mancha\_urbana

---

**SELECT st\_intersection(r.geom, m.geom) AS  
intersection\_geom,**

- Esta função retorna a geometria da interseção entre r.geom (rs\_manchas\_urbanas) e m.geom de (rs)
- Intersection\_geom vai armazenar apenas a parte comum das duas geometrias

# CRIAÇÃO DA TABELA `sm_mancha_urbana`

---

**`r.*`**

- Inclui todas as colunas da tabela `rs_manchas_urbanas` (apelidada como `r`) no resultado

**FROM `rs_manchas_urbanas` AS `r`, `rs` AS `m`**

- Utiliza duas tabelas espaciais:
  - ❖ `rs_manchas_urbanas` contém dados de áreas urbanas
  - ❖ `rs` contém todos os municípios do Rio Grande do Sul

# CRIAÇÃO DA TABELA sm\_mancha\_urbana

---

## WHERE cláusulas

### r.geom && m.geom

- Operador de retângulo envolvente
- Verifica se as geometrias potencialmente se intersectam com base em seus limites
- É uma verificação rápida e eficiente para reduzir o número de comparações



# CRIAÇÃO DA TABELA `sm_mancha_urbana`

---

## **WHERE cláusulas**

### **`st_intersects(r.geom, m.geom)`**

- Confirma se as geometrias de fato se intersectam (não apenas seus retângulos envolventes)
- Garante que o resultado da `st_intersection` não será nulo

### **`m.nome = 'Santa Maria'`**

- Filtra para considerar apenas a geometria correspondente ao município de Santa Maria na tabela `rs`

## RELACIONAMENTOS ESPACIAIS:

---

- **ST\_Length (geometria):** retorna o comprimento da cadeia de linhas.
- **ST\_Intersects (geometria, geometria)** Retorna verdadeiro se duas geometrias compartilham algum espaço.  
Testa se duas geometrias se cruzam (se elas tem pelo menos um ponto em comum)

## RELACIONAMENTOS ESPACIAIS:

---

- **ST\_Intersection (geometria A, geometria B)** função retorna a área espacial (ou linha, ou ponto) que ambos os argumentos têm em comum. Se os argumentos forem disjuntos, a função retorna uma geometria vazia.

Calcula uma geometria que representa a parte compartilhada das geometrias A e B

# TABELA sm\_mancha\_urbana

```
ALTER TABLE sm_mancha_urbana  
ALTER COLUMN intersection_geom  
TYPE geometry(MULTIPOLYGON, 4326)  
USING ST_SETSRID(intersection_geom, 4326);
```

ALTER TABLE – altera a estrutura da tabela chamada sm\_mancha\_urbana

ALTER COLUMN – altera a coluna de geometria chamada intersection\_geom

Após a execução do comando:

A coluna intersection\_geom terá tipo MultiPolygon;

O SRID será fixado como 4326;

A tabela aparecerá corretamente na view geometry\_columns;

A geometria estará coerente e utilizável em análises espaciais com base no SRID.

# Operações com Arquivos Shape

---

hidrografia\_linhas\_latlong.shp

- Tabela: rs\_hidrografia
- Tabela: sm\_hidrografia

# CRIAÇÃO DA TABELA sm\_hidrografia

```
CREATE TABLE sm_hidrografia as  
SELECT st_intersection(r.geom, m.geom) AS  
       intersection_geom,  
       ST_length(r.geom) AS rd_orig_length, r.*  
FROM rs_hidrografia AS r, rs AS m  
WHERE r.geom && m.geom  
AND ST_intersects(r.geom, m.geom)  
AND m.nome = 'Santa Maria';
```

**ST\_LENGTH(r.geom) AS rd\_orig\_length**

Calcula o comprimento original da geometria da linha r.geom.  
Útil para comparar depois com o comprimento da interseção.

# TABELA sm\_hidrografia

---

```
ALTER TABLE sm_hidrografia  
ALTER COLUMN intersection_geom  
TYPE geometry(MULTILINESTRING, 4326)  
USING ST_SETSRID(intersection_geom, 4326);
```

# Operações com Arquivos Shape

---

curvas\_de\_nivel\_latlong.shp

- Tabela: rs\_curvas
- Tabela: sm\_curvas



# CRIAÇÃO DA TABELA sm\_curvas

---

```
CREATE TABLE sm_curvas as
SELECT ST_Intersection(r.geom, m.geom) AS
    intersection_geom,
ST_length(r.geom) AS rd_orig_length, r.*
FROM rs_curvas AS r, rs AS m
WHERE r.geom && m.geom
AND ST_intersects(r.geom, m.geom)
AND m.nome = 'Santa Maria';
```

# TABELA sm\_curvas

---

```
ALTER TABLE sm_curvas  
ALTER COLUMN intersection_geom  
TYPE geometry(MULTILINESTRING, 4326)  
USING ST_SETSRID(intersection_geom, 4326);
```

# CRIAÇÃO DA TABELA sm\_sistema\_viario

---

```
CREATE TABLE sm_sistema_viario as
SELECT st_intersection(r.geom, m.geom) AS
intersection_geom,
st_length(r.geom) AS rd_orig_length, r.*
FROM rs_viario AS r, rs AS m
WHERE r.geom && m.geom
AND st_intersects(r.geom, m.geom)
AND m.nome = 'Santa Maria';
```

# CRIAÇÃO DA TABELA sm\_sistema\_viario

---

```
ALTER TABLE sm_sistema_viario  
ALTER COLUMN intersection_geom  
TYPE geometry(MULTILINESTRING, 4326)  
USING ST_SETSRID(intersection_geom, 4326);
```

# REFERÊNCIAS

---

<https://postgis.net/documentation/>