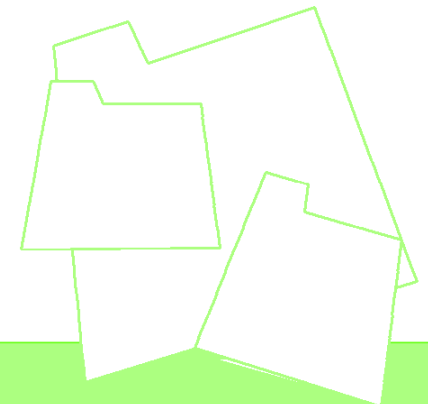


Skript basiertes geoprozessieren

am Beispiel einer

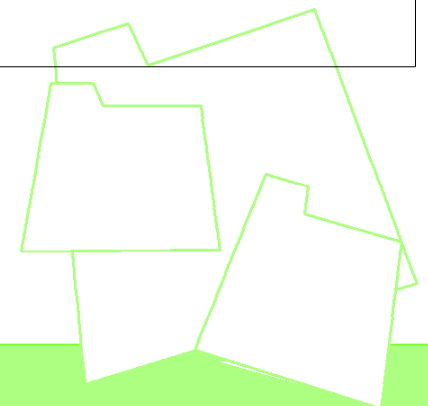
Flächenermittlung von Grundwasserverunreinigungen aus
Punktquellen

mit einem Datenbankmanagementsystem
(PostgreSQL/PostGIS)



Summe aus Einzelflächen ermitteln

| | | |
|----------|---|--|
| |  | Fahnenkartierungen |
| + |  | Altlastflächen |
| + |  | Flächenbildung aus einzelnen Messstellen |
| Σ |  | Gesamtfläche |



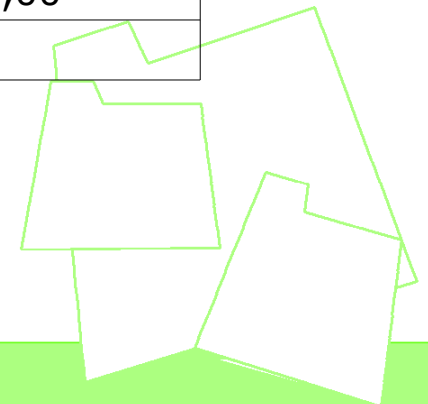
Datenstruktur und Vorlagen:

Liste mit ca. 900.000 Datensätzen (Analysen)

| BNR | Prodat | Ifd. Nr. | Labor | Stoffgruppe | Stoffkürzel | Wert | Merkmal | Anzahl Nachkommast. |
|------------|---------------|-----------------|--------------|--------------------|--------------------|----------------------|----------------|----------------------------|
| DD01 | 16.08.2000 | 0 | 51 | 4 | PER | ,5000000000000000E-3 | | 4 |
| DD01 | 16.08.2000 | 0 | 51 | 4 | TETRACLME | 0,10 | < | 1 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... |

Liste mit ca. 9.000 Datensätzen (Raumdaten)

| BNR | KOR | KOH |
|------------|------------|------------|
| 00036 | 2556732,00 | 5679287,00 |
| 100218 | 2552887,00 | 5680343,00 |
| ... | ... | ... |



Geometrie "Simple Features for SQL" erzeugen

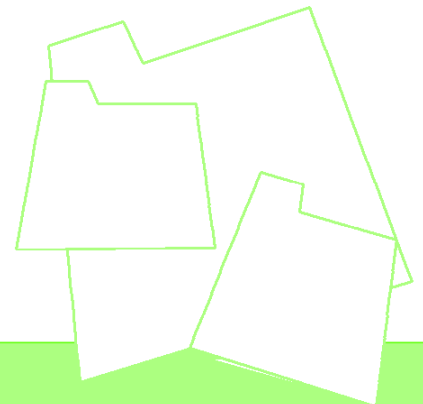
```
UPDATE messstellen SET the_geom = SetSRID(ST_MakePoint(x, y), 31466);
```

