



FOSSGIS 2018

ALKIS mit Spatialite



GKG-ALKIS-Konverter - ein Skript-Konverter zur Aufbereitung von **NAS-XML** für **QGIS** auf Basis von **ogr2ogr** und **Spatialite**

The screenshot displays the QGIS interface with a cadastral map. A dialog box titled "flurstueck_vollinfo - Objektattribute" is open, showing details for a specific parcel. The dialog has three tabs: "Flurstueck", "Eigentümer", and "Nutzung". The "Eigentümer" tab is active, showing the following information:

Gesamt	
Anzahl	4
Eigentümer	Müller, Dieter; Schönbach-Wagner, Barbara; Schönbach, Stefanie; Senn, Axel
Adressen	Friedrich-Silcher-Str. 18, 88008 Waldstadt; Friedrich-Silcher-Str. 18, 88008 Waldstadt; Friedrich-Silcher-Str. 18, 88008 Waldstadt
Beziehung	DEBWLT010000r0DP
Buchungsart	1101

Below the main form, there is a "Liste" section with a tree view for "eigentum" and a sub-dialog for "Eigentuemmerliste".

Eigentuemmerliste	
Nachname	Müller
Vorname	Dieter
Straße	Friedrich-Silcher-Str.
Hnr	18
PLZ	88008
Ort	Waldstadt
Beziehung	DEBWLT010000r0DP
Buchungsart	1101

The background map shows a cadastral plan with various colored parcels (orange, yellow, green, pink, red) and parcel numbers (e.g., 540, 541/2, 557, 558, 559, 561, 562, 563, 564, 565, 566, 567, 568, 569, 642, 643, 644, 645, 646, 647, 648, 649, 650, 651, 652, 653, 654, 660, 661, 662, 663, 664, 665, 666, 667, 668, 669, 670).



NAS-XML ist für QGIS lesbar aber, die Verarbeitung ist mühevoll

Ausgangslage:

- **NAS-XML lässt sich direkt ins QGIS laden**
- **Um ALKIS zu nutzen, müssen Beziehungen zwischen den Tabellen hergestellt werden**
- **Einfache Tabellenjoins reichen nicht – N:N-Beziehungen sind über SQL auszuwerten.**
- **Ich versuchte mich an einer portablen Lösung ohne Datenbankserver**
- **Alle Daten werden in einer SQLite-Datei verarbeitet**



ALKIS mit Spatialite

Zielgruppe

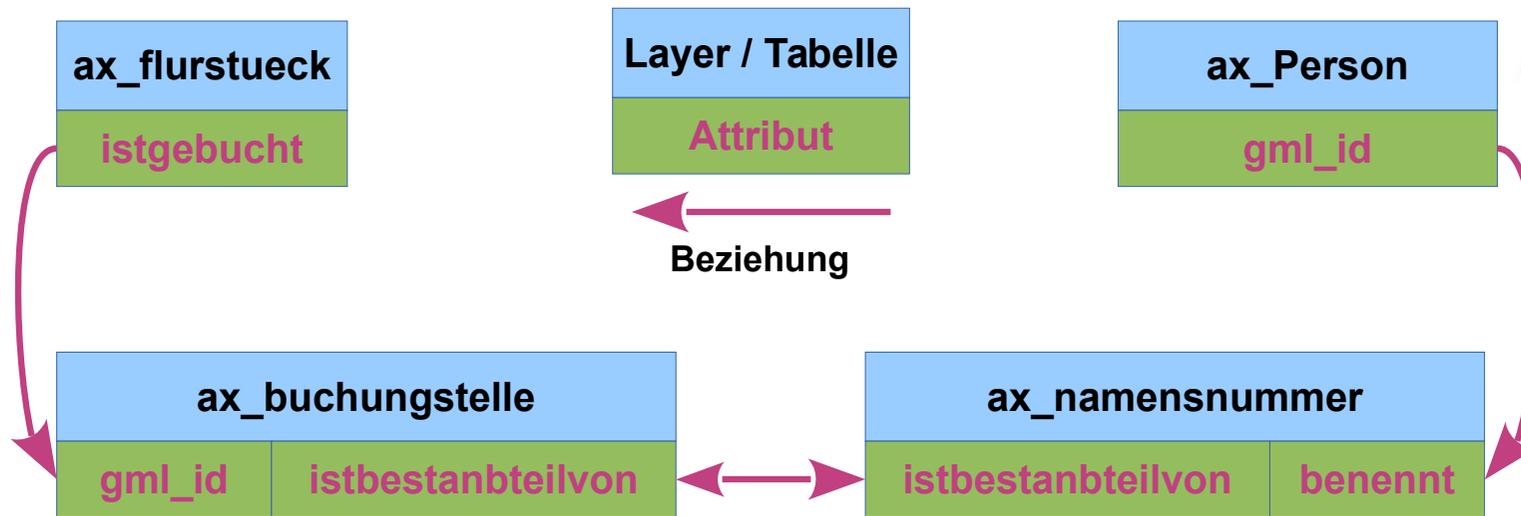
- **Planungsbüros, die regelmäßig kleinere NAS-XML-Dateien verarbeiten**
- **Alle, die die Administration einer PostGis-Datenbank überfordert**
- **Anwender, welche Ihr ALKIS-Projekt in transportabler Form benötigen**
- **Anwender, die die Eigentümerliste als Eigenschaft des Flurstücks benötigen**
- **Anwender, die eine Excelliste aller Buchungen benötigen**

Ausgangsproblem:

Eigentümer und Flurstücke verbinden

- AX_Flurstueck > (AX_Buchungsblatt) > AX_Buchungsstelle > AX_Namensnummer > AX_Personen

