

JasperReports als Reportingwerkzeug für QGIS

Bernhard Ströbl

Anwendungsbetreuer GIS

Kommunale Immobilien Jena

Am Anger 26, 07743 Jena

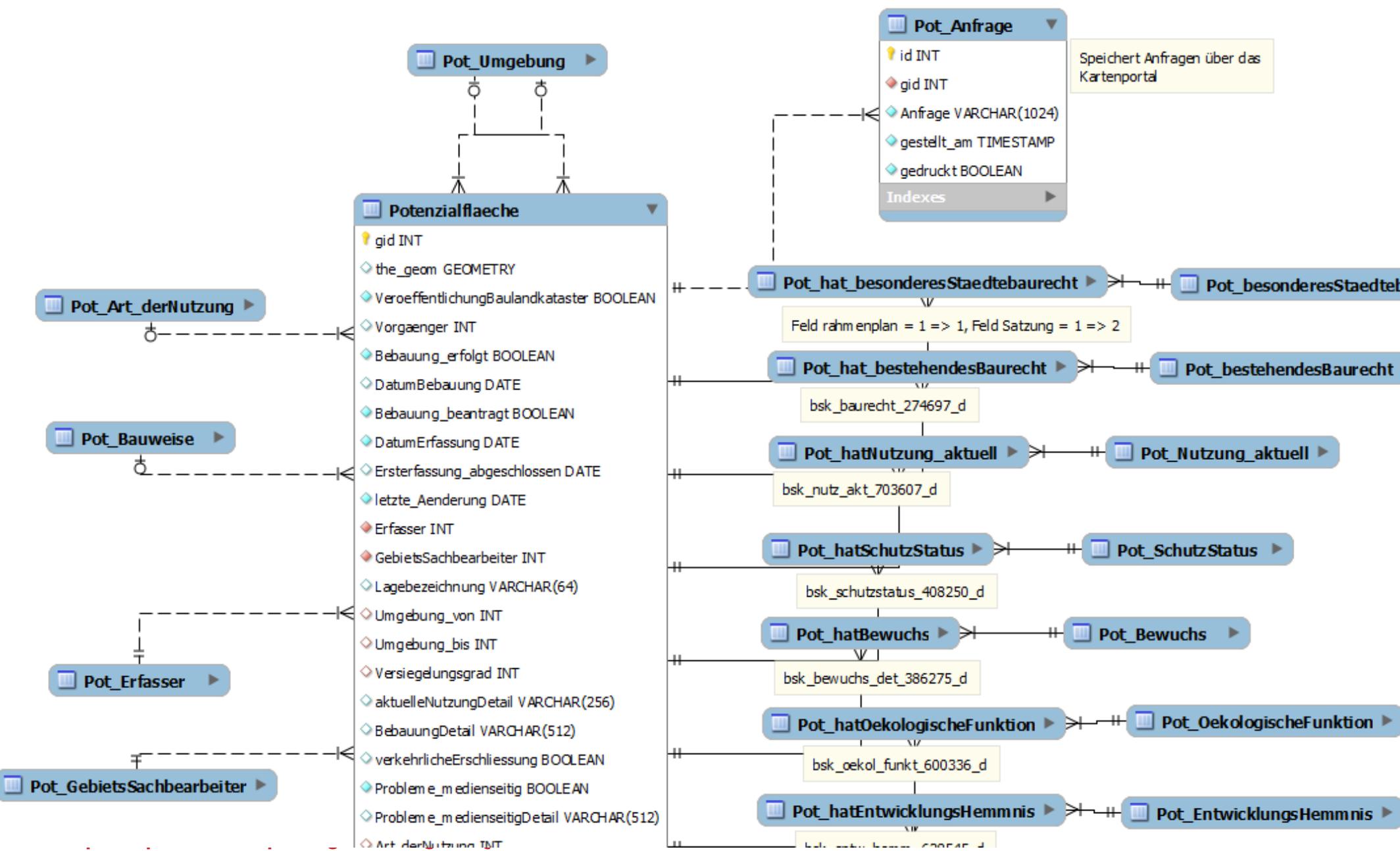


Problemstellung

- Häufige Kombination QGIS mit Datenhaltung in einer relationalen Datenbank z.B. PostgreSQL/PostGIS
- In einer relationalen Datenbank werden die Daten normalisiert gehalten
 - sollten sie zumindest :-)
- Es ist möglich aber umständlich, die Daten aus QGIS heraus abzufragen



Datenmodell Baulandkataster (Ausschnitt)