Lage auf der Referenzfläche untersuchen

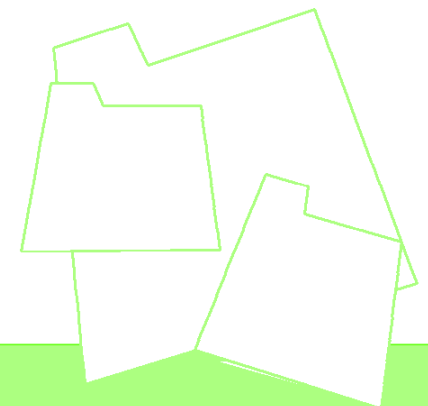
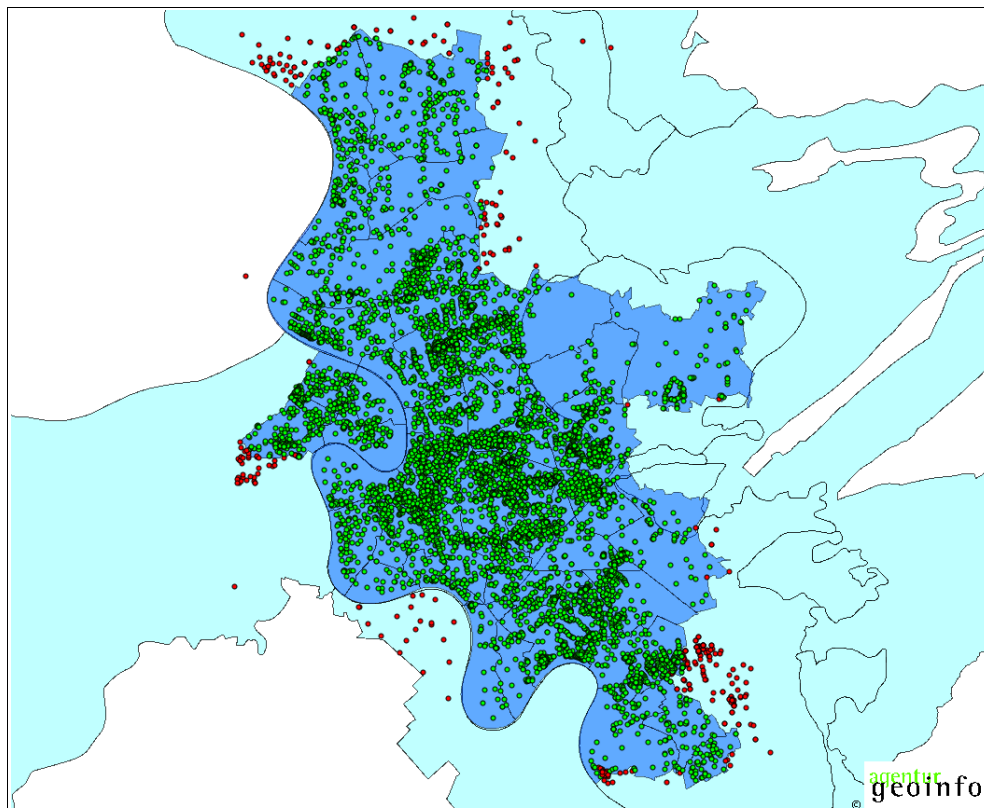
```
UPDATE messstellen c SET in_dssd = TRUE WHERE
  (SELECT ST_Within(a.the_geom, ST_Union(b.the_geom))
   FROM messstellen a, die_stadt b
   GROUP BY a.gid, a.the_geom
   HAVING c.gid = a.gid );
```

Bezug zum Grundwasserkörper bestimmen

```
UPDATE messstellen b SET tkoerper =
  (SELECT tkoerper FROM gwkoerper a WHERE
   (SELECT ST_Within(b.the_geom, a.the_geom))
   AND a.tkoerper = 'der_eine')
WHERE b.tkoerper IS NULL;
```

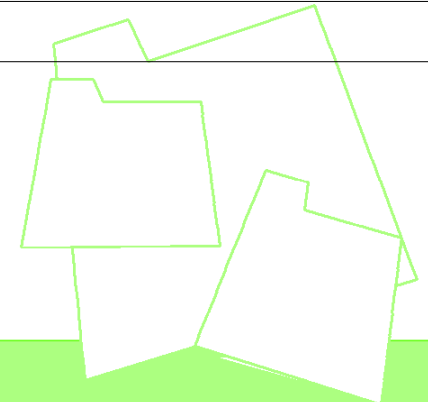


| gid | bnr | X | Y | in_town | tkoerper |
|------|--------|------------|------------|---------|----------|
| 97 | DU108 | 2549610 | 5689758 | f | 27_10 |
| 2768 | 100546 | 2552658.92 | 5678722.62 | t | 27_18 |
| ... | ... | ... | ... | ... | ... |



Grundwasseranalysen gruppieren

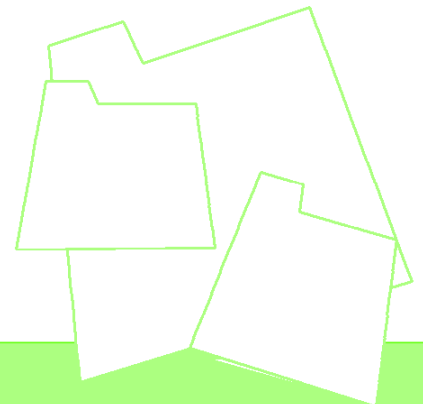
| Fortlaufende Nr. | Neue Stoffkürzel | aufsummierende Stoffe |
|------------------|-------------------|--|
| 1 | Summe PAK | PAK1, PAK2, PAK3, PAK4, PAK5, PAK6, PAK8, PAK9, PAK10, PAK11, PAK12, PAK13, PAK14, PAK15, PAK16, PAK17, PAK18, PAK19, PAK20, PAK21, PAK22, PAK23, PAK24, PAK25 |
| 2 | Summe Naphthaline | PAK7, PAK7A, PAK7B, PAK7C, PAK7D |
| 3 | Summe BTX | BENZ, TOL, XYL, EB, CUMOL, STYR, ETOL, HEM, MES, MPTOL, MPXYL, OETOL, OXYL, PSEUDOC, STYR |
| 4 | Summe LHKW | BROMDICHLOR, CISDI, CLDIM, DIBROMCHLOR, DIBROME, DICLE11, DICLE12, DICLM, DICLME, DI11, FRIGEN 11, FRIGEN 113, MONOETHAN, PCE, TCLE1112, TCLE1122, TETRACLME, TRANSDI, TRIBROM, TRICLE111, TRICLE112, TRICLFLE112, TRICLME, TRICLP, TRIFL112, VC, TRI, PER, VC |
| ... | ... | ... |



```

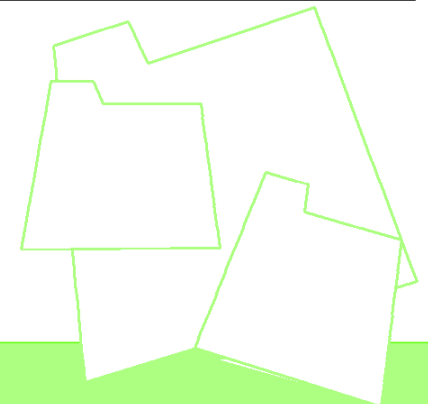
CREATE OR REPLACE VIEW sum_lhkw AS
  SELECT bnr, prodat::date, lfd_nr, labor, stoffgruppe, 'sum_LHKW'::varchar AS stoffkuerzel,
    sum((wert)::numeric)::varchar AS wert,
    (max(anzahl_nachkomma)::int2)::varchar AS anzahl_nachkomma
  FROM analy_00_07_step1
  WHERE (stoffkuerzel = 'BROMDICHLOR'
    OR stoffkuerzel = 'CISDI'
    OR stoffkuerzel = 'CLDIM'
    OR ...
    OR stoffkuerzel = 'VC')
  AND bnr = bnr
  AND prodat = prodat
  AND lfd_nr = lfd_nr
  AND labor = labor
  AND stoffgruppe = stoffgruppe
GROUP BY stoffgruppe, labor, lfd_nr, prodat, bnr
ORDER BY bnr, prodat, lfd_nr, labor
;