NAS-XML: Relationen zwischen Flurstück und Personen

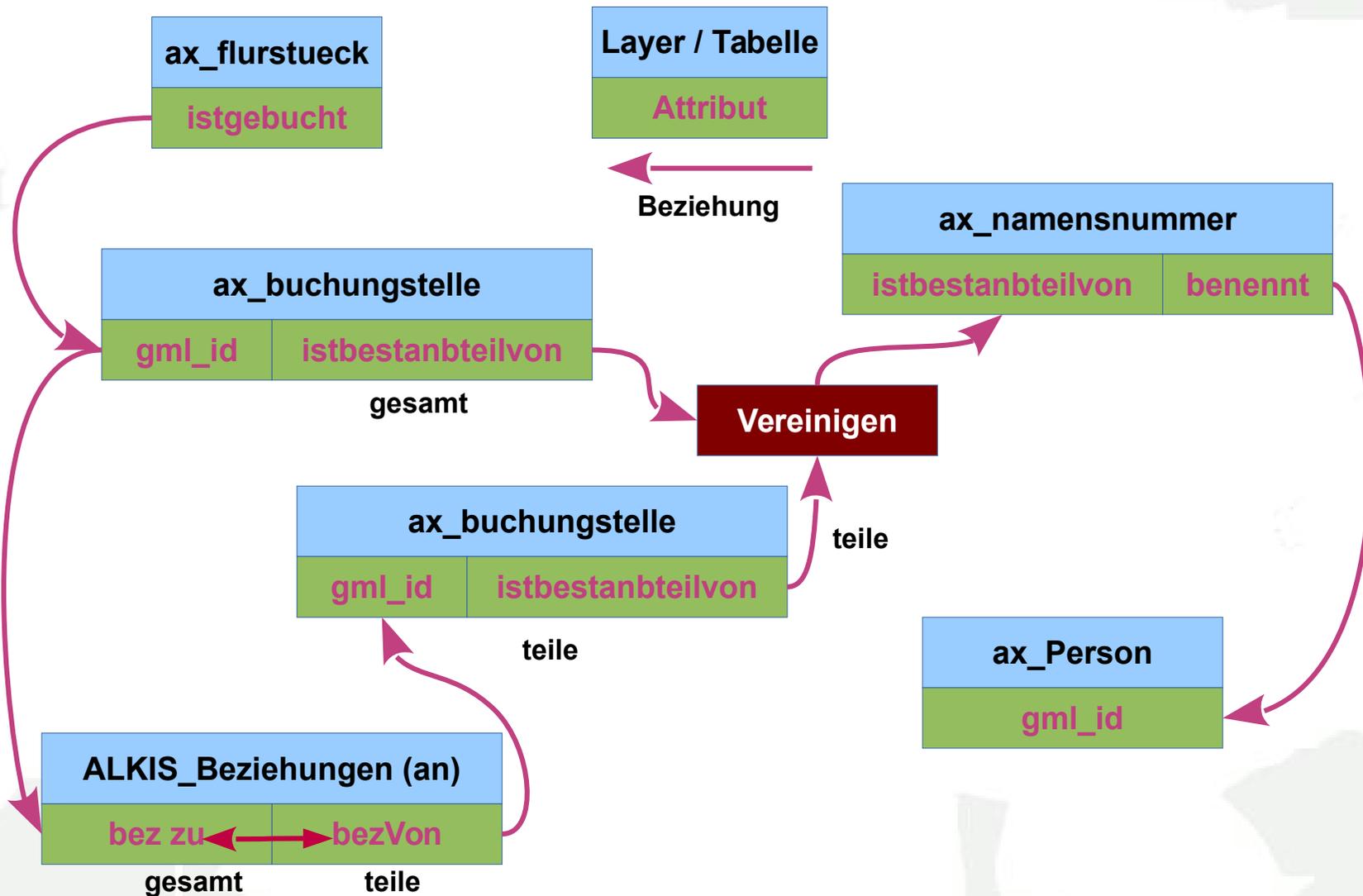


Aber ca. 5 % der Flurstücke bleiben ohne Eigentümer.

Die Eigentümergemeinschaften haben eine eigene Buchungsstelle.
Die Buchungsstellen der einzelnen Mitglieder sind so nicht erreichbar

ALKIS mit Spatialite

NAS-XML: Relationen zwischen Flurstück und Personen bei aufgeteilten Flurstücken





ALKIS mit Spatialite



Nach einiger Mühe konnten alle Eigentümer angebunden werden

- Ca 100 Zeilen SQL um die Eigentümer an die Flurstücke anzubinden
- Eigentümerliste lässt sich mit Flurstück verbinden
- Eigentümerliste als Attribut des Flurstücks

Zunächst gab es eine 18-seitige PDF-Anleitung mit den einzelnen SQL-Befehlen zur Ausführung in der Erweiterung QSpatialite.

flurstueck_vollinfo - Objektattribute

Flurstueck | Eigentümer | Nutzung

Gesamt

Anzahl: 2

Eigentümer: Cantarella, Hubert; Eckhardt, Kerstin

Adressen: Rheingaustraße 25, 65439 Flörsheim; Rheingaustraße 25, 65439 Flörsheim

Beziehung: DEHE0620000005fy

Buchungsart: 1100

Liste

▼ eigentum

Ausdruck

Cantarella

Eckhardt

Eigentüemerliste

Nachname: Cantarella

Vorname: Hubert

Straße: Rheingaustraße

Hnr: 25

PLZ: 65439

Ort: Flörsheim

Beziehung: DEHE0620000005fy

Buchungsart: 1100

Abbrechen | OK



ALKIS mit Spatialite



Ausgangspunkt Eigentümeranbindung

Nutzer mussten folgendermaßen vorgehen, um das Skript zu nutzen:

Konvertierung der einzelnen XML-Dateien mit ogr2ogr:

- OSGeo4w-Kommandozeile öffnen
- Mit CD In das Verzeichnis mit der XML-Datei wechseln.
- Anschließend den ogr2ogr-Befehl ausführen

```
ogr2ogr --config OGR_SQLITE_SYNCHRONOUS OFF --config OGR_SQLITE_CACHE 2048 -progress -f "SQLite" \  
-dsco SPATIALITE=yes -lco SRID=25832 -lco COMPRESS_GEOM=YES -nlt PROMOTE_TO_MULTI -skipfailures -append \  
-gt unlimited -a_srs EPSG:25832 alkis.sqlite alkis.xml 2>> ogrmeldungen.txt
```

Anschließend die resultierende SQLite-Datei im *Spatialite-Gui* öffnen
und das SQL-Skript dort hineinkopieren und ausführen

Aufwändig und für die Nutzer oft überfordernd



ALKIS mit Spatialite

Flächennutzung als Eigenschaft der Flurstücke

Weitere Kundenwünsche

- Ein Layer mit der gesamten Flächennutzung und Klartext-Attributen
- Flächennutzung als Eigenschaft der Flurstücke
- Gebäude mit Klartext-Nutzung, Adressen und Flurstücks-Information

ALKIS mit Spatialite

Gebäude zusammenfassen

```

create table gebaeude_info as
SELECT DISTINCT
a.gml_id as gml_id,
castmultipolygon(a.geometry) as geometry,
case
when a.gebaeuefunktion like '10%' then 'Wohnen'
when a.gebaeuefunktion in ('1121', '1122', '1123', '1131', '1222', '1223') then 'Wohnen mit sonstiger Nutzung'
when a.gebaeuefunktion like '13%' then 'Wochenend oder Gartenhaus'
when a.gebaeuefunktion like '20%' then 'Gewerbe: Büro und Geschäftsgebäude'
when a.gebaeuefunktion like '21%' then 'Gewerbe: Fabriken, Werkstätten, Betriebsgebäude'
when a.gebaeuefunktion like '24%' then 'Garagen, Parkhäuser'
when a.gebaeuefunktion like '3%' then 'Öffentliche Gebäude'
Else 'Sonstige Gebäude'
End as nutzung,
a.gebaeuefunktion as gebaeuefunktion,
name as name,
b.beziehung_zu as beziehung_zu
FROM
--- unterabfrage spezielgebaeude einbinden
(select
gml_id as gml_id,
geometry as geometry,
bauwerksfunktion as gebaeuefunktion,
NULL as name
from ax_bauwerkoderanlagefuerindustriueundgewerbe
union
select
gml_id as gml_id,
geometry as geometry,
bauwerksfunktion as gebaeuefunktion,
NULL as name
from ax_bauwerkimverkehrsbereich
union
select
gml_id as gml_id,
geometry as geometry,
bauwerksfunktion as gebaeuefunktion,
NULL as name
from ax_bauwerkoderanlagefuersportfreizeitunderholung
union
select
gml_id as gml_id,
geometry as geometry,
bauwerksfunktion as gebaeuefunktion,
NULL as name
from ax_turm
union
select
gml_id as gml_id,
geometry as geometry,
gebaeuefunktion as gebaeuefunktion,
name as name
from ax_gebaeude)
--- ende unterabfrage
--- Beziehungsspalte fuer anbindung der Lage joinen
as a left join alkis_beziehungen AS b on (a.gml_id = b.beziehung_von);

