



PostGIS

die PostgreSQL Erweiterung zur Geodatenhaltung

Astrid Emde



Astrid Emde

-  WhereGroup, Bonn
- Projektumsetzung im Bereich WebGIS
- Projekte mit PostgreSQL/PostGIS, MapServer, GeoServer, Quantum GIS
- Aktiv in **OSGeo** und **FOSSGIS e.V.**



Was ist PostGIS?



Was ist PostGIS ?

- PostGIS ist ein räumlicher Aufsatz zur Speicherung und Verwaltung von Geodaten in PostgreSQL
- konform mit der OGC Simple Feature Spezifikation für SQL (SFSQL)
- Orientierung an der ISO Spezifikation SQL/MM Teil 3



Was ist PostGIS?

- Erweitert um PostGIS-eigene Funktionen
- Programmierung in C++, PL/pgSQL
- nutzt GEOS-, Proj4-, GDAL, LibXML2-Bibliothek



Vorteile der Geodatenhaltung in der Datenbank



Vorteile

- Geometrie und Sachdaten können in einer Tabelle abgelegt werden
- Alles in einem System
- Berechnungen direkt in der Datenbank über PostGIS Funktionen - GIS wird nicht benötigt



Die ersten Schritte



Installation

- PostgreSQL, GEOS, PROJ4

```
tar xvzf postgis-1.5.3.tar.gz
```

```
cd postgis-1.5.3
```

```
./configure
```

```
make
```

```
make install
```



Installation

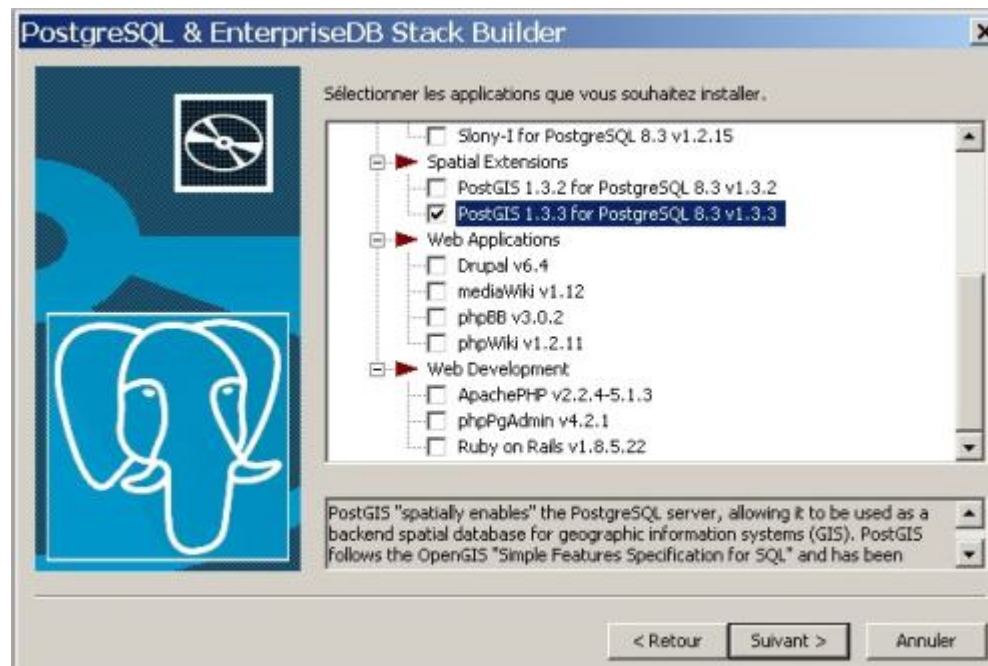
```
apt-get install postgresql  
postgresql-client postgresql-  
contrib pgadmin3
```

```
apt-get install postgresql-8.4-  
postgis
```



Installation unter Windows

- PostgreSQL Stack-Builder





PostGIS Datenbank

- prozedurale Sprache PL/pgSQL
- PostGIS Funktionen
- Metadatatabelle geometry_columns
- Tabelle mit Projektionsangaben
spatial_ref_sys



PostGIS Datenbank anlegen

```
createlang -U postgres plpgsql  
postgis_template;
```

```
psql -U postgres geodatabase -f  
/usr/share/postgresql/8.4/contrib/post  
gis-1.5/postgis.sql
```

```
psql -U postgres geodatabase -f  
/usr/share/postgresql/8.4/contrib/post  
gis-1.5/spatial_ref_sys.sql
```



PostGIS Template verwenden

```
createdb -U postgres  
-T postgis_template  
geodatabase
```



PostGIS Funktionen

- > 300 Funktionen
- C++ oder PL/pgSQL
- OGC Simple Feature for SQL Standard (SFSQL)
- SQL/MM Standard Teil 3
- PostGIS-eigene Funktionen



Tabelle geometry_columns

- Metadaten zu den Geometriespalten
 - Geometrietyp
 - Projektionssystem
 - Dimension

Ab 2.0 als View

	table_catalog	table_schema	table_name	geometry_column_name	coord_dimension	srid	type
1		public	poi	the_geom	2	31467	POINT
2		public	leitung	the_geom	2	31466	LINESTRING



Tabelle spatial_ref_sys

- > 3000 EPSG Codes
- Umrechnungsparameter für Transformation

	srid	auth_name	auth_srid	srtext	proj4text
1	3819	EPSG	3819	GEOGCS["HD1909...	+proj=longlat +ellps=bessel...
2	3821	EPSG	3821	GEOGCS["TWD67"...	+proj=longlat +ellps=aust_S...

srid	auth_name	auth_srid	srtext	proj4text
25832	EPSG	25832	PROJCS["ET...	+proj=utm +zone=32 +ellps=GRS80 +units=m +no_defs
31466	EPSG	31466	PROJCS["DH...	+proj=tmerc +lat_0=0 +lon_0=6 +k=1 +x_0=2500000 +y_0=0 +ellps=bessel +datum=potsdam +units=m +no_defs

5	3906	EPSG	3906	GEOGCS["MGI 190...	+proj=longlat +ellps=bessel...
6	4001	EPSG	4001	GEOGCS["Unknow...	+proj=longlat +ellps=airy +...
7	4002	EPSG	4002	GEOGCS["Unknow...	+proj=longlat +a=6377340....
8	4003	EPSG	4003	GEOGCS["Unknow...	+proj=longlat +ellps=aust_S...



Geometrietypen

- POINT, LINESTRING, POLYGON
- MULTIPOINT, MULTILINESTRING, MULTIPOLYGON
- GEOMETRYCOLLECTION
- CIRCULARSTRING, COMPOUNDCURVE, CURVEPOLYGON, MULTICURVE, MULTISURFACE
- GEOGRAPHY (ab 1.5)
- ab 2.0: TIN, POLYHEDRALSURFACE, TRIANGLE
- Ab 2.0: Raster
- Ab 2.0: TopoGeometry



WKT Repräsentation

`POINT(0 0)`

`POINT(0 0 0)`

`LINESTRING(0 0,1 1,1 2)`

`POLYGON((0 0,4 0,4 4,0 4,0 0),(1 1, 2 1, 2 2, 1 2,1 1))`

`MULTIPOINT(0 0,1 2)`

`MULTILINESTRING((0 0,1 1,1 2),(2 3,3 2,5 4))`

`GEOMETRYCOLLECTION(POINT(2 3),LINESTRING(2 3,3 4))`



Die erste Tabelle



Tabelle mit Punktobjekten

```
CREATE TABLE poi (gid  
serial, name varchar);
```

```
SELECT AddGeometryColumn(  
'public' , 'poi' ,  
'the_geom' , 31466 ,  
'POINT' , 2  
);
```



ST_GeomFromText

```
INSERT INTO poi
  (name , the_geom)
VALUES (
  'Rathaus' ,
  ST_GeomFromText(
    'POINT(2568793 5631376)' ,
    31466
  )
);
```





Name

ST_GeomFromText — Return a specified ST_Geometry value from Well-Known Text representation (WKT).

Synopsis

geometry **ST_GeomFromText**(text *WKT*);

geometry **ST_GeomFromText**(text *WKT*, integer *srid*);

Description

Constructs a PostGIS ST_Geometry object from the OGC Well-Known text representation.



There are 2 variants of ST_GeomFromText function, the first takes no SRID and returns a geometry with no defined spatial reference system. The second takes a spatial reference id as the second argument and returns an ST_Geometry that includes this srid as part of its meta-data. The srid must be defined in the spatial_ref_sys table.

- ✓ This method implements the [OpenGIS Simple Features Implementation Specification for SQL 1.1](#). s3.2.6.2 - option SRID is from the conformance suite.
- ✓ This method implements the SQL/MM specification. SQL-MM 3: 5.1.40
- ✓ This method supports Circular Strings and Curves

Examples

```
SELECT ST_GeomFromText('LINESTRING(-71.160281 42.258729,-71.160837 42.259113,-71.161144 42.25932)');
SELECT ST_GeomFromText('LINESTRING(-71.160281 42.258729,-71.160837 42.259113,-71.161144 42.25932)',4269);

SELECT ST_GeomFromText('CIRCULARSTRING(220268 150415,220227 150505,220227 150406)');
```

See Also

[ST_GeomFromEWKT](#), [ST_GeomFromWKB](#), [ST_SRID](#)

[Prev](#)
[ST_GMLToSQL](#)

[Up](#)
[Home](#)

[Next](#)
[ST_GeomFromWKB](#)



ST_MakePoint

```
INSERT INTO poi
  (name , the_geom)
VALUES (
  'Schule',
  ST_SetSRID(
    ST_MakePOINT(2568833.6,5631230.1),
    31466
  )
);
```




ST_POINT

```
INSERT INTO poi
  (name , the_geom)
VALUES (
  'Museum',
  ST_SetSRID(
    ST_POINT(2569104.4 , 5631326.4) ,
    31466
  )
);
```





WKB und WKT

WKB – Well Known Binary

WKT – Well Known Text

```
SELECT the_geom, astext(the_geom) from poi limit 1;
```

the_geom

0101000020EA7A0000000000802C99434100000000647B5541

astext

POINT(2568793 5631376)



ST_AsEWKT

```
SELECT ST_AsText(the_geom), ST_AsEWKT(the_geom)
FROM poi limit 1;
```

```
st_astext
```

```
-----
```

```
POINT(2568793 5631376)
```

```
st_asewkt
```

```
-----
```

```
SRID=31466;POINT(2568793 5631376)
```



LINESTRING

```
CREATE TABLE leitung (gid serial, name varchar);
```

```
SELECT AddGeometryColumn( 'public' , 'leitung' ,  
'the_geom' , 31466 , 'LINESTRING' , 2);
```

```
INSERT INTO leitung (name , the_geom) VALUES  
( 'Wasserleitung',  
GeometryFromText( 'LINESTRING(2568996.2 5631464 ,  
2568849 5631296)', 31466));
```



Länge ermitteln mit ST_Length

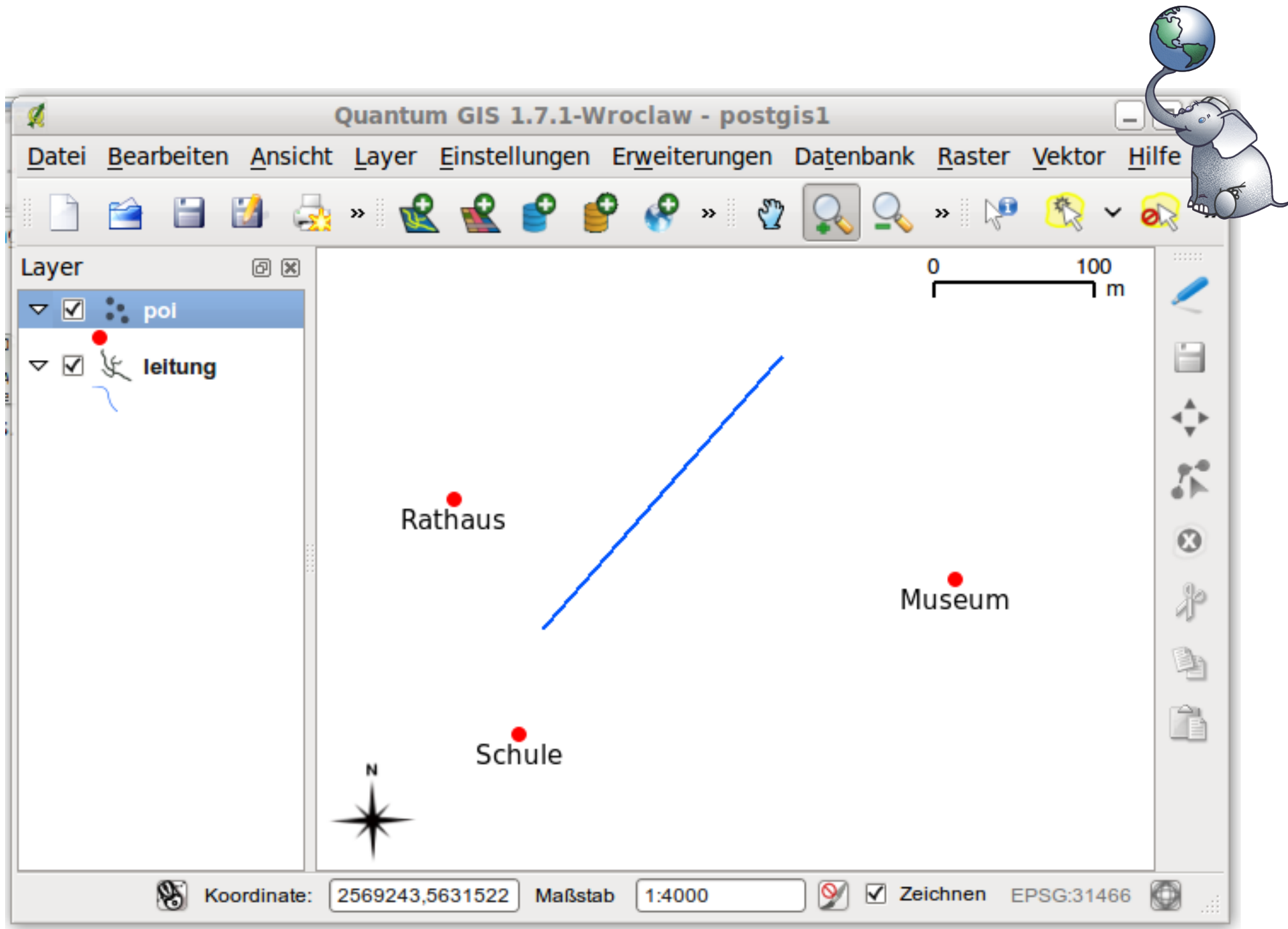
```
SELECT name ,  
       ST_Length(the_geom)  
FROM leitung;
```

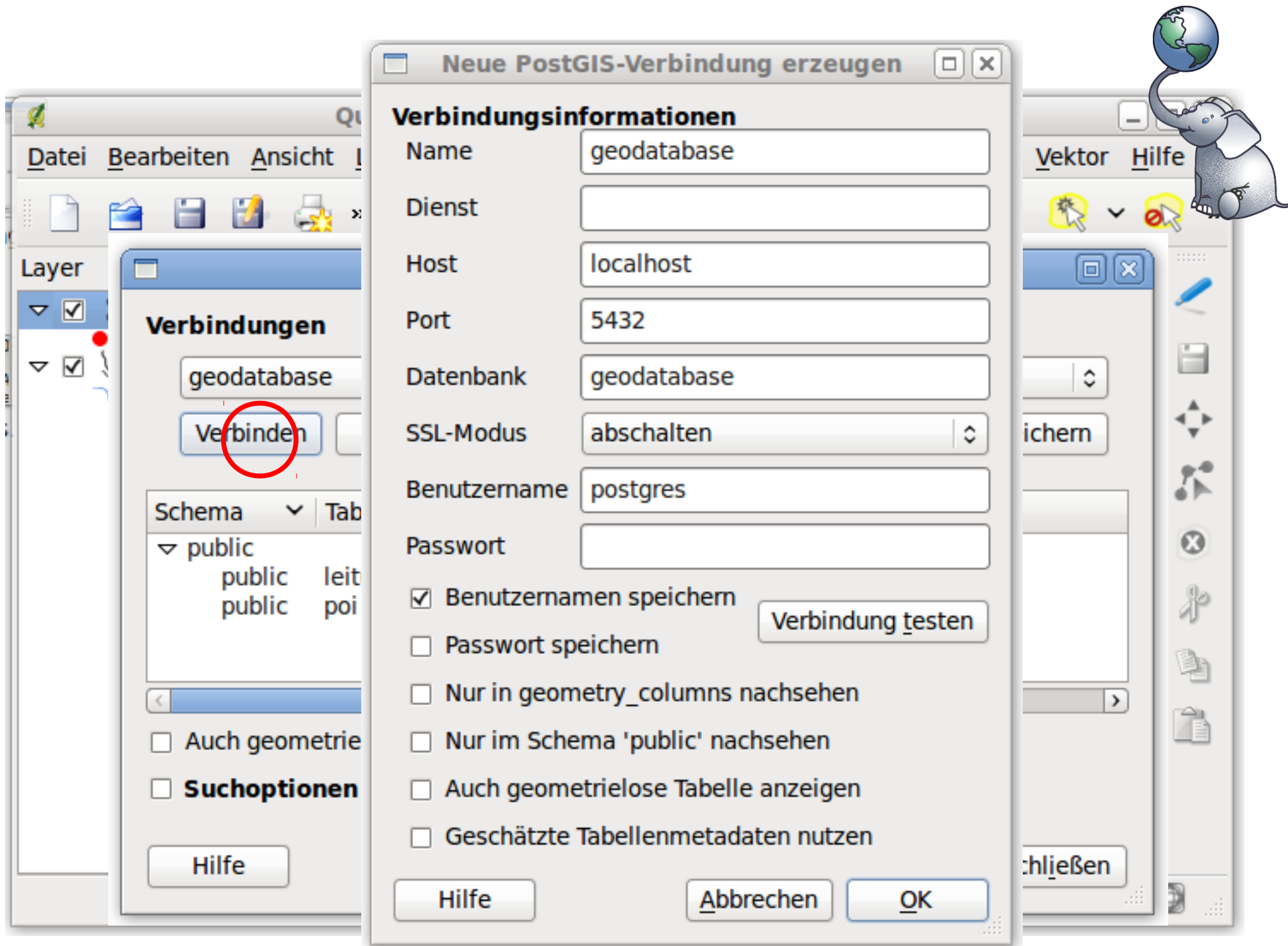
name	st_length
-----+-----	
Wasserleitung	223.364813701834



Datenvisualisierung

- Quantum GIS
- GvSIG
- uDig
- Jump
- SAGA
- Spatial Commander
- UMN MapServer
- GeoServer
- Deegree
- MapInfo 10+
- AutoCAD
- ArcGIS über SDE
- ...







QGIS PostGIS Feature

- Visualisierung
- Abfragen über den Abfragemanager
- Import von Shape nach PostgreSQL über Plugin Spit oder PostGIS Manager
- Digitalisierung
- Offline Digitalisierung
- 3D Visualisierung über Plugin
- PostGIS Raster Plugin



Datenimport

- shp2pgsql
- shp2pgsql-gui
- ogr2ogr (GDAL)
- Quantum GIS Spit Plugin
- osm2pgsql, osmosis, Imposm
- FME



shp2pgsql

```
shp2pgsql -I -s 4326 -W LATIN1  
countries.shp laender > laender.sql
```

-I Erstellt einen Geometrie-Index
-s <epsgcode> SRID wird gesetzt
-W <encoding> Encoding



shp2pgsql-gui

Shape File to PostGIS Importer

PostGIS Connection

Username: postgres

Password:

Server Host: localhost 5441

Database: postgis20_sampler

Test Connection... Connection succeeded.

Shape File

Add File

Shapefile	Schema	Table	Geometry Column	SRID	Mode	Rm
C:\projects\postgis\data\world_borders.shp	public	world_borders	geom	4326	Create	☐
C:\projects\postgis\data\place.shp	public	place	geom	4326	Create	☐

Options... Import About Cancel

Import Log

connecting to postgis20_sampler
Connection succeeded.
pgui_action_open_file_dialog called.
pgui_action_open_file_dialog called.



ogr2ogr

```
ogr2ogr -f PostgreSQL PG:"host=localhost  
dbname=geodatabase user=postres password=xxx"  
poi.shp -nln poi -a_srs EPSG:31466
```

```
ogr2ogr --formats
```

```
-f   Ausgabeformat
```

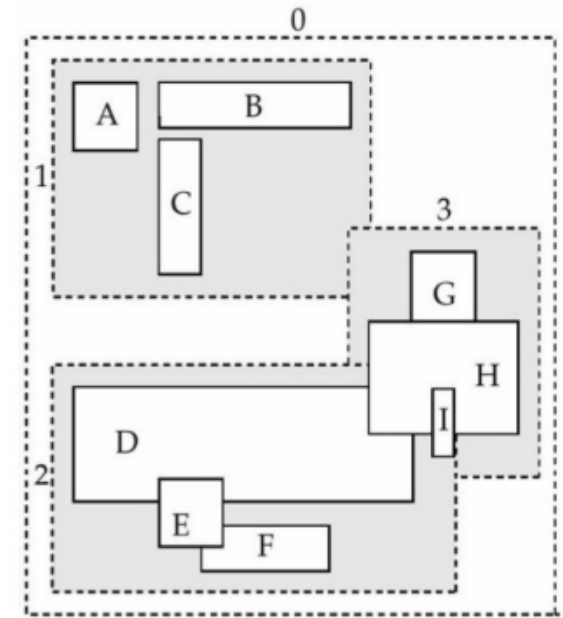
```
-nln <tabellenname> gewünschter Tabellenname
```

```
-a_srs EPSG:4326 Angabe des Referenzsystems
```



GIST Index

- Generalized Search Tree
- Schnellere räumliche Abfragen
- Nutzt BBOX



```
CREATE INDEX gix_poi
ON poi
USING GIST ( the_geom
GIST_GEOMETRY_OPS );
```

Räumliche Funktionen in PostGIS



- Generierung, Analyse, Manipulation via SQL
- Mehr als 300 räumliche Funktionen
- Operatoren (z.B. &&)
- OGC SFSQL (konform ab Version 0.9, 2005)
- ISO Spezifikation SQL/MM Teil 3
- Eigene PostGIS Funktionen



Eine Auswahl

- ST_EXTENT(geometry)
- ST_Touches(geometry,geometry)
- ST_Intersects(geometry,geometry)
- ST_Overlaps(geometry,geometry)
- ST_Union(geometry,geometry)
- ST_ExteriorRing(polygon)
- ST_Centroid(geometry)
- ST_IsValid(geometry)

<http://postgis.org/documentation/>



Leitungsverlegung benötigt eine 20 m breite Baustelle



Puffern mit ST_Buffer

```
Create View st_buffer_10 AS  
SELECT oid, gid,  
       ST_Buffer(the_geom, 10)  
       AS the_geom  
FROM leitung;
```

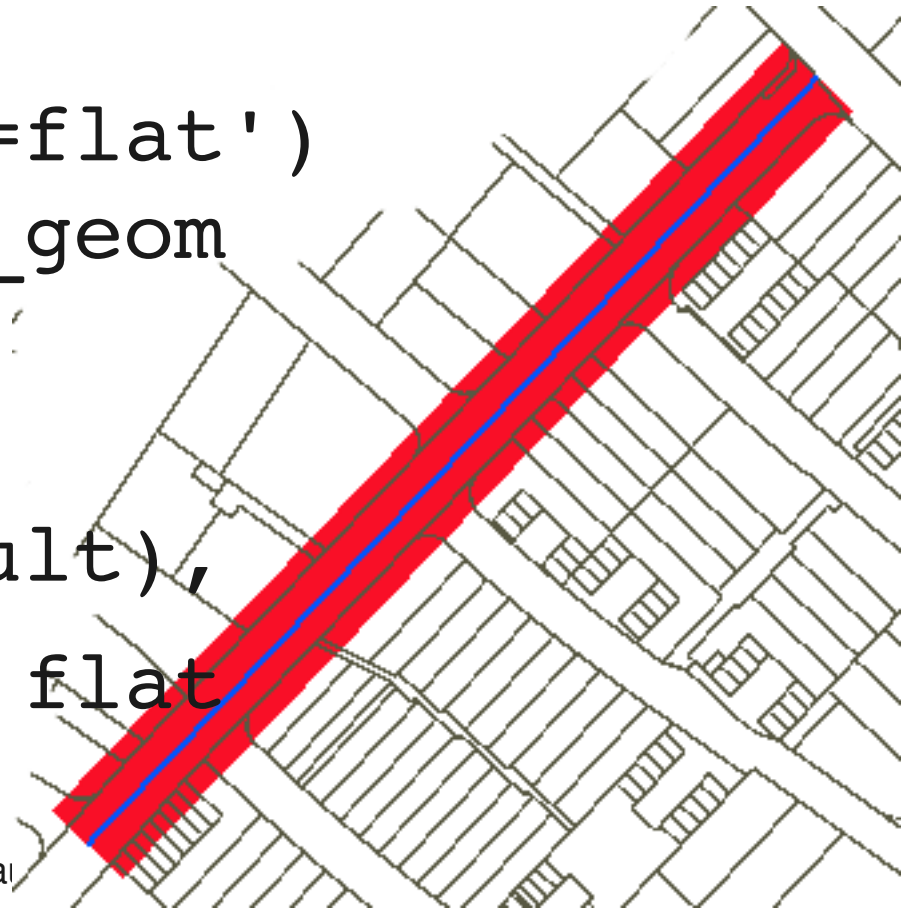




ST_Buffer mit endcap=flat

```
Create View st_buffer_10 AS
SELECT oid, gid,
       ST_Buffer(the_geom,
                 10 ,
                 'endcap=flat')
       AS the_geom
FROM leitung;
```

endcap – round (default),
square oder flat





Wie groß ist die Fläche?



Fläche berechnen mit ST_Area

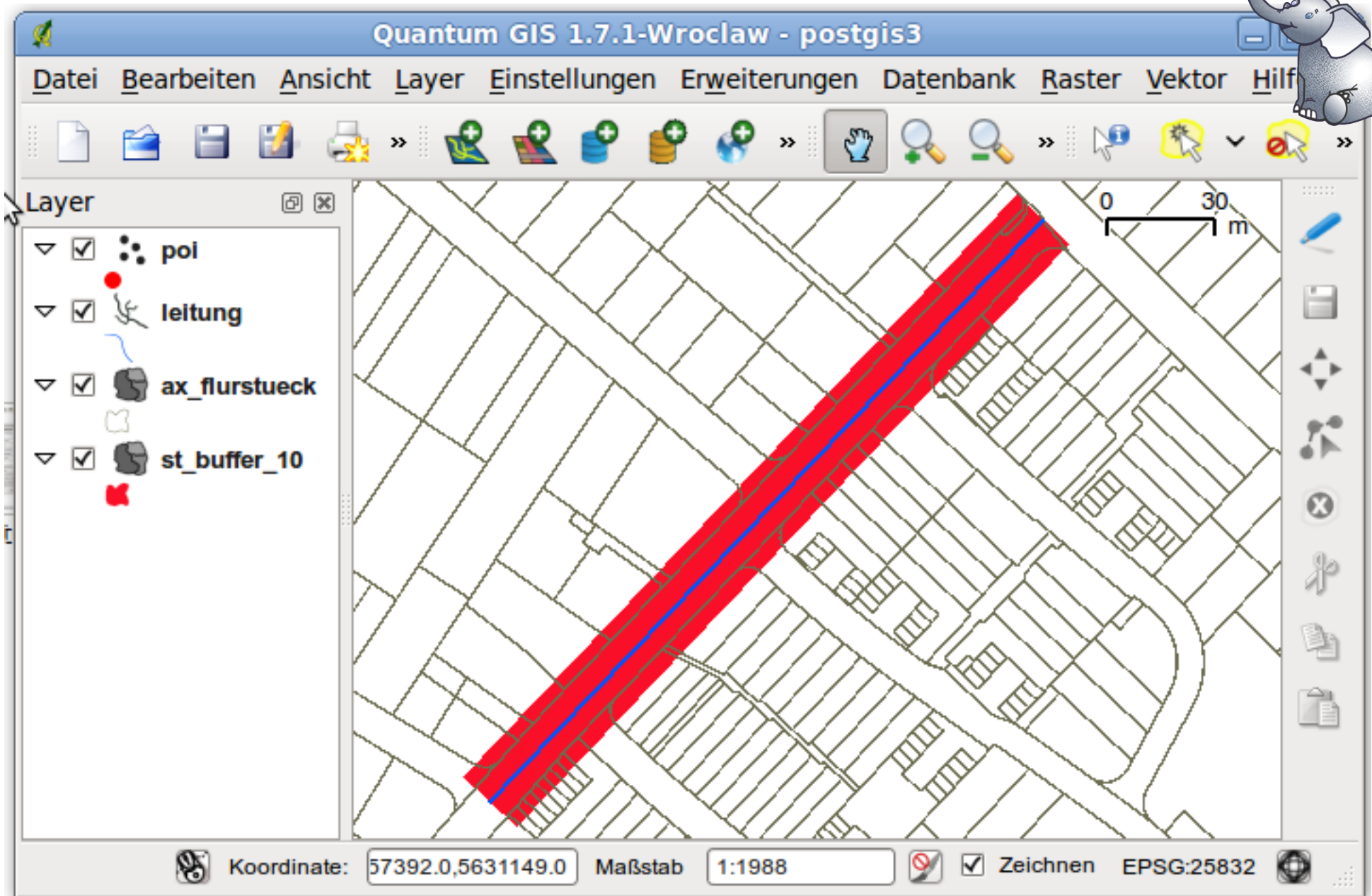
```
SELECT ST_Area(  
    ST_Buffer(the_geom,  
              10 ,  
              'endcap=flat')  
    ) AS flaeche  
FROM leitung;
```

flaeche

4467.29627418518



Welche Flurstücke sind betroffen?





ST_Intersect

```
SELECT
```

```
flur
```

```
FROM
```

```
WHER
```

```
ST_I
```

FEHLER

ung

```
    ST_Buffer(the_geom,10 , 'en  
dcap=flat' ),  
    wkb_geometry);
```




FEHLER: Operation on two geometries with different SRIDs

**KONTEXT: SQL-Funktion
»st_intersects« Anweisung 1**



ST_SRID

```
SELECT ST_SRID(wkb_geometry)
from ax_flurstueck limit 1;
```

st_srid

25832



ST_Transform

```
SELECT flurstueckskennzeichen
FROM ax_flurstueck, leitung
WHERE
  ST_Intersects(
    ST_Buffer(
      ST_Transform(the_geom, 25832)
      , 10 , 'endcap=flat' ),
    wkb_geometry
  ) ;
```



Quantum GIS 1.7.1-Wroclaw - postgis

Datei Bearbeiten Ansicht Layer Einstellungen Erweiterungen Datenbank Raster Vektor Hilfe

Layer

srid	auth_name	auth_srid	srtxt	proj4text
25832	EPSG	25832	PROJCS["ET...]	+proj=utm +zone=32 +ellps=GRS80 +units=m +no_defs
31466	EPSG	31466	PROJCS["DH...]	+proj=tmerc +lat_0=0 +lon_0=6 +k=1 +x_0=2500000 +v_0=0 +ellps=bessel +datum=notsdam +units=m +no_defs +towgs84=582,105,414,1.04,0.35,-3.08,8.3

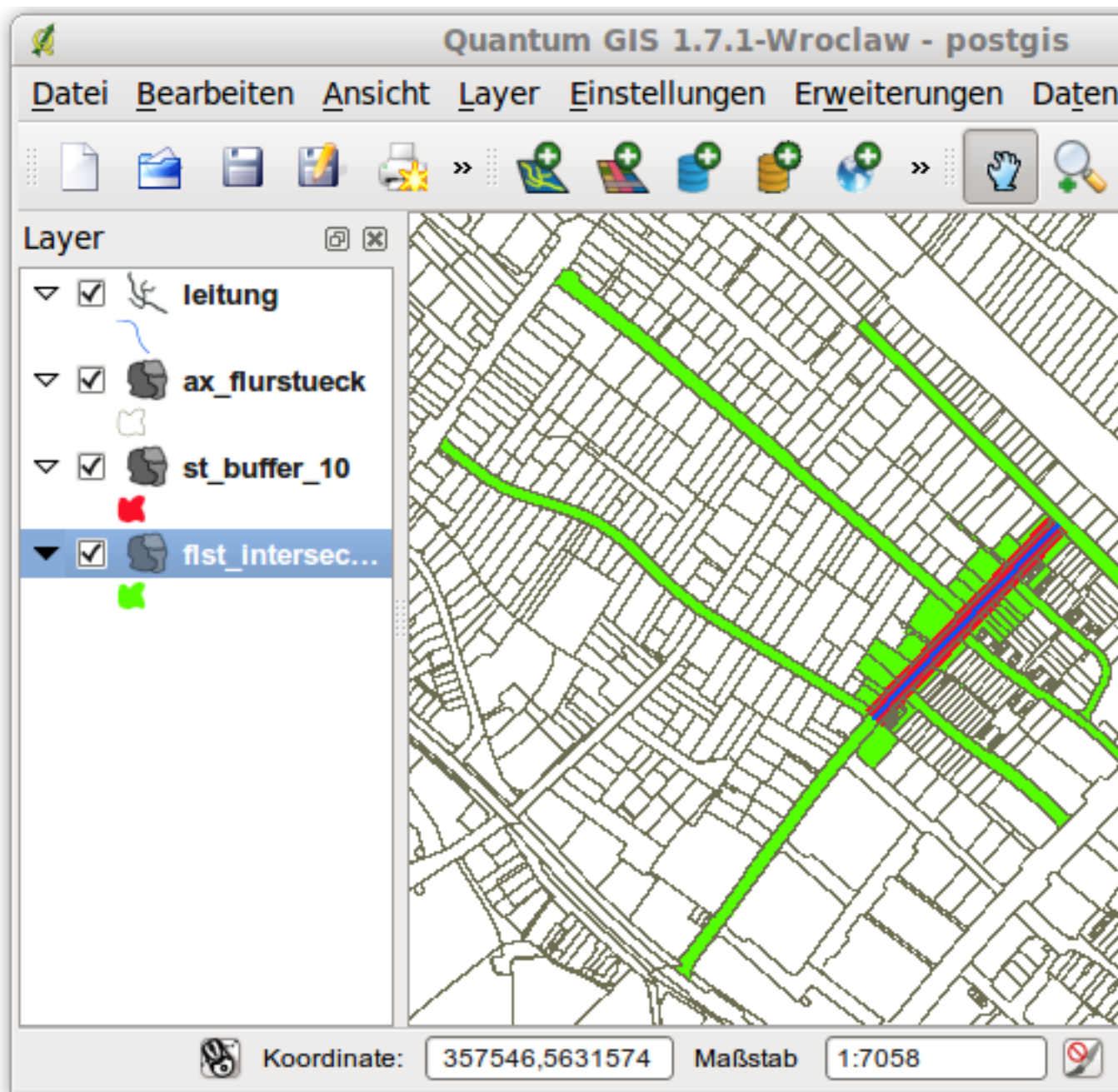
st_buffer_10

flst_intersec...

?

Transformationsparameter prüfen!

Koordinate: 357720,5631386 Maßstab 1:7058 Zeichnen EPSG:25832



flurstueckskennzeichen

05469701600171
05469701600186
05469901701661
05469901703924
05469901701658
05469901704653
05469901701660
05469901701659
05469901701663
05469901701662
05469701600199
05469701600189
05469700700359
05469701600241
05469701600242
05469901701664
05469901701666

...
(42 Zeilen)



ST_Distance

```
select p.name,  
distance(p.the_geom, ST_Buffer(l.the_geom, 10  
, 'endcap=flat')) AS entfernung  
FROM poi p, leitung l;
```

name		entfernung
-----+-----		
Rathaus		84.8403629422498
Schule		63.5849795875077
Museum		162.06076177808



Auf geht's!

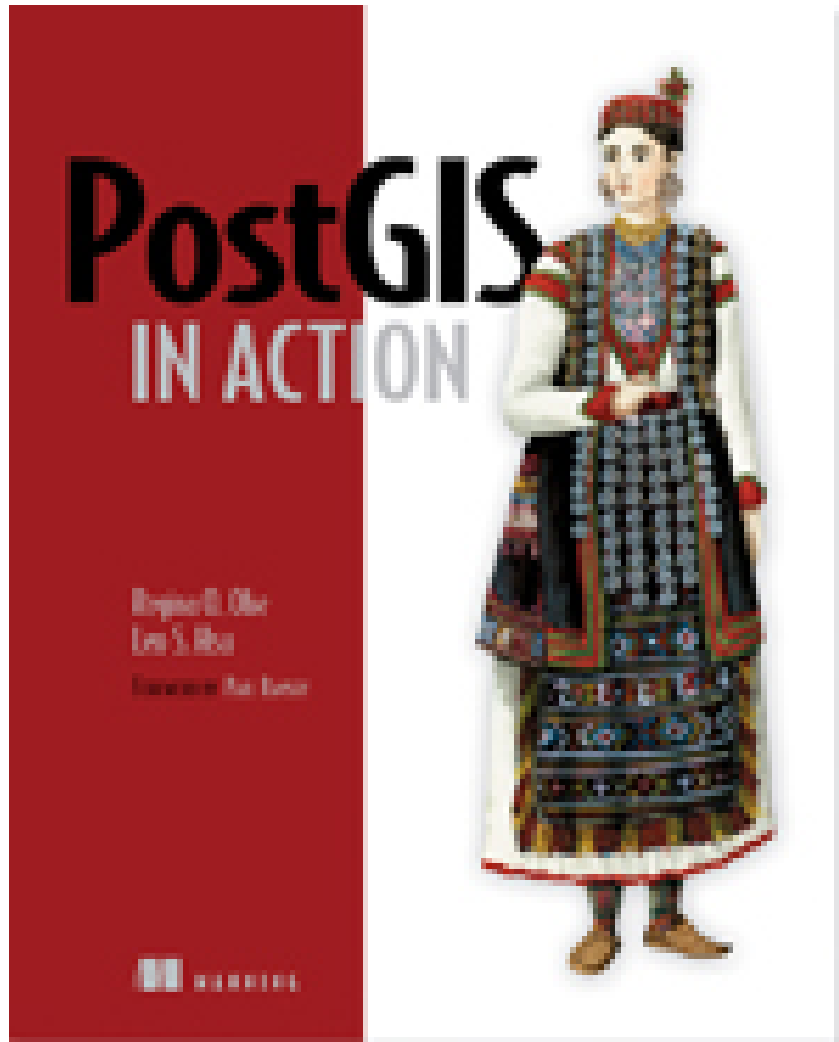


PostGIS Dokumentation

- sehr gute und ausführliche Dokumentation mit SQL-Beispielen und Grafiken
- HTML oder PDF-Version
- <http://postgis.org/documentation/>
- **PostGIS Wiki** (Tutorials, Präsentationen, Videos)



PostGIS in Action



<http://www.manning.com/obe/>

Regina O. Obe und Leo S.
Hsu

Vorwort Paul Ramsey

April 2011, 520 Seiten

ISBN 9781935182269

PostGIS auf



- <http://live.osgeo.org>
- GIS Software Kollektion
- > 40 GeoSpatial Open Source Anwendungen
- Beispieldaten
- Dokumentationen
- basiert auf Xubuntu
- bootfähige DVD, USB-Stick oder virtuelle Maschine
- iso zum Download unter <http://live.osgeo.org/de/download.html>



Vielen Dank !

Fragen?

Feedback

<https://www.postgresql.eu/events/feedback/pgconfde2011/>

Astrid Emde



WhereGroup, Bonn

astrid.emde@wherogroup.com