```



Analysen mit Geringfügigkeitsschwellenwert verknüpfen Maximalwert je Messstelle und Stoff separieren

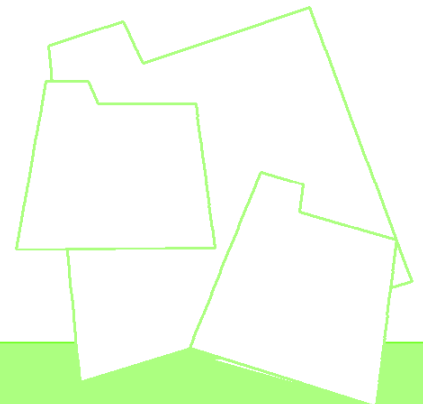
| Fortlaufende Nr. | Stoffkürzel | GFS |
|------------------|-----------------|-----|
| 1 | sum_pak | 0.2 |
| 2 | sum_naphthaline | 1 |
| 3 | sum_btx | 20 |
| 4 | sum_lhkw | 20 |
| ... | ... | ... |
| 31 | VC | 0.5 |
| 32 | MTBE | 15 |
| 33 | PESTALL | 0.5 |
| ... | ... | ... |



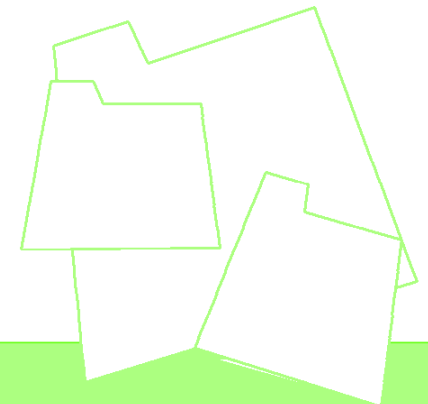
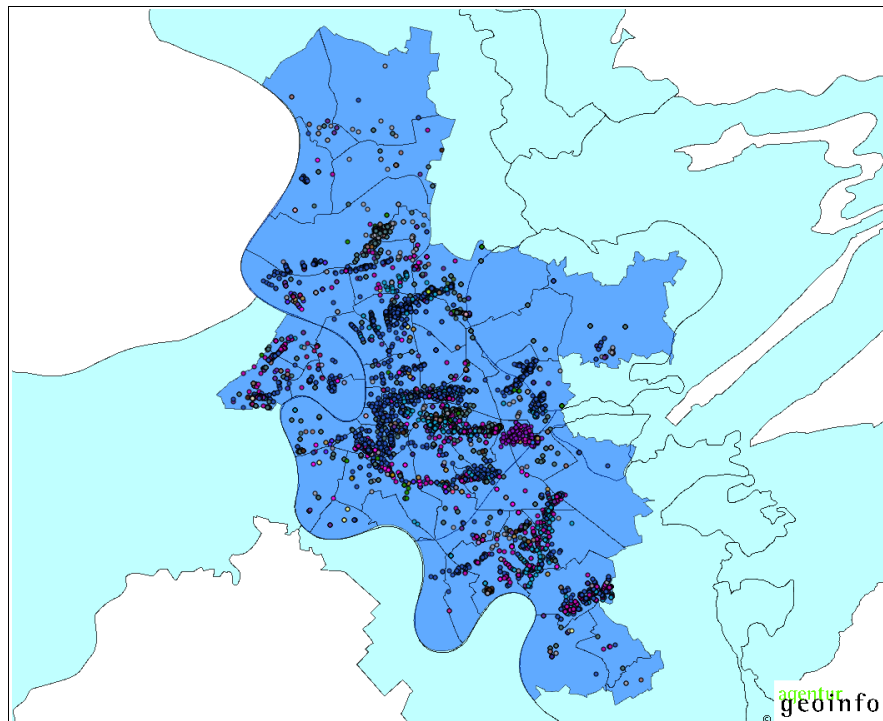

```

INSERT INTO die_tabelle (bnr, prodat, lfd_nr, labor, stoffgruppe, stoffkuerzel, wert, gid, the_geom)
  (SELECT a.bnr, a.prodat, a.lfd_nr, a.labor, a.stoffgruppe, a.stoffkuerzel,
        a.wert, c.gid, c.the_geom
   FROM analy_00_07_step1 a JOIN messstelle c ON a.bnr = c.bnr, analy_00_07_step1 b
   WHERE a.bnr = b.bnr AND a.stoffgruppe = b.stoffgruppe
        AND a.stoffkuerzel = b.stoffkuerzel
        AND ((a.stoffkuerzel = 'AS' AND a.wert::numeric >= 10)
            OR (a.stoffkuerzel = 'PB' AND a.wert::numeric >= 7)
            OR (a.stoffkuerzel = 'CD' AND a.wert::numeric >= 0.5)
            OR (a.stoffkuerzel = 'CR' AND a.wert::numeric >= 7)
            OR ...
            OR (a.stoffkuerzel = 'SPEST9' AND a.wert::numeric >= 0.07))
   GROUP BY a.bnr, a.wert, a.lfd_nr, a.prodat, a.labor, a.stoffgruppe,
            a.stoffkuerzel, c.gid, c.the_geom
   HAVING a.wert::numeric = max(b.wert::numeric)
   ORDER BY a.bnr)
AS foo
;

```



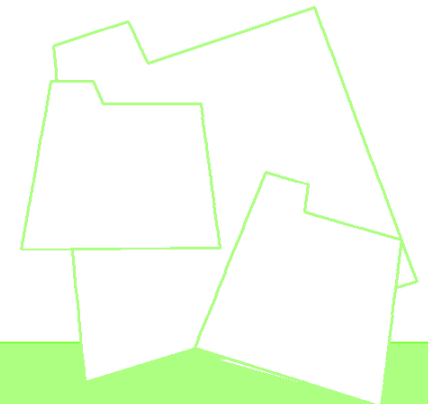
| bnr | prodat | lfd_nr | labor | stoffgruppe | stoffkuerzel | wert | tkoerper |
|-------|------------|--------|-------|-------------|--------------|------|----------|
| DU606 | 24.11.2000 | 0 | 51 | 1 | SO4 | 250 | 27_10 |
| DU606 | 23.05.2000 | 0 | 57 | 1 | PAK12 | 0.01 | 27_10 |
| ... | ... | ... | ... | ... | ... | ... | ... |



ausfiltern der Messstellen innerhalb von GW-Fahnen und Altlastflächen

```
DELETE FROM stoffe_typ_a c WHERE
  (SELECT ST_Within(a.the_geom, ST_Union(ST_Buffer(b.the_geom, 30, 15))) FROM
    stoffe_typ_a a, gw_fahnen b
  GROUP BY a.gid, a.the_geom
  HAVING c.gid = a.gid);
DELETE FROM stoffe_typ_b c WHERE
  (SELECT ST_Within(a.the_geom, ST_Union(ST_Buffer(b.the_geom, 30, 15))) FROM
    stoffe_typ_b a, gw_fahnen b
  GROUP BY a.gid, a.the_geom
  HAVING c.gid = a.gid);
DELETE FROM stoffe_typ_c c WHERE
  (SELECT ST_Within(a.the_geom, ST_Union(ST_Buffer(b.the_geom, 30, 15))) FROM
    stoffe_typ_c a, gw_fahnen b
  GROUP BY a.gid, a.the_geom
  HAVING c.gid = a.gid);
```

...

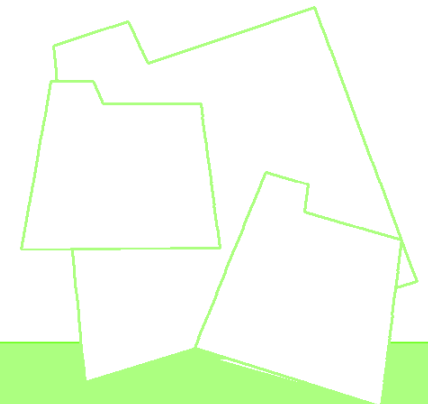


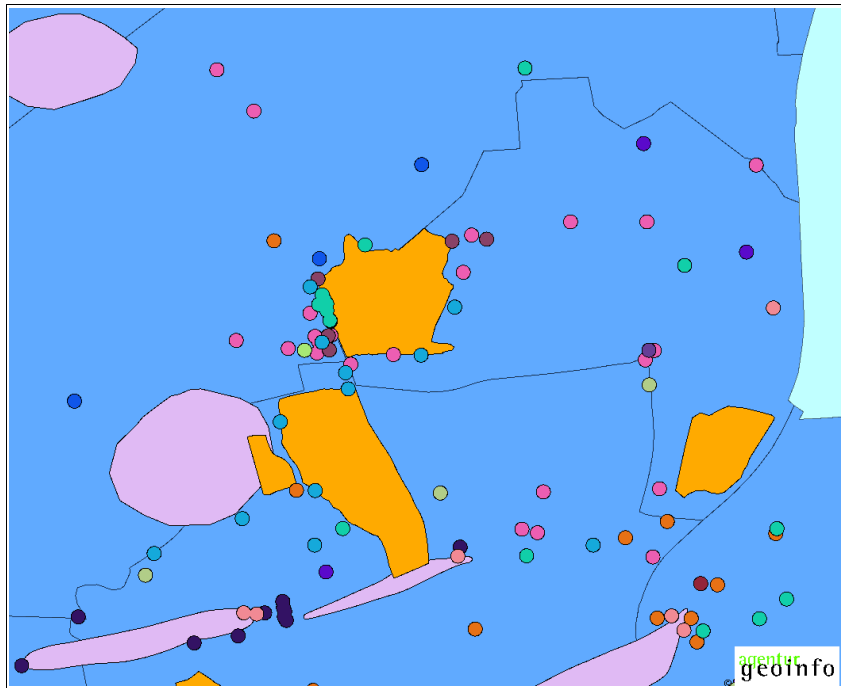
pauschale Flächenbildung auf den Messstellen

```

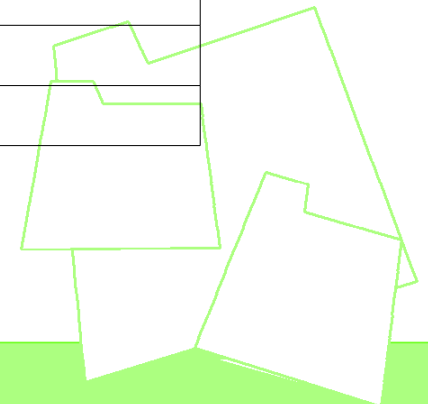
INSERT INTO die_box (stoffgruppe, stoffkuerzel, the_geom)
  (SELECT stoffgruppe, stoffkuerzel, ST_Multi(ST_Union(ST_Buffer(the_geom, 30, 15))) FROM
    stoffe_typ_a
  WHERE stoffkuerzel = 'PESTALL'
  GROUP BY stoffgruppe, stoffkuerzel);
INSERT INTO die_box (stoffgruppe, stoffkuerzel, the_geom)
  (SELECT stoffgruppe, stoffkuerzel, ST_Union(ST_Buffer(the_geom, 30, 15)) FROM
    stoffe_typ_b
  WHERE stoffkuerzel = 'AS'
  GROUP BY stoffgruppe, stoffkuerzel);
INSERT INTO die_box (stoffgruppe, stoffkuerzel, the_geom)
  ...

```



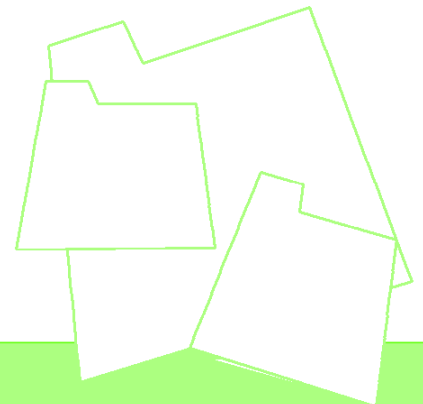


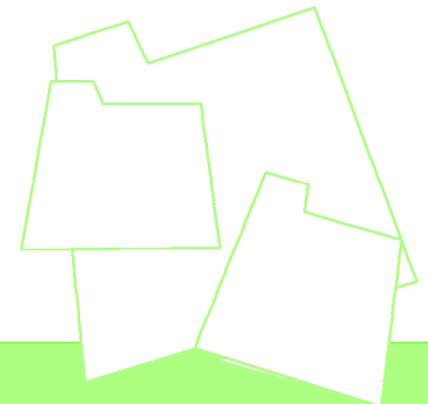
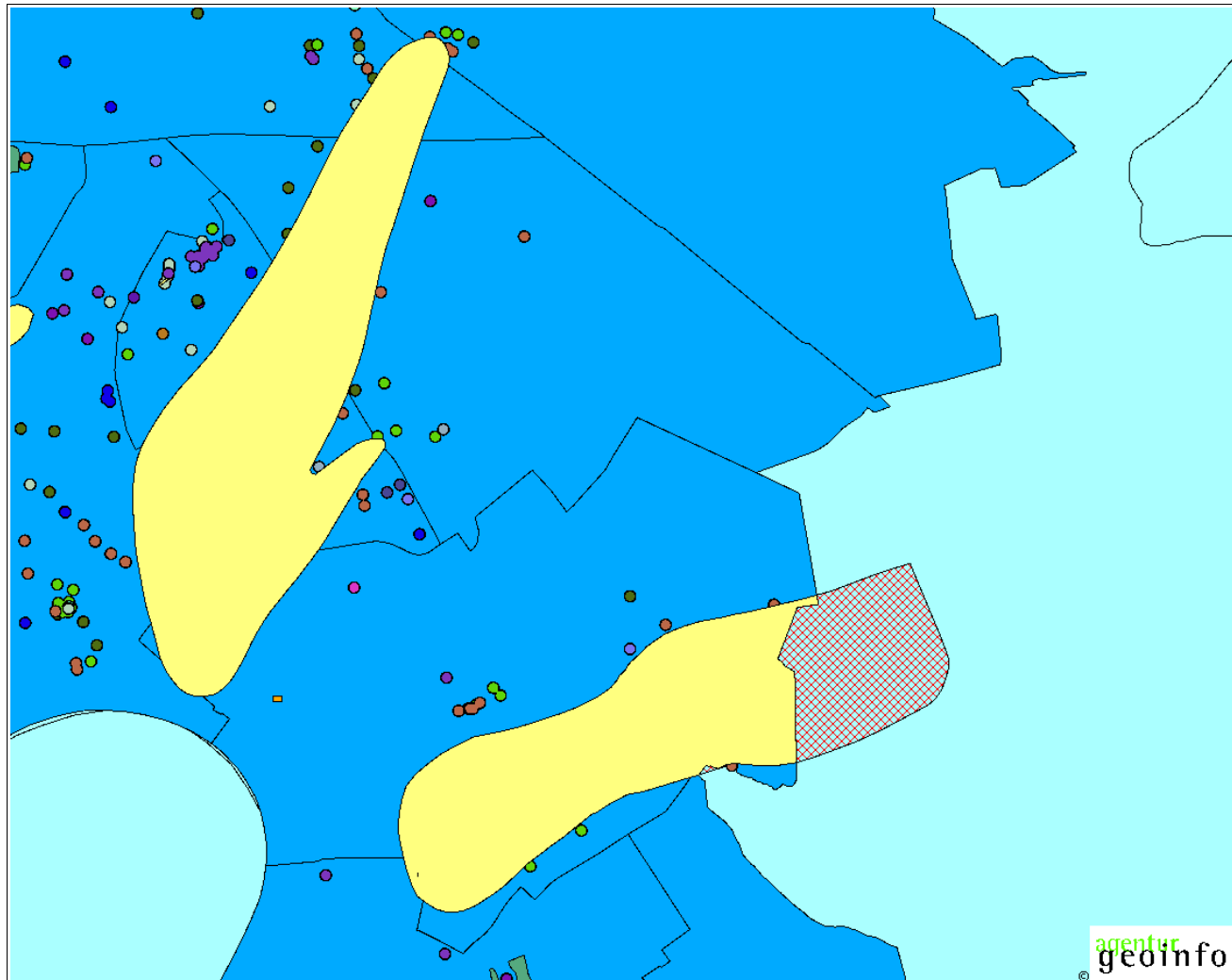
| GID | stoffgruppe | stoffkuerzel |
|------------|--------------------|---------------------|
| 1 | 6 | PAK6 |
| 2 | 0 | alle_as |
| ... | ... | ... |



Flächenüberhang abschneiden

```
UPDATE die_box SET the_geom =  
  (SELECT ST_Multi(ST_Intersection(ST_Union(a.the_geom), ST_Union(b.the_geom)))  
    FROM stadtteile a, gw_fahnen b  
   GROUP BY b.the_geom  
  )  
;
```





Aggregat über die Flächen bilden

Messstellen

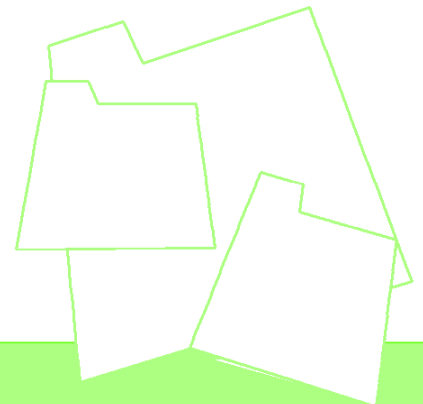
```
INSERT INTO die_box (stoffgruppe, stoffkuerzel, the_geom)
VALUES ('0', 'alle_BNR_Wolken',
(SELECT ST_Union(the_geom) FROM die_box WHERE stoffgruppe::int > 0));
```

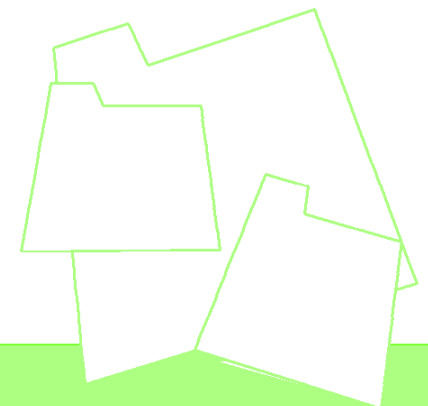
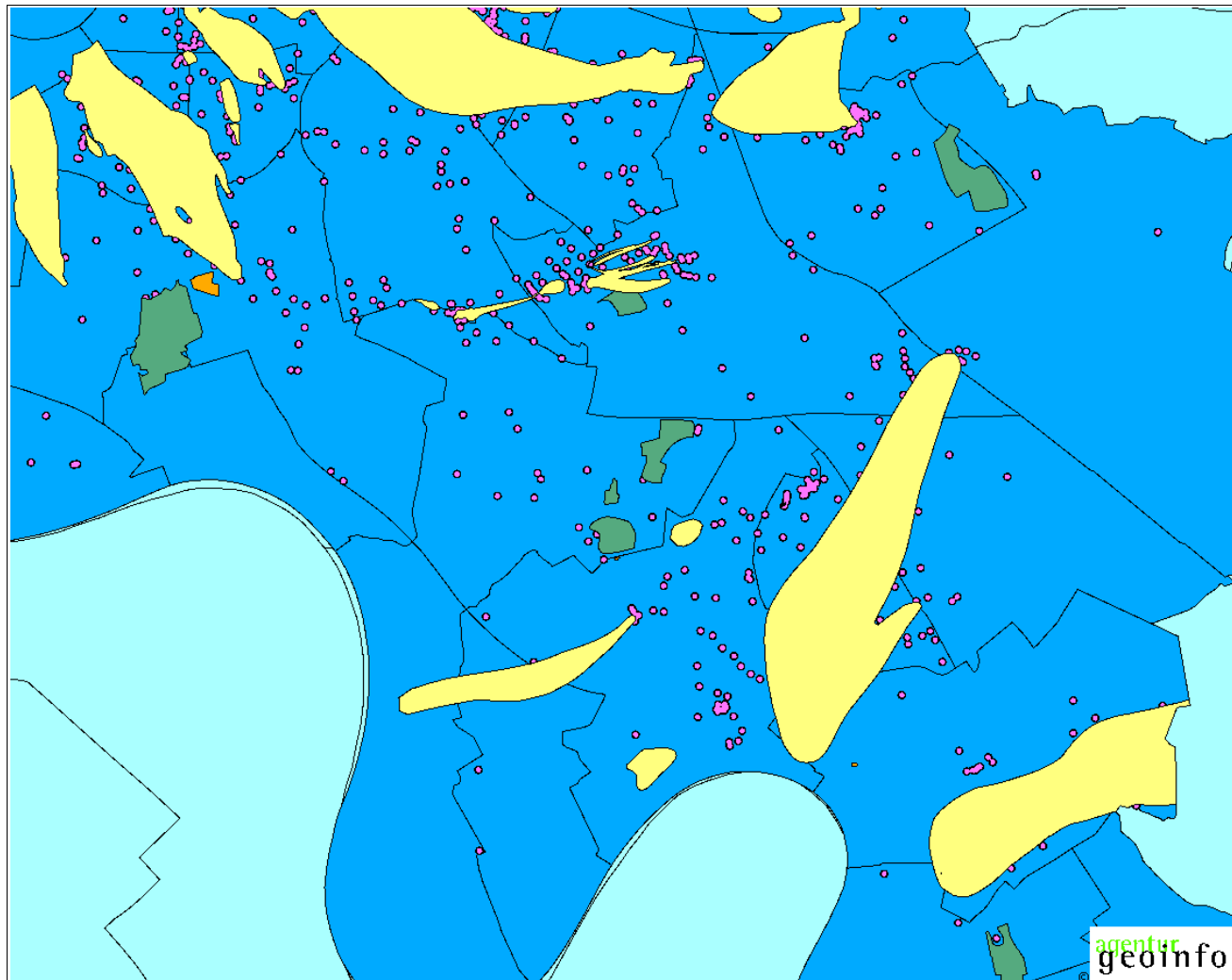
GW-Fahnen

```
INSERT INTO die_box (stoffgruppe, stoffkuerzel, the_geom)
VALUES ('0', 'alle_Fahnen',
(SELECT ST_Union(the_geom) FROM gw_fahnen));
```

Altlastflächen

```
INSERT INTO die_box (stoffgruppe, stoffkuerzel, the_geom)
VALUES ('0', 'alle_as',
(SELECT ST_Union(the_geom) FROM as_altlasten));
```



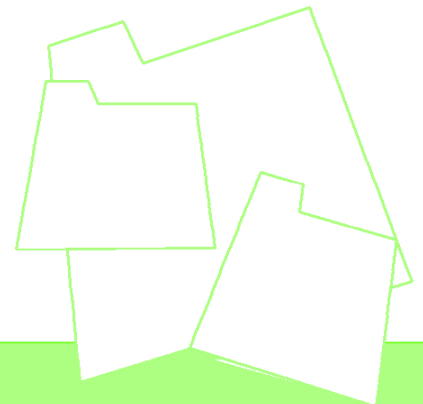


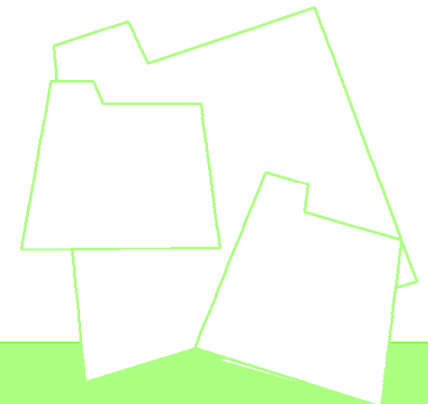
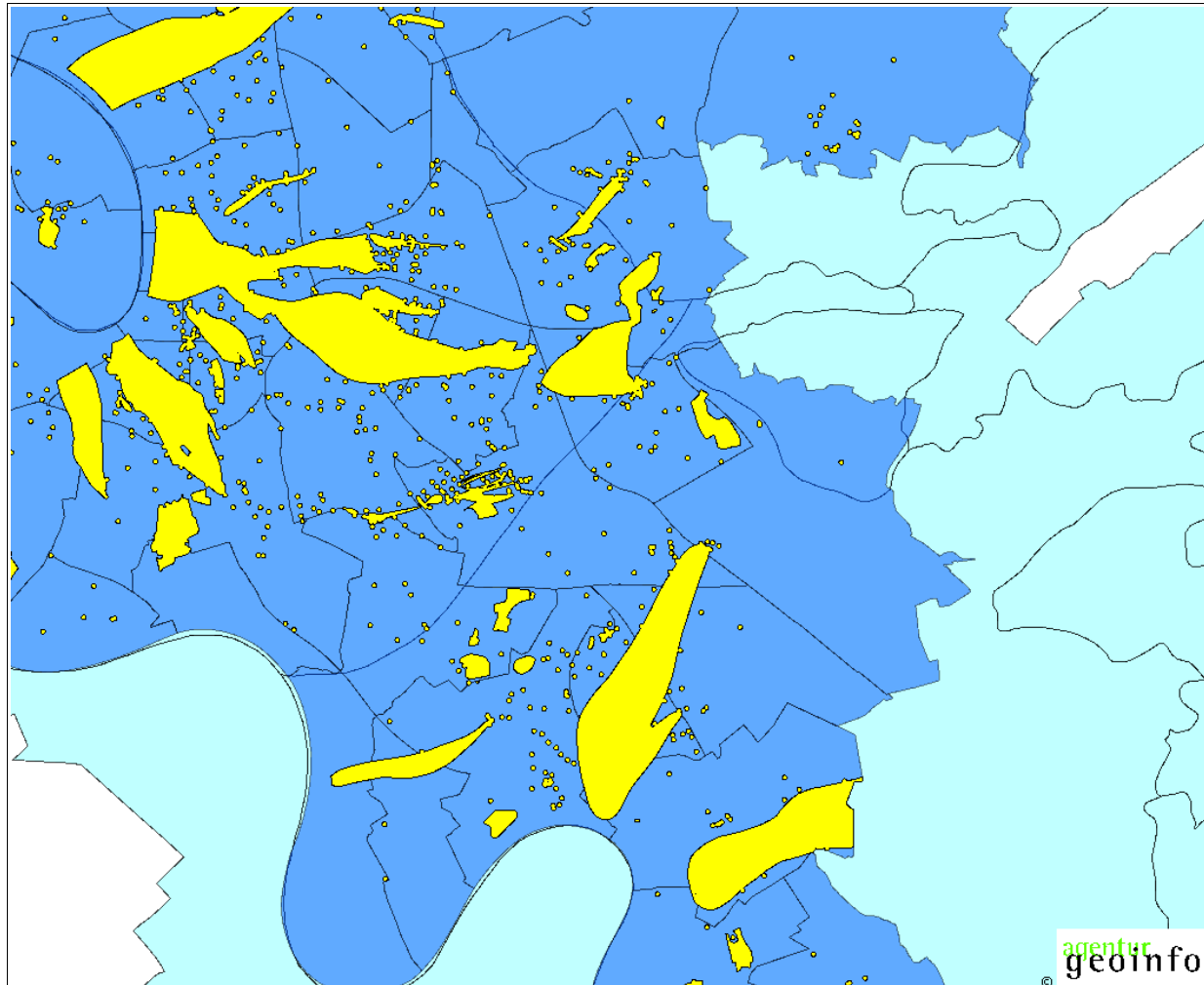
21. September 2008

< harald.schwenk@agentur-geoinfo.de >

Aggregat über alle Teilflächen (GW-Fahnen + Altlastflächen + Messstellen)

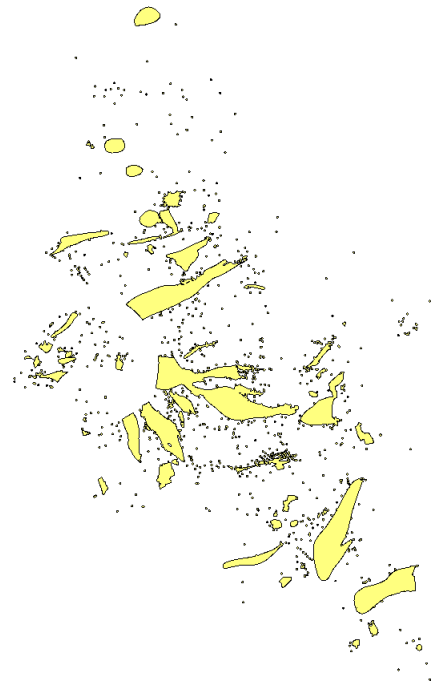
```
INSERT INTO die_box (stoffgruppe, stoffkuerzel, the_geom)
VALUES ('0', 'alles',
(SELECT ST_Union(the_geom) FROM die_box
WHERE stoffgruppe::int = 0
AND stoffkuerzel <> 'alle_Fahnen'
AND stoffkuerzel <> 'alles'
)
);
```



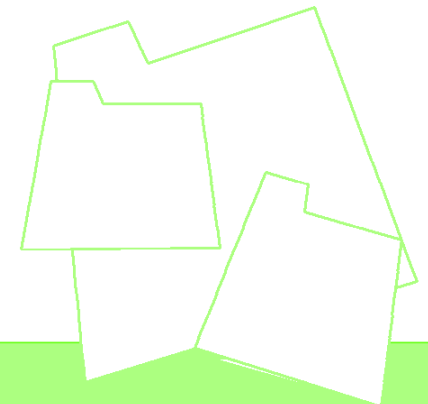


die Fläche

```
SELECT ROUND((ST_Area(the_geom) / 1000000)::numeric, 2) || ' km2'
FROM die_box
WHERE stoffkuerzel = 'alles'
;
```



agentur
geoinfo

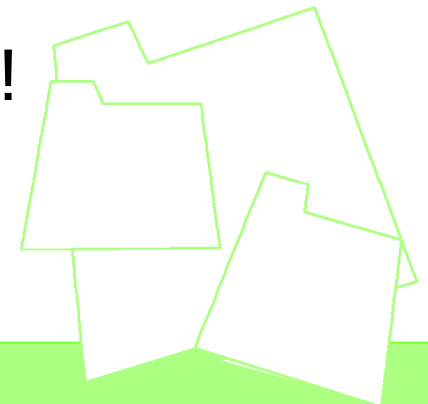


die Aufgabenstellung ist hier Beispielhaft

andere Fragestellungen sind ohne weiteres möglich

- räumliche Konzentrationen?
 - Verursacher lokalisieren?
- ...
- sozialräumliche Fragestellungen!
- wirtschaftliche Fragestellungen!

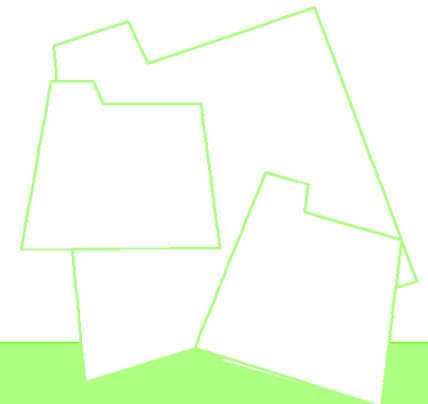
- ...



der Dateninput ist hoch flexibel

alles was in das Datenbankmanagementsystem rein geht und einen logisch Zusammenhang hat, ist prozessierbar

das gilt sowohl für die Geometrien als auch die Fachdaten

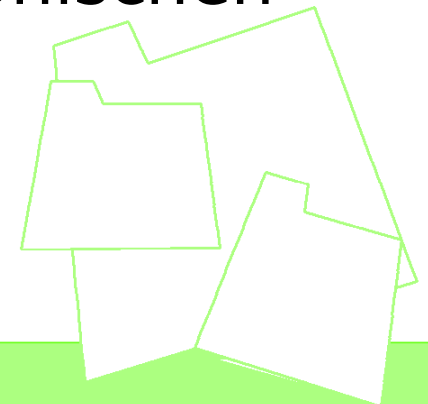


der gesamte Funktionsumfang zum Prozessieren stellt das
Datenbankmanagementsystem bereit

der Funktionsumfang ist erweiterbar

ein solcher Prozess kann voll/teil automatisiert geführt werden

der Prozess kann auch hinter einer graphischen
Nutzeroberfläche laufen



Vielen Dank für Ihre Aufmerksamkeit!