Eingabemaske

Potenzialfläche - Feature 316

Potenzialfläche | Bestand | Umwelt/Erschliessung | Planung | Mobilisierung | Ziele | Baulandkataster

lfd. Nr. 316

Vorgänger ✕ leer

Bebauung_erfolgt

Bebauung erfolgt am Freitag, 31. Dezember 7999 ✕ leer

Bebauung_beantragt

erfasst am Donnerstag, 4. Juni 2009

Ersterfassung_abgeschlossen Freitag, 31. Dezember 7999 ✕ leer

letzte Änderung Donnerstag, 23. Januar 2014

erfasst von SiemerM

Gebietssachbearbeiter Steger

Lagebezeichnung Kernbergstraße/ Friedrich-Engels-Straße leer

OK Abbrechen Hilfe



Eingabemaske

Potenzialfläche - Feature 316

Potenzialfläche Bestand Umwelt/Erschliessung Planung Mobilisierung Ziele Baulandkataster

Geschossigkeit Umgebung von 1 + D leer

Geschossigkeit Umgebung bis 2 + D leer

Versiegelungsgrad der Potenzialfläche niedrig leer

aktuelle Nutzung

- Garagen
- Grünland
- Bebauung vorhanden
- Brache
- gärtnerische Nutzung
- Kleingarten

Details zur aktuellen Nutzung Grünland mit Garagen leer

Beschreibung der Bebauung massive Doppelgarage; betonierte Einfahrt; massive Einfriedungsmauer leer

Beschreibung des Bewuchses

- Bäume
- Sträucher
- gepflegte Gartennutzung
- verwildert

OK Abbrechen Hilfe



Problemstellung

- Nutzer interessieren sich nicht für Technik und Datenmodelle, sondern für Informationen
 - zu einem Geoobjekt mit allen Relationen
 - zu benachbarten Geoobjekten (z.B. Flurstücke, auf denen eine Fläche liegt)
- Diese Informationen müssen einfach „auf Knopfdruck“ zur Verfügung stehen
- Und das alles in einer „hübschen“ Form, zum ausdrucken oder weitergegeben



Lösung: Reportingwerkzeug

- JasperReports lib (Java, LGPL lizenziert)
<http://community.jaspersoft.com/project/jasperreports-library>
- Aus QGIS über PyJasper (Python, BSD-Lizenz)
ansprechbar
 - <https://github.com/hudora/pyJasper>
→ veraltet, wird offensichtlich nicht mehr gepflegt
 - <https://github.com/bstroebel/pyJasper>
→ fork mit aktueller JasperReports lib



Funktionsumfang JasperReports

- Bilder über URI (z.B. WMS GetMap-Request) oder Dateipfad
- Torten-, Balken- usw. -graphiken
- Tabellen, Crosstabellen
- Unterreports (n:m-Verknüpfungen)
- ...



Für eine QGIS/PyJasper-Lösung benötigen Sie

- Reportvorlagen (JRXML) erzeugt mit



Jaspersoft Studio
WYSIWYG



WYSIWYG-Editor Jaspersoft Studio

The screenshot shows the JasperReports WYSIWYG editor interface. At the top, there are three tabs: 'baulandkataster_sub.jrxml', 'baulandkataster_sub_planung.jrx', and 'Baulandk'. Below the tabs is a 'Main Report' tab. A horizontal ruler at the top shows values from 0 to 500. A vertical ruler on the left shows values from 0 to 400. The main area displays a report design on a grid background. The report is titled 'Planung' and is divided into several sections:

- planungsrechtliche Situation**: Includes fields for 'bestehendes Baurecht:', 'zulässiges Maß der Nutzung:', and 'besondere Städtebaurecht:'. Each field has a small icon to its left.
- Bebauungsziel**: Includes 'mögliche Bauweise:' and 'mögliche Bebauungsform:'. The 'mögliche Bauweise:' field contains the text '\$F{Bauweise}'.
- angestrebte Nutzung und Anzahl möglicher Einheiten**: Includes 'potenzielle Wohnbaufläche, Schätzung Wohneinheiten [WE]' and 'Einfamilien-/Doppelhauspotential:'. The latter field contains the text '"von " + \$F{EFH_von} + " bis " + \$F{EFH_bis}' and a conditional expression '(((\$F{EFH_von} != "k.A." || \$F{EFH_bis} != "k.' followed by '"von " + \$F{MFH_von} + " bis " + \$F{MFH_bis}'.
- Hinweise zur Verwertung und Mobilisierung (unverbindliche bzw. unvollständige Vorabereinschätzung; Prüfung am konkreten Bauvorhaben erforderlich)**: Includes 'mögliche Entwicklungshemmnisse:' and 'Grundstücksneuordnung erforderlich:'. The 'mögliche Entwicklungshemmnisse:' field contains the text '\$F{EntwicklungshemmnisDetail}'. Below this, there are two checkboxes: 'städtebauliche Planung erforderlich' and 'bewusstes Freihalten von Bebauung'.



Für eine QGIS/PyJasper-Lösung benötigen Sie

- Reportvorlagen (JRXML) erzeugt mit



Jaspersoft Studio
WYSIWYG

- Applikationsserver mit



pyJasper
Backend



PyJasper-Backend



- Jetty-Server
- Subreports müssen vom Server aus erreichbar sein

kompiliert
Reportvorlage
→ *.jasper



erzeugt
Report mit
Daten
→ *.pdf



gibt Report
an Client
zurück



Für eine QGIS/PyJasper-Lösung benötigen Sie

- Reportvorlagen (JRXML) erzeugt mit



Jaspersoft Studio
WYSIWYG

- Applikationsserver mit



pyJasper
Backend

- Python-Plugin für QGIS



pyJasper Client

QGIS Python-Plugin



liest Daten
aus PostGIS
Datenbank



formatiert
Daten als xml

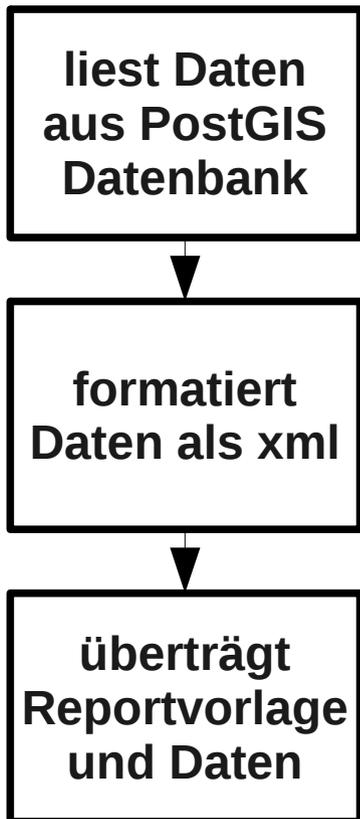
- Implementiert Python-Layeraktion
- z.B. QtSql, pycopg2
- Im Report wird ein XPath-Ausdruck benutzt



Beispiel XML-Daten

```
<QgsJasperReport reportfile="/Baulandkataster/Baulandkataster.jasper">
  <creator>QuantumGIS 2.0.1-Dufour</creator>
  <data>
    <dataset fid="640">
      <letzteAenderung>23.01.2014</letzteAenderung>
      <Daten>
        <Vorgaenger />
        <Bebauung_erfolgt>Bebauung ist erfolgt</Bebauung_erfolgt>
        <DatumBebauung>k.A.</DatumBebauung>
        <Bebauung_beantragt>es liegt kein Bauantrag vor</Bebauung_beantragt>
        <DatumErfassung>17.07.2009</DatumErfassung>
        <Ersterfassung_abgeschlossen>k.A.</Ersterfassung_abgeschlossen>
        <letzteAenderung>23.01.2014</letzteAenderung>
        <Erfasser>SiemerM</Erfasser>
        <GebietsSachbearbeiter>Thierolf</GebietsSachbearbeiter>
        <VeroeffentlichungBaulandkataster>False</VeroeffentlichungBaulandkataster>
        <Veroeffentlichung_widersprochen>False</Veroeffentlichung_widersprochen>
        <DatumWiderspruch />
        <Flaeche>1278</Flaeche>
        <Lagebezeichnung>Ringwiese</Lagebezeichnung>
        <Planungsraum>Winzerla</Planungsraum>
        <alk>
          <item flaeche="832">
            <ort>Burgau</ort>
            <flur>3</flur>
            <flstnummer>99/1</flstnummer>
            <oeff>privat</oeff>
          </item>
        </alk>
      </Daten>
    </dataset>
  </data>
</QgsJasperReport>
```

QGIS Python-Plugin



- Implementiert Python-Layeraktion
- z.B. QtSql, psycopg2
- Im Report wird ein XPath-Ausdruck benutzt

QGIS Python-Plugin



- Speichert pdf-Stream als Datei auf dem Rechner
- Startet Defaultprogramm zum Anzeigen der gespeicherten pdf-Datei



Beispiele

- Baulandkataster
- Gewerbekataster



Hinweise

- Kartendarstellung im Report
 - Über WMS
 - Aus QGIS („Bild speichern als...“) als png und Übertragen des png auf den Server
- Verschneidungen auf der Datenbank als View vorhalten
 - Damit muß im Bedarfsfall nur der View und nicht der Quelltext des Plugins angepasst werden.



Hinweise

- Formatierung des Ausgabebetextes
 - z.B. Bebauung = true/false wird zu „Bebauung ist erfolgt / ist nicht erfolgt“
 - z.B. NULL wird zu „k.A.“
- Vorhalten von universell einsetzbaren Subreports für wiederkehrende Aufgaben
 - z.B. Verschneidungsergebnis Flurkarte
 - z.B. für n:m – verknüpfte Daten



Nutzungsmöglichkeiten

- Datenblätter
- Serienbriefe
- Schadensberichte
-