```

Nutzung als Klartext

Spezialgebäude über Unterabfrage anbinden



ALKIS mit Spatialite

Auszug Flächen zusammenfassen

/*Skript fasst die Flaechennutzung aus den verschiedenen Tabellen der NAS_XML-Datei in eine Tabelle zusammen und schreibt die Klartextnutzung Eine Tabelle mit Primarschlüssel wird erzeugt, in welche die Daten zusammengefasst werden
In der Windows-version flaechennutzung_anlegen.sql und flaechennutzung.sql →
*/

```
CREATE TABLE flaechennutzung(
ogc_fid INTEGER PRIMARY KEY AUTOINCREMENT, ----- 17-11-23
geometry geometry,
gml_id text(255),
uebersicht text(255),
gruppe text(255), →
nutzung text(255),
funktion integer(12),
vegetationsmerkmal integer(12));
Select RecoverGeometryColumn('flaechennutzung', 'Geometry', $kbs, 'Multipolygon');
```

/*Dummytabellen werden erzeugt, wenn der betreffende Inhalt im Datensatz nicht vorhanden sind, damit Abfragen nicht wegen nicht gefundener Tabelle abbrechen.*/

```
create table if not exists ax_moor(
geometry geometry,
gml_id text(255) PRIMARY Key,
nutzung text(255),
funktion integer(12),
vegetationsmerkmal integer(12));
create table if not exists ax_sumpf(
geometry geometry,
gml_id text(255) PRIMARY Key,
nutzung text(255),
funktion integer(12),
vegetationsmerkmal integer(12));
create table if not exists ax_heide(
geometry geometry,
gml_id text(255) PRIMARY Key,
nutzung text(255),
funktion integer(12),
vegetationsmerkmal integer(12));
create table if not exists ax_gehoelz(
geometry geometry,
gml_id text(255) PRIMARY Key,
nutzung text(255),
funktion integer(12),
vegetationsmerkmal integer(12));
create table if not exists ax_unlandvegetationsloseflaeche(
geometry geometry,
gml_id text(255) PRIMARY Key,
nutzung text(255),
funktion integer(12),
vegetationsmerkmal integer(12));
create table if not exists ax_halde(
geometry geometry,
gml_id text(255) PRIMARY Key,
nutzung text(255),
funktion integer(12),
vegetationsmerkmal integer(12));
create table if not exists ax_bergbaubetrieb(
geometry geometry,
gml_id text(255) PRIMARY Key,
nutzung text(255),
funktion integer(12),
vegetationsmerkmal integer(12));
```

Dummytabellen
anlegen

```
SELECT distinct casttomultipolygon(Geometry) AS Geometry,
"ax_industrieundgewerbeflaeche".gml_id AS gml_id,
'Siedlung'.as.uebersicht,
'Industrie und Gewerbe'.as.gruppe,
case
when funktion in ('1700', '1701') then 'Industrie und Gewerbe, Allgemein'
when funktion = 1740 then 'Industrie und Gewerbe, Lagerplatz'
when funktion = 1400 then 'Handel und Dienstleistung'
when funktion = 2501 then 'Versorgung'
when funktion in ('2601', '2602') then 'Entsorgung'
Else 'Industrie und Gewerbe Allgemein'
End AS nutzung,
funktion as funktion,
null as vegetationsmerkmal
FROM "ax_industrieundgewerbeflaeche"
union
SELECT distinct casttomultipolygon(Geometry) AS Geometry,
"ax_wohnbauflaeche".gml_id AS gml_id,
'Siedlung'.as.uebersicht,
'Wohnbauflaeche'.as.gruppe,
'Wohnbauflaeche'.AS.nutzung,
null as funktion,
null as vegetationsmerkmal
FROM "ax_wohnbauflaeche"
union
SELECT distinct casttomultipolygon(Geometry) AS Geometry,
"ax_flaechegemischternutzung".gml_id AS gml_id,
'Siedlung'.as.uebersicht,
'Flaeche gemischter Nutzung'.as.gruppe,
'Flaeche gemischter Nutzung'.AS.nutzung,
null as funktion,
null as vegetationsmerkmal
FROM "ax_flaechegemischternutzung"
union
SELECT distinct casttomultipolygon(Geometry) AS Geometry,
"ax_flaechebesondererfunktionalerpraegung".gml_id AS gml_id,
'Siedlung'.as.uebersicht,
'Bes. funktionale Praegung'.as.gruppe,
case
when funktion = 1100 then 'Öffentliche Zwecke'
when funktion = 1300 then 'Historische Anlage'
Else 'Siedlung, Bes. funktionale P.'
End AS nutzung,
funktion as funktion,
null as vegetationsmerkmal
FROM "ax_flaechebesondererfunktionalerpraegung"
union
SELECT distinct casttomultipolygon(Geometry) AS Geometry,
"ax_platz".gml_id AS gml_id,
'Siedlung'.as.uebersicht,
'Verkehr'.as.gruppe,
'Platz'.AS.nutzung,
null as funktion,
null as vegetationsmerkmal
FROM "ax_platz"
union
SELECT distinct casttomultipolygon(Geometry) AS Geometry,
"ax_strassenverkehr".gml_id AS gml_id,
'Siedlung'.as.uebersicht,
'Verkehr'.as.gruppe,
'Straßenverkehr'.AS.nutzung,
null as funktion,
```



ALKIS mit Spatialite

Flächennutzung als Eigenschaft der Flurstücke

Flächennutzung und Flurstücke mit ST_Intersection verschneiden und Fläche der resultierenden Polygone berechnen

```

1250 /*
1251 Dieses Teilskript verschneidet die Flächennutzung mit den Flurstücken,
1252 um die Anteilige Nutzung der Flurstücke zu ermitteln.
1253 In der Windows-version die skripte
1254 flaechenverschneidung.sql, nutzung_influr_stuecke.sql, flurstueck_vollinf_gen.sql, hilfstabellen_drop.sql
1255 */
1256 -- Nutzung mit Flurstuecken verschneiden
1257 create table flst_nutzung as
1258 select distinct
1259 casttomultipolygon(ST_intersection(a.geometry,b.geometry)) as geometry,
1260 round(area(casttomultipolygon(ST_intersection(a.geometry,b.geometry)))) as qm,
1261 a.gml_id as gml_id,
1262 b.nutzung as nutzung
1263 from ax_flurstueck as a inner join flaechnutzung as b on ST_intersects(a.geometry,b.geometry) = 1
1264 where qm > 0 and a.ROWID IN
1265 ..... (SELECT ROWID FROM SpatialIndex WHERE f_table_name='ax_flurstueck' AND search_frame=b.geometry)
1266 ;
1267 Select RecoverGeometryColumn('flst_nutzung','Geometry',$kbs,'MULTIPOLYGON');
1268 Select CreateSpatialIndex('flst_nutzung','Geometry');
1269 SELECT UpdateLayerStatistics('flst_nutzung');
1270

```

Verschneidung via ST_intersection

Flächenberechnung

Räumlichen Index verwenden



Flächennutzung als Eigenschaft der Flurstücke

Die Flächennutzungen und deren qm-Wert werden aus dem verschnittenen Layer via Join über die gml_id an die Flurstücksgeometrie angebunden. Über group_concat werden alle Nutzungen eines Flurstücks mit ihrer Fläche aufgezählt.

```
1271 --Die anteilige Flächennutzung in die Flurstuecke schreiben
1272 create table flurstueck_zwischeninfo as
1273 select distinct
1274 a.gml_id as gml_id,
1275 a.Geometry as geometry,
1276 a.lage as lage,
1277 a.adresse as adresse,
1278 a.hausnr as hausnr,
1279 group_concat((b.nutzung || ':' || b.qm || ' qm'), ';' ) as nutzungen
1280
1281 from flurstueck_lage as a left join
1282
1283 .. (select
1284 .. gml_id,
1285 .. nutzung,
1286 .. sum(qm) as qm
1287 .. from
1288 .. flst_nutzung
1289 ..
1290 .. group by gml_id, nutzung)
1291
1292 as b on (a.gml_id = b.gml_id)
1293 where qm > 0
1294 group by a.gml_id;
1295
1296 Select RecoverGeometryColumn('flurstueck_zwischeninfo', 'Geometry', $kbs, 'MULTIPOLYGON');
1297 Select CreateSpatialIndex('flurstueck_zwischeninfo', 'Geometry');
1298 SELECT UpdateLayerStatistics('flurstueck_zwischeninfo');
```

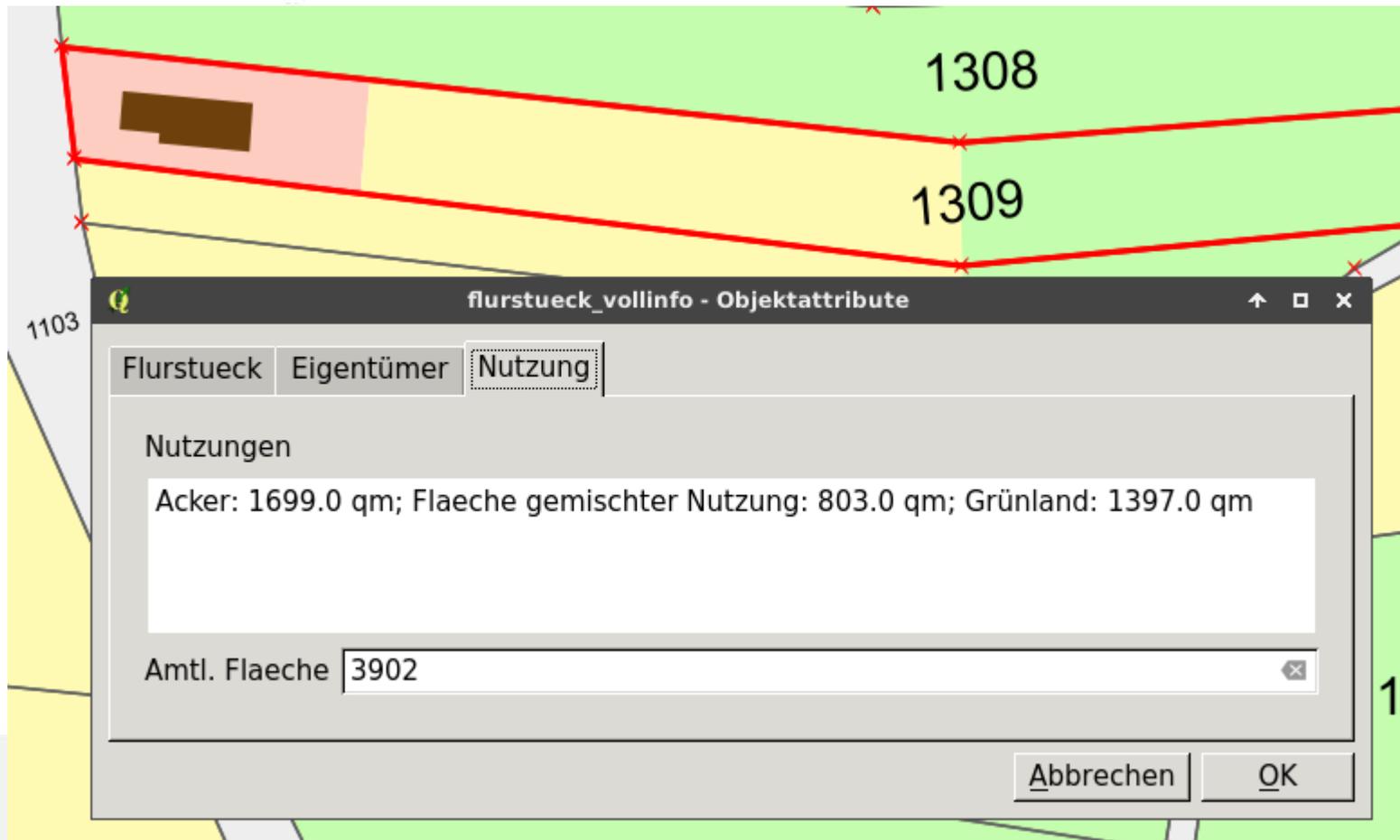
Flächennutzungen eines Flurstücks als Liste

Unterabfrage um die Flächensumme jeder Nutzung je Flurstück zu ermitteln

Flächennutzung der Flurstücke an den Flurstückslayer anjoinen

Flächennutzung als Eigenschaft der Flurstücke

Die Flächennutzungen und deren qm-Wert werden aus dem verschnittenen Layer via Join über die `gml_id` an die Flurstücksgeometrie angebunden. Über `group_concat` werden alle Nutzungen eines Flurstücks mit ihrer Fläche aufgezählt.





ALKIS mit Spatialite

Jetzt wäre es schön alles in einen Ablauf zusammenzufassen

Ziel:

Ein Shellskript ruft ogr2ogr und die SQL-Skripte auf für Windows (cmd-BAT) und Linux (bash).

Ziele und Herausforderungen

- ogr2ogr und SQL-Skripte sollen nacheinander aufgerufen werden, ohne das die Anwender weitere Programme öffnen müssen.
- Am Ende entsteht eine SQLite-Datei die einem fertigen QGIS-Projekt zugeordnet ist
- Länderspezifische Besonderheiten (spezielle Koordinatenbezugssysteme) sind zu berücksichtigen.
- Es müssen ganze Verzeichnisse von NAS-XML-Dateien angesprochen werden und nicht nur einzelne Dateien.
- Fehlende Tabelle oder Spalten dürfen nicht zum Skript-Abbruch führen
- Das Werkzeug soll auf jedem Windows-Rechner bei unterschiedlichen QGIS-Installationen laufen und den Ort der QGIS-Installation finden.
- Das Werkzeug soll über einen Desktop-Starter geöffnet werden und keine Kenntnisse über Kommandozeilen-Befehle erfordern.
- Eine Umsetzung in unterschiedliche Koordinatensysteme muss möglich sein.
- Das Skript muss die Anwender fragen, ob eine Anbindung von Eigentümern erforderlich ist



ALKIS mit Spatialite

Auswahl Koordinatenbezugssystem

```
echo -n "Koordinatensystem wählen"
echo
echo
# Ausgabekoordinatensystem wählen
# Zunächst Erläuterungen via Echo ausgeben
echo -n "In welchem Koordinatensystem sollen die Daten ausgegeben werden? (Mit Ziffer wählen)"
echo
echo -n "31467 ist Gauß-Krüger 3: Tippen Sie 1 und RETURN"
echo
echo -n "31468 ist Gauß-Krüger 4: Tippen Sie 2 und RETURN"
echo
echo -n "25832 ist (ETRS89 / UTM 32): Tippen Sie 3 und RETURN"
echo
echo -n "25833 ist (ETRS89 / UTM 33): Tippen Sie 4 und RETURN"
echo
echo
# Mit dem select Kommando die Auswahl der vier Ausgabe-KBS anbieten, funktioniert wie for
select kbs in 31467 31468 25832 25833
do
kbs=$kbs
break
done
if [ $kbs = 31467 -o $kbs = 25832 -o $kbs = 25833 ]; then
cp ~/bin/alkisqml/alkis_ges_$kbs.qgs "$pfad"
else
cp ~/bin/alkisqml/alkis_ges.qgs "$pfad"
fi

# Das Land wählen. In einigen Ländern wird nicht EPSG:25832 verwendet...
# Für jede Variante gibt es einen eigenen ogr2ogr-Befehl
# Zunächst die Auswahl erläutern
echo
echo -n "Aus welchem Land stammen die Daten?:"
echo
echo -n "Tippen Sie 1, 2, 3, 4 oder 5 und RETURN"
echo
# Mit dem Select Kommando die Variablen der Länder für den angepassten ogr2ogr-Befehl setzen
select laender in "Baden-Württemberg" "Bayern" "BrandenburgBerlinMeckpom" "Sachsen" "Sonstige"
do
if [ $laender = "Baden-Württemberg" ]; then
...clear
...for xmls in *.xml; do
```



ALKIS mit Spatialite

Auswahl Koordinatenbezugssystem

```
echo =====
echo In welchem Koordinatensystem sollen die Daten ausgegeben werden? (Mit Ziffer wählen)
echo Ohne Auswahl werden die Daten nach (ETRS89 / UTM 32) geschrieben.
echo.
echo [1] 31467 ist Gauß-Krüger 3:  →  →  →  Tippen Sie 1 und RETURN
echo [2] 31468 ist Gauß-Krüger 4:  →  →  →  Tippen Sie 2 und RETURN
echo [3] 25832 ist (ETRS89 / UTM 32): →  →  →  Tippen Sie 3 und RETURN
echo [4] 25833 ist (ETRS89 / UTM 33): →  →  →  Tippen Sie 4 und RETURN
echo.
set asw=3
set /p asw="Bitte eine Auswahl treffen: "
if %asw%==1 set kbs=31467
if %asw%==2 set kbs=31468
if %asw%==3 set kbs=25832
if %asw%==4 set kbs=25833
echo KBS ist %kbs%
echo =====
echo.

::Das Land wählen. In einigen Ländern wird nicht EPSG:25832 verwendet.
::Für jede Variante gibt es einen eigenen ogr2ogr-Befehl
::Zunächst die Auswahl erläutern
echo.
echo =====
echo Aus welchem Land stammen die Daten?
echo.
echo [1] Baden-Württemberg: →  →  →  Tippen Sie 1 und RETURN
echo [2] Bayern: →  →  →  Tippen Sie 2 und RETURN
echo [3] BB, MeckP, Berlin →  →  →  Tippen Sie 3 und RETURN
echo [4] Sachsen: →  →  →  Tippen Sie 4 und RETURN
echo [5] Sonstige: →  →  →  Tippen Sie 5 und RETURN
echo.
set asw=5
set /p asw="Tippen Sie 1 (BaWü), 2 (Bay), 3 (BraB), 4 (Sach) oder 5 (Sonstige) und RETURN: "

if %asw%==1 goto baw
if %asw%==2 goto bay
if %asw%==3 goto bra
if %asw%==4 goto sax
if %asw%==5 goto son
```



```

ALIKS Converter
*****
QGIS ist installiert unter C:\OSGeo4W64
*****
Spatialite und ogr2ogr müssen in C:\OSGeo4W64 installiert sein!
Spatialite ist installiert
ogr2ogr ist installiert
  1 Datei(en) kopiert.
*****
Auf welchem Laufwerk liegt Ihr Datenverzeichnis?
Bitte Laufwerksbuchstaben mit : eingeben z.B: D: C:
In welchem Verzeichnis liegen Ihre XML-Dateien?
Vollständigen Verzeichnispfad eingeben und return drücken
Bitte Pfad eingeben oder via rechte Maustaste aus der Zwischenablage einfügen: C:\Users\
Ihre Daten liegen in C:\Users\claas\Desktop\moehltest
*****
Wenn Eigentüemerdaten vorhanden sind bitte j eingeben ansonsten n: j
*****
Folgende zwei Optionen kosten ein wenig Zeit, im Zweifelsfall n für nein
Excelltabelle mit Buchungsblattinfos schreiben? j eingeben ansonsten n: j
Wenn Grundstücke modelliert werden sollen j eingeben ansonsten n: j
*****
Die fertige SQLITE-Datei ist unter folgendem Pfad zu finden:
C:\Users\claas\Desktop\moehltest\alkis_ges.sqlite
*****
=====
In welchem Koordinatensystem sollen die Daten ausgegeben werden? (Mit Ziffer wählen)
Ohne Auswahl werden Die daten nach (ETRS89 / UTM 32) geschrieben.
[1] 31467 ist Gauß-Krüger 3:           Tippen Sie 1 und RETURN
[2] 31468 ist Gauß-Krüger 4:           Tippen Sie 2 und RETURN
[3] 25832 ist (ETRS89 / UTM 32):       Tippen Sie 3 und RETURN
[4] 25833 ist (ETRS89 / UTM 33):       Tippen Sie 4 und RETURN
Bitte eine Auswahl treffen: 4
KBS ist 25833
=====
Aus welchem Land stammen die Daten?
[1] Baden Württemberg:                Tippen Sie 1 und RETURN
[2] Bayern:                            Tippen Sie 2 und RETURN
[3] BB, MeckP, Berlin                   Tippen Sie 3 und RETURN
[4] Sachsen:                            Tippen Sie 4 und RETURN
[5] Sonstige:                           Tippen Sie 5 und RETURN
Tippen Sie 1 (Bawü), 2 (Bay), 3 (BraB), 4 (Sach) oder 5 (Sonstige) und RETURN:

```

Installationsorte geprüft

Pfad eingeben

Eigentümer anbinden?

Buchungsblattinfos verarbeiten?

Ausgabe-KBS wählen

Aus welchem Land?





ALKIS mit Spatialite

OGR2OGR-Aufrufe für unterschiedliche Länder

```

echo
# Mit dem Select-Kommando die Variablen der Ländern für den angepassten ogr2ogr-Befehl setzen
select laender in "Baden-Württemberg" "Bayern" "BrandenburgBerlinMeckpom" "Sachsen" "Sonstige"
do
  if [ "$laender" = "Baden-Württemberg" ]; then
    clear
    for xmls in *.xml; do
      echo -n "*****"
      echo
      echo -n "Bitte etwas Geduld: Die Konvertierung via ogr2ogr kann bei großen XML-Dateien einige Zeit dauern -"
      echo
      echo -n "so lange die SQLITE-Datei und das Journal geschrieben wird, arbeitet das Skript"
      echo
      ogr2ogr --config OGR_SQLITE_SYNCHRONOUS OFF --config OGR_SQLITE_CACHE 2048 --progress -f "SQLite" --dsco SPATIALITE=yes --lco SRID=$kbs --lco COMPRESS_GEOM=YES --nlt PROMOTE_TO_MULTI \
      --skipfailures --addfields --gt unlimited -s_srs "+proj=tmerc +lat_0=0 +lon_0=9 +k=1 +x_0=500000 +y_0=0 +ellps=bessel +datum=potsdam +nadgrids=BETA2007.gsb +units=m +no_defs" \
      -t_srs EPSG:$kbs $alkis.sqlite $xmls.2>> ogrammeldungen.txt
      echo
      echo
      echo -n "XMLs dieser GFS-Dateien sind konvertiert und in die SQLITE geschrieben:"
      echo
      ls -C *.gfs
      echo
      echo
    done
  elif [ "$laender" = "Bayern" ]; then
    clear
    for xmls in *.xml; do
      echo -n "*****"
      echo
      echo -n "Bitte etwas Geduld: Die Konvertierung via ogr2ogr kann bei großen XML-Dateien einige Zeit dauern -"
      echo
      echo -n "so lange die SQLITE-Datei und das Journal geschrieben wird, arbeitet das Skript"
      echo
      ogr2ogr --config OGR_SQLITE_SYNCHRONOUS OFF --config OGR_SQLITE_CACHE 2048 --progress -f "SQLite" --dsco SPATIALITE=yes --lco SRID=$kbs --lco COMPRESS_GEOM=YES --nlt PROMOTE_TO_MULTI \
      --skipfailures --update --addfields --gt unlimited -s_srs EPSG:25833 -t_srs EPSG:$kbs $alkis.sqlite $xmls.2>> ogrammeldungen.txt
      echo
      echo
      echo -n "XMLs dieser GFS-Dateien sind konvertiert und in die SQLITE geschrieben:"
      echo
      ls -C *.gfs
      echo
      echo
    done
  elif [ "$laender" = "BrandenburgBerlinMeckpom" ]; then
    clear
    for xmls in *.xml; do
      echo -n "*****"
      echo
      echo -n "Bitte etwas Geduld: Die Konvertierung via ogr2ogr kann bei großen XML-Dateien einige Zeit dauern -"
      echo
      echo -n "so lange die SQLITE-Datei und das Journal geschrieben wird, arbeitet das Skript"
      echo
      ogr2ogr --config OGR_SQLITE_SYNCHRONOUS OFF --config OGR_SQLITE_CACHE 2048 --progress -f "SQLite" --dsco SPATIALITE=yes --lco SRID=$kbs --lco COMPRESS_GEOM=YES --nlt PROMOTE_TO_MULTI \
      --skipfailures --update --addfields --gt unlimited -s_srs EPSG:25833 -t_srs EPSG:$kbs $alkis.sqlite $xmls.2>> ogrammeldungen.txt
      echo
      echo
      echo -n "XMLs dieser GFS-Dateien sind konvertiert und in die SQLITE geschrieben:"
      echo
      ls -C *.gfs
      echo
      echo
    done
  fi
done

```

addfields statt append

Merkwürdiges BaWü-KBS



ALKIS mit Spatialite

Aufrufe der SQL-Befehle aus dem Skript

Unter Linux Aufruf mit dem Spatialite-CLI direkt aus dem Skript
`spatialite -silent $alkis.sqlite "Begin Transaction;SQL-Befehle;commit;"`

Unter Windows Aufruf von Skript-Dateien mit dem Spatialite-CLI
`spatialite -silent %alkis%.sqlite < "%HOMEDRIVE%\%HOMEPATH%\gkg\sqlskript.sql"`



ALKIS mit Spatialite

Ergebnislayer

- **Flächennutzung** enthält die zusammengefasste Flächennutzung mit detaillierter thematischer Klartext-Attributierung.
- **Gebäudevollinfo** enthält die Gebäudegeometrien mit Gebäudeadresse, Hausnummer und der Gebäudenutzung im Klartext sowie das Flurstückskennzeichen. Es sind nicht nur die Gebäude aus ax_gebaeude, sondern auch die Sondergebäude erfasst.
- Die Tabelle **bs_nn_eigentuemmer** stellt eine Liste sämtlicher Buchungsstellen mit den zugehörigen Eigentümern und ihrer Adresse zur Verfügung.
- **Flurstueckvollinfo** umfasst die Flurstücksgeometrien mit Flurstückskennzeichen und Adresse samt Hausnummer oder Flurbezeichnung sowie eine Auflistung sämtlicher Eigentümer. Außerdem werden die Flächennutzungen der Flurstücke mit Quadratmeterangabe als Attribut aufgelistet.



ALKIS mit Spatialite



- bs_nn_eigentuemer
- flurstueck_vollinfo
- gebaeude_vollinfo
- nutzung
 - Wohnen
 - Wohnen mit sonstiger N...
 - Gewerbe: Büro und Ges...
 - Gewerbe: Fabriken, We...
 - Öffentliche Gebäude
 - Wochenend oder Garte...
 - Garagen, Parkhäuser
 - Sonstige Gebäude
- flaechennutzung
- gruppe
 - Wohnbauflaeche
 - Flaeche gemischter Nut...
 - Industrie und Gewerbe
 - Bes. funktionale Praegung
 - Verkehr
 - Grünflächen
 - Ackerland
 - Grünland
 - Garten- und Weinbau
 - LW Sonstiges
 - Gehölz
 - Wald
 - Naturnah - offen, trocken
 - Fließgewässer
 - Stehendes Gewässer
 -

```

*****
Die Shapes sollten im Verzeichnis C:\Users\claas\Desktop\moehlttest\shapes liegen.
C:\Users\claas\Desktop\moehlttest\alkis_ges.sqlite enthält sämtliche Daten
*****

Alle Prozesse sind durchgeführt das Programm kann nun beendet werden.

#####
#
# GKG-ALKIS-Konverter
#
# Autor: Dr.-Ing. Claas Leiner
#
# http://www.gkg-kassel.de
# Mail: claas.leiner@gkg-kassel.de
#
# Konvertiert sämtliche Inhalte aller NAS-XML-Dateien eines Verzeichnisses
# in eine Spatialite-Datenbank
#
# Fasst die Landnutzung in einen Layer zusammen und
# und trägt die Gebäudenutzung mit Klartext ein.
#
# Verbindet Hausnummern und Adressen mit Flurstücken und Gebäuden.
# Schreibt die Flurstückskenung als Attribut zu jedem Gebäude.
# Schreibt die anteilige Flächennutzung jedes Flurstücks als Attribut
#
# Listet sämtliche Eigentümer jedes Flurstücks mit Adresse
# als Attribut in den Flurstückslayer.
#
# Erzeugt eine Tabelle sämtlicher Eigentümer mit direkter Beziehung zu den
# Flurstücken und schreibt die Adressen jedes Eigentümers als Attribut
#
# Exportiert Gebäude, Flächennutzung und Flurstücke mit
# Eigentümern sowie Nutzungsanteilen als Shapefile
#
#
# Weitere Inhalte können beim Autor beauftragt werden
#
#####

*****
Alle Prozesse sind durchgeführt das Programm kann nun beendet werden.
*****

Soll QGIS gestartet werden, um das Projekt zu öffnen? ja oder nein ja

*****

Das Zoomen auf den Layer gelingt erst, wenn in den Layer-Eigenschaften unter
"Allgemein" einmal auf Schaltfläche "Ausdehnung aktualisieren" geklickt wird

QGIS starten und Projekt mit Layern aus alkis_ges.sqlite öffnen
Alternativ kann auch die andere Projektdatei mit Verknüpfung zu den Shapes geöffnet werden.

Extrahierter Wert: 348499,5796488,356990,5801290.

```



ALKIS mit Spatialite



The screenshot shows the QGIS interface with a Spatialite query window open. The query window is titled "flurstueck_vollinfo - Objektattribute" and has three tabs: "Flurstueck", "Eigentümer", and "Nutzung". The "Eigentümer" tab is active, showing the following details:

- Gesamt**
 - Anzahl: 4
 - Eigentümer: Müller, Dieter; Schönbach-Wagner, Barbara; Schönbach, Stefanie; Senn, Axel
 - Adressen: Friedrich-Silcher-Str. 18, 88088 Waldstadt; Friedrich-Silcher-Str. 18, 88088 Waldstadt; Friedrich-Silcher-Str. 18, 88088 Waldstadt
 - Beziehung: DEBWLT010000r0dP
 - Buchungsart: 1101
- Liste**
 - eigentum**
 - Müller
 - Schönbach-Wagner
 - Schönbach
 - Senn
 - Eigentümerliste**
 - Nachname: Müller
 - Vorname: Dieter
 - Straße: Friedrich-Silcher-Str.
 - Hnr: 18
 - PLZ: 88088
 - Ort: Waldstadt
 - Beziehung: DEBWLT010000r0dP
 - Buchungsart: 1101

The background map shows a cadastral map (ALKIS) with various colored parcels and their identification numbers. A red box highlights a specific parcel (664/5) which is the subject of the query window.



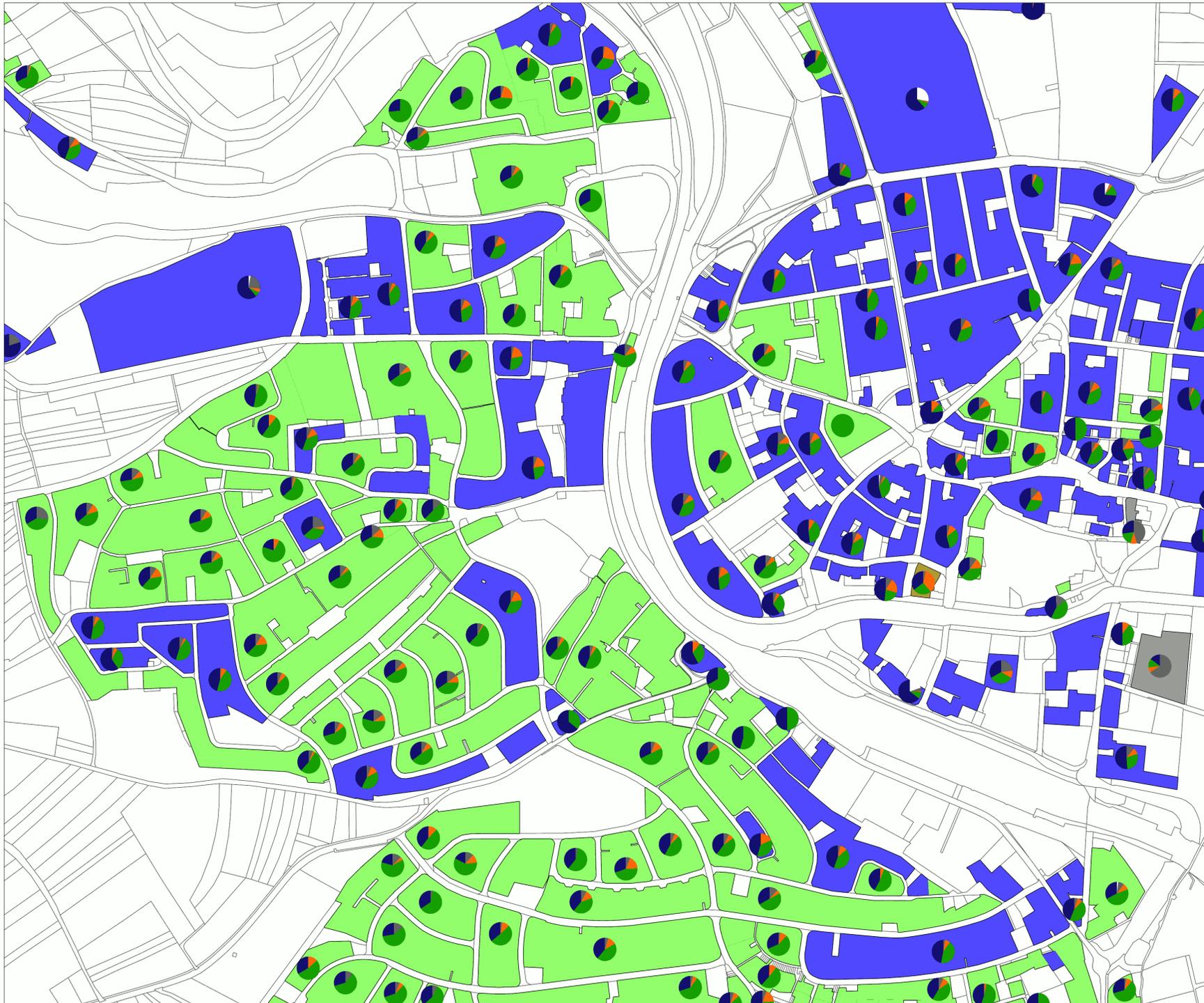
ALKIS mit Spatialite

Weitere Anpassungen

- Exceltabelle mit Buchungsblattinformationen
- Anbindung von Einwohnerstatistiken an Flurstücke und Gebäude

```
/*Baublücke zusammenfassen
Gruppirt nach den ersten Zeichen der Spalte g_klg
Klg musste mit der Gemeinenummer verbunden werden, weil es identische klg
in verschiedenen gemeinden gab
*/
create table klg_baublock as
select
substr(a.g_klg,1,9) as klg_baublock,
casttomultipolygon(buffer(GUnion(Buffer(Geometry,0.1)),-0.1)) as geometry,
--- Geometrie wird verschmolzen, Pufferfunktion ist nötig, weil es unkorrekte Topologien gibt
sum(a.anz) as einwohner,
sum(a.alleinige_wohnung) as alleinige_wohnung,
sum(a.haupt_wohnung) as einwohner_hauptwohnsitz,
sum(a.neben_wohnung) as einwohner_nebenwohnsitz,
sum(a.seit_geburt) as seit_geburt,
casttointeger(CastToDouble(sum(a.seit_geburt)/(sum(a.anz))*100) as seit_geburt_anteil,
casttointeger(CastToDouble(sum(a.anz*a.wohndauer_gemeinde)/sum(a.anz)) as wohndauer_gemeinde,
casttointeger(CastToDouble(sum(a.anz*a.wohndauer_wohnung)/sum(a.anz)) as wohndauer_wohnung,
casttointeger(CastToDouble(sum(a.anz*a.durchschnittsalter)/sum(a.anz)) as durchschnitt_alter,
casttointeger(sum(a.anz)/(st_area(casttomultipolygon(buffer(GUnion(Buffer(a.Geometry,0.1)),-0.1)))/10000)) as dichte,
sum(a.deutsche) as deutsche,
casttointeger(CastToDouble(sum(a.deutsche)/(sum(a.anz))*100) as de_anteil,
sum(a.Frau) as Frau,
sum(a.Mann) as Mann,
casttointeger(CastToDouble(sum(a.Frau)/(sum(a.anz))*100) as Frauen_anteil,
sum(a.fs_ledig) as fs_ledig,
sum(a.fs_verheiratet) as fs_verheiratet,
sum(a.fs_verwitwet) as fs_verwitwet,
sum(a.fs_geschieden) as fs_geschieden,
sum(a.fs_unbekannt) as fs_unbekannt,
casttointeger(CastToDouble(sum(a.fs_ledig)/(sum(a.anz))*100) as fs_ledig_anteil,
casttointeger(CastToDouble(sum(a.fs_verheiratet)/(sum(a.anz))*100) as fs_verheiratet_anteil,
casttointeger(CastToDouble(sum(a.fs_verwitwet)/(sum(a.anz))*100) as fs_verwitwet_anteil,
casttointeger(CastToDouble(sum(a.fs_geschieden)/(sum(a.anz))*100) as fs_geschieden_anteil,
casttointeger(CastToDouble(sum(a.fs_unbekannt)/(sum(a.anz))*100) as fs_unbekannt_anteil,
sum(a.u_18) as u_18,
casttointeger(CastToDouble(sum(a.u_18)/(sum(a.anz))*100) as u_18_anteil,
casttointeger(sum(a.u_18)/(st_area(casttomultipolygon(buffer(GUnion(Buffer(a.Geometry,0.1)),-0.1)))/10000)) as u_18_dichte_ha,
sum(a.evangelisch) as evangelisch,
sum(a.katholisch) as katholisch.
```

Karte Familienstand nach Blöcken



Vorherrschender Familienstand

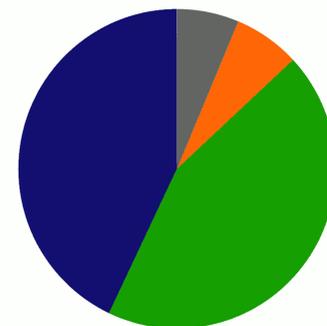
Flächen

- ledig
- verheiratet
- geschieden
- verwitwet

Kreisdiagramm

- ledig
- verheiratet
- geschieden
- verwitwet

Stand	Anzahl	Anteil
ledig	15440	43
verheiratet	15810	44
geschieden	2423	7
verwitwet	2269	6



Verteilung Stadtwei

100 0 100 200 300 m





ALKIS mit Spatialite

Was fehlt?

- Insbesondere gibt es noch keine ALKIS-Stile gemäß Signaturenkatalog, sondern nur eine selbst gestrickte Symbolisierung und Flurstücksbeschriftung.
- Die Auswertung der anteiligen Flächennutzung kann bei einzelnen Flurstücken unkorrekt sein, da die SQL-Verschneidung im Spatialite über Geometriefehler stolpert.
- Viele Attribute, die für die Gebäude in einigen Ländern vorhanden sind (Stockwerke, Zustand etc.) werden aktuell noch nicht ausgewertet.
- Die Erb-Pachverhältnisse der Flurstücke werden aktuell nicht ausgewertet.
- Es fehlt auch noch eine ansprechende grafische Benutzeroberfläche
- Oder eine Umsetzung als QGIS-Plugin.

etc.....



ALKIS mit Spatialite

Wie gehts weiter?

Wie ist es zu nutzen und welche Pläne gibt es?

Der Autor Claas Leier stellt das Werkzeug allen Interessierten für die private Nutzung kostenfrei zur Verfügung. Für **Mitglieder des QGIS.DE-Anwendervereins** und des **FOSSGIS e.V.** ist auch die kommerzielle oder behördeninterne Nutzung uneingeschränkt und kostenfrei möglich.

Sonstigen behördlichen und kommerziellen Nutzern wird das Tool bis 2019 nur gegen Honorar zur Verfügung gestellt. **Spätestens ab dem 31. Dezember 2019 möchte ich das Werkzeug als freie Software veröffentlichen.**