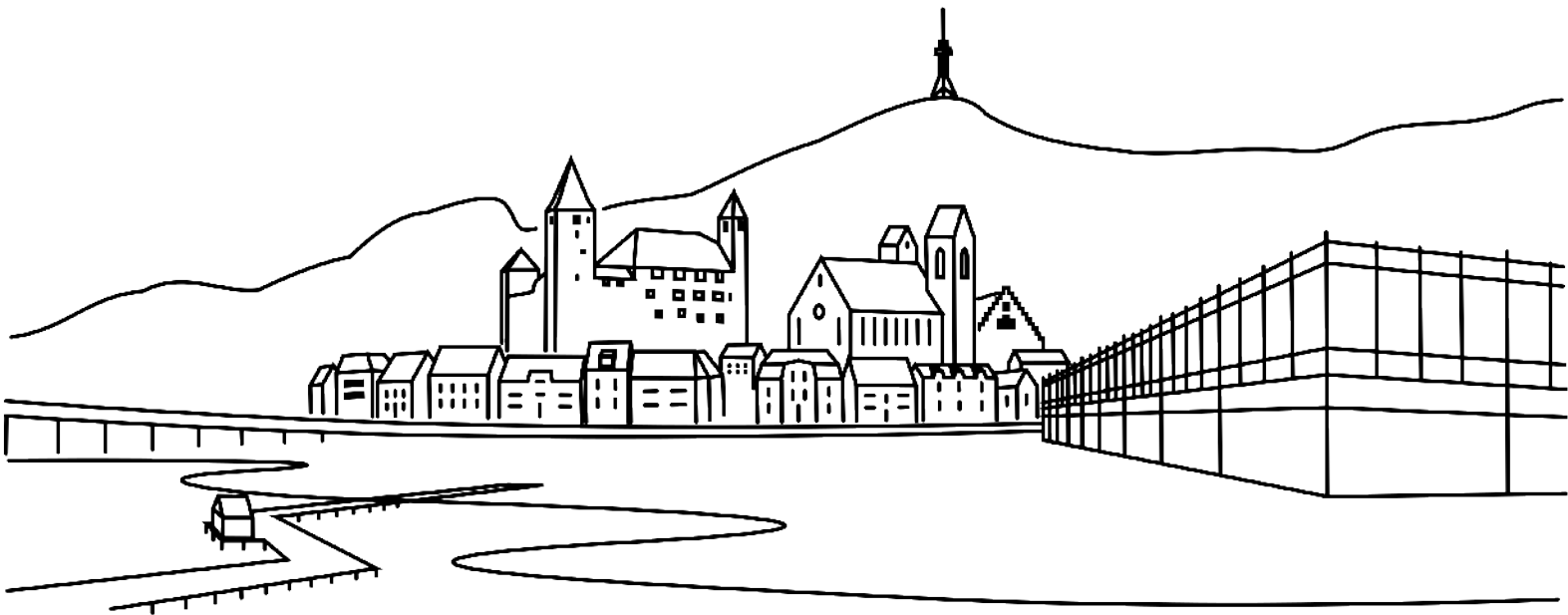




**Anwenderkonferenz für Freie  
und Open Source Software  
für Geoinformationssysteme**

Rapperswil 12. - 14. Juni 2013





Anwenderkonferenz für Freie  
und Open Source Software  
für Geoinformationssysteme



# Inhaltsverzeichnis

Redaktioneller Hinweis:.....	5
3D Landschaftsmodelle aus OSM und SRTM.....	7
3D SlippyMap.....	8
3D-Stadtmodelle in PostGIS mit der 3D City Database.....	9
Administratives PostgreSQL Tuning.....	15
Analyse strukturgeologischer 3D-Daten in PostGIS.....	15
Atlas der Schweiz – online.....	16
Aufbau eines webbasierten Netzinformationssystems mit CAD und freier Software.....	17
Bereitstellung hochaufgelöster Wettervorhersagen für Geoportale.....	19
Cartaro – Geospatial CMS.....	22
CouchWFS Light.....	23
Das OpenStreetMap-Datenmodell.....	32
Das QGEP Abwasser Projekt.....	33
Das SHOGun-WebGIS Framework.....	39
Datenqualität von OSM in der Schweiz.....	40
Die gemeinsame Nutzung von MS-SQLServer und PostgreSQL/PostGIS sowie weiterer OSGeo-Software in einer Fachdatenbank für kulturelles Erbe mit GIS-Anbindung.....	41
Effizientes Mappen von Hausnummern.....	42
Eine Karte in jeder Sprache.....	43
Einführung in das WebFramework GeoDjango.....	44
Entwicklung und Anwendung von Downloaddiensten im Kontext der europäischen Geodateninfrastruktur INSPIRE.....	45
Flexibles System zur geostatistischen Visualisierung am Beispiel von „Personalized Pollen Profiling“ Daten.....	49
FOSS-GIS im Kanton Basel-Landschaft.....	54
Frauen in OpenStreetMap. Oder: Vielfältigere Gruppen für OSM begeistern.....	55
GeoCouch.....	58
Geodaten absichern mit MapProxy.....	59
GeoNetwork 2.8 - Neuerungen.....	63
GeoPackage, das Shapefile der Zukunft.....	65
Geoprocessing mit OpenSource Desktop-GIS.....	67
Georeferenzierung alter Karten mit Grid Shift Files anhand der Tranchot-Karte.....	68
GeoServer.....	69
Im- und Export von dxf-Daten mit OGR.....	70
Kollaboration um ein Open Source generisches Mapping System.....	71
Kort.....	72
Mapbender3 - nimmt Fahrt auf!.....	73
MapFish.....	75
Mapfish Appserver - Funktionsreiches Web-GIS mit Standard-Protokollen.....	76
MATSim: Open-Source-Verkehrssimulation mit offenen Daten.....	77
Metadatenmanagement mit CKAN.....	79
Mobile Karten erstellen mit OSM, OpenLayers und Overpass Api.....	80

Netzwerkanalysen mit GRASS GIS.....	87
Neuentwicklungen in Quantum-GIS.....	87
Offene Standards und frei Software – Zusammenspiel, Entwicklungen, Unterschiede.....	88
Open Standards, Open Source und Open Data: Zuviel des Guten?.....	89
Open Street Map in Freiheit erstart?.....	91
OpenStreetMap- und ALK-Daten gemeinsam nutzen – Eine Karte für Mecklenburg-Vorpommern .....	97
OpenTopoMap.....	100
Ort-Suche mit PostgreSQL/PostGIS.....	101
OSGeo-Live rocks!.....	102
OSM Buildings.....	103
OSM Daten und MariaDB / MySQL.....	104
overpass turbo – Einfache Analyse von OpenStreetMap Daten.....	105
Performanceoptimierte WMS-Dienste mit QGIS Server.....	109
QGIS-Eingabemasken für PostGIS-Layer leicht gemacht.....	115
Qualitätssicherung von Geodaten auf der Basis von Web Processing Services.....	119
Rendering und Bereitstellung massiver Geodaten unter Verwendung von OpenWebGlobe und MapCache in der Cloud.....	120
Schweizmobil Geodatawarehouse.....	126
Semi-automatische Adressdatenerfassung.....	127
Spatial Data Mining: Wenn, dann Open Source.....	128
Stadtplan aus OSM-Daten.....	129
Tile based map publishing with WMTS TileServer, MapTiler an TileMill.....	130
Umstellung auf bw. Einsatz von FOSSGIS in der Voralberger Landesverwaltung: Technische Betrachtung.....	131
Von Vektordaten zum Rasterkartenwerk mit QGIS Server.....	133
WebGL für die Geomatik.....	136
Wer ist der Boss bei OpenStreetMap?.....	136
Werkzeuge zur Nutzung von OpenStreetMap-Daten in GIS.....	137
Wird das Geoportal der Verwaltung genutzt?: Metrik der Dienste von geo.admin.ch.....	139
Geo-Metadaten in OpenData Portalen.....	141
OSGeo-Live rocks!.....	153

Redaktioneller Hinweis:

Diese Downloadversion des Tagungsbandes wurde um die nachgereichten Beiträge *Geo-Metadaten in OpenData Portalen* und *OSGeo-Live rocks!* ergänzt.

## 3D Landschaftsmodelle aus OSM und SRTM

### Wie OSM2World die OpenStreetMap-Daten um die dritte Dimension ergänzt

TOBIAS KNERR

Speziell im deutschsprachigen Raum enthält OpenStreetMap zunehmend hochdetaillierte Informationen, dank derer sich eine dreidimensionale Szene etwa mit Gebäudemodellen, spurgenaue Straßendarstellungen und Straßenmöblierung ausstatten lässt. Auch relative Höheninformationen wie Steigungen, Stockwerke oder die vertikale Anordnung von Brücken und Tunneln werden bereits vielerorts eingetragen.

Die Erfassung absoluter Höheninformationen ist jedoch unüblich und bleibt in der Praxis vereinzelt Landschaftspunkten wie Gipfeln vorbehalten. Weder die Bedienkonzepte der OSM-Editoren, noch die üblichen Datenquellen (Luftbilder und GPS-Daten) eignen sich gut für diese Aufgabe. Daher muss hier ein anderer Ansatz gewählt werden: Der Rückgriff auf externe Quellen wie die SRTM-Satellitendaten der NASA.

Der Vortrag stellt einen Ansatz zur Verschmelzung von OSM- und SRTM-Daten vor und beschreibt dessen Umsetzung in OSM2World, einem Open-Source-Werkzeug zur Erzeugung von 3D-Landschaftsmodellen: Mithilfe von passend gewählten Interpolationsverfahren wird zunächst aus den SRTM-Messpunkten und gegebenenfalls zusätzlichen Landschaftsmerkmalen eine Geländeoberfläche berechnet. Für diesen Schritt wurde u.a. mit Least-Squares- und Natural-Neighbor-Algorithmen experimentiert. Anschließend werden Modelle mittels eines Optimierungsverfahrens so in das Terrain eingepasst, dass implizit und explizit vorhandene Informationen etwa zur Steigung in Längs- und Querrichtung oder zum lichten Raum über Verkehrswegen berücksichtigt werden.

Links:

<http://osm2world.org/>

<http://tobias-knerr.de/publications/>

### 3D SlippyMap

#### Rendering einer 3D-Deutschland-Karte mit Hilfe von OSM2World

PETER BARTH

Die Darstellung von OpenStreetMap-Daten in Webseiten geschieht in den sogenannten Slippymaps. Diese Javascript-Anzeige holt sich dabei dynamisch die anzuzeigenden Kacheln von einem Server. Je nach Kartenart und Stil sind dies unterschiedliche Dienste. So auch die OSM2World-Karte, eine 3D-Darstellung von Deutschland.

Das Ausliefern der Kacheln geschieht dabei oft mit renderd oder tirex. Beiden gemein ist das effektive Caching und on-the-fly Rendern der benötigten Kacheln. Auf diese Art muss nur ein sehr kleiner Teil der Kacheln (unter 2%) auf der Platte vorgehalten werden.

Was aber, wenn das verwendete Renderprogramm zu langsam für einen Livebetrieb ist? Dieser Vortrag bietet interessante Einblicke in die Herausforderungen, Probleme und Lösungsansätze im Falle der OSM2World-SlippyMap. So wird zum einen die Toolchain von Datenaufbereitung bis zur fertigen Kachel vorgestellt und auch einige interessante Ansätze gezeigt, die in der Vergangenheit immer wieder die Performance verbesserten. Zum anderen werden identifizierte Hardwaregrenzen besprochen und Möglichkeiten, diese auszureizen bzw. zu überwinden erklärt. Im Ausblick werden Ideen zur Performancesteigerung genannt, die in weiteren Entwicklungen angedacht sind.

Links:

<http://maps.osm2world.org/>

# 3D-Stadtmodelle in PostGIS mit der 3D City Database

FELIX KUNDE

Mit der 3D City Database, in der Folge mit 3DCityDB abgekürzt, ist es möglich CityGML-basierte virtuelle 3D-Stadtmodelle in PostgreSQL/PostGIS oder Oracle Spatial zu speichern. Die Software umfasst zum Einen ein relationales Datenbank-Schema, welches auf dem Datenbank Management System (DBMS) aufsetzt, und zum Anderen die Java-basierte Client-Anwendung Importer/Exporter, mit dessen Hilfe Stadtmodell-Daten im CityGML Format unabhängig von Größe und Komplexität hochperformant in die Datenbank hinein und wieder heraus geschrieben werden können. Die 3DCityDB stellt damit eine einfach zu bedienende, leistungsfähige und vor allem offene Schnittstelle zum Verwalten, Analysieren und Austauschen von 3D-Stadtmodellen bereit. Die Software ist ursprünglich aus dem Forschungsprojekt „Geodatenmanagement in der Berliner Verwaltung - Amtliches 3D-Stadtmodell für Berlin“ [1] hervorgegangen und wurde maßgeblich von dem Institut für Kartographie und Geoinformation (IKG) an der Uni Bonn und dem Institut für Geodäsie und Geoinformationstechnik (IGG) an der TU Berlin unter der Leitung von Thomas H. Kolbe entwickelt [2]. Das Projekt hatte zum Ziel eine Software-Schnittstelle für die Arbeit mit CityGML-Daten zu schaffen. Die Software ist frei verfügbar unter [3dcitydb.net](http://3dcitydb.net) [3] und quelloffen nach LGPL v3.

## Was ist CityGML?

CityGML ist ein Anwendungsschema der Geography Markup Language (GML). Es stellt ein offenes Austauschformat für 3D-Stadtmodelle dar und wird seit 2002 von der Special Interest Group 3D (SIG 3D), einem Gremium aus Vertretern der Wirtschaft, Wissenschaft und Öffentlichen Verwaltung ([sig3d.de](http://sig3d.de)), konzipiert und entwickelt. Seit 2008 ist CityGML ein offizieller OGC-Standard [4]. Es deckt mit seinem modular aufgebauten Datenmodell verschiedenste thematische Bereiche einer urbanen Infrastruktur ab (neben Gebäuden u.a. auch Verkehr, Vegetation, Landnutzung, Gewässer), lässt sich darüber hinaus aber noch über eine generische Schnittstelle (Application Domain Extension – ADE) erweitern.

Eine entscheidende Stärke von CityGML gegenüber anderen 3D-Formaten ist die semantische Tiefe, welche den Modellen verliehen werden kann. Bestehen Stadtobjekte bei gängigen Visualisierungsformaten meist aus einer nicht näher definierten Ansammlung von Geometrien und Texturen, können bei CityGML nicht nur die Objekte selbst sondern auch einzelne Komponenten der Objekte thematisch klassifiziert werden, wie z.B. Wände, Dächer oder Fenster eines Gebäudes etc. [5]. Dieser hierarchische Ansatz kann sich je nach Zweck und Anforderung an Genauigkeit auf bis zu fünf unterschiedliche Maßstabsebenen, so genannter „Levels of Detail“ (LOD), ausdehnen. Die Breite an semantischer Qualifizierung ermöglicht die Integration von 3D-Geodaten aus heterogenen Quellen in einem Stadtmodell und macht es somit für verschiedenste Anwendungen nutzbar. Die Beispiele reichen von der einfachen Visualisierung über komplexe räumliche Analysen bis hin zu Simulationsanwendungen in unterschiedlichsten Fachdomänen wie beispielsweise Stadt- und Umweltplanung, Energiewirtschaft, Katastrophenmanagement oder Facility Management.



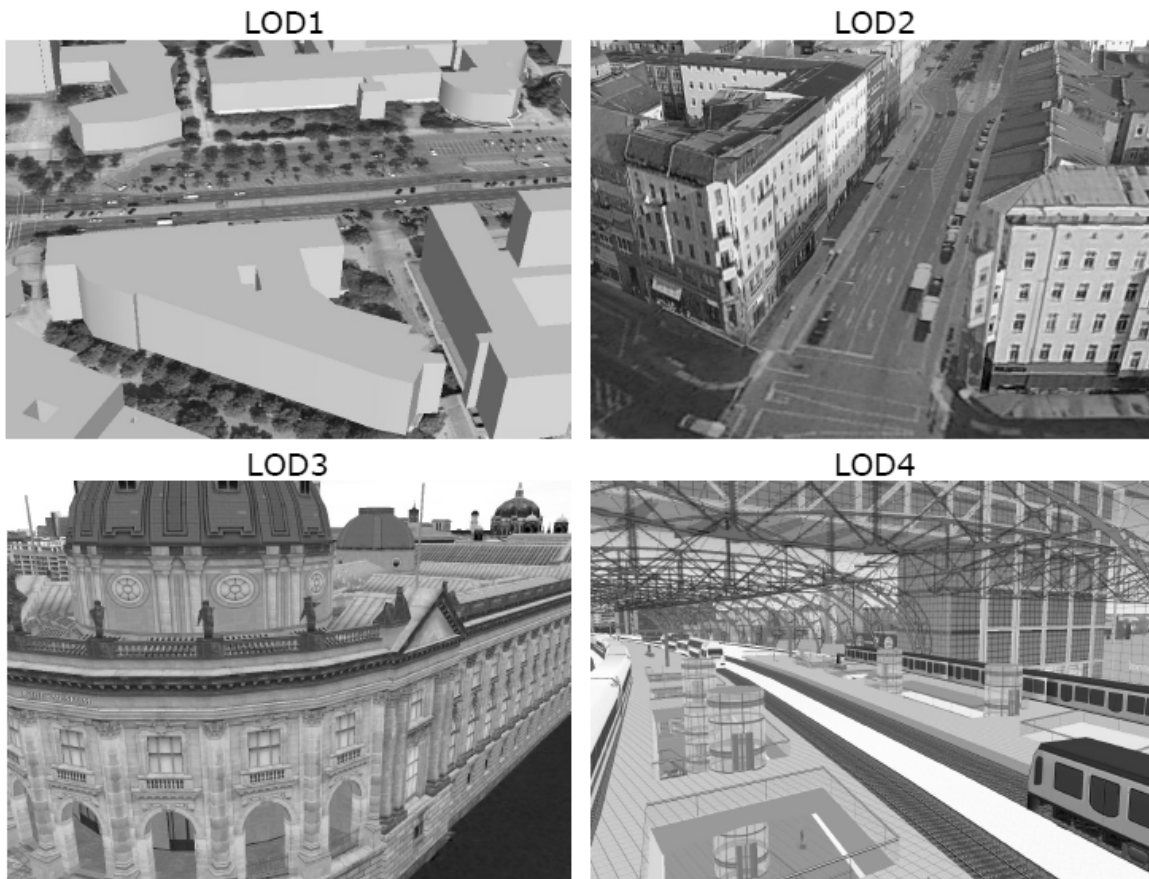


Abbildung 1: CityGML Gebäudepräsentation in verschiedenen Level of Detail (LOD). Die Beispiele entstammen dem Berliner 3D Stadtmodell. [12]

Aufgrund seiner breiten Akzeptanz in der Fachwelt, dient CityGML auch als Grundlage für weitere Datenmodelle im Bereich der dreidimensionalen Gebäudemodellierung, z.B. INSPIRE Building 3D [6], Adv-CityGML-Profil (ALKIS 3D) [7], IMGeo (Niederländische GDI) [8].

## Features

### 3DCityDB:

- Verfügbar für *Oracle Spatial* ab 10g und für *PostgreSQL 8,4+* mit *PostGIS 2.0+*
- Geometrisch-semantische Modellierung in fünf LOD-Stufen
- Speicherung von zusätzlichen Oberflächeneigenschaften wie Texturen, Materialien, aber auch nicht sichtbare Phänomene wie Wärmeabstrahlung, Lärmemission, -etc.
- Speicherung von digitalen Geländemodellen (DGM) als Raster, TIN, 3D-Punktwolke oder 3D-Bruchkanten
- Speicherung von Luftbildern
- Prototypische Repräsentation von Objekten, die in gleicher Form an vielen Stellen im Stadtmodell existieren (Bäume, Stadtmöbel, Straßenlaternen, Schilder etc.)
- Referenzierung von externen Datenquellen
- Geometrische Erfassung von 3D-Objekten anhand ihrer Begrenzungsflächen um die verschiedenen Ausprägungen in Solids, Surfaces oder ihrer Aggregation in einer Tabelle organisieren zu können
- Rekursive Gruppierungsmöglichkeit von Objekten

### 3D-Stadtmodelle in PostGIS mit der 3D City Database

- Historien-Management der Datenbank, u.a. auch für das Erstellung von Planungsszenarios (dieses Feature existiert nur für die *Oracle*-Version)

#### Importer/Exporter:

- Unterstützt derzeit CityGML 0.4.0 und 1.0.0
- Exportiert auch in das KML/COLLADA Format
- XML Validierung von CityGML Dokumenten
- Anlegen von benutzer-definierten Koordinatensystemen in 2D und 3D
- Affine Transformation beim CityGML Import
- Koordinatentransformation beim CityGML Export
- Kartenfenster zum Bestimmen eines Auswahlrechtecks (Bounding Box)
- Inkrementell erweiterbar durch Plugins
- Matching / Merging von Gebäuden
- Spreadsheet Im- und Export
- Technische Details:
  1. Programmiert in Java
  2. Verwendete Frameworks: citygml4j, JDBC, JAXB, SAX, Swing
  3. Modulare Softwarearchitektur
  4. Liest und schreibt CityGML-Dateien unabhängig von Größe und Komplexität
  5. Nebenläufige Programmierung für hoch-performantes Prozessieren der Daten
  6. Vor- und Rückwärtsauflösung von Xlinks
  7. Unterstützte Betriebssysteme sind *Windows*, *MAC OS*, *Linux/UNIX* und seine Derivate
  8. Quellcode verfügbar unter: <http://opportunity.bv.tu-berlin.de/software>

### Aufbau der 3DCityDB

Das relationale Schema der 3DCityDB ist aus dem Datenmodell von CityGML abgeleitet. Für das Mapping wurde der Anspruch verfolgt das objektorientierte Modell von CityGML zugunsten einer optimalen Datenbank-Performance zu vereinfachen, die Inhalte aber weitestgehend verlustfrei abzubilden [9].

Das geometrische Modell von CityGML wurde im Datenbankschema auf die Speicherung von flächenhaften Geometrien reduziert. Komplexe Geometrien wie Solids, TINs und sonstige Aggregationen von Polygonen werden in ihre einzelnen Flächenprimitive zerlegt, die über zusätzliche Tabellenspalten mit ihren Eltern- und Wurzelementen verknüpft werden können (siehe Abb.2).

## 3D-Stadtmodelle in PostGIS mit der 3D City Database

```

<bldg:lod1Solid>
  <gml:Solid>
    <gml:exterior>
      <gml:CompositeSurface gml:id="lod1Surface">
        <gml:surfaceMember>
          <gml:Polygon gml:id="Left1">
            <gml:exterior>
              <gml:LinearRing gml:id="LeftRing1">
                <gml:posList srsDimension="3"> 0.0 0.0 0.0 10.0
                0.0 0.0 10.0 0.0 4.0 0.0 0.0 4.0 0.0 0.0 0.0
              </gml:posList>
            </gml:LinearRing>
          </gml:exterior>
        </gml:Polygon>
      </gml:surfaceMember>
      ...
      <gml:surfaceMember>
        <gml:Polygon gml:id="Roof1">
          <gml:exterior>
            <gml:LinearRing gml:id="RoofRing">
              <gml:posList srsDimension="3">
                0.0 4.0 10.0 5.0 4.0 0.0 5.0 4
              </gml:posList>
            </gml:LinearRing>
          </gml:exterior>
        </gml:Polygon>
      </gml:surfaceMember>
    </gml:CompositeSurface>
  </gml:exterior>
</gml:Solid>
</bldg:lod1Solid>

```

SURFACE_GEOMETRY						
ID	GMLID	PARENT_ID	ROOT_ID	IS_SOLID	IS_COMPOSITE	GEOMETRY
1	UUID		1	1	0	
2	lod1Surface	1	1	0	1	
3	Left1	2	1	0	0	SDO_GEOMETRY
4	Front1	2	1	0	0	SDO_GEOMETRY
5	Right1	2	1	0	0	SDO_GEOMETRY
6	Back1	2	1	0	0	SDO_GEOMETRY
7	Roof1	2	1	0	0	SDO_GEOMETRY

Abbildung 2: Verwaltung komplexer GML-Geometrien in der Datenbank [2]

Auf diese Weise können den Einzelflächen beliebig viele Oberflächeneigenschaften wie Texturen oder Materialinformationen zugewiesen werden. Durch die Referenzierung des Wurzelements von jeder Flächengeometrie können komplexe Geometrieobjekte mit einem einzigen SELECT Statement abgefragt werden.

Neben den Tabellen umfasst die 3DCityDB auch Datenbank Prozeduren, die hauptsächlich für Funktionalitäten des Importer/Exporter verwendet werden. Diese sind in Oracle (PL/SQL) in Packages und in PostgreSQL (PL/pgSQL) in einem separaten Schema organisiert. Für die Oracle Version besteht darüber hinaus die Möglichkeit auf Basis des mit dem Oracle Workspace Managers Planungsszenarien in der 3DCityDB anzulegen, um verschiedene Versionen und Planungsalternativen der Stadtmodelldaten zu verwalten. Leider bietet PostgreSQL bis dato kein vergleichbares Werkzeug an, so dass die PostgreSQL/PostGIS Version der 3DCityDB derzeit ohne dieses Feature auskommen muss.

### Datenaustausch mit der 3DCityDB

Auch wenn das relationale Datenbankschema aus dem Datenmodell von CityGML abgeleitet wurde, ist die 3DCityDB nicht nur auf den Datenaustausch von CityGML-Daten beschränkt. Mit Hilfe von Konvertern, wie z.B. FME von Safe Software, können auch andere semantisch weniger umfangreiche 3D-Formate (3D Shape, 3ds, dxf, osm3D) in der Datenbank aufgehen bzw. aus ihr exportiert werden [10]. Die 3DCityDB kann deshalb als zentrale Datenhaltungsschnittstelle eingesetzt werden, die einerseits Stadtmodelldaten aus verschiedenen Quellen in sich vereint und andererseits zweckorientierte Ausgaben für Desktop- oder Webanwendungen liefert.

### 3DCityDB in Action

Neben der Verwendung in zahlreichen Forschungsprojekten, u.a. dem Energie Atlas Berlin (siehe Abb.3) [11], wird die 3DCityDB auch in einem kommerziellen Umfeld von den Firmen virtualcitySYSTEMS und M.O.S.S. Computer Grafik Systeme eingesetzt, u.a. in Stadtverwaltungen (Berlin, Potsdam, Nürnberg, Zürich, Dresden) und Landesvermessungs-ämtern (Baden-Württemberg, Bayern, Hessen, Rheinland-Pfalz, Sachsen, Thüringen).

### 3D-Stadtmodelle in PostGIS mit der 3D City Database

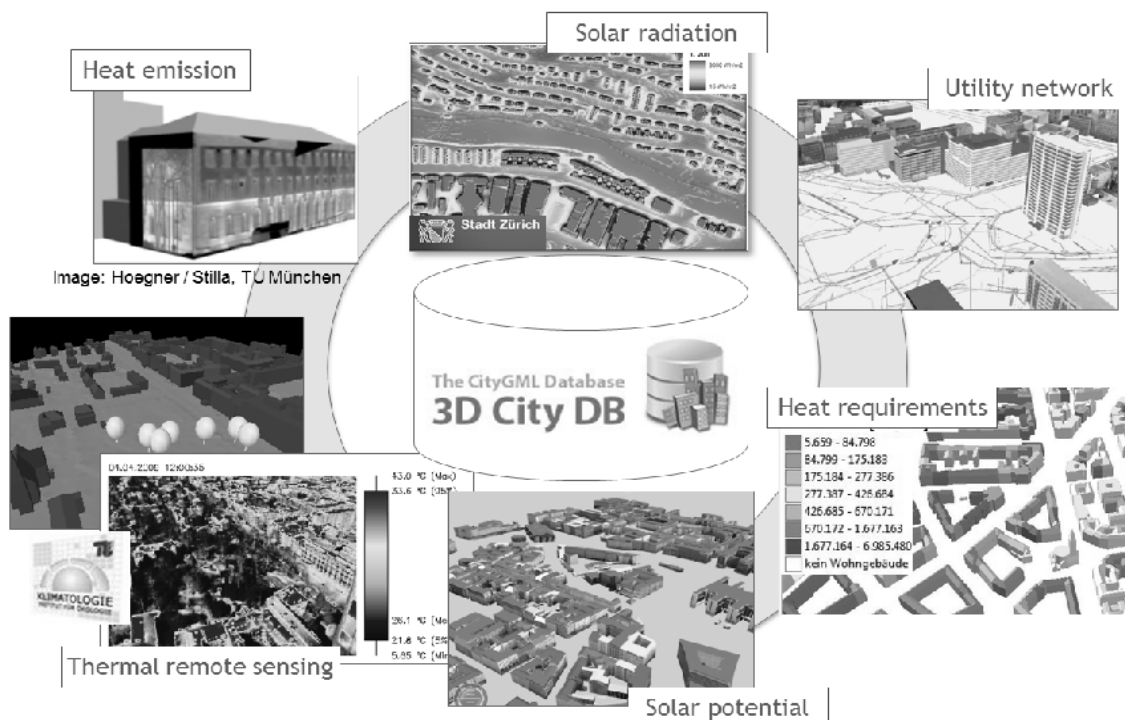


Abbildung 3: Zusammenführung energierelevanter Informationen im Energieatlas Berlin [11]

Die Freie und Hansestadt Hamburg wird innerhalb der ersten Jahreshälfte 2013 als erste Stadt ihr 3D-Stadtmodell mit der seit Juli 2012 erhältlichen PostgreSQL/PostGIS-Version der 3DCityDB [12] im Pilotbetrieb verwalten. Angesichts der Bestimmungen der Arbeitsgemeinschaft der Vermessungsverwaltungen der Länder (AdV), gemäß denen in den kommenden Jahren flächendeckend für Deutschland LOD1- und LOD2-Modelle von den zuständigen Katasterämtern zu führen sind [14], kann davon ausgegangen werden, dass die Nutzung einer gänzlichen kostenfreien Datenbanklösung für weitere Kommunen in Betracht kommen wird.

#### Zukünftige Entwicklungen

Geplant ist, dass die 3DCityDB in der zweiten Jahreshälfte 2013 einen neuen Major-Release erfahren wird. Es folgt die Anpassung auf CityGML 2.0.0, das im April 2012 vom OGC verabschiedet wurde, sowie weitere Optimierungen des Datenbankschemas. Für das Importer/Exporter Tool ist vorgesehen die derzeit parallel entwickelten Versionen für Oracle und PostgreSQL/PostGIS in einer Software zusammenzuführen. Langfristig ist außerdem eine generische Unterstützung beliebiger ADEs geplant.

Im Februar dieses Jahres haben Prof. Kolbe, virtualcitySYSTEMS und M.O.S.S. in einer Kooperationsvereinbarung beschlossen, die Weiterentwicklung der 3DCityDB gemeinsam voranzutreiben. Dadurch wurde ein Forum geschaffen, dem sich einerseits weitere Entwickler und Unternehmen anschließen können, das andererseits aber auch die Nutzer der 3DCityDB vernetzen soll [14].

### 3D-Stadtmodelle in PostGIS mit der 3D City Database

Kontakt zum Autor:

Felix Kunde  
virtualcitySYSTEMS GmbH  
Cicerostraße 21  
+49(0)30 890 48 71-42  
fkunde[at]virtualcitysystems.de

Ko-Autoren: Claus Nagel, Javier Herrerruela, Lutz Ross, Thomas H. Kolbe

#### Literatur

- [1] Casper, E. (2008): Geodatenmanagement in der Berliner Verwaltung – Amtliches 3D Stadtmodell für Berlin. Projektbericht 2007. Verfügbar unter:  
[http://www.businesslocationcenter.de/imperia/md/content/3d/efre\\_ii\\_projektdokumentation.pdf](http://www.businesslocationcenter.de/imperia/md/content/3d/efre_ii_projektdokumentation.pdf)
- [2] Kolbe, T.H.; König, G.; Nagel, C.; Stadler, A. (2009): 3D-Geo-Database for CityGML. Version 2.0.1. Documentation. Berlin. Verfügbar unter:[www.3dcitydb.net](http://www.3dcitydb.net)
- [3] <http://www.3dcitydb.net/>
- [4] <http://www.opengeospatial.org/standards/citygml>
- [5] Stadler, A. (2008): Kohärenz von Geometrie und Semantik in der Modellierung von 3D Stadtmodellen. In: Clemen, C. (Hrsg.): Entwicklerforum Geoinformationstechnik 2008. Shaker, Aachen. 167-181.
- [6] INSPIRE Thematic Working Group Building (TWG BU) (2011): INSPIRE Data Specification on Building – Draft Guidelines. Version D2.8.III.2.
- [7] Gruber, U. (2012): Die dritte Dimension im amtlichen Vermessungswesen Deutschlands. Vortrag am 29.03.2012. Geoinformatik 2012. Verfügbar unter:  
[http://www.kreis-re.de/dok/Katasteramt/File/Geoinformatik2012\\_Gruber.pdf](http://www.kreis-re.de/dok/Katasteramt/File/Geoinformatik2012_Gruber.pdf)
- [8] <http://www.geonovum.nl/>
- [9] Nagel, C. ; Stadler, A. (2008): Die Oracle-Schnittstelle des Berliner 3D-Stadtmodells. In: Clemen, C. (Hrsg.): Entwicklerforum Geoinformationstechnik 2008. Shaker, Aachen. 197-221.
- [10] <http://www.business-geomatics.com/online/unternehmen-a-maerkte/47-unternehmen-a-maerkte/1119-3d-auf-knopfdruck.html>
- [11] Kolbe, T.H. (2012): Neue Entwicklungen zu CityGML und der 3DCityDB. Vortrag am 31.01.2012. Räumliche GDI und 3D-GIS-Technologien, Dresden. Verfügbar unter:  
[http://gdi-sachsen.de/web/2012/Workshop/1\\_CityGML\\_3DCityDB\\_Kolbe\\_TU\\_Berlin.pdf](http://gdi-sachsen.de/web/2012/Workshop/1_CityGML_3DCityDB_Kolbe_TU_Berlin.pdf)
- [12] Kunde, F (2012): CityGML in PostGIS: Portierung, Anwendung und Performanz-Analyse am Beispiel der 3D City Database von Berlin. Masterarbeit. Verfügbar unter:  
[urn:nbn:de:kobv:517-opus-63656](http://nbn:de:kobv:517-opus-63656)
- [13] GeoInfoDok (2009): Dokumentation zur Modellierung der Geoinformationen des amtlichen Vermessungswesens. Hauptdokument Version 6.0.1. Arbeitsgemeinschaft der Vermessungsverwaltungen der Länder der Bundesrepublik Deutschland.
- [14] <http://gispoint.de/news-einzelansicht/514-stadtmodellspezialisten-schliessen-kooperationsvereinbarung.html>

## Administratives PostgreSQL Tuning

SUSANNE EBRECHT

Viele sehen PostGIS als das ideale Tool zur Verwaltung von Kartendaten. PostGIS ist ein Aufsatz auf das Datenbanksystem PostgreSQL. Der PostgreSQL-Server hat mehr als 100 Stellschrauben, an denen gedreht werden kann, doch für ein gutes Tuning auf administrativer Ebene sind nur ein paar wenige interessant. Der Vortrag gibt Aufschluss darüber, was auf administrativer Ebene für ein gutes PostgreSQL-Tuning notwendig und was zu beachten ist.

Viele sehen PostGIS als das ideale Tool zur Verwaltung von Kartendaten. PostGIS ist ein Aufsatz auf das Datenbanksystem PostgreSQL. Der PostgreSQL-Server hat mehr als 100 Stellschrauben, an denen gedreht werden kann, doch für ein gutes Tuning auf administrativer Ebene sind nur ein paar wenige interessant. Der Vortrag gibt Aufschluss darüber, was auf administrativer Ebene für ein gutes PostgreSQL-Tuning notwendig und was zu beachten ist.

## Analyse strukturgeologischer 3D-Daten in PostGIS

### Eine Entscheidungsgrundlage bei der Bewilligung von Erdwärmesonden

ULI MÖLLER

Im Zuge steigender Energiekosten werden geothermische Heizanlagen in der Schweiz immer beliebter. Um Risiken bei der Anlage von Erdwärmesonden zu vermeiden ist es entscheidend, neben zweidimensionalen Oberflächendaten auch die räumliche Geometrie der im Untergrund befindlichen Schichten zu kennen. Zur Entscheidungsfindung für die Bewilligung von Erdsonden wurde eine Abfrageroutine in PostGIS entwickelt. Die geologischen Schichten wurden in GOCAD, einer proprietären geologischen Software, modelliert und als 3D-Geometrien in PostGIS importiert. Die Abfrageroutine liefert für jeden Punkt auf Basis einer langen Reihe von Kriterien als Ergebnis die Zulässigkeit einer Erdwärmesonde, die zulässige Bohrtiefe sowie einen spezifischen erläuternden Text mit Begründungen und allfälligen Auflagen. Der Vortrag demonstriert einerseits die Abfrageroutine und geht weiter auf die Möglichkeiten und Grenzen der Ablage und Analyse von geologischen 3D-Daten in PostGIS ein.

## Atlas der Schweiz – online

### 3D Visualisierung von Geodaten in Echtzeit

RENÉ SIEBER

Digitale Atlanten und damit auch der ATLAS DER SCHWEIZ müssen neue Wege einschlagen, um gegenüber frei verfügbaren, kostenlosen Kartendiensten, Geo-portalen und Virtuellen Globen konkurrenzfähig zu bleiben. Eine Analyse dieser Geovisualisierungsprodukte hat gezeigt, dass die Mehrzahl der Anwendungen primär für den Einsatz im Web konzipiert ist. Die Attraktivität solcher Produkte basiert auf dem unmittelbaren Nutzen im Alltag, der Aktualität der Daten und den integrativen Möglichkeiten. Anwendungen auf der Basis von Virtuellen Globen bestechen zudem durch die intuitiv navigierbare Drei-dimensionalität. Die inhaltliche und kartografische Qualität dieser Applikationen ist hingegen sehr heterogen; oft fehlt auch eine atlaspezifische Funktionalität. Die grosse Herausforderung und das Ziel des ATLAS DER SCHWEIZ ist es, einen Webatlas zu entwickeln und bereitzustellen, welcher hochqualitative 2D- und 3D-Visualisierungen und Interaktionen in Echtzeit unterstützt. Als Visualisierungs-Engine wird dabei die OpenSource-Software osgEarth eingesetzt. Dieser auf OpenSceneGraph basierende Virtuelle Globus bietet u.a. eine breite Palette an Visualisierungs- und Navigationstechniken, die Kombination von Höhenmodellen, die Einbindung diverser Datenformate und Webservices, sowie verschiedene geografische Referenzsysteme an. Der Vortrag zeigt anhand von Beispielen und einer Live-Demo unseres online 3D-Atlas-Prototyps auf, wie kartografisch aufbereitete Geoinformationen mit osgEarth visualisiert werden können und welche Chancen und Herausforderungen sich dadurch ergeben.

## Aufbau eines webbasierten Netzinformationssystems mit CAD und freier Software

OLAF KNOPP

Ausgangssituation Die Werraenergie GmbH in Bad Salzungen ist als regionaler Energiedienstleister zuständig für die Belieferung von rund 20.000 Haushalten in zwei Thüringer Landkreisen mit Gas und Wärme. Anfang 2013 übernahm das Unternehmen zudem den Betrieb des regionalen Stromnetzes und beliefert seitdem zusätzlich rund 6.000 Haushalte mit Strom. Insgesamt werden zur Zeit 826 km Rohrnetz betreut, die Nacherfassung des Stromnetzes hält noch an.

Neben der Aufgabe als reiner Energielieferant agiert die Werraenergie auch als technischer Betreiber der Energienetze sowie als Beratungsstelle für Energie und Ressourcenschutz. Dementsprechend umfangreich sind auch die Anforderungen der verschiedenen Fachbereiche an Software zur Wahrnehmung der jeweiligen Aufgaben.

Im Zuge der Übernahme des Stromnetzes und den daraus resultierenden Anforderungen an Netzdokumentation, Beauskunftung und Planung, entschied sich das Unternehmen für die Ablösung der bestehenden dezentralen Softwarelösung durch ein zentrales webbasiertes Netzdokumentationssystem. Folgende Module sollten umgesetzt werden:

Erfassung und Pflege des Netzes über bewährte CAD-Software  
Speicherung der Daten in einer zentralen Datenbank  
Umsetzung verschiedener Webapplikationen für OGC-konformes WebGIS  
Netzauskunft  
Wartungsmanagement  
Dokumentenmanagement  
Trassenplanung  
Störungsmanagement  
mobiles GIS

### Umsetzung

Die Umsetzung erfolgte als OGC-konforme Geodateninfrastruktur, da sowohl die Übernahme von Daten der Landesvermessung (Luftbilder, Liegenschaftsdaten) als auch die Abgabe von Daten in Form von Diensten (z.B. für die Webauskunft) vorgesehen ist. Zum Einsatz kamen neben OpenSource-Software wie MapServer, PostgreSQL/PostGIS und Mapbender3 auch AutoCAD Map.

### Datenübernahme

Die Erfassung der Netzdaten erfolgte bisher ausschließlich über CAD in eine Oracle-Datenbank. Das vorliegende Datenmodell musste daher zunächst in eine OGC-konforme Struktur überführt werden. Zum Import der Daten in die PostgreSQL-Datenbank wurde die freie Software GeoKettle verwendet, ein räumliches ETL-Programm, mit dem Geodaten extrahiert, transformiert und geladen werden können. Ziel war es, die im alten Datenmodell in verschiedenen Tabellen verteilt vorliegenden Daten in einer Tabelle zu aggregieren. Sachdaten und Geometrien (als Textfelder mit X/Y-Werten) waren zudem auf verschiedene Tabellen verteilt und sollten ebenfalls zusammengeführt werden. Ein weiteres „CAD-typisches“ Problem war, dass in den Tabellen neben dem eigentlichen Objekt auch dessen Beschriftung als eigener Datensatz vorlag. Diese sollten voneinander getrennt und in einer separaten Annotation-Tabelle abgelegt werden. Für die verschiedenen datenmodellierenden Aufgaben wurden mit GeoKettle Transformationsmodelle erstellt, die für alle zukünftigen Importe eingesetzt werden können. Diese Modelle enthalten ähnlich einem Skript alle notwendigen Prozesse für die Datenmigration, angefangen beim Einlesen der Daten aus den verschiedenen Datenquellen über die Transformationen der Daten in das neue Modell bis hin zur Erzeugung der neuen Datenbankstruktur und dessen Befüllung.

Ergebnis der Migration war ein objektbasiertes Datenmodell mit klarer Zuordnung von Lage, Attributen und Geometrien im OGC simple feature model.



### Erfassung und Pflege mit AutoCAD Map

Die Erfassung und Pflege der Objekte sollte auch weiterhin an leistungsstarken Desktop-Arbeitsplätzen mittels CAD erfolgen. Die Wahl fiel auf AutoCAD Map mit konfektionierten Erfassungsmasken. Durch Nutzung der OSGeo-Software FDO Data Access Technology kann das CAD direkt in die zentrale PostgreSQL-Datenbank speichern. Durch die Software konnte ein direkter Datenaustausch zwischen Erfassungsplätzen und Fachanwendungen eingerichtet werden, der die permanente Aktualität der Datengrundlage für alle Anwendungen sicherstellt.

### Webapplikationen

Bei den bisher eingesetzten Fachanwendungen handelte es sich um Desktop-Programme, die zum Teil auf separate Datenquellen zugriffen. Eine zentrale Datenbank für alle Anwendungen existierte nicht. Ziel des neuen Netzinformationssystems sollte es dementsprechend sein, den aktuellen Leitungsdatenbestand für die oben genannten Anwendungen verfügbar zu machen. Die Umsetzung sollte webbasiert erfolgen; zum einen, um arbeitsplatzunabhängig auf alle Applikationen zugreifen zu können, zum anderen um zukünftig weitere Anwendungen wie eine Internetauskunft oder mobile Lösungen realisieren zu können.

Die Umsetzung erfolgte mit UMN MapServer und Mapbender3.

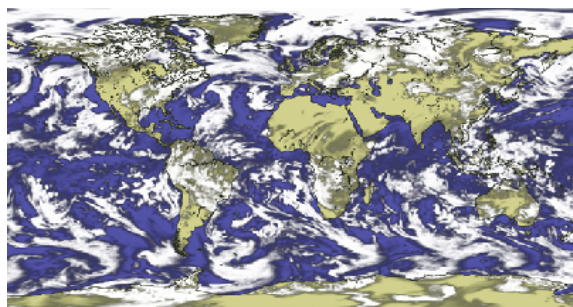
## Bereitstellung hochaufgelöster Wettervorhersagen für Geoportale

FRANK SCHWARZBACH, ANDREAS BUBLAK, THOMAS KLOSS

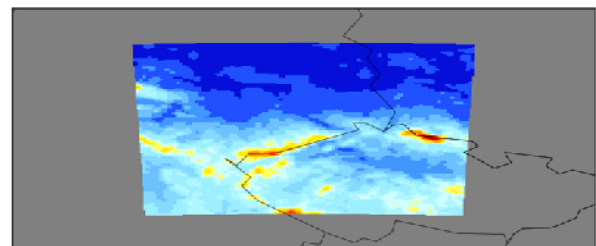
In dem Beitrag wird gezeigt, wie auf der Grundlage frei verfügbarer meteorologischer Rohdaten räumlich und zeitlich hochaufgelöste Vorhersagekarten für verschiedene Wetterphänomene bereitgestellt und in Geoportalen genutzt werden können. Der gesamte Workflow ist mit FOSS realisiert.

Die Motivation für die Auseinandersetzung mit dem Thema ist zunächst aus zwei Aspekten heraus entstanden: Zu einen bestand der Wunsch, auf möglichst „genaue“ Wettervorhersagen (kosten)frei zugreifen zu können. Zum anderen sollten diese Vorhersagen im Rahmen einer Geodateninfrastruktur interoperabel bereitgestellt werden und so auch in Geoportalen nutzbar sein. Vorab ist anzumerken, dass die Autoren keine Meteorologen, sondern bezüglich Wetterprognosen lediglich „interessierte Laien“ sind. Primär ging es darum zu zeigen, wie frei zugängliche Wetterdaten mit FOSS aufbereitet und somit für die interessierte Öffentlichkeit verfügbar gemacht werden können.

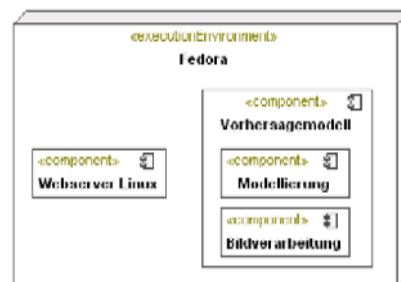
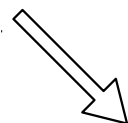
Die Datenrecherche und -analyse ergab zunächst fast erwartungsgemäß, dass Wetterdaten und -vorhersagen ein wichtiges Wirtschaftsgut sind und infolgedessen kaum geeignete Ausgangsdaten verfügbar sind, welche die oben beschriebenen Anforderungen an die Vorhersagequalität und die gewünschte freie Nutzung gleichermaßen erfüllen. Die Recherche führte jedoch zu einem alternativen Lösungsansatz: Die Berechnung eines „eigenen“ regionalen Wettermodells. Dies erfolgt durch die Implementierung einer Dienstkette, die vollständig auf FOSS basiert und nachfolgend stark vereinfacht dargestellt wird:



Globale Vorhersagedaten (NOAA)

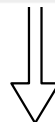


Oberflächendaten (USGS)



regionales Modell  
WRF-EMS  
(Quelle: NOAA/NWS  
Betrieb: HTW Dresden)

Betrieb: HTW Dresden



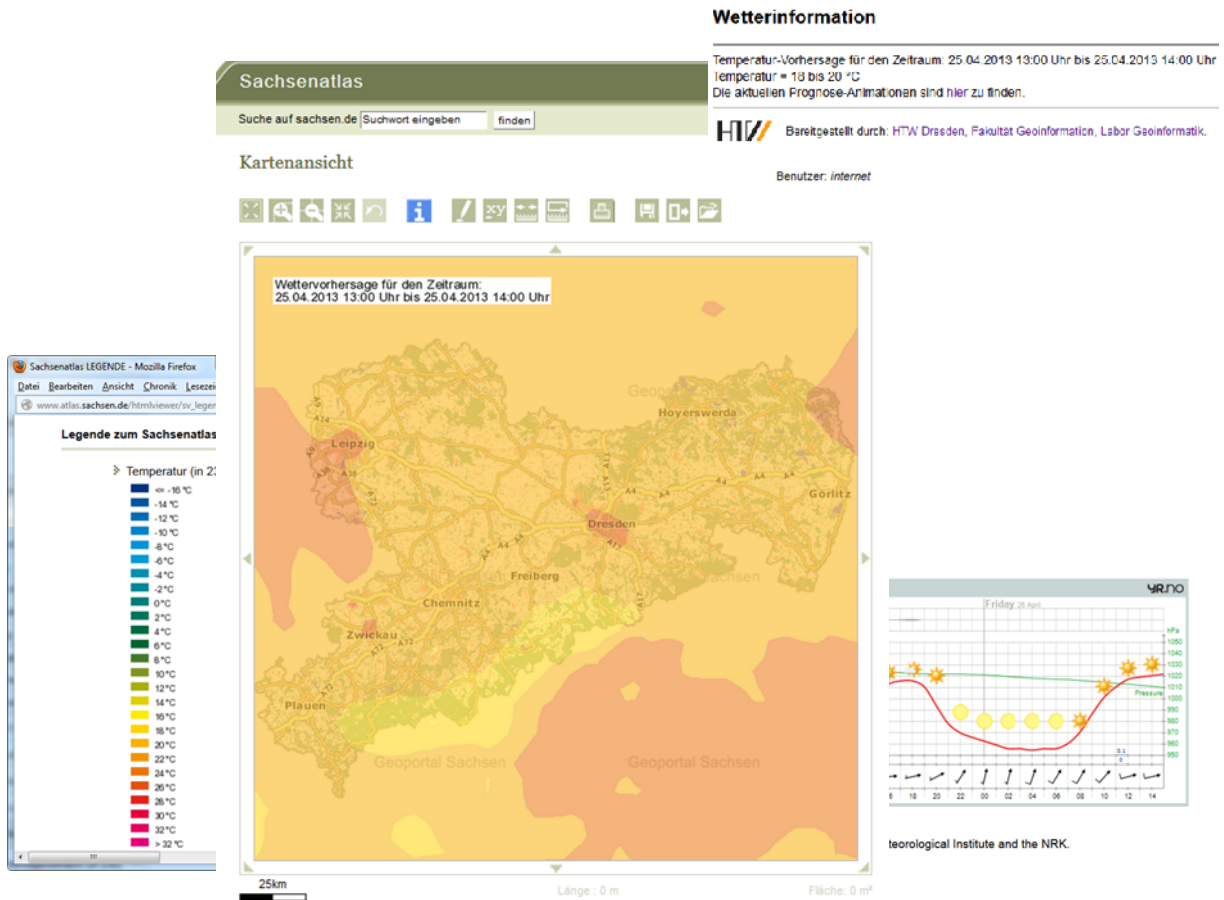
<http://geoinformatik.htw-dresden.de/cgi-bin/wetter?>

## Bereitstellung hochaufgelöster Wettervorhersagen für Geoportale

Die Grundlage des Workflows bilden zunächst globale Vorhersagedaten der US-amerikanischen National Oceanic and Atmospheric Administration (NOAA) mit einer Auflösung von 0,5 Grad. Die Berechnung des globalen Vorhersagemodells erfolgt auf der Grundlage weltweiter meteorologischer Messwerte und benötigt ca. 3-4 Stunden. Diese Daten werden anschließend auf den NOAA-Servern zum Download bereitgestellt und dienen, neben Oberflächendaten des U.S. Geological Survey, als Input für die Implementierung des WRF-EMS. Dabei wird das globale Modell für einen kleinräumigen Bereich verfeinert („Downscaling“). Die Berechnung der regionalen Vorhersagedaten beansprucht auf der aktuellen Plattform ca. 3 weitere Stunden. In einem abschließenden Verarbeitungsschritt werden die zunächst als Rasterdaten vorliegenden Ergebnisse transformiert, mit Legenden versehen und als WMS unter der oben angegebenen URL veröffentlicht. Zusätzlich zu den Karten werden die konkreten Vorhersagewerte für die jeweilige Bildposition über das GetFeature-Interface des WMS bereitgestellt. Aus den Vorhersagekarten wurden zudem zeitliche Animationen [1] abgeleitet. Außerdem werden frei verfügbare Zeitreihendaten (sog. Meteogramme) des norwegischen Wetterdienstes ([www.yr.no](http://www.yr.no)) angeboten. Animationen und Meteogramme sind ebenfalls über das GetFeature-Interface erreichbar.

Das abgedeckte Gebiet, der Vorhersagezeitraum und die räumliche und zeitliche Auflösung sind parametrierbar. Gegenwärtig werden für ein 3,5 \* 3,5 km - Raster, welches das Gebiet des Bundeslandes Sachsen abdeckt, Vorhersagekarten für einen Zeitraum von jeweils 48 Stunden (in Stundenintervallen) veröffentlicht. Berechnet werden Vorhersagen für Temperatur, Luftdruck, Niederschlag, Luftfeuchte, Bewölkung, Schneefall, Windrichtung und –stärke.

Der Dienst, dessen Metadaten im sächsischen GeoMIS ([www.geomis.sachsen.de](http://www.geomis.sachsen.de)) recherchierbar sind, kann mit beliebigen WMS Clients genutzt werden (hier [www.atlas.sachsen.de](http://www.atlas.sachsen.de)):



## **Bereitstellung hochaufgelöster Wettervorhersagen für Geoportale**

Performance, Kapazität und Verfügbarkeit des WMS werden durch ein externes Monitoring überwacht. Die Qualität der Vorhersage kann letztlich nur durch einen kontinuierlichen Vergleich zwischen den Prognosen und dem tatsächlichen Wetter beurteilt werden. Erste Ergebnisse lassen den Schluss zu, dass die Vorhersagen insgesamt brauchbar sind, es jedoch nennenswerte Unterschiede zwischen den einzelnen Wetterphänomenen gibt.

### **Dank**

Die Autoren bedanken sich bei Dr. Valeri Goldberg von der Professur für Meteorologie der TU Dresden für die fachliche Unterstützung.

Kontakt zum Autor:

Frank Schwarzbach  
Hochschule für Technik und Wirtschaft Dresden  
Fakultät Geoinformation  
Friedrich-List-Platz 1, D-01069 Dresden  
0049 351 4623134  
schwarzbach@htw-dresden.de

### Literatur

[1] *Büchner, Martin*: Dienstbasierte Bereitstellung animierter Darstellungen für die Wettervorhersage (Bachelorarbeit), Dresden, 2013 (unveröffentlicht).

## Cartaro – Geospatial CMS

### Ein Content Management System für räumliche Daten

ULI MÜLLER

Cartaro ist ein Content Management System, mit dem räumliche Daten nahtlos mit Texten und Bildern kombiniert, verwaltet und publiziert werden. Cartaro verbindet dabei die Stärken des CMS Drupal mit etablierten Open Source GIS-Projekten, allen voran GeoServer, GeoWebCache, PostGIS und OpenLayers. Mit Cartaro lassen sich somit über die mächtige Administrationsoberfläche von Drupal ohne Programmierung sowohl allgemeine Webseiten wie auch spezialisierte Geoportale und komplette Geodateninfrastrukturen entwickeln. Vor allem dank der Integration von GeoServer stehen alle im CMS verwalteten Inhalte als OGC-Webservices zur Verfügung – innerhalb und ausserhalb der Drupal-Webseite. In folgenden Bereichen zeigt sich die enge Verzahnung von räumlichen und nicht-räumlichen Inhalten des CMS besonders. Alle wichtigen Digitalisierungsfunktionen, wie sie aus einem Desktop-GIS bekannt sind, stehen in der Drupal-Oberfläche zur Verfügung. Die Persistierung räumlicher Daten beruht auf PostGIS und erlaubt somit umfangreiche räumliche Abfrage- und Analysemöglichkeiten direkt im CMS. Die ausgefeilten Benutzer- und Rollenkonzepte von Drupal lassen sich komplett auf Geodaten anwenden. Beginnend mit der Steuerung von Privilegien auf Datenebene bis zur Absicherung von Webdiensten erfolgen alle Einstellungen über die Drupal-Oberfläche. Eine wichtige Stärke von Cartaro liegt in der Erweiterbarkeit über eine Vielzahl an Drupal-Modulen. Zum Beispiel lassen sich mit Modulen für Redaktionssysteme komplexe Workflows bei der Erfassung und Freigabe räumlicher Daten abbilden. Dies ist für die interne Verwaltung von Geodaten ebenso interessant wie für Portale, die im Rahmen der Bürgerbeteiligung oder des Crowdsourcings Daten erheben. Die Präsentation stellt die Architektur von Cartaro, die Funktionalität und verschiedene Anwendungsfälle vor.

Links

<http://cartaro.org/>

## CouchWFS Light

MARKUS MAYR

### Grundlagen

Der in diesem Beitrag beschriebene Programmcode kann unter <http://www.github.com/scubbx/> heruntergeladen werden. Da der gesamte WFS Server ausschließlich aus einem einzigen Datenbankeintrag besteht, kann dieser auch direkt unter der Adresse <http://www.gisforge.com/files/couchwfs.js> geladen und in eine beliebige CouchDB Datenbank kopiert werden.

Um den WFS anzusprechen, muss eine der folgenden URLs verwendet werden:

```
http:// [...] /_design/wfs/_spatial/_list/wfspoint/point?  
http:// [...] /_design/wfs/_spatial/_list/wfsline/line?  
http:// [...] /_design/wfs/_spatial/_list/wfspolygon/polygon?
```

### CouchDB

CouchDB ist eine dokumentenbasierte Datenbank, welche Datensätze als JSON Entitäten in einem Sammel Speicher ablegt und mithilfe von map - reduce Abfragen zugänglich macht. So oder ähnlich beginnen die meisten Beschreibungen von CouchDB und schließen auch gleich mit diesem Thema ab. Unter Umständen findet noch die robuste Replikationsfähigkeit und Eignung für verteilte Datenbanken Erwähnung.

Tatsächlich besitzt eine CouchDB eine Reihe von weiteren Eigenschaften, welche nicht nur für die Web-Entwicklung interessant sind sondern auch ihrem anpassungsfähigen Charakter zugute kommen. Folgend sollen ein paar dieser für uns nützlichen Funktionalitäten beschrieben werden, aufgrund derer die Umsetzung eines CouchWFS überhaupt erst ermöglicht wird.

GeoCouch, entwickelt von [2], ist ein räumlicher Index für CouchDB. Dieser erlaubt es, map - Anfragen an mit einem räumlichen Index abgelegte Daten zu stellen. Diese Abfragen werden in sogenannten spatial views innerhalb eines für diese Zwecke speziell benannten design Dokuments abgespeichert und können bei Bedarf mittels GET request ausgeführt werden. Eine optionaler BoundingBox Parameter dient dabei als räumlicher Filter.

```
GET http:// ... /_DESIGN/designdok/_SPATIAL/view?BBOX=0,0,180,90
```

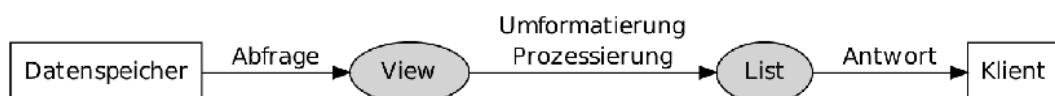


Abbildung 1: Aufschlüsselung des Abfrageprozesses unter Zuhilfenahme einer *list* Funktion

Das Ergebnis wird in Form eines JSON Objekts [1] an den Klienten geliefert. Innerhalb der CouchDB kann mit Hilfe einer list Funktion dieses zurückgegebene JSON Objekt zuvor verändert und ein beliebiges Ausgabeformat erzeugt werden, bevor es an den Klienten weitergereicht wird. In [2] wird so beispielsweise die Möglichkeit angeführt, direkt aus einer Anfrage heraus einen GeoRSS Feed zu generieren.

## CouchWFS Light

Eine list Funktion wird ebenfalls in einem design Dokument der Datenbank abgelegt und ist per GET Anfrage erreichbar. Zu beachten ist, dass man nun nicht nur die list Funktion zum Umwandeln der Abfrage, sondern im Anschluss daran auch die spatial view zur Selektion der Daten für die Abfrage angeben muss.

```
GET http:// ... /_DESIGN/designdok/_SPATIAL/_LIST/list/view?BBOX=0,0,180,90
```

Anzumerken ist noch, dass etwaige Parameter, welche an die GET Anfrage angehängt werden, innerhalb der vom Nutzer erstellten list Funktion abrufbar sind und in deren Programmcode verwendet werden können.

## Web Feature Server

WFS ist ein Standard, welcher ein Protokoll und Datenformat zum Austausch von geographischen Daten über HTTP definiert. Die Kommunikation zwischen Klient und Server erfolgt mittels POST oder GET, abhängig vom Implementierungslevel mit verschiedenen Typen von Anfragen (Operationen).

Auf Seite 3 in [3] werden drei verschiedene Implementationslevels definiert:

- **BasicWFS:** Es müssen ausschließlich die Operationen *GetCapabilities*, *DescribeFeatureType* und *GetFeature* implementiert werden (read-only).
- **XLink WFS:** Um XML-XLinks folgen zu können, ist zusätzlich zu den zuvor genannten Operationen noch *GetGmlObject* zu realisieren.
- **Transaction WFS:** Hier ist *Transaction* obligatorisch, zusätzlich zu den oben genannten Operationen. (read / write)

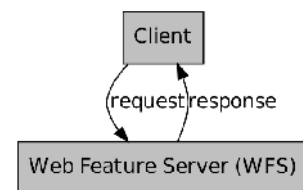


Abbildung 2: Kommunikation zwischen Server und Klient eines WFS Systems

Wie auf Seite 10 in [3] beschrieben, muss jeder WFS geographische Elemente zumindest als GML formulieren können.

## Implementierung

Der Server, so wie er in diesem Beitrag beschrieben wird, beschränkt sich auf einen Basic WFS. Folglich werden nur Leseoperationen unterstützt. Da mit einer CouchDB üblicherweise mittels eines REST Interfaces per GET Anfragen kommuniziert wird, findet bei einem CouchWFS ebenfalls nur GET Verwendung. Laut Seite 10 in [3] ist es möglich, entweder GET oder POST oder auch beides zu implementieren.

Auf Seite 19 sieht die WFS Spezifikation [3] bestimmte Eigenschaften und Fähigkeiten vor, deren Einhaltung für die exemplarische Implementierung des CouchWFS nicht zwingend notwendig war und daher darauf verzichtet wurde.

Mithilfe einer list Funktion der CouchDB ist es möglich, eine Datenbankabfrage strukturell in jedes beliebige Format umzuwandeln. Diese Funktion ist in JavaScript programmiert und hat Zugriff auf alle Parameter, die in der zugehörigen GET Anfrage mitgeliefert werden. Somit ist es innerhalb der Funktion möglich, zwischen den drei möglichen Anfragetypen eines WFS zu unterscheiden und die dafür spezifisch formatierten XML Dokumente zurückzuliefern.

Innerhalb der list Funktion sieht die Unterscheidung der einzelnen Anfragen beispielsweise folgendermaßen aus:

```
if (req.query.request == "GetCapabilities"){ ... };
```

## CouchWFS Light

Die für die Datenbankabfrage notwendige URL samt Parametern kann genau dem entsprechen, was für die Kommunikation mit einem WFS Server notwendig ist. Speziell der BBOX Parameter weist sowohl bei GeoCouch als auch bei einem WFS das gleiche Format auf und kann ohne weitere Manipulation verwendet werden.

Folglich kann eine bestimmte Anfrage an die CouchDB Datenbank als valide WFS Anfrage formatiert erfolgen. Nun liegt es an der list Funktion, für eine gültige Antwort zu sorgen.

### Organisation

Es sind mehrere CouchDB Operationen nacheinander erforderlich, um das gewünschte Ergebnis zu erzielen. Für die Vorselektion und Abfrage der Daten ist die view Funktion notwendig. Die Formatierung und weitere Verarbeitung übernimmt die list Funktion.

Die endgültige Ausgabe sollte aus Kompatibilitätsgründen nur einen FeatureTyp umfassen (Point, LineString oder Polygon). Die Auswahl des FeatureTyps erfolgt durch die verwendete view Funktion. Es wäre zwar möglich, den Filterprozess erst in der list Funktion durchzuführen, dies würde jedoch mehr Prozessierungsaufwand bedeuten.

Die Folge davon, dass pro FeatureTyp jeweils eine andere view Funktion verwendet wird, ist, dass für jeden FeatureTyp eine eigene URL existiert. Eine GetCapabilities - Anfrage wird in der Standardkonfiguration also immer nur einen Layer zur Verfügung stellen.

Wenn gewünscht, lassen sich innerhalb jeder list Funktion weitere Ausgabelayer definieren, welche die von der zugehörigen view Funktion erzeugten Daten gefiltert oder modifiziert ausgeben.

Alle von der view Funktion im value Feld zurückgegebenen Einträge (eine view liefert key-value Paare als Ergebnis) werden als Attribute betrachtet und von dieser CouchWFS Implementierung auch als solche ausgegeben. Somit liegt es an der view Funktion, vorab auszuwählen, welche Werte als Eigenschaften ausgegeben werden sollen.

Das key Feld sollte als GeoJSON formatiert sein.

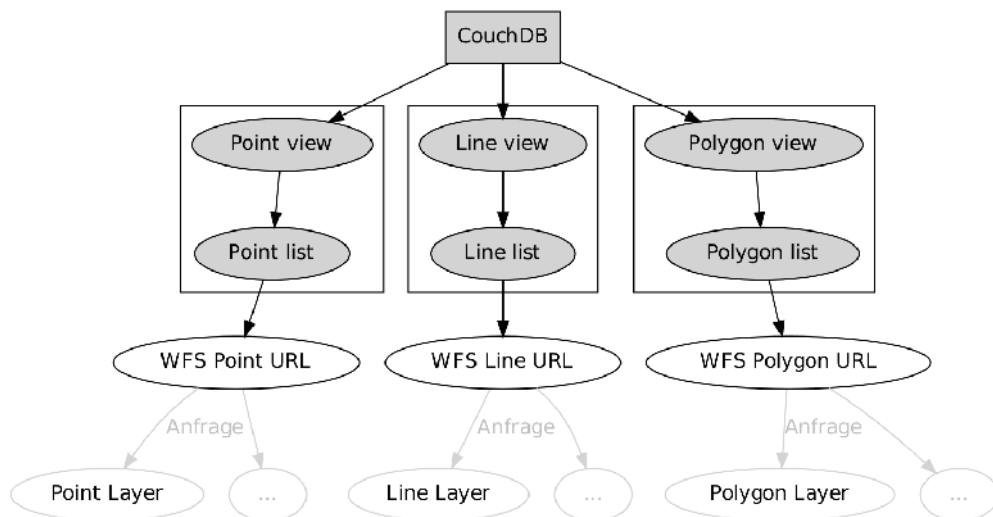


Abbildung 3: Die drei unterschiedlichen WFS Adressen für die einzelnen Feature Typen



## CouchWFS Light

### Requests

Bei einem WFS wird der Anfragetyp mit dem Parameter "request=" beschrieben.

CouchWFS implementiert den 'BasicWFS' Standard und hat daher mindestens drei verschiedene Typen von Requests zu verstehen und entsprechend zu beantworten.

Die Antworten auf die Anfragen DescribeFeatureType und GetFeature haben folgenden MIME Header Typ aufzuweisen:

```
text/xml;subtype=gml/3.1.1
```

Es ist wichtig zu erwähnen, dass für egal welche der drei hier beschriebenen Anfragen, eine unvermeidbare Abfrage auf die Datenbank ausgeführt wird. Dies liegt daran, dass mit einer *list* Funktion immer auch die damit assoziierte *view* ausgeführt wird, egal ob deren Daten später benötigt werden oder nicht.

### GetCapabilities

Vereinfacht gesagt wird mit dieser Anfrage eine Liste aller am Server verfügbaren Layer und unterstützten Koordinatensystemen übermittelt. Dies ist in der *list* Funktion einfach zu implementieren. Um ein Durchlaufen aller von der *view* Funktion übergebenen Elemente zwecks Berechnung der korrekten Bounding-Box zu vermeiden, wird die Welt als Ausdehnung angegeben ( [-180,-90,180,90] ).

Die Generierung der Antwort zu dieser Art Anfrage erfolgt ausschließlich statisch.

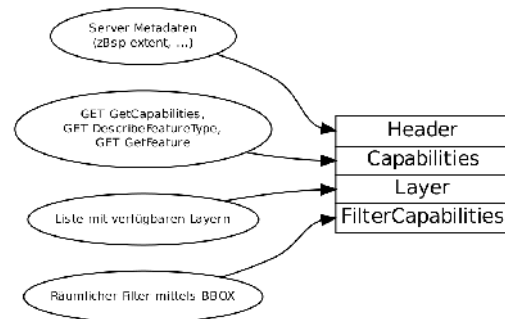


Abbildung 4: Generierung der Antwort zur "GetCapabilities" Anfrage (weiß: hardcoded)

### DescribeFeatureType

Dieser Request beschreibt mithilfe eines GML Schemas [4], mit welcher GML Struktur die Daten des selektierten Layers übermittelt werden und welche Attribute zu erwarten sind. Es ist laut Spezifikation zwar möglich, beliebige Ausgabeformate anzugeben, der WFS Standard sieht aber zumindest GML als Mindeststandard vor.

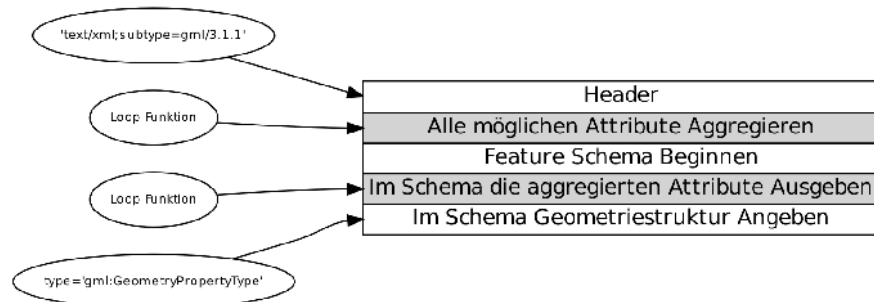


Abbildung 5: Generierung der Antwort zur "DescribeFeatureType" Anfrage (weiß: hardcoded, grau: prozedural)

In der *list* Funktion wird zunächst einmal das statische Grundgerüst der Antwort erzeugt. Um nun alle möglichen Attribute der abgefragten Daten bereitstellen zu können, müssen diese bereits in diesem Schema beschrieben werden. Zu diesem Zweck werden alle Attribute der im GeoJSON vorliegenden Datenbankabfrage gesammelt und die auszugebende Schemabeschreibung entsprechend ergänzt.

## GetFeature

Mit dieser Anfrage werden die tatsächlichen Geodaten angefordert. Laut WFS Standard sind hier eine Vielzahl an Parametern möglich, welche in dieser exemplarischen Implementation jedoch nicht umgesetzt wurden. Bei installierter GeoCouch Erweiterung wird der *bbox* Parameter ausgewertet. Ansonsten liefert der Aufruf dieser Anfrage ungeachtet etwaiger weiterer Parameter ohne Erweiterung des aktuellen Programmcodes immer das gleiche Resultat.

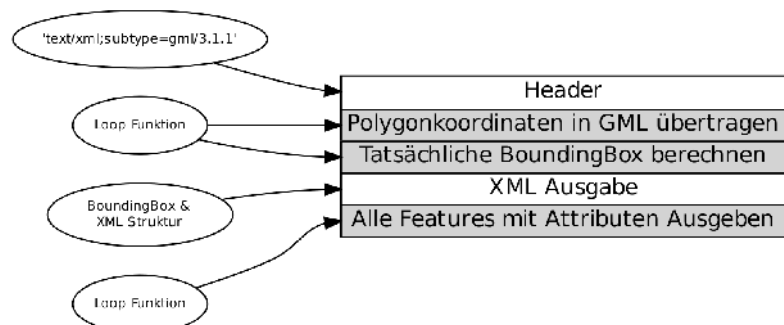


Abbildung 6: Generierung der Antwort zur "GetFeature" Anfrage, (weiß: hardcoded, grau: prozedural)

Es wird nach der Ermittlung der tatsächlichen Ausdehnung aller Elemente durch eine Iteration über diese für jedes Feature das entsprechende XML für die Ausgabe erzeugt. Dabei werden neben den Koordinaten auch alle Attribute ausgegeben.

## Nebeneffekte

Die speziellen Eigenschaften einer CouchDB beeinflussen die CouchWFS Implementierung in negativem als auch positivem Sinn.

### Positive Effekte

Folgende Punkte sind implizite Eigenschaften einer CouchDB und wirken sich automatisch positiv auf den Betrieb des CouchWFS aus:

- **ETags:** Ein ETag ist ein ID Wert, welcher, wenn vorhanden, bei einem Request nach einer bestimmten Ressource an den Server übermittelt wird. Falls der Server anhand dieses Wertes erkennt, dass sich die angeforderten Daten seit dem letzten Aufruf nicht verändert haben, schickt er nicht abermals die gleichen Daten an den Klienten, sondern weist darauf hin, dass dieser auf seinen Cache zurückgreifen kann, um den angeforderten Response zu erhalten. Im Falle der CouchWFS Implementierung bedeutet dies, dass nicht bei jedem Request zwangsweise die gesamte angeforderte Datenmenge übertragen werden muss, sondern automatisch ein caching - Mechanismus in Kraft tritt, welcher sich empirisch als sehr effektiv herausgestellt hat.
- **BoundingBox:** Da der Parameter für eine BoundingBox bei GeoCouch und dem WFS Standard identisch ist und in der URL angegebene Parameter einfach an die CouchDB weitergeleitet werden, kann ohne weiteren Aufwand räumlich gefiltert werden. Einzig, wenn die GeoCouch Erweiterung nicht installiert ist, muss auf diese Funktion verzichtet werden.

### Negative Effekte

Genau so wie spezielle Eigenschaften einer CouchDB positiven Einfluss auf den CouchWFS haben, gibt es auch negative Seiten:

## CouchWFS Light

- **Redundanz der Abfrage:** Für jede Anfrage an den Server wird auch die *view* Funktion ausgeführt und deren gesamten Ergebnisse bereitgestellt, egal ob dies für die Antwort notwendig ist oder nicht. Dies hat eine Verlangsamung des gesamten Abfrageprozesses zur Folge.
- **Geschwindigkeit:** Der CouchWFS Server ist aus einer Kombination von Erlang (CouchDB) und JavaScript (*view* und *list* Funktion) umgesetzt. Die hier vorgestellte Implementierung ist nicht sehr effizient und weist Potential zur Verbesserung auf. Doch selbst in einer optimierten Form sind die Möglichkeiten stark eingeschränkt und es wird nie zu niedrigen Antwortgeschwindigkeiten kommen können. Durch Anpassen der *view* und *list* Funktionen an spezielle Anforderungen (zBsp. Filtern von nur gewünschten Features durch eine spezielle *view*) kann die Zugriffszeit noch am ehesten optimiert werden.

## Anwendungsfälle

Um die tatsächliche Funktionalität des CouchWFS zu testen, wurde mit unterschiedlichen GIS darauf zugegriffen. Zwei dieser Systeme, mit denen der Zugriff funktioniert, werden hier kurz beschrieben. Ein Test mit gvSIG [5] war nicht erfolgreich, das Programm erkennt zwar die angebotenen Layer, lässt sie allerdings nicht dem Projekt hinzufügen.

### QuantumGIS

Der Zugriff auf den CouchWFS per QuantumGIS in der Version 1.8 [6] erfolgt problemlos. Es werden alle drei URLs der FeatureTypen geladen und angezeigt.

Das Hinzufügen des ausgewählten Layers kann bei einer großen Datenmenge mitunter einige Zeit in Anspruch nehmen, da bei jeder Abfrage die *view* Funktion zur Gänze und die *list* Funktion teilweise abgearbeitet werden muss.



Abbildung 7: Ergebnis der "GetCapabilities" Anfrage von QuantumGIS

Sind die Layer einmal geladen, können sie mit allen von QuantumGIS für WFS Layer bereitgestellten Funktionen bearbeitet werden. Am interessantesten dürfte die Möglichkeit sein, den Layer mitsamt seinen Attributen in beliebige Formate (unter anderem Shapefiles) zu exportieren.

## CouchWFS Light

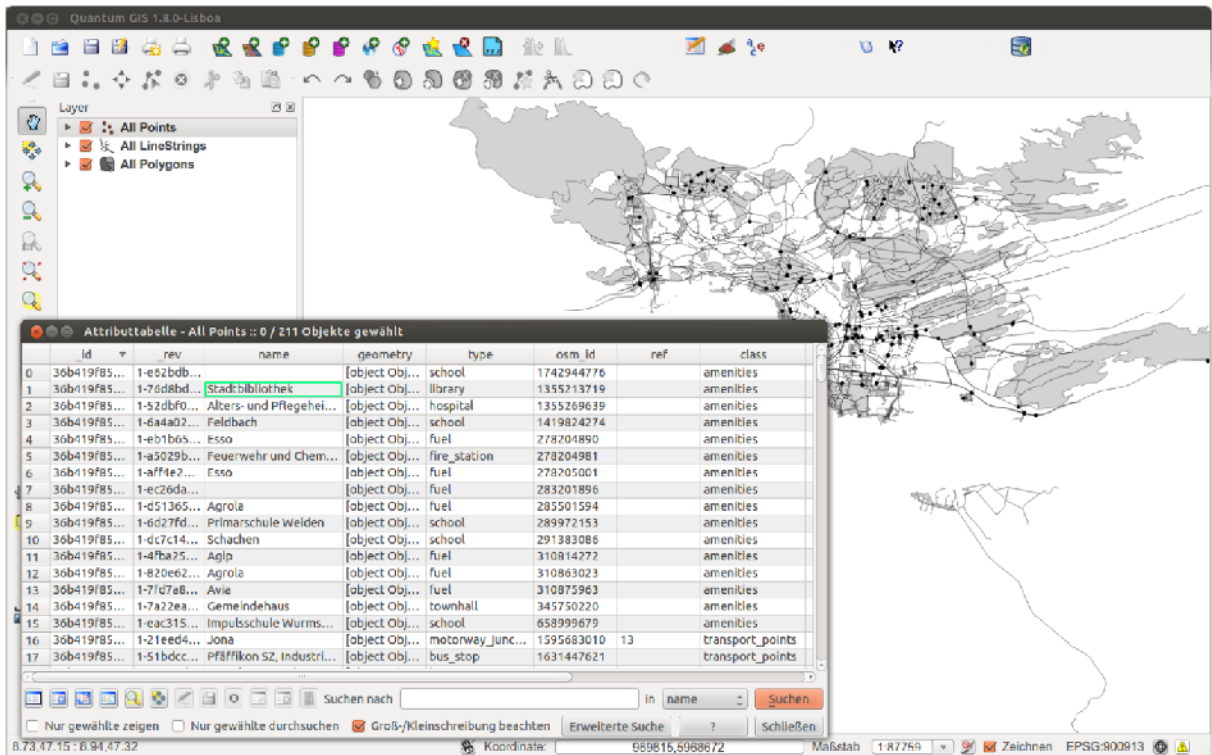


Abbildung 8: Alle verfügbaren Layer aus dem CouchWFS mitsamt Attributen in QuantumGIS

## OpenLayers

Speziell um die Funktionalität des BoundingBox Parameters zu überprüfen, wurde der CouchWFS mittels OpenLayers in der Version 2.12 [7] auf einer Website eingebunden. Dies erfolgte mit einem positiven Resultat.

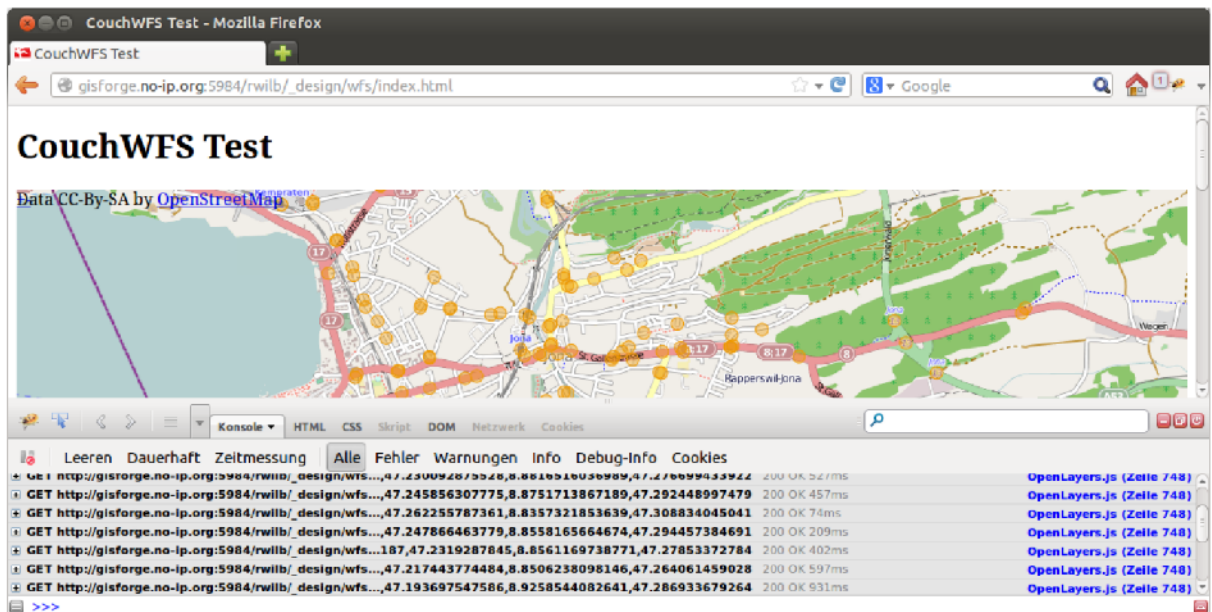


Abbildung 9: Der Point-Layer in OpenLayers geladen und mit BoundingBox Parameter abgefragt

## CouchWFS Light

Eine Schwierigkeit, die sich herausgestellt hat, ist, dass der WFS Provider von OpenLayers standardmäßige nur POST Anfragen an den CouchWFS stellt, welche dieser wiederum nicht korrekt verarbeitet. Unter Verwendung des HTTP Providers werden jedoch GET Anfragen ausgeführt, mit welchen sich OpenLayers mit dem CouchWFS verwenden lässt.

## Weitere Möglichkeiten

Durch Anpassen der *view* oder *list* Funktionen können auf spezielle Bedürfnisse zugeschnittene Ausgaben erzeugt werden. Dabei gilt es zu überlegen, ob diese Anpassungen bereits in der Vorauswahl datenbankseitig erfolgen sollen (*view*), oder ob ein serverseitiges nachträgliches Bearbeiten (*list*), welches die Datenmenge erst kurz vor der Ausgabe reduziert, mehr Sinn macht.

Der hier beschriebene WFS ist in seiner gegenwärtigen Form nicht kompatibel mit allen getesteten GIS Programmen. Dies dürfte vor allem an der noch nicht genau nach WFS Spezifikationen generierten XML Ausgabe liegen. Hier besteht Potential zur Verbesserung.

## Conclusio

Mit der hier vorgestellten Methode lässt sich unkompliziert ein "graphisches Fenster" auf eine CouchDB öffnen. Diese muss nur dann zwangsweise mit einer GeoCouch Erweiterung ausgestattet sein, wenn man mittels BoundingBox filtern will. Unter Anbindung weiterer GIS Software können so Daten aus der Datenbank in beliebige Formate exportiert oder dargestellt werden.

Da der WFS Standard nicht zu 100 Prozent eingehalten werden kann ist ein universeller Einsatz dieser Methode nur bedingt zu empfehlen, kann im konkreten Fall allerdings eine willkommene Erleichterung sein, da sie keinen nennenswerten Overhead für ihre Implementierung aufweist und auch wieder rückstandsfrei zu entfernen ist.

## CouchWFS Light

Kontakt zum Autor:

(DI) Markus Mayr  
TU Wien  
Department für Geodäsie und Geoinformation  
Gusshausstraße 27-29, 1040 Wien  
markusmayr@gmx.net

## Literatur

- [1] Douglas Crockford. The application/json media type for javascript object notation (json), 2006. UR: <http://tools.ietf.org/html/rfc4627>.
- [2] Volker Mische, Pierre Giraud, and Filipe Manana. Geocouch - readme, 2012. URL : <https://github.com/couchbase/geocouch/blob/master/README.md>
- [3] Panagiotis A. Vretanos. Web feature service implementation specification 1.1.0. Technical report, Open Geospatial Consortium Inc., 2005.
- [4] Panagiotis A. Vretanos. Geography markup language (gml) simple features profile 3.1.1. Technical report, Open Geospatial Consortium Inc., 2006.
- [5] gvSig , 2013. URL: <http://www.gvsig.org/web/>
- [6] QuantumGIS, QuantumGIS Development Team, 2013. GNU General Public License. URL: <http://www.qgis.org/>
- [7] OpenLayers Contributors, OpenLayers, 2012. URL: <http://openlayers.org/>

## Das OpenStreetMap-Datenmodell

JOCHEN TOPF

Auf den ersten Blick scheint OpenStreetMap ein sehr einfaches Datenmodell zu benutzen. Es gibt Nodes, Ways und Relations mit freien Tags. Das war es schon. Aber wenn man etwas genauer schaut, dann findet man hinter diesem einfachen Grundmodell ein komplexes Gebilde von Konventionen, die manchmal dokumentiert sind, vielfach aber nur in den Köpfen der Mapper existieren. Daneben führt die Interpretation der Daten durch vorhandene Software vielfach zu weiteren Regeln, die man eigentlich dem Datenmodell zuordnen muss. All dies führt zu einem ziemlich komplexen Datenmodell, das jeder verstanden haben muss, der intensiver mit OSM-Daten arbeitet. Die simple Welt der OSM-Grundobjekte mit ein paar Tags ist eine Illusion.

Der Vortrag zeigt anhand der existierenden OSM-Daten, welche Einflüsse unser Datenmodell geprägt haben, was damit ausgedrückt werden kann und wo seine Beschränkungen liegen. Er versucht aus den einzelnen Aspekten von Tagging-Konventionen, über komplexe Relations bis zu implizierten, aber nicht wirklich modellierten Zusammenhängen einen Eindruck davon zu geben, wie unser Datenmodell eigentlich aussieht und wie es sich weiterentwickeln kann. Der Vortrag richtet sich an alle "Mapper" und alle, die OSM-Daten nutzen oder nutzen wollen und die sich schon mal gefragt haben, warum das eigentlich oft so schwierig ist.

## Das QGEP Abwasser Projekt

ANDREAS NEUMANN, MATTHIAS KUHN

Das QGIS-Projekt erfreut sich zunehmender Beliebtheit bei Behörden und Firmen. Allerdings gibt es noch kaum grössere Fachschalen (z.B. für die Leitungsdokumentation, Vermessung, oder Raumplanung). Gegen Ende des Jahres 2011 formierte sich eine Gruppe von QGIS Anwendern und Abwasserfachpersonen (Fischer Ingenieure, Arbon; Holinger, Thun; SjiB Consulting, Zürich) um gemeinsam mit der Firma Sourcepole und einigen Gemeinden eine Fachschale auf QGIS-Basis für die Bereiche Abwasserwerkplan-Dokumentation und GEP (Genereller Entwässerungsplan) zu entwickeln. Ziel der Gruppe ist es eine bestehende und gut funktionierende Fachapplikation auf der technologischen Basis von Arcview 3.x abzulösen. Ein weiteres Ziel ist es die Vorgaben des VSA (Verband Schweizer Abwasserfachleute) möglichst getreu und vollständig abzubilden. Einige GIS-Hersteller, die auch international tätig sind, setzen diese nationalen Richtlinien nur mangelhaft um.

Das QGEP Projekt, der Name leitet sich von QGIS und GEP ab, nimmt insofern auch eine Pionierrolle ein, da wie eingangs erwähnt noch wenige komplexe Fachschalen auf Basis von QGIS entwickelt wurden. So müssen im Zuge der QGEP-Entwicklung einige technologische Probleme gelöst werden deren Behebung später auch anderen Fachschalen zugute kommen (siehe Abschnitt „Grundlagenarbeiten im QGIS Kern“).

Das Ziel von QGEP ist es zuerst produktiv den Werkplan Abwasser editieren und nachführen zu können. Danach soll die Applikation laufend ergänzt werden durch Analysen und Auswertungen aus dem GEP-Bereich. Die Präsentation zeigt den aktuellen Stand, sowie die Baustellen und Probleme bei der Umsetzung der Fachschale.

### Datenmodell

Das Datenmodell wurde vom VSA-DSS Modell [1] übernommen und in die englische Sprache übersetzt. Sämtliche Tabellen- und Spaltennamen werden in englisch geführt, es gibt Lokalisierungen der Tabellennamen und Attributwerte für deutsch und französisch. Weitere Sprachen könnten hinzugefügt werden. Technologisch gesehen wird auf eine Postgis-Datenbank gesetzt. Mit einem gewissen Zusatzaufwand sollten später auch Portierungen auf andere Datenbanken möglich sein. Das Datenmodell ist bereits fertig umgesetzt – im Zuge der Entwicklung der Editierfunktionen werden später jedoch noch weitere Trigger und Regeln eingeführt werden.

### Grundlagenarbeiten im QGIS Kern

Unabhängig von der Abwasserfachschale fehl(t)en in QGIS einige wichtige Funktionalitäten die zur Implementierung von komplexeren Fachschalen notwendig sind. Diese Arbeiten wurden und werden von Schweizer QGIS Anwendern teilweise unabhängig vom QGEP Projekt finanziert und kommen allen zukünftigen QGIS Fachschalen zugute.

### Komplexe datendefinierte Symbolisierung

QGIS führte vor einigen Monaten eine neue Generation für die Symbolisierung ein: Symbology 2 mit der Möglichkeit für die Repräsentation von Geometrien mehrere Symbolebenen (sogenannte Mehrfachsymbole) zu verwenden. Ebenso wurde die Möglichkeit von Symbolregeln (rule-based renderer) eingeführt. Damit kann nicht nur ein Attribut, sondern eine beliebige Kombination von Attributen und Regeln für die Klassenbildung verwendet werden. Was aber noch fehlte war die „datendefinierte“ Symbologie mittels derer grafische Eigenschaften (wie z.B. Strichstärke, Strichfarbe, Linienversatz, etc.) direkt von Datenbankfeldern oder von auf Datenbankfeldern beruhenden Berechnungsausdrücke (Expressions) abgeleitet werden können. Damit kann z.B. ein Schacht mit der tatsächlichen Dimension und Rotation dargestellt werden, oder eine Haltung mit der tatsächlichen Breite. Diese Werte sind in



## Das QGEP Abwasser Projekt

Datenbankfeldern abgeleitet. Es können auch verschiedene Einheiten gemischt werden: z.B. Strichstärke der Umrandung in mm und Breite/Höhe in m. Für Berechnung kann der QGIS Expression-Editor verwendet werden. Diese Verbesserungen sind bereits umgesetzt und werden Teil von QGIS 2.0 sein.

Eigenschaft	Feld	Ausdruck
<input checked="" type="checkbox"/> Füllfarbe	hex_color	...
<input checked="" type="checkbox"/> Symbolhöhe	dimension_1	...
<input type="checkbox"/> Versatz		...
<input type="checkbox"/> Randfarbe		...
<input type="checkbox"/> Umrandungs...		...
<input checked="" type="checkbox"/> Drehung	orientierung	...
<input type="checkbox"/> Symbolname		...
<input checked="" type="checkbox"/> Symbolbreite	dimension_2	...

Abbildung 1: Einstellungsdialog für datendefinierte Symbolisierung (Punktsymbole)

### Verschmelzung von Formular und Tabelle („Dual View“)

Bisher konnten die Objektattribute entweder in Tabellen oder in separaten Formularen betrachtet und editiert werden. Beim Formular konnte nur ein Objekt betrachtet und editiert werden. Neu wurden die Tabelle und die Formularansicht in ein gemeinsames „Dual View“ Fenster zusammengeführt. Mit den Icons rechts unten im Fenster kann zwischen der Tabellen- und der Formularansicht hin- und hergeschaltet werden. Filter und Selektionen greifen simultan in der Tabellen- und Formularansicht. Neu werden Filter von Selektionen unterschieden. Der Filter- und Selektionsstatus wird in der Kopfzeile des Fensters eingeblendet. Grundsätzlich neu ist in der Formularansicht, dass von Objekt zu Objekt geblättert werden kann ohne dass das Formular zuerst von der Kartenansicht oder der Tabelle aufgerufen werden muss. Eine Spalte links der Formularansicht wird für die Identifikation der einzelnen Records und zur Navigation verwendet. In ihr werden eindeutige Schlüssel oder Ausdrücke (Expressions) aufgelistet. Zusammen mit den funktionalen Verbesserungen wurden auch Performance-Verbesserungen umgesetzt. Die „Dual View“-Ansicht ist bereits umgesetzt und wird in QGIS 2.0 verfügbar sein.

## Das QGEP Abwasser Projekt

The screenshot shows a software interface with a 'Dual View' layout. On the left, a list of objects is displayed, with 'GID 291016 ( 615.2.102)' selected. On the right, a form displays the details for the selected object. The form includes the following fields and values:

Schachtnummer	615.2.102	Baujahr	
Schachttyp	Kontrolle, Standardschacht	Status	In Betrieb
Eigentümer	Private Erschliessung	Betreiber	Private Erschliessung
Nutzungsart	Regenabwasser	Zugänglichkeit	NULL
Durchmesser 1	900	Durchmesser	900
Schachttiefe		Sohlenkote	
Rückstaukote			
Einstiegshilfe	Leiter	Bankettart	NULL
Bemerkung	NULL		
Deckel			
Einläufe		Ausläufe	
ID Schacht	22857		
letzte Änderung	2012-02-06		

At the bottom right of the form area, there is a button labeled 'Dataver starten'. At the bottom left of the window, there is a dropdown menu labeled 'Alle Objekte anzeigen'.

Abbildung 2: "Dual View" Ansicht - Formularteil - mit Icon rechts unten kann auf Tabelle umgeschaltet werden.

### Datenbankrelationen

Bisher konnte QGIS nur mit sehr einfachen 1:1 Verknüpfungen umgehen. Fast alle Fachschalen benötigen allerdings auch 1:n und n:m Verknüpfungen. Obwohl diese Relationen über Fremdschlüssel bereits in Postgis definiert wurden haben wir uns dazu entschieden diese Relationen in QGIS noch einmal zu definieren. QGIS Fachschalen sollen nicht nur mit Postgis-Datenquellen umgehen können sondern auch Relationen über Datenbanken- und Datenformate hinweg ermöglichen. Dazu ist es unumgänglich die Relationen mit Hilfe eines „Relations Manager“ innerhalb von QGIS zu definieren. Zu einem späteren Zeitpunkt könnte allenfalls noch ein Werkzeug entwickelt werden welches bereits bestehende Relationen aus Datenbanken übernehmen kann. Dieser Relations Manager befindet sich derzeit in Entwicklung und wird mit QGIS 2.1 veröffentlicht werden.

### Verschachtelte Formulare

Die mit dem Relations Manager erzeugten Verknüpfungen sollen später dazu genutzt werden um verschachtelte Formulare anzeigen und bearbeiten zu können. Dadurch sollen 1:1, 1:n und n:m Beziehungen durch verschachtelte „Master/Detailformulare“ abbildbar sein. Beispielweise können damit zu einem Schacht, das Bauwerk, sämtliche Bauwerksteile, das Abwassernetzelement und die zugehörigen Unterhaltseignisse (Zustandsaufnahmen, Sanierungen, Planungen, etc.) zusammen in einem Formular dargestellt und editiert werden, obwohl diese aus verschiedenen Tabellen stammen. Diese Entwicklung ist noch nicht im Gange (Finanzierung ist gesichert) und die Ergebnisse sollen in QGIS 2.1 verfügbar sein.

## Das QGEP Abwasser Projekt

### DXF Export und Import

Das DXF Format ist in QGIS erst rudimentär unterstützt (ohne Symbolisierung und nur ein Geometrietyp pro Layer). Dieses Format ist populär für die Abgabe an Architekten und Planer. Erste Arbeiten für einen verbesserten DXF Export welcher möglichst viel der Symbologie vom QGIS Projekt übernimmt wurden bereits umgesetzt. So können Linienfarben, Strichstärken und Strichlierungen bereits korrekt in das DXF exportiert werden (verfügbar in QGIS 2.0). Anderes muss noch entwickelt werden: Übernahme Beschriftung, Übernahme Punktsymbole und Konvertierung in DXF-Blöcke, Einflussnahme auf die Layersteuerung.

Beim Import wurden noch keine Verbesserungen vorgenommen. Import und Export soll in QGIS 2.1 wesentlich verbessert sein. Bis dahin können die QGEP-Daten auch über OGR oder FME exportiert werden.

### Interlis Schnittstelle

Die Schweizer Interlis-Schnittstelle dient zum verlustfreien Austausch von Modellen und Daten zwischen verschiedenen GIS-Systemen. Das Austauschformat ist entweder text-basiert (Interlis 1) oder XML basiert (Interlis 2). QGIS kann Interlis 1 Dateien bereits recht gut lesen. Schlechter sieht es beim Lesen von Interlis 2-Daten und beim Export (insbesondere Mapping zwischen verschiedenen Datenstrukturen) aus. Es wird Anstrengungen geben die Interlis-Fähigkeiten von OGR zu verbessern und ein Mapping zwischen Datenstrukturen mit Hilfe des SEXTANTE Frameworks zu ermöglichen. Diese Arbeiten wurden noch nicht gestartet, daher gibt es auch noch keine Termine für die Fertigstellung. Bis dahin muss über FME oder Infogrips-Werkzeuge importiert und exportiert werden.

### Seriendruck und Reporting

In QGIS 2.0 wird erstmals das Atlas-Serienwerkzeug verfügbar sein welches teilweise auch im Rahmen des QGEP Projektes finanziert wurde. Es erlaubt den Seriendruck über Kartenrahmen oder über alle oder eine gefilterte Zahl von Objekten. Texte können aus Attributwerten oder Ausdrücke (Expressions) automatisch befüllt werden. Daneben können Fotos und HTML-Blöcke in die Kartenlayouts oder Reports eingefügt werden. Im Laufe des weiteren QGEP Projektes werden sicherlich noch einige weitere Verbesserungen im Atlas-Seriendruck durchgeführt werden, beispielsweise für die Integration von verknüpften aggregierten Objekten.

### QGEP-spezifische Editierfunktionen

Neben den bereits in QGIS vorhandenen Editierfunktionen müssen die QGEP-Erweiterungen vor allem die Konsistenz des Datenmodells sicherstellen. Beim Erstellen, Löschen oder Ändern eines Elements müssen über Trigger und Regeln sämtliche damit verbundene Tabellen nachgeführt werden. Daneben müssen Konstruktionswerkzeuge aus den CAD-Tools angepasst oder ergänzt werden. Eine detaillierte Liste der Editierfunktionen würde den Rahmen dieser Publikation sprengen. Die QGEP spezifischen Editierfunktionen sind noch nicht entwickelt. Deren Entwicklung soll begonnen werden wenn die Datenbankrelationen und verschachtelten Formulare in QGIS Master zur Verfügung stehen. Bei den Editierfunktionen sollen die Standardabläufe möglichst effizient abgedeckt werden: z.B. Erstellung, Verändern und Löschen von Bauwerken; Auftrennen und Zusammenführen von Haltungen; Massenänderung von Attributdaten auf selektierten Objekten.

### Netzverfolgung und Interaktives Längenprofil

Eine Netzverfolgung wurde bereits auf Basis von NetworkX [3] (Python) umgesetzt. Diese erfolgt nicht auf den Geometrien selbst sondern über die Verknüpfungen im Datenmodell (oberes und unteres Bauwerk). Es kann der kürzeste Weg zwischen 2 Punkten gefunden werden oder von einem Punkt aus eine Verfolgung „upstream“ oder „downstream“ (zur Kläranlage). Ausserdem wurde eine Profilgenerierung zur Darstellung des Gefälles und der Dimensionen der Abwasserbauwerke im Kontext der Geländeoberfläche umgesetzt (siehe QGEP Plugin unter [2]). Für die Darstellung des interaktiven Kanalpro-

## Das QGEP Abwasser Projekt

files kommt d3.js [4] innerhalb eines Webkitbrowser-Bereichs zum Einsatz. Die Profilsicht ist mit der Kartenansicht verknüpft. Die aktuell ausgewählte Haltung im Profil wird in der Kartenansicht hellgrün hervorgehoben. Zudem werden die wichtigsten Haltungsattribute in einem Tooltip zusammengefasst.

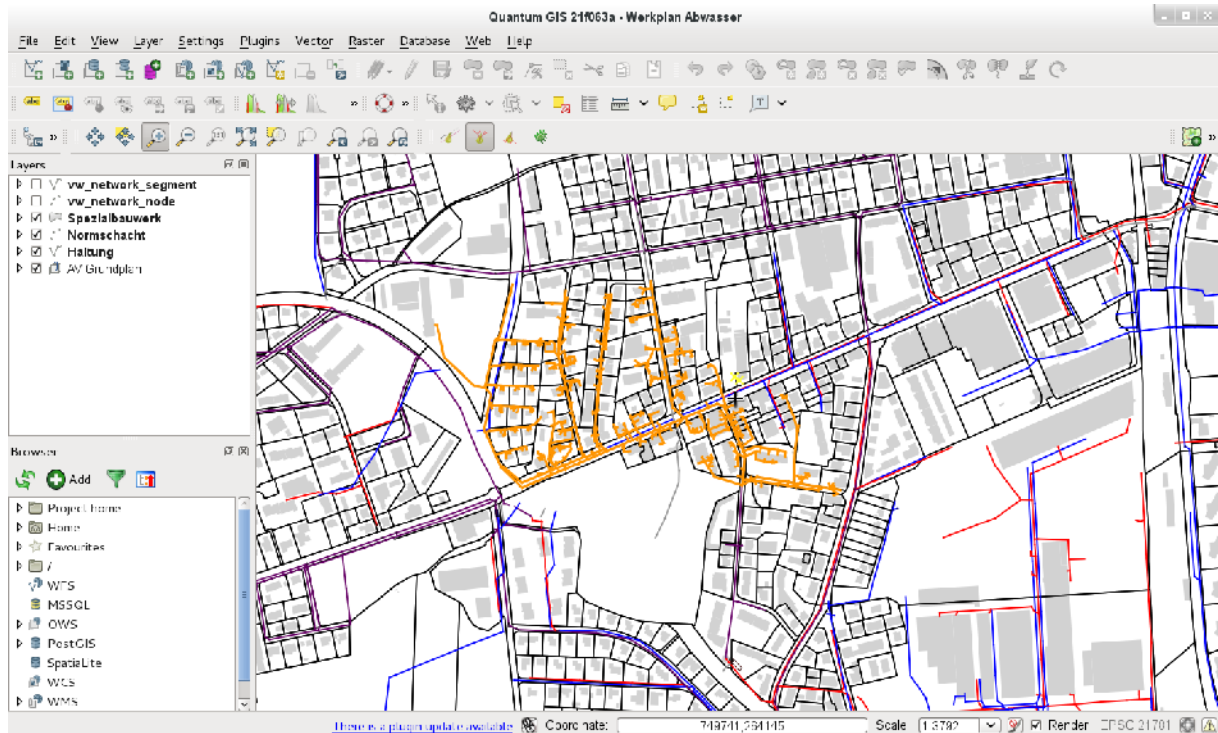


Abbildung 3: Netzverfolgung - Resultat von Upstream-Suche (orange hervorgehobene Haltungen)

### Ausblick

Bis zu einer ersten produktiven Version sind noch einige Arbeiten zu erledigen – insbesondere bei den Datenbankrelationen, bei den verschachtelten Formularen und bei den Editierfunktionen. Nach der ersten stabilen Release sollen vor allem die Exporte und Importe (Hydrauliksoftware (Mouse, SWIMM), VSA-KEK für Erhaltungsereignisse) vervollständigt werden. Zudem sollen weitere Analysefunktionen und Reportingfunktionen folgen. Auf selektierten Bauwerken (z.B. aus der Netzverfolgung) sollen Aggregationsfunktionen und Auswertungen möglich sein.

Die erste stabile Version ist im Laufe des Jahres 2014 zu erwarten. Ein genaues Releasedatum gibt es noch nicht. Falls Sie Interesse am Projekt haben, sich finanziell beteiligen oder fachlich mithelfen können würden wir uns über Ihre Kontaktaufnahme freuen. Als Open Source Projekt ist QGEP offen für die Mitbeteiligung von weiteren Firmen und Behörden.

## Das QGEP Abwasser Projekt

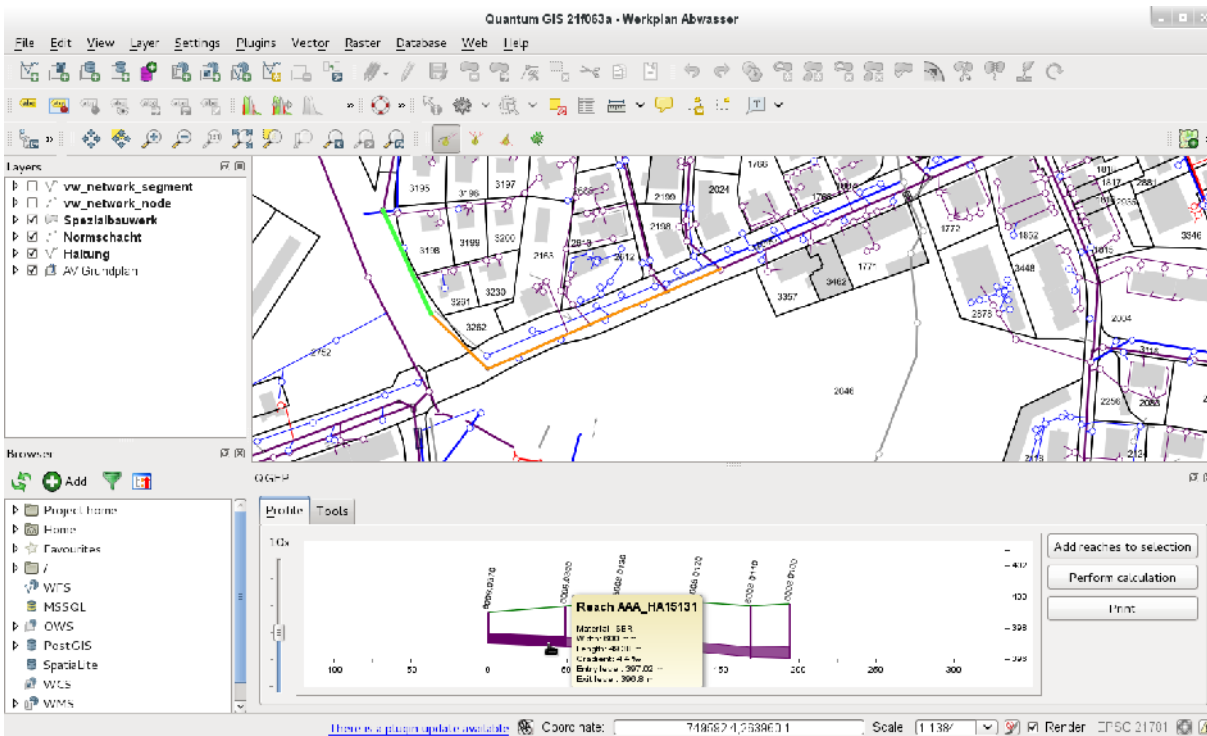


Abbildung 4: Profildarstellung entlang eines Kanalverlaufs (orange Haltungen in Kartenansicht)

Kontakt zum Autor:

Dr. Andreas Neumann  
Stadt Uster  
Oberlandstrasse 78, CH-8610 Uster  
++41-44-944 72 66  
andreas.neumann@uster.ch

Matthias Kuhn  
Geiselweidstrasse 43, CH-8400 Winterthur  
matthias.kuhn@gmx.ch

Literatur

- [1] VSA (Verband Schweizer Abwasser- und Gewässerschutzfachleute): Datenstruktur Siedlungsentwässerung des VSA, <http://www.vsa.ch/fachbereiche-cc/siedlungsentwaesserung/vsa-dss/>
- [2] QGEP Team: QGEP Projekthomepage mit Code, Wiki, Issues, <https://github.com/qgеп/QGEP>
- [3] NetworkX: High-productivity software for complex networks, <http://networkx.github.com/>
- [4] Bostock, Mike: d3.js – data-driven documents, <http://d3js.org/>

## Das SHOGun-WebGIS Framework

TILL ADAMS

Die Anforderungen an WebGIS-Lösungen werden vielseitiger und komplexer. Oft werden verwaltungsweite Lösungen benötigt, ein umfassendes Berechtigungskonzept möglichst mit Anbindung an externe Authentifizierungsdienste wie LDAP nachgefragt oder auch die Integration von Geo- und nicht-Geo-Diensten bei gleichzeitiger Absicherung gefordert.

Solche komplexen WebGIS-Lösungen erfordern ein reibungsloses Zusammenspiel und eine hohe Flexibilität seitens aller eingesetzten Komponenten. Als eine Lösung stellt terrestris seit Ende 2012 das Framework „SHOGun“ („Spring, Hibernate, OpenLayers, GeoExt und... d weitere“) als OpenSource Software bereit, gleichzeitig kommt SHOGun auch in Projekten zum Einsatz. SHOGun besteht aus einem Java-Backend, welches neben der Benutzer- und Rechteverwaltung ein Framework zum Aufsetzen von eigenen Diensten bietet. Dazu liefert SHOGun einen frei konfigurierbaren WebGIS-Client auf Basis von ExtJS, GeoExt und OpenLayers.

Durch den Einsatz dieser renommierten und hochqualitativen Open Source-Frameworks wird die Nachhaltigkeit, Wartbarkeit sowie die Qualität der Anwendung gewährleistet.

Der Vortrag stellt den Stand des OpenSource Projektes SHOGun vor. Dabei wird neben dem aktuellen Stand auch die Roadmap vorgestellt. Weiterhin werden aktuelle Projekte und dafür entwickelte Erweiterungen, die SHOGun verwenden vorgestellt. SHOGun ist aktuell in einem großen Forschungsprojekt sowie bei einer Lösung für die gesamte Wasserwirtschaft eines Bundeslandes im Einsatz. Highlights dabei sind beispielsweise das automatisierte Anlegen von WMS-Layern über die GeoServer-API mit SLD-Editor oder die Erstellung eigener GIS-Client-Instanzen.

## Datenqualität von OSM in der Schweiz

MICHAEL SPRENG

Eine immer wiederkehrende Frage ist, wie gut sind die OSM Daten? Dieser Vortrag soll einige Aspekte der Qualität und Vollständigkeit der OSM Daten in der Schweiz beleuchten. Die Schweiz ist kein grosses Land, und trotzdem zeigt sich die für OSM typische grosse Bandbreite: Es gibt sehr detailliert gemappte Gebiete, aber auch solche, wo erst das Hauptstrassennetz und ein paar von Luftbildern mit schlechter Auflösung abgezeichnete Wälder verfügbar sind.

OSM bietet vielen geographischen Features Platz. Wichtig ist sicherlich das Strassennetz, welches Routing ermöglicht und häufig auch den Hintergrund für Spezialkarten darstellt. Für die Adresssuche braucht es Strassennamen, aber auch die Administrativen Einheiten. Dann gibt es noch eine breite Palette von Points of Interest: Bushaltestellen, Restaurants, Hotels. Aber auch Brunnen (mit oder ohne Trinkwasser) oder markante Bäume. Dazu kommen noch Informationen die für Spezialkarten Verwendung finden wie Wanderwege, Skipisten oder Skatingrouten.

Im Vortrag wird für einige ausgewählte Aspekte versucht eine quantitative Bewertung der Qualität des OSM-Datenbestandes in der Schweiz darzulegen und deren Entwicklung aufzuzeigen. Als Vergleichsgrundlage dazu werden wo möglich offizielle Daten verwendet wie zum Beispiel das Eidgenössische Gebäude- und Wohnungsregister für die Strassennamen.

## Die gemeinsame Nutzung von MS-MSQLServer und PostgreSQL/PostGIS sowie weiterer OSGeo-Software in einer Fachdatenbank für kulturelles Erbe mit GIS-Anbindung

CHRISTOF SCHUPPERT, KATRIN WOLTERS, ULRICH HIMMELMANN

Die Generaldirektion Kulturelles Erbe Rheinland-Pfalz setzt im Bereich der Geodatenverarbeitung seit 2010 auf Open Source Software wie PostgreSQL/PostGIS, Quantum GIS, Mapserver und Mapbender. Zur Verwaltung der Fachdaten wird der MS SQL-Server eingesetzt, die clientseitig installierten Datenbankfrontends basieren auf .NET. Um die Vorteile aus beiden „Welten“ nutzen zu können, wird seit 2011 an einer Verknüpfung gearbeitet, die vor allem auf dem Datenaustausch zwischen den Datenbanksystemen SQLServer und PostgreSQL/PostGIS aufbaut. Das Ergebnis ist eine Fachanwendung, die hinsichtlich der Darstellung und Bearbeitung von Geodaten je nach Erfordernissen des Users mit unterschiedlichen GIS-Clients genutzt werden kann. Dazu zählen ein in die Datenbank integrierter, OpenLayers-Kartenviewer ebenso wie eine Anbindung von QGIS sowie eines weiteren proprietären GIS. Derzeit wird zudem eine Verknüpfung mit dem auf Open Source Komponenten aufbauenden GeoPortal.rlp entwickelt, so dass dort in Zukunft WMS- und WFS-Dienste unmittelbar durch den User freigeschaltet bzw. aktualisiert werden können. Im Vortrag soll nicht nur das Ergebnis dieser Symbiose präsentiert werden, sondern auch einige der Hürden, die auf diesem Weg genommen werden mussten (z.B. Codierungsprobleme, Einrichten eines DB-Verbindungsservers mit ODBC, Nutzung von OpenQuery und ogr2ogr). Durch die Präsentation des noch in der Entwicklungsphase befindlichen Projektes erhoffen sich die Entwickler ein Feedback aus der FOSSGIS-Community und den Kontakt zu ähnlichen Projekten.



## Effizientes Mappen von Hausnummern

### Keypad-Mapper 3 in der Praxis

MARKUS SEMM

Von den 30 Millionen Adressen in Deutschland sind derzeit in OSM nur ca. 10% erfasst. In anderen Ländern ist der Anteil meist noch geringer. Adress-Suchmaschinen wie NOMINATIM, aber auch Navigationssysteme, LBS-Anwendungen u.v.m. sind auf hinreichend genaue Adressangaben angewiesen. Steve Coast, der Gründer von OSM, hat erst kürzlich darauf hingewiesen, daß die noch am Anfang befindliche Erfassung von Adressen in OSM der letzte ernstzunehmende Grund ist, proprietären Karten gegenüber OSM in manchen Mainstream-Anwendungen den Vorrang zu geben. Eine Unterstützung von Mappern beim effizienten Erfassen von Hausnummern und Adressen ist daher wünschenswert.

Keypad-Mapper 3 ist eine Android-Software, die hocheffizientes Mappen von Hausnummern und Adressen ermöglicht.

Die neue Version 3 wartet mit produktivitätssteigernden Features auf, die qualitativ hochwertige Datenerfassung mit hohem Durchsatz verbinden. Die App ist auch von OSM-Einsteigern leicht zu bedienen und daher massentauglich, z.B. für Anfängerkurse in Schulen, VHS etc.

Im Vortrag soll aufgezeigt werden, wie Keypad-Mapper 3 das Mappen von Hausnummern und Adressen unterstützt.

Referent: Dipl.-Ing. Markus Semm Gründer und Geschäftsführer der ENAiKOOON GmbH, Berlin (Anbieter von Telematiklösungen für gewerbliche Kunden) Herr Semm ist einer der Top 1500 Mapper weltweit. ENAiKOOON setzt bei seinen Lösungen ausschließlich auf OpenStreetMap und unterstützt die Community inhaltlich wie auch finanziell

Warum Hausnummern / Adressen mappen? Welche Datenmodelle gibt es für Hausnummern und Adressen in OSM? Historie der Software Features der Software Backend-Unterstützung des Hausnummern-Mappers durch ENAiKOOON OSM-Dienste Seiteneffekt: Beitrag zum OpenSource Projekt [opencellids.org](http://opencellids.org) Gewinnung neuer Mapper durch einfachen Einstieg: Kursmaterial

## Eine Karte in jeder Sprache

JOCHEN TOPF, TIM ALDER



OpenStreetMap ist mit dem Anspruch angetreten, eine Karte der ganzen Welt zu schaffen, eine Karte, die von jederman genutzt werden kann. Die Karten, die aus OSM-Daten erstellt wurden, haben aber häufig das Problem, dass die Namen von Länder, Städte und POIs für die Nutzer der Karten nicht lesbar sind. Der europäische Tourist, der seine Asienreise plant, sieht oft nur unverständliche Namen in unbekanntem Schriftzeichen. Andersherum, kann ein Schulkind in Nepal die europäischen Namen nicht verstehen. Außerdem gibt es viele Konflikte in mehrsprachigen Regionen, welche Namen denn nun auf der Karte

erscheinen sollen.

Diesen Problemen kann nur mit einer mehrsprachigen Karte abgeholfen werden. Jedem Nutzer soll eine OSM-Karte in seiner Sprache zur Verfügung gestellt werden. Die Daten dazu sind bei OSM häufig schon vorhanden, aber es braucht auch die Software um damit umgehen zu können. Der Wikimedia Deutschland e.V. hat im Jahre 2012 die Entwicklung von Software für eine solche Karte finanziert und damit wichtige Schritte ermöglicht. Seither arbeiten wir daran, die mehrsprachige Karte allgemein zur Verfügung zu stellen.

Der Vortrag stellt den Stand des Projektes dar und beschreibt was für technische und soziale Probleme zu überwinden waren (und noch sind). Er geht auf die technischen Hintergründe des Kartenservers auf Basis von Mapnik und der MapQuest-Render-Software ein und zeigt wie mit der manuellen Datenerfassung durch tausende OSM-Enthusiasten in Verbindung mit automatischer Transliteration und schnellem Kartenrendering eine wirklich überall nutzbare Karte möglich wird.

<http://mlm.jochentopf.com/>

## Einführung in das WebFramework GeoDjango

### The Web framework for perfectionists with deadlines

STEPHAN WAGNER

Django ist ein in Python programmiertes, quelloffenes Web-Framework, welches dem Model-View-Controller-Schema (MVC) folgt und auf die explizite Konfiguration der Anwendung setzt. GeoDjango erweitert die Django-Basis zu einem leistungsfähigen geographischen Web-Framework, von neuen Feldern für OGC-Geometrien im objektrelationalen Django-Mapper (ORM) über abstrahierte Python-Schnittstellen für Geometrie-Operationen und -Datenformate bis hin zur automatisch generierten, produktionsstauglichen Administrationsoberfläche, welche direkt die kartographische Bearbeitung von Geometrien erlaubt.

Nach dieser Einführung werden gängige Architekturen einer GeoDjango-Weblösung skizziert, von der Datenbank bis zum http-Server und dem Webbrowser. Als Einstieg ins Handwerk wird aufgezeigt wie typischerweise eine GeoDjango-Entwicklungsumgebung aufzusetzen ist. Der Umgang mit den verschiedenen GeoDjango-APIs bildet den Kern der Betrachtungen. Es wird auf die zentralen Aspekte der Model-API, der Database-API und der GeoQuerySet-API eingegangen. Ein erster Blick in die GEOS- und GDAL-API runden den Block ab.

Das GeoDjango Framework liefert Python-Objekte. Es wird beispielhaft gezeigt, wie diese im Django-Template-Engine in eine vom Web-Browser interpretierbare Form aufgearbeitet werden, oft als JSON-Objekte, welche der JavaScript-Mapping-Library der Wahl übergeben werden. Hinweise zum Deploying einer GeoDjango-Applikation runden die Einführung ab.

## Entwicklung und Anwendung von Downloaddiensten im Kontext der europäischen Geodateninfrastruktur INSPIRE

JÜRGEN WEICHAND

### Einleitung:

Der Zeitplan der europäischen Geodateninfrastruktur INSPIRE sieht die Bereitstellung von Downloaddiensten ab dem Jahr 2012 vor (vgl. Abb. 1). Downloaddienste übertragen originäre, unpräzentierte Geodaten. Die heruntergeladenen Geodaten können – im Gegensatz zu den über einen Darstellungsdienst bereitgestellten Präsentationsgrafiken – individuell analysiert, verarbeitet und präsentiert werden.

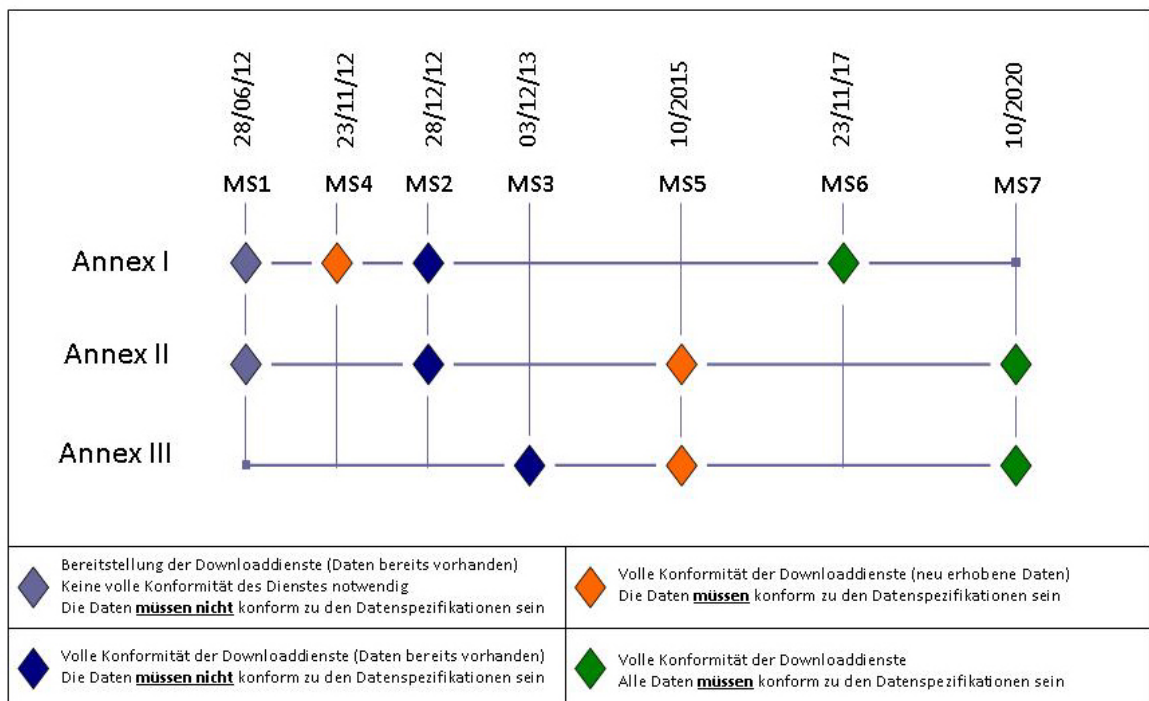


Abb. 1: Zeitplan INSPIRE-Downloaddienste [1]

Im Vortrag werden die rechtlichen Vorgaben und technischen Empfehlungen zu INSPIRE-Downloaddiensten sowie die hierfür verwendeten Normen und Standards erörtert. Die aktuellen Technical Guidance 3.0 [2] empfehlen die drei Umsetzungsvarianten Pre-defined Atom, Pre-defined WFS und Direct WFS auf Basis der Standards Atom (GeoRSS) und WFS 2.0 (vgl. Abb. 2).

## Entwicklung und Anwendung von Downloaddiensten im Kontext der europäischen Geodateninfrastruktur INSPIRE

Pre-defined Atom	Pre-defined WFS	Direct WFS
<ul style="list-style-type: none"> <li>▪ <b>Standards:</b> Atom, GeoRSS, OpenSearch</li> <li>▪ <b>Eignung:</b> Raster- und Vektordaten</li> </ul>	<ul style="list-style-type: none"> <li>▪ <b>Normen / Standards:</b> WFS 2.0 (ISO 19142) FES 2.0 (ISO 19143) GML 3.2 (ISO 19136)</li> <li>▪ <b>Konformitätsklasse:</b> Simple WFS (WFS 2.0)</li> <li>▪ <b>Eignung:</b> Vektordaten</li> </ul>	<ul style="list-style-type: none"> <li>▪ <b>Normen / Standards:</b> WFS 2.0 (ISO 19142) FES 2.0 (ISO 19143) GML 3.2 (ISO 19136)</li> <li>▪ <b>Konformitätsklasse:</b> Basic WFS (WFS 2.0)</li> <li>▪ <b>Eignung:</b> Vektordaten</li> </ul>

Abb. 2: Umsetzungsvarianten INSPIRE-Downloaddienste nach [2]

Im Folgenden werden verschiedene OpenSource-Lösungen hinsichtlich ihrer Eignung für die Bereitstellung von INSPIRE-Downloaddiensten betrachtet. Hierbei wird insbesondere auf die unterschiedlichen WFS 2.0-Fähigkeiten der etablierten OpenSource-Produkte UMN MapServer, GeoServer und deegree eingegangen.

Abschließend werden Anwendungsmöglichkeiten von INSPIRE-Downloaddiensten vorgestellt. Aufgrund der speziellen Semantik des INSPIRE-Atom-Profiles ist eine Nutzung bestehender GeoRSS-Clients nur eingeschränkt möglich. Ferner wird WFS 2.0 noch von sehr wenigen Desktop-GIS unterstützt. Im Rahmen einer Masterarbeit wurde QuantumGIS durch Plugins erweitert, die eine Nutzung von INSPIRE-Downloaddiensten im freien Geoinformationssystem ermöglichen. Der „INSPIRE Atom Client“ und der „WFS 2.0 Client“ stehen über das QGIS Plugin-Repository [3] zur Verfügung.

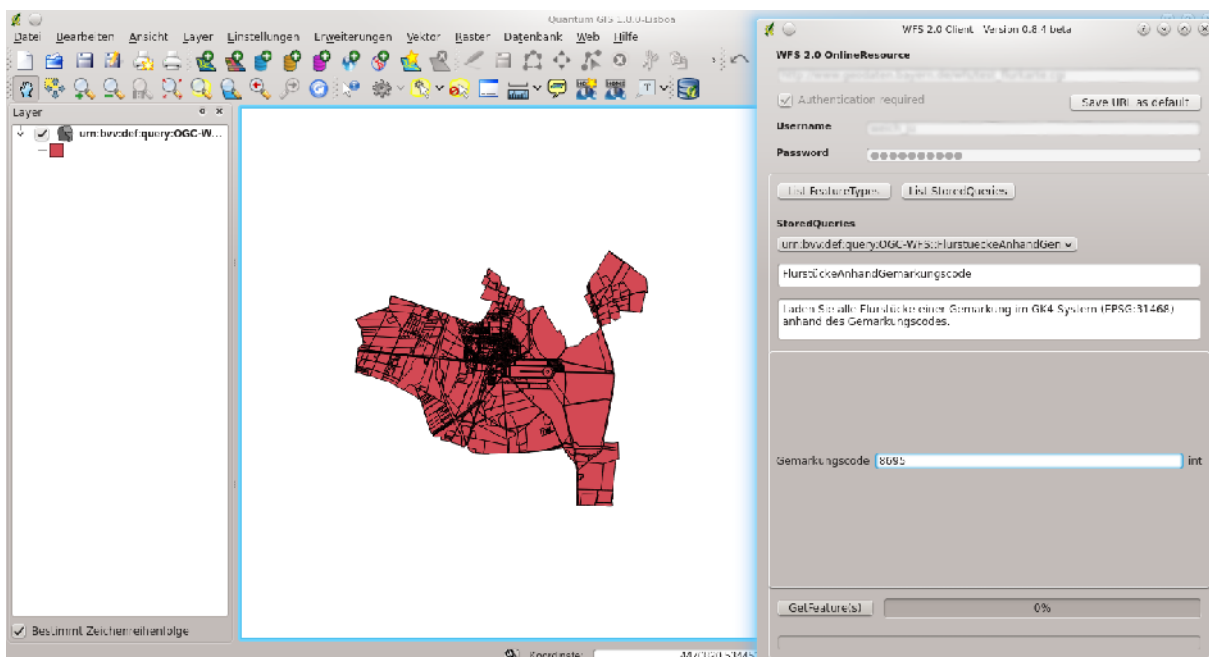


Abb. 3: QGIS WFS 2.0 Client – WFS auf Flurstücke und Gebäude (geodaten.bayern.de)

## Entwicklung und Anwendung von Downloaddiensten im Kontext der europäischen Geodateninfrastruktur INSPIRE

Der Vortrag basiert auf der gleichnamigen Masterarbeit [4], die im Januar 2013 bei der Hochschule Anhalt im Onlinestudiengang Geoinformationssysteme eingereicht wurde sowie auf Erfahrungen bei der Bereitstellung von INSPIRE-Downloaddiensten durch die Bayerische Vermessungsverwaltung.

Kontakt zum Autor:

Jürgen Weichand  
Landesamt für Vermessung und Geoinformation  
Alexandrastraße 4, 80538 München  
+ 49-89-2129-2984  
juergen.weichand@lvg.bayern.de  
<http://www.geodaten.bayern.de>

### Literatur

- [1] *Geodateninfrastruktur Deutschland, AK-Geodienste*: Handlungsempfehlungen für die Bereitstellung von INSPIRE konformen Downloaddiensten, Version 1.0, 2012, [http://www.geoportal.de/SharedDocs/Downloads/DE/GDI-DE/Handlungsempfehlungen\\_INSPIRE\\_Downloaddienste.pdf](http://www.geoportal.de/SharedDocs/Downloads/DE/GDI-DE/Handlungsempfehlungen_INSPIRE_Downloaddienste.pdf), letzter Zugriff: 22.04.2013
- [2] *Initial Operating Capability Task Force*: Technical Guidance for the implementation of INSPIRE Download Services, Version 3.0, 2012, [http://inspire.jrc.ec.europa.eu/documents/Network\\_Services/Technical\\_Guidance\\_Download\\_Services\\_3.0.pdf](http://inspire.jrc.ec.europa.eu/documents/Network_Services/Technical_Guidance_Download_Services_3.0.pdf), letzter Zugriff: 22.04.2013
- [3] *Quantum GIS*: Plugins Repository <http://plugins.qgis.org>, letzter Zugriff: 22.04.2013
- [4] *Weichand, Jürgen*: Entwicklung und Anwendung von Downloaddiensten im Kontext der europäischen Geodateninfrastruktur INSPIRE, Masterarbeit, Hochschule Anhalt, Dessau, 2013, <http://www.weichand.de/masterarbeit-inspire-downloaddienste/>, letzter Zugriff: 23.04.2013

## **Flexibles System zur geostatistischen Visualisierung am Beispiel von „Personalized Pollen Profiling“ Daten**

NIKOLAI BOCK, KLAUS BÖHM

### **Einleitung**

In Deutschland reagieren 41 Prozent der Kinder und Jugendlichen, sowie immer öfter Erwachsene auf Atmungs- oder Lebensmittelallergene. Auch Kreuzallergien treten häufiger auf. Das heißt, dass Menschen gleichzeitig zum Beispiel gegen Birkenpollen und Äpfel bzw. Nüsse allergisch sind [1]. Zudem wird der Prozentsatz aller Personen mit Allergien weltweit als hoch eingeschätzt. Man geht von 10% bis 20% aus [2]. Die Relevanz ist weiter steigend. Deshalb sind die Möglichkeiten der genauen Messung der Pollenkonzentration in der Luft ein zentraler Teil der palynologischen Entwicklungen. Derzeit werden die Informationen von stationären Pollenfallen erfasst und müssen dann anschließend von Experten identifiziert und klassifiziert werden. Dabei sind die Messungen örtlich beschränkt und geben keine Information über die persönliche Belastung eines einzelnen Patienten. In dem Forschungsprojekt 3P-GM – Personalized Pollen Profiling and Geospatial Mapping wird hierzu ein tragbares Pollensammelgerät entwickelt, mit dessen Hilfe persönliche Pollenbelastungsprofile erstellt werden können. Dieses gerät saugt Luft in der Umgebung des Anwenders an und bläst diese auf einen beschichteten Objektträger. Auf diesem Objektträger bleiben die Pollen haften. Zusätzlich misst das Gerät noch weitere sogenannte Metainformation, wie eine GPS-Position, Luftfeuchtigkeit, Luftdruck und die Temperatur. Die Objektträger werden anschließend gescannt. Zur Identifizierung und Klassifizierung der Pollen werden Bildanalysealgorithmen entwickelt, welche eine Segmentierung und Klassifizierung vornehmen [3].

Die so gesammelten Polleninformationen haben einen temporalen/räumlichen Bezug, so dass sie abschließend auch in diesen beiden Kontexten visualisiert werden können. Hierfür werden neue, innovative Dienste zur geostatistischen Visualisierung entwickelt. Um die Prozessketten hierfür offen zu halten, wird ein flexibles System auf Basis von OGC-Standards konzipiert und umgesetzt, welches möglichst zu großen Teilen vom Client entkoppelt ist. Diese Umsetzung ist der Fokus dieses Beitrags.

### **Personalized Pollen Profiling**

Das vorgestellte System soll, wie bereits erwähnt, im Bereich von Personal Pollen Profiling genutzt werden. Abbildung 1 zeigt ein vereinfachtes Modell der in dem oben beschriebenen Forschungsprojekt entwickelten Komponenten.

Nach dem Scannen der Objektträger werden die Bilder von den Algorithmen des „Pollen Detection System“ analysiert. Hierbei werden zunächst die Pollenpartikel heraussegmentiert und anschließend klassifiziert. Die Partikel, welche nicht eindeutig klassifiziert werden konnten, werden dann von Experten a posteriori manuell der korrekten Pollenart zugeordnet. Eine detaillierte Erläuterung des „Pollen Detection System“ findet man in [3].

Mit Hilfe von geostatistischen Visualisierungen können nun vor allem auch personalisierte Belastungsanalysen unterstützt werden. Zur geostatistischen Auswertung gehören die in Abbildung 1 eingerahmten Komponenten: ein Geostatistik-Service und ein Kartenclient für den Analyse-Nutzer. Diese werden nachfolgend näher erläutert.

## Flexibles System zur geostatistischen Visualisierung am Beispiel von „Personalized Pollen Profiling“ Daten

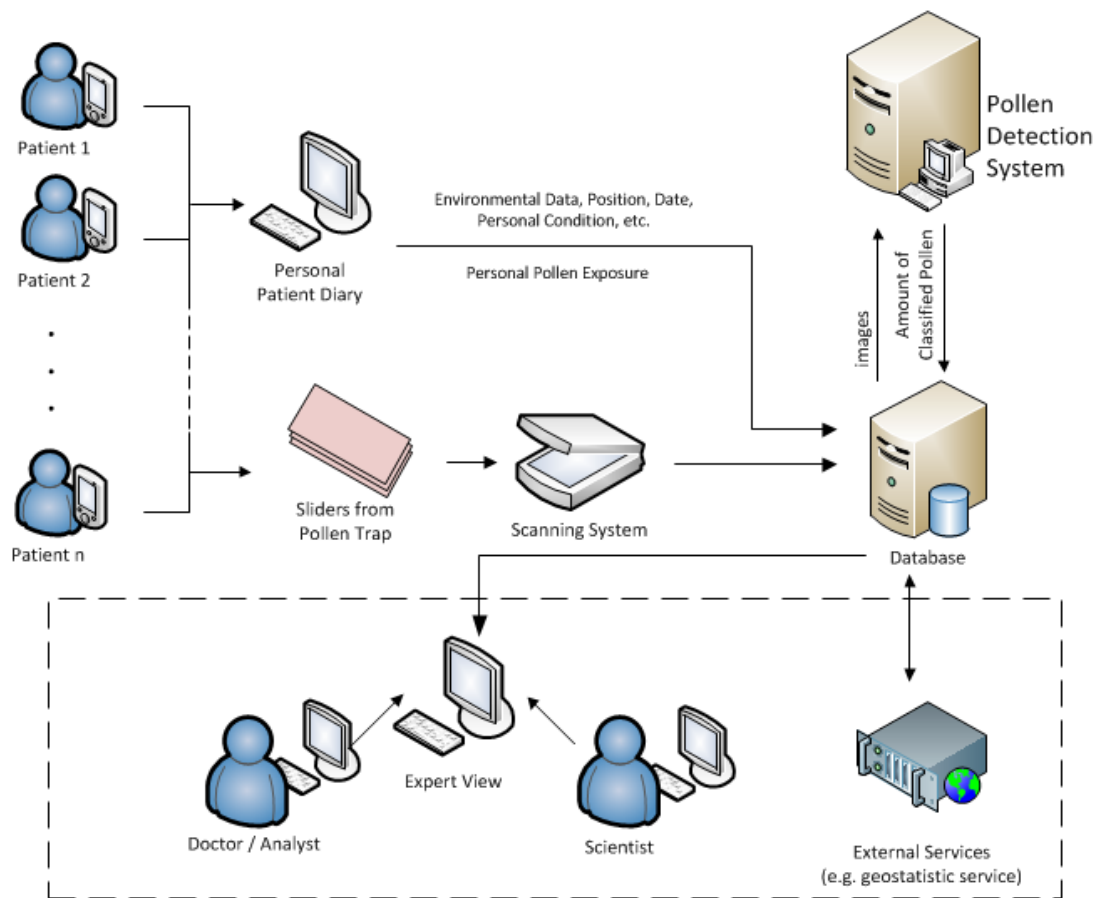


Abbildung 1: Systemmodell von 3P-GM

## Flexibles System zur geostatistischen Visualisierung

Ziel bei der Entwicklung der geostatistischen Visualisierungskomponenten ist die Konzeption und Umsetzung eines möglichst flexiblen Systems. Zu Grunde liegt der Gedanke einer Architektur, die die jeweiligen Stärken bzw. vorhandenen Funktionalitäten unterschiedlicher Systemkomponenten nutzt. Zudem sollte das System hinsichtlich weiterer Clients oder Nutzungskonzepte offen sein, so dass eine möglichst einfache und etablierte Schnittstelle zwischen der Client- und Serverschicht genutzt werden kann.

Die Schnittstelle wurde durch die Nutzung des Web Mapping Service (WMS) Standards gelöst. Der Schwerpunkt von WMS-Services liegt allerdings bei Rasterkarten-Rendering, daher bieten sie keinerlei Möglichkeiten der Prozessierung der Ausgangsdaten. Dieser Bedarf ist allerdings in dem System gegeben. In einer klassischen OGC-Service-Architektur wird dies durch Web Processing Services (WPS) gelöst.

Um die Client-Funktionalität auf dem WMS zu belassen, wird die Prozessierung in dem konzipierten System, wie in Abbildung 2 zu sehen, in den Darstellungsprozess integriert. Hierfür wird das Framework Geoserver genutzt. Dieses bietet die Möglichkeit, sogenannte „Transformations“ in ihrer SLD-basierten Darstellungsdefinition zu integrieren. Somit lassen sich Prozesse zwischen RAW-Daten-Zugriff und Renderingprozess schalten (vgl. [4]). Ein Nachteil ist allerdings, dass nur interne Geoserver-Prozess-Funktionen genutzt werden können. Das Geoserver Framework bietet einen eigenen WPS-Dienst an, welcher genutzt werden kann. Allerdings gibt es andere Frameworks, wie das in der



**Flexibles System zