



HSR

HOCHSCHULE FÜR TECHNIK
RAPPERSWIL

FHO Fachhochschule Ostschweiz

Swiss PG Day 2018

PostGIS für Fortgeschrittene

sfkeller@hsr.ch



Einführung

- PostGIS
- Use Cases
 - Areas-of-Interest
 - Some shamelessly copied from mastermind Paul Ramsey
 - ... and some spatial SQL picks
- Diskussion

Motivation

- Data Mangling (ELT - ETL)
- Visualization (and Reporting)
- **Analysis!**



PostGIS - An Extension of PostgreSQL

- CREATE EXTENSION postgis;
- Contains
 - Data types (OGC)
 - Indexes
 - Executables (Data Loaders)
- Dependencies
 - GEOS, PROJ
 - Data (e.g. CRS/SRS)

PostGIS - An Extension of PostgreSQL ff.

- <https://medium.com/@tjukanov/why-should-you-care-about-postgis-a-gentle-introduction-to-spatial-databases-9eccd26bc42b>



Topi Tjukanov

Follow

I mostly put lines, points and polygons on maps 🌍

Jun 7 · 15 min read

Why should you care about PostGIS?—A gentle introduction to spatial databases

Databases? Not very interesting.

So might an average person working with GIS or data visualizations think. I must admit that databases aren't the sexiest thing in the world (sorry DBA's), but if you are claiming (or aiming) to do analytics or visualization with (spatial) data in a more serious manner, you definitely shouldn't ignore them. I hope this blog post can give you an idea what kind of benefits the efficient use of spatial databases could offer you.

Getting started with PostGIS?



I have been reading a lot about PostGIS, but I am struggling to find a good beginner's guide.

85

Which tutorials/books do you recommend?

postgis

references



share edit flag

edited Mar 7 '16 at 7:36

community wiki

relima



55



<http://workshops.boundlessgeo.com/postgis-intro>

<https://github.com/CartoDB/carto-workshop/tree/master/04-database>

PostGIS – Related Extensions

- `postgis_sfcgal`
- `postgis_topology`
- `pgrouting`
- `ogr_fdw`
- `pointcloud` and `pointcloud_postgis`

PostGIS – Some dev tools

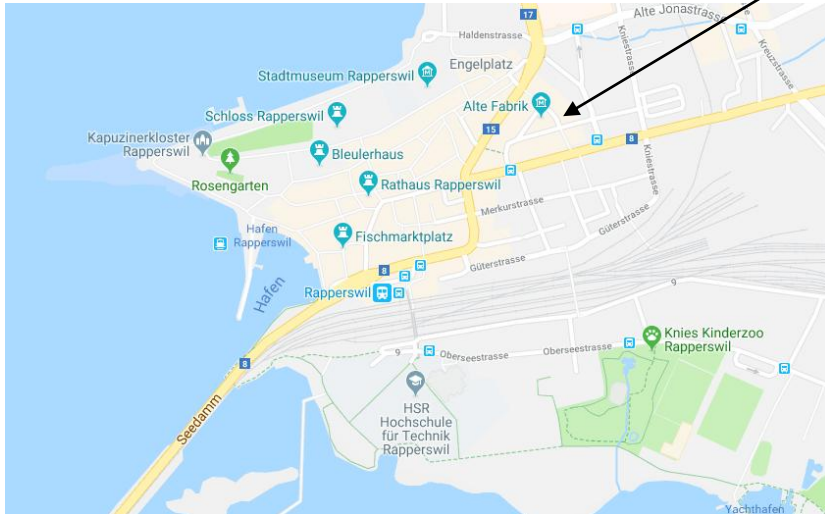
- GUI
 - pgAdmin
 - QGIS
 - postgis-editor (by Lukas Martinelli)
- CLI
 - psql
 - (and probably more to come in the future with command completion...)
- Web-based
 - CARTO – <https://carto.com/>
 - QGIS Cloud - <http://qgiscloud.com/>

Use Case **Areas-of-Interest**

by **Philipp Koster**
cand. MSc, HSR, 2018

Koster Thesis: Areas-of-Interest

- Areas-of-Interest (AOI)
- Based on OpenStreetMap data
- and a reproducible algorithm



Google Maps: “*Highly frequented and worth seeing areas*” (light orange)

Our def.: “*Urban area at city or neighbourhood level with a high concentration of POI, and typically located along a street of high spatial importance*”

Koster Thesis: Areas-of-Interest ff.

- Open Source Tools and SQL plus preferably Python
- Implement AOI with a robust database (and explore it's limits)
 - PostgreSQL/PostGIS
 - Python
- Implement AOI with a “Big Data Framework” which supports SQL
 - GeoSpark
 - DataFrames (SQL+Scala) with fallback to RDD (Scala)

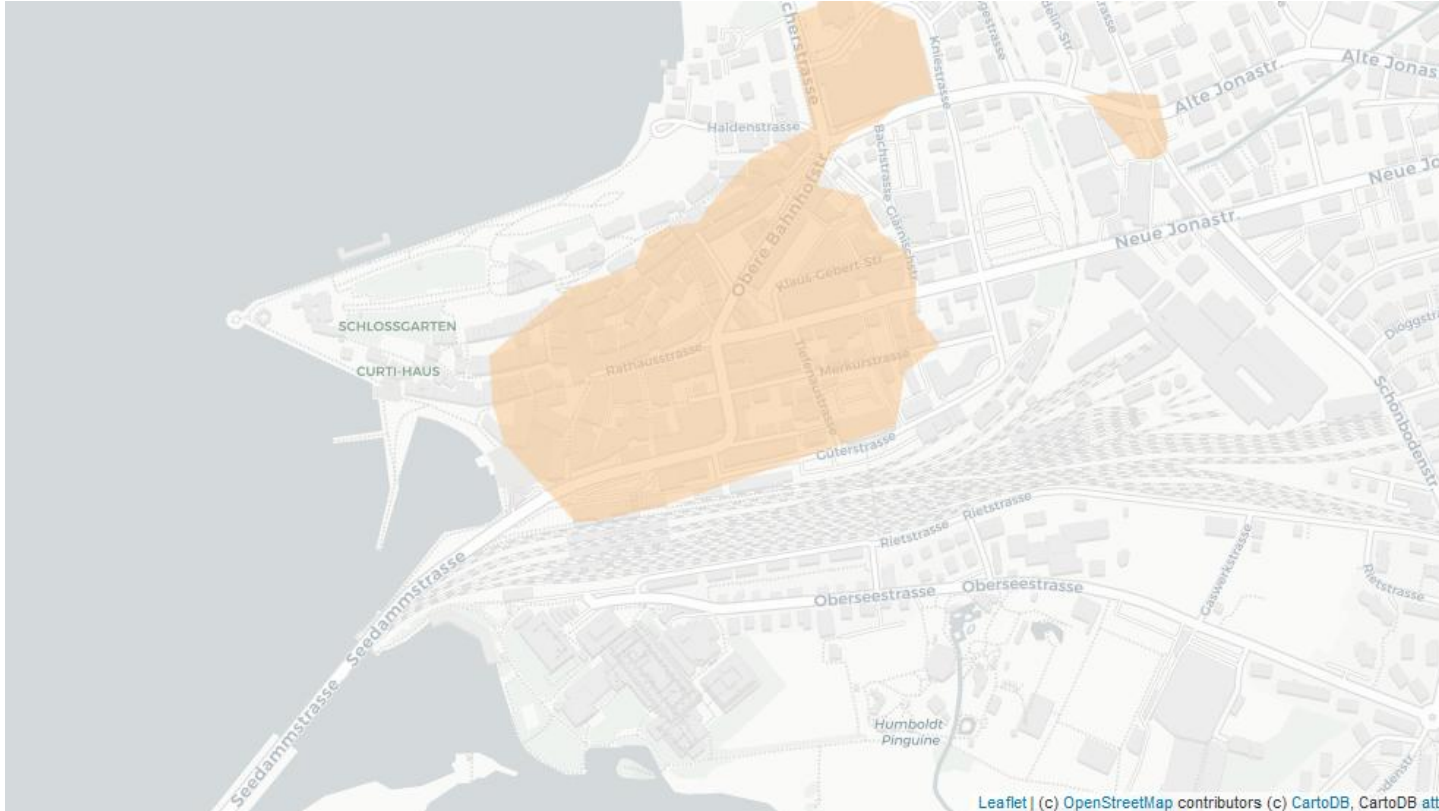


Koster Thesis: Areas-of-Interest ff.

Processing steps:

1. Get polygons – filter data from OSM
 2. Cluster polygons – with ST_ClusterDBSCAN
 3. Create hulls around clusters – with ST_ConcaveHull
 4. Extend using network centrality –
get street network from OSM, apply ST_Buffer, ST_Intersection, ST_Union
 5. Exclude waterways and water (enhance) and sanitize
- Done!

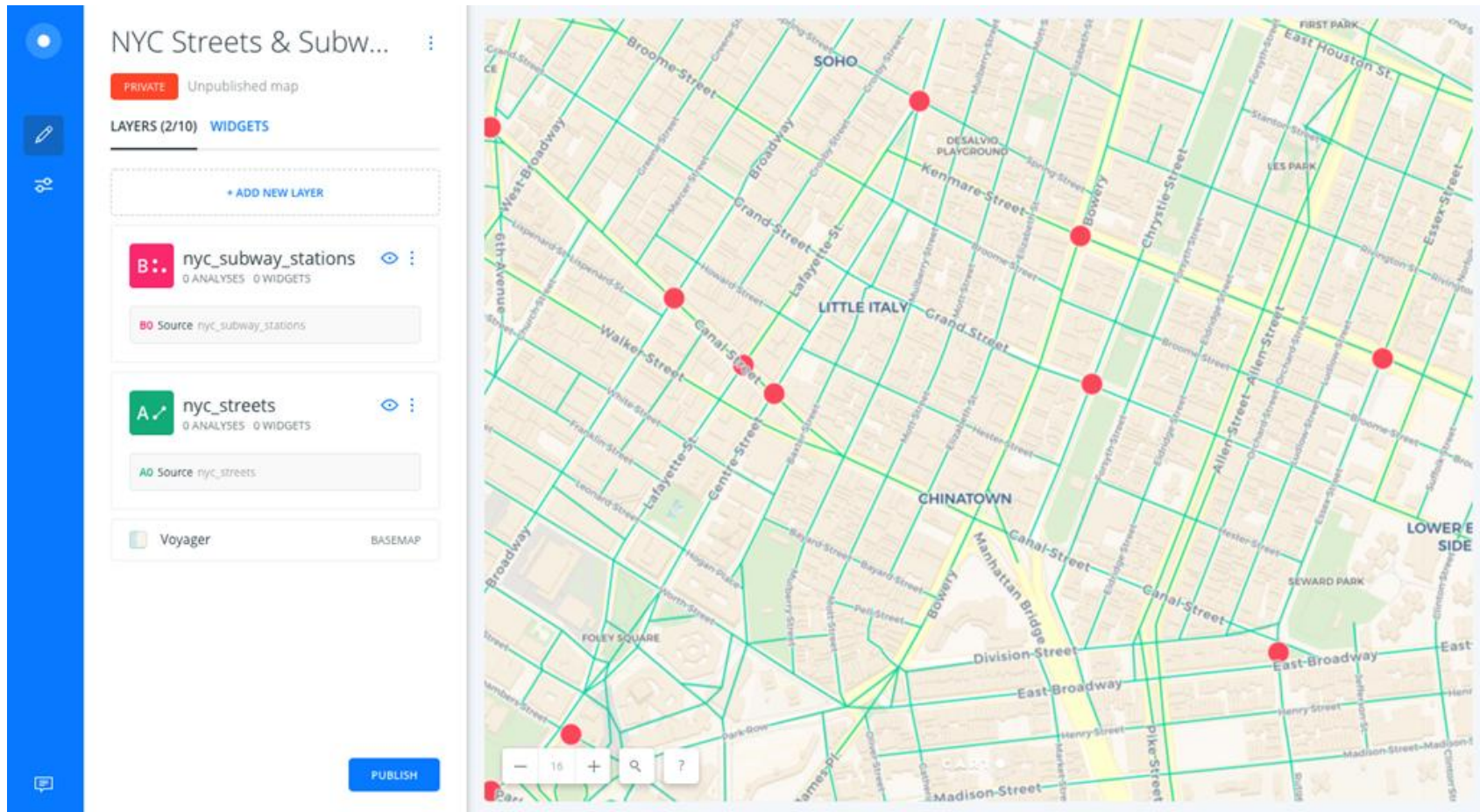
Koster Thesis: Areas-of-Interest Demo



To be
announced
soon.

See e.g.
<https://2018.stateofthemap.org/2018/A30-Areas-of-Interest-for-OpenStreetMap-with-Big-Spatial-Data-Analyses/>

**Use Case „Location Analytics“
(taken from Paul Ramsey's talk „SQL
Festival“, FOSS4G NA 2018)**



← Back / Layer options

nyc_streets

nyc_streets

DATA

ANALYSIS

STYLE

POP-UP

LEGEND

```
1 SELECT
2   DISTINCT ON (sbw.cartodb_id)
3   str.*
4 FROM nyc_streets str
5 JOIN nyc_subway_stations sbw
6 ON ST_DWithin(
7   str.the_geom_webmercator,
8   sbw.the_geom_webmercator,
9   200)
10 ORDER BY
11 | sbw.cartodb_id,
12 | ST_Distance(
13 |   str.the_geom_webmercator,
14 |   sbw.the_geom_webmercator)
```

CMD + S to apply your query. CTRL + Space to autocomplete.

VALUES ☒ SQL

↶ ↷

CLEAR

APPLY

<http://files.boundlessgeo.com/workshopmaterials/postgis-workshop-201401.zip>

```
SELECT
  DISTINCT ON (sbw.cartodb_id)
  str.*
FROM nyc_streets str
JOIN nyc_subway_stations sbw
ON ST_DWithin(
  str.the_geom_webmercator,
  sbw.the_geom_webmercator,
  200)
ORDER BY
  sbw.cartodb_id,
  ST_Distance(
    str.the_geom_webmercator,
    sbw.the_geom_webmercator)
```


subway	street	distance
B	32nd Street	123.5
A	23nd Street	23.4
B	Reed Avenue	12.4
A	24th Street	34.4
B	25th Street	102.2
A	Reed Avenue	45.5
C	State Street	32.1
B	State Street	56.4

```
SELECT
  DISTINCT ON (sbw.cartodb_id)
  str.*
FROM nyc_streets str
JOIN nyc_subway_stations sbw
ON ST_DWithin(
  str.the_geom_webmercator,
  sbw.the_geom_webmercator,
  200)
ORDER BY
  sbw.cartodb_id,
  ST_Distance(
    str.the_geom_webmercator,
    sbw.the_geom_webmercator)
```

subway	street	distance
A	23nd Street	23.4
A	24th Street	34.4
A	Reed Avenue	45.5
B	Reed Avenue	12.4
B	State Street	56.4
B	25th Street	102.2
B	32nd Street	123.5
C	State Street	32.1

```
SELECT
  DISTINCT ON (sbw.cartodb_id)
  str.*
FROM nyc_streets str
JOIN nyc_subway_stations sbw
ON ST_DWithin(
  str.the_geom_webmercator,
  sbw.the_geom_webmercator,
  200)
ORDER BY
  sbw.cartodb_id,
  ST_Distance(
    str.the_geom_webmercator,
    sbw.the_geom_webmercator)
```

subway	street	distance
A	23nd Street	23.4
B	Reed Avenue	12.4
C	State Street	32.1

**Use Case „Selected PostGIS Q&A”
(taken from Paul Ramsey’s talk „SQL
Festival”, FOSS4G NA 2018)**

Some popular PostGIS questions on
Stack Exchange?

[https://gis.stackexchange.com/questions
/tagged/postgis?
sort=votes&
pageSize=15](https://gis.stackexchange.com/questions/tagged/postgis?sort=votes&pageSize=15)

66

votes

4

answers

21k views

What are the pros and cons of PostGIS geography and geometry types?

My company uses geometry (the_geom) data type for storing geospatial data. I've recently been acquainted to the concept of geography (the_geog) data type which, as I understand it, stores the SRID ...

postgis

geometry-data-type

geography-data-type

asked Mar 2 '11 at 13:31



Adam Matan

3,925



5



28



45

Geography Pros

- Exact
- Understood
- “Simple”

Geography Cons

- Slow
- Incomplete
- Confusing

Spatial clustering with PostGIS



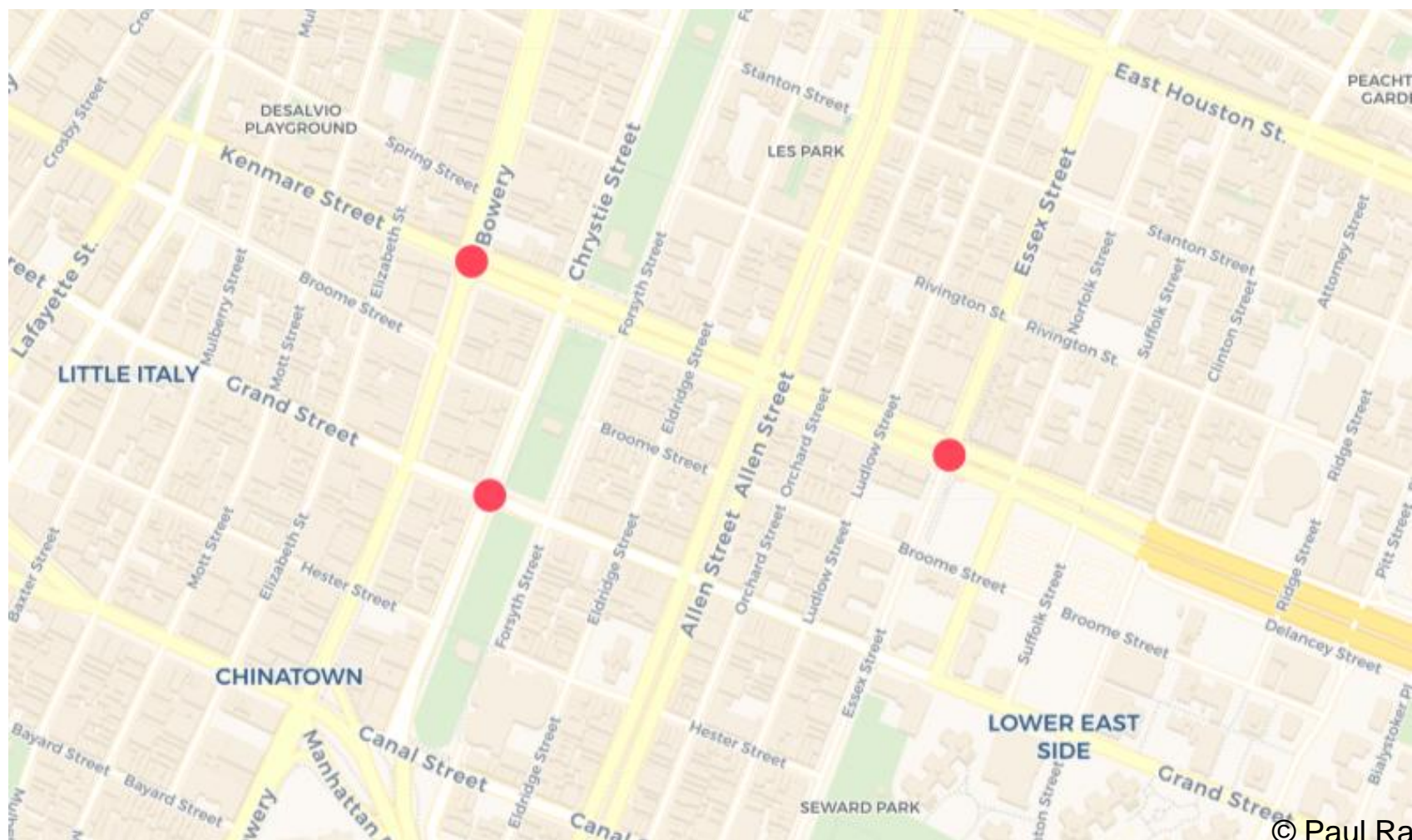
84



I'm looking for spatial clustering algorithm for using it within PostGIS-enabled database for point features. I'm going to write plpgsql function that takes distance between points within the same cluster as input. At the output function returns array of clusters. The most obvious solution is to build buffer zones specified distance around the feature and search for features into this buffer. If such features exist then continue to build a buffer around them, etc. If such features not exist that means cluster building is completed. Maybe there are some clever solutions?

postgis

clustering



UNIFORM

```
CREATE TABLE nyc_subway_stations_gaussian AS
WITH u AS (
  SELECT random() AS u1,
         random() AS u2
  FROM generate_series(1,1000)
),
```

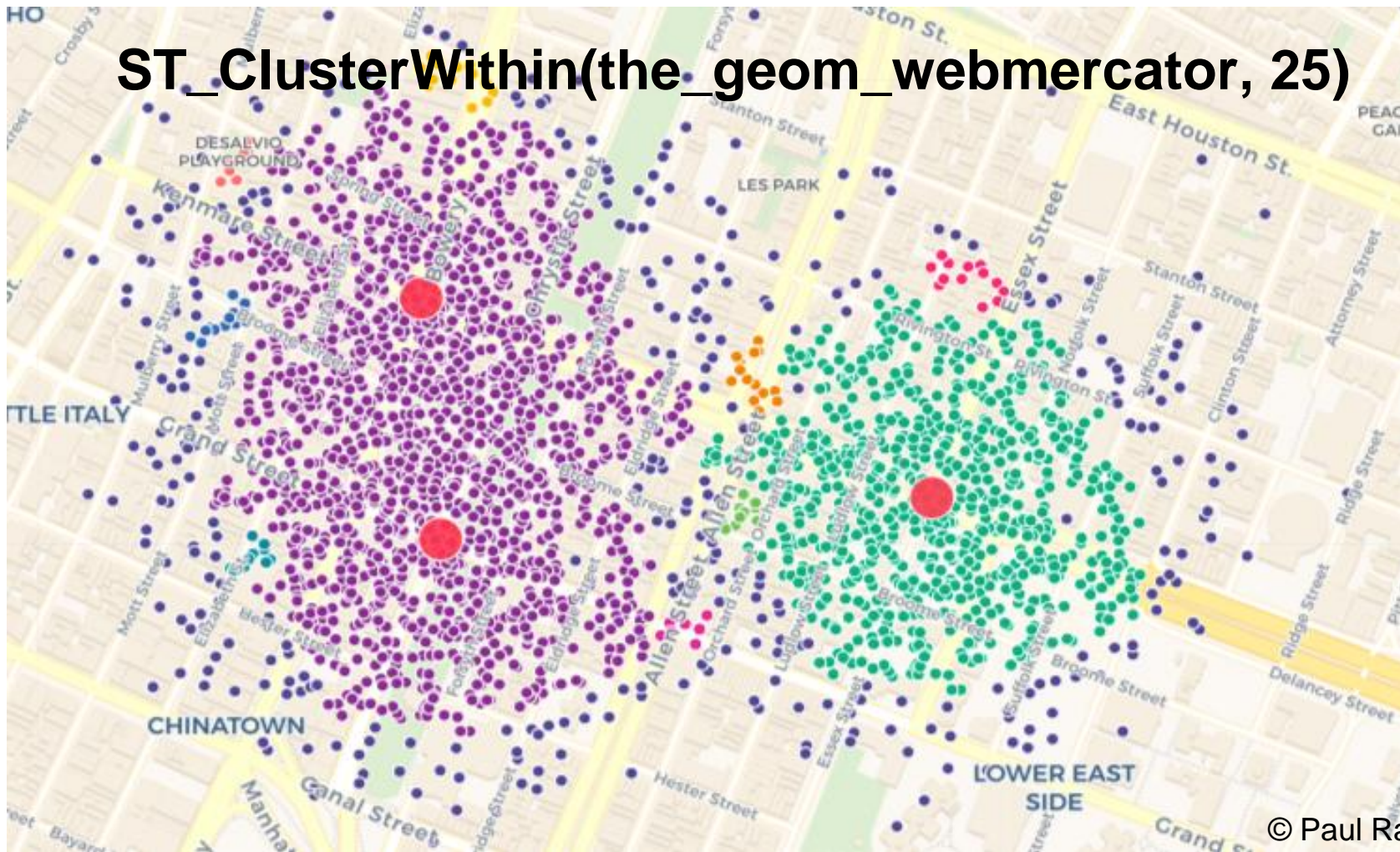
GAUSSIAN

```
off AS (
  SELECT
    sqrt(-2*ln(u1))*cos(2*pi()*u2) AS x_off,
    sqrt(-2*ln(u1))*sin(2*pi()*u2) AS y_off
  FROM u
)
```

JOIN/OFFSET

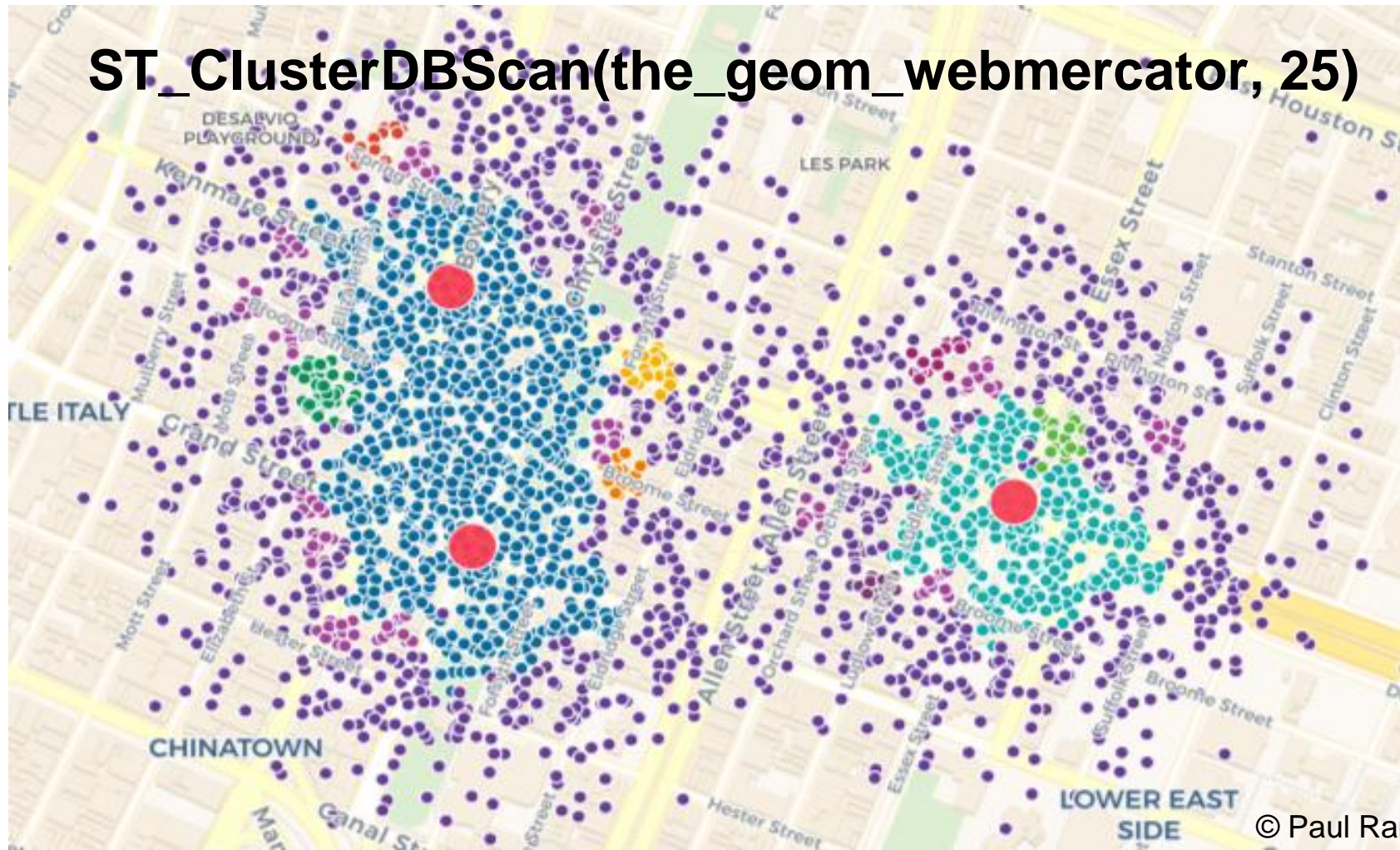
```
SELECT row_number() OVER () AS cartodb_id, name, routes,
       ST_Translate(the_geom_webmercator, 150*x_off, 150*y_off)
       AS the_geom_webmercator
FROM nyc_subway_stations sub
CROSS JOIN off
WHERE sub.name IN ('Grand St', 'Bowery', 'Essex St')
```


ST_ClusterWithin(the_geom_webmercator, 25)



```
WITH c AS (  
    SELECT  
        unnest(ST_ClusterWithin(the_geom_webmercator, 23))  
        AS the_geom_webmercator  
    FROM nyc_subway_stations_gaussian  
)  
d AS (  
    SELECT row_number() OVER () AS cluster_id,  
        ST_Dump(the_geom_webmercator) AS dump  
    FROM c  
)  
SELECT  
    row_number() OVER () AS cartodb_id,  
    cluster_id,  
    (dump).geom AS the_geom_webmercator  
FROM d
```

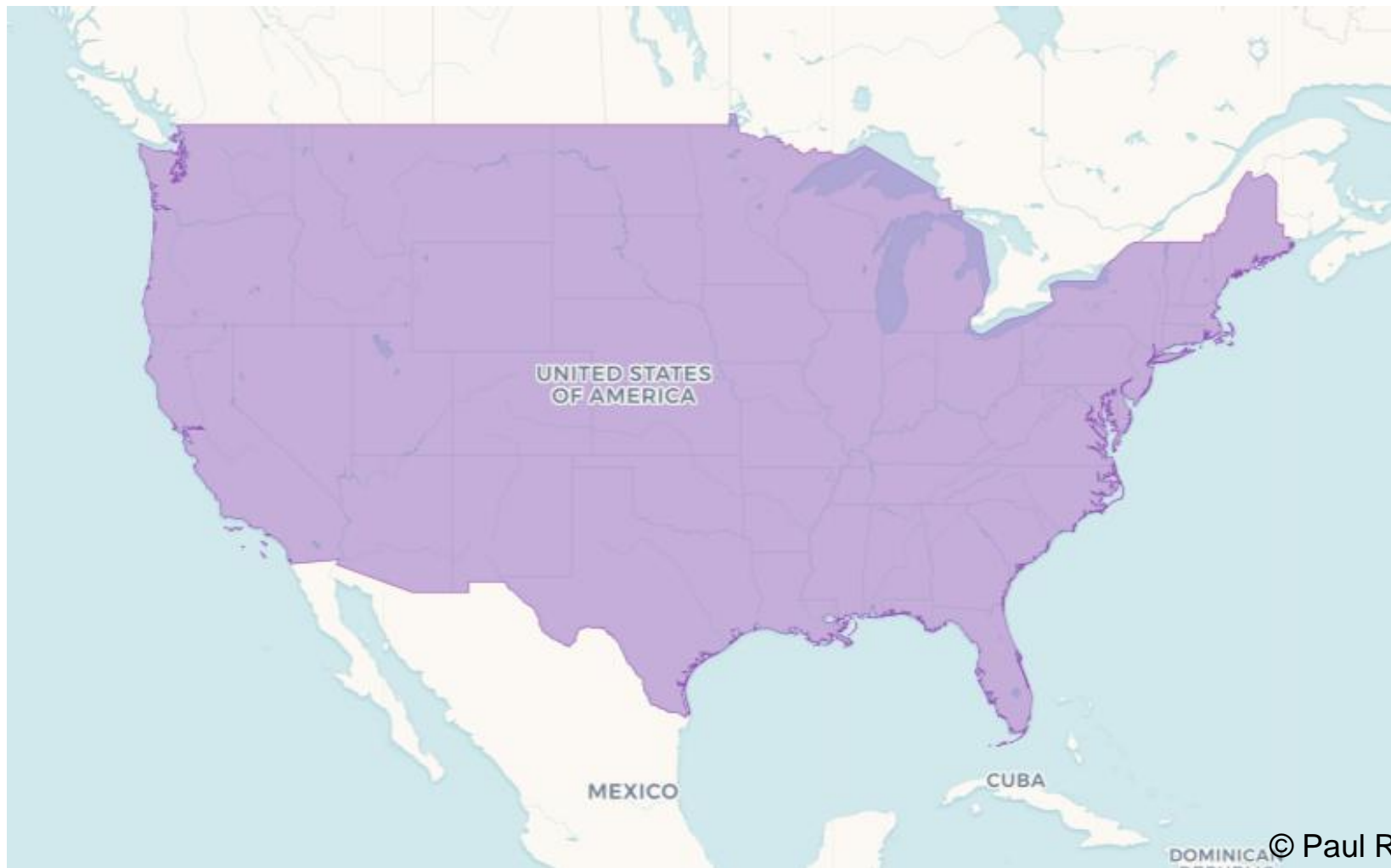

ST_ClusterDBScan(the_geom_webmercator, 25)



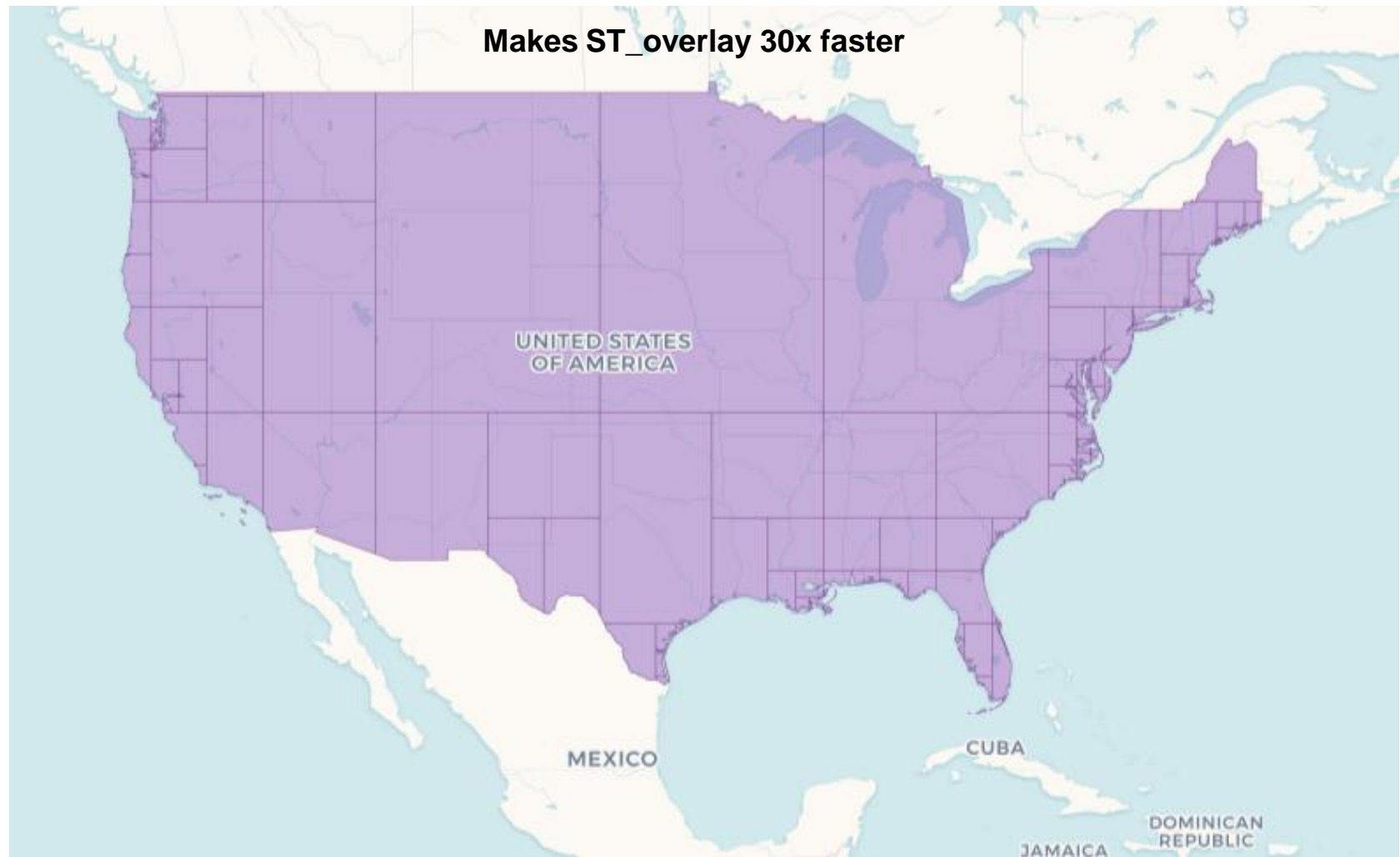
Point Clustering

- GISPunkt Wiki HSR
- https://giswiki.hsr.ch/PostGIS_Terminal_Examples#Point_Clustering

**Spatial SQL Pick „ST_Subdivide()”
(taken from Paul Ramsey’s talk „SQL
Festival”, FOSS4G NA 2018)**

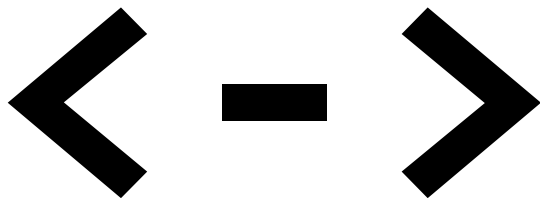


Makes ST_overlay 30x faster



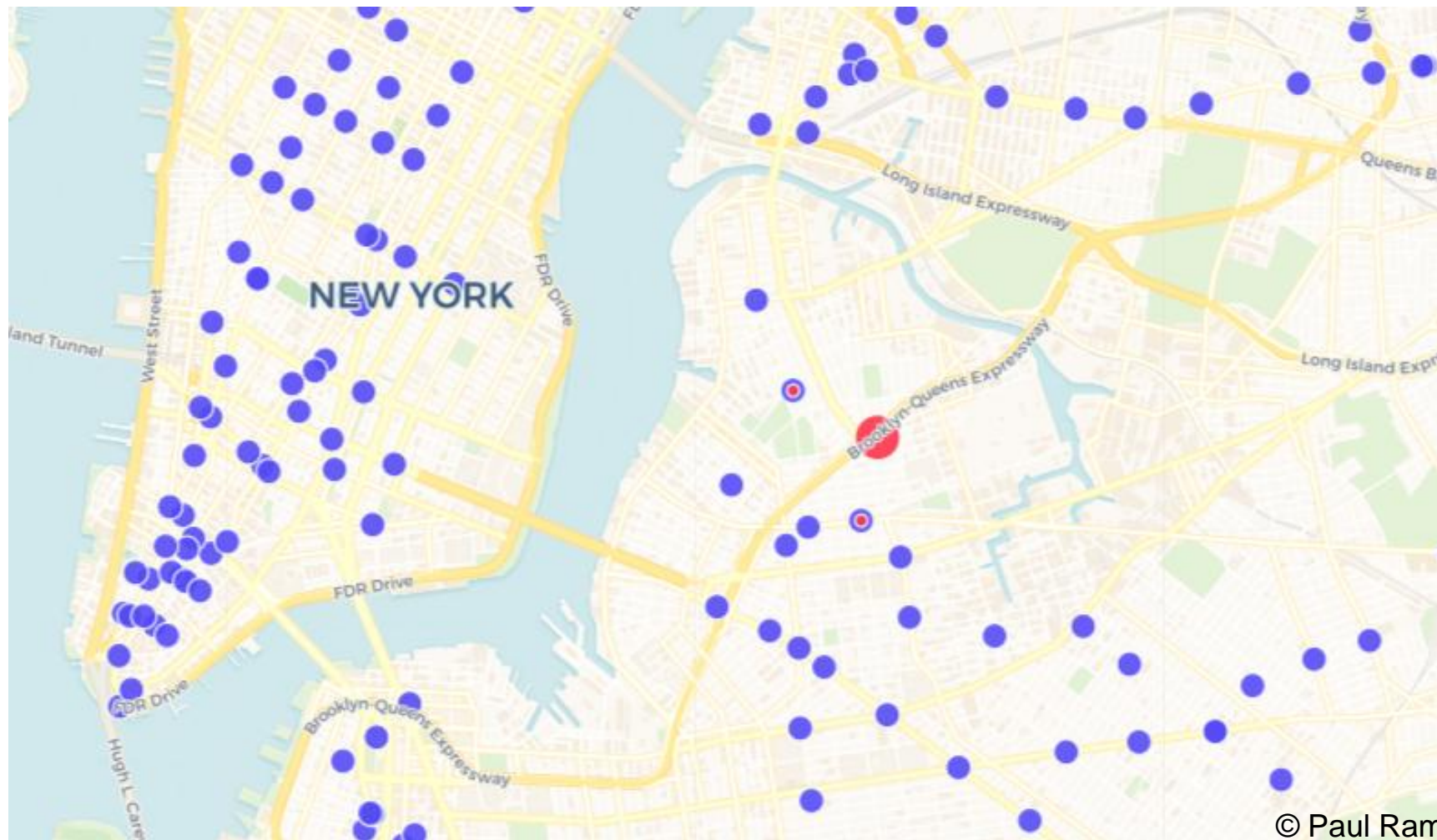
Spatial SQL Pick „K Nearest Neighbours” Operator and Index (also taken from Paul Ramsey’s talk „SQL Festival”, FOSS4G NA 2018)

“Find the nearest from...”



Returns the 2D distance between A and B.

```
SELECT *  
FROM nyc_subway_stations  
ORDER BY  
    the_geom <->  
    CDB_LatLng(40.72, -73.94)  
LIMIT 2;
```



**“Find the nearest that
is not yourself.”**

```
SELECT
    row_number() OVER () AS cartodb_id,
    ST_Transform(ST_MakeLine(a.the_geom, b.the_geom), 3857)
    AS the_geom_webmercator,
    ST_DistanceSphere(a.the_geom, b.the_geom)
    AS distance,
    a.name AS name_a,
    b.name AS name_b
FROM nyc_subway_stations AS a
CROSS JOIN LATERAL (
    SELECT subq.the_geom, subq.name
    FROM nyc_subway_stations subq
    WHERE a.cartodb_id != subq.cartodb_id
    ORDER BY
        a.the_geom <-> subq.the_geom
    LIMIT 1) AS b
```


SELECT

...

FROM table_a AS a

CROSS JOIN LATERAL (

SELECT table_b.*

FROM table_b

ORDER BY

a.the_geom <-> table_b.the_geom

LIMIT 1) AS b

```
SELECT
    row_number() OVER () AS cartodb_id,
    ST_Transform(ST_MakeLine(a.the_geom, b.the_geom), 3857)
    AS the_geom_webmercator,
    ST_DistanceSphere(a.the_geom, b.the_geom)
    AS distance,
    a.name AS name_a,
    b.name AS name_b
FROM nyc_subway_stations AS a
CROSS JOIN LATERAL (
    SELECT subq.the_geom, subq.name
    FROM nyc_subway_stations subq
    WHERE a.cartodb_id != subq.cartodb_id
    ORDER BY
        a.the_geom <-> subq.the_geom
    LIMIT 1) AS b
```



PostGIS Terminal Examples

Curated by Stefan Keller

- „Die nächsten 10 Bars in der Nähe von 'mylocation' (ungeachtet der Distanz)“
- „Die nächsten 100 Restaurants in der Nähe...“

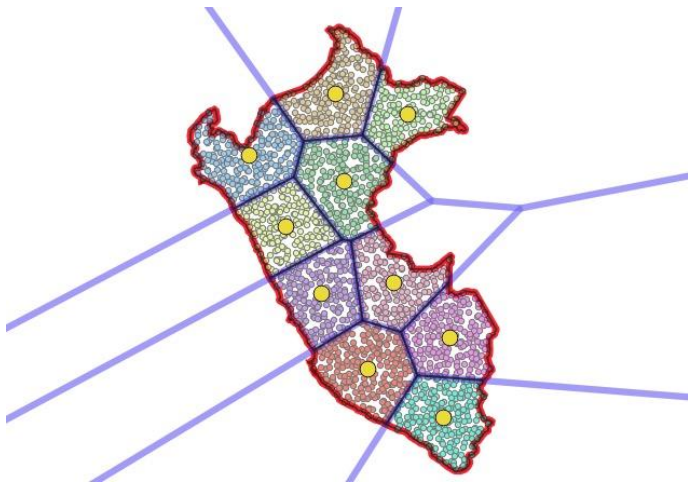
https://giswiki.hsr.ch/PostGIS_Terminal_Examples

PostGIS Polygon Splitting

(by Darafei Praliaskouski and Paul Ramsey)

PostGIS Polygon Splitting

- Blog post from Paul Ramsey inspired by Darafei Praliaskouski
- “Is there a way to split a polygon into sub-polygons of more-or-less equal N areas (where N is predefined, e.g. 10)?”



PostGIS Polygon Splitting ff.

<http://blog.cleverelephant.ca/2018/06/polygon-splitting.html>

Paul Ramsey

Open source software
developer and information
technology professional.
Occasional blowhard.

Home

About

Archive

Projects

Talks & Writing

PostGIS Polygon Splitting

21 Jun 2018

One of the joys of geospatial processing is the variety of tools in the tool box, and the ways that putting them together can yield surprising results. I have been in the guts of PostGIS for so long that I tend to think in terms of primitives: either there's a function that does what you want or there isn't. I'm too quick to ignore the power of combining the parts that we already have.

A community member on the users list asked (paraphrased): "is there a way to split a polygon into sub-polygons of more-or-less equal areas?"

I didn't see the question, which is lucky, because I would have said: "No, you're SOL, we don't have a good way to solve that problem." (An [exact algorithm](#) showed up in the Twitter thread about this solution, and maybe I should implement that.)

PostGIS developer [Darafei Praliaskouski](#) **did** answer, and [provided a working solution](#) that is absolutely brilliant in combining the parts of the PostGIS toolkit to solve a pretty tricky problem. He said:

Trajectories

(by Anita Graser)

Trajectories

- Def. Spatiotemporal trajectory is the periodic/non-periodic recording of spatial locations for a moving object
- „Movement data in GIS“ by Anita Graser
<https://anitagraser.com/category/gis/movement-data-in-gis/>
- PG-Trajectory - <http://pg-trajectory.dmlab.cs.gsu.edu/>

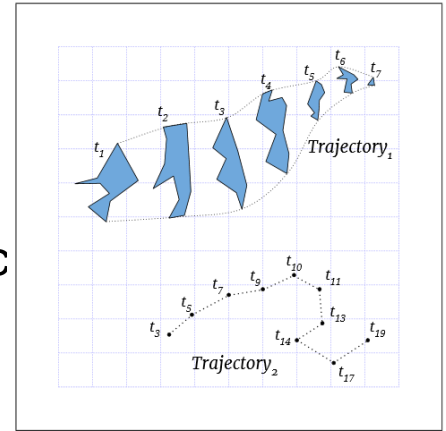


Fig. 1: Example of point and region trajectory

Welcome to PG-Trajectory

PG-Trajectory is a PostgreSQL/PostGIS extension for spatiotemporal data. Thanks to trajectory data model and functions, it is possible to develop complicated spatiotemporal applications. PG-Trajectory is developed by DMLAB @ GSU.

[Documentation!](#)

Weitere Ressourcen

Weitere Ressourcen...

- “Advanced Spatial Analysis with PostGIS” by Pierre Racine FOSS4G 2017 Workshop: https://github.com/pedrogit/postgis_workshop
- “Creating Centerlines with PostGIS and ArcGIS”, Phillip Penn, June 2017: <https://www.udcus.com/blog/2017/06/28/creating-centerlines-postgis-and-arccgis>
- “PostGIS Scaling” by Paul Ramsey - <http://blog.cleverelephant.ca/2017/12/postgis-scaling.html>

Diskussion

- Geodaten-Verwaltung optimieren
- Anfragen beschleunigen
- Grenzen von PostGIS
- Weitere Entwicklung (Parallelisierung, GPU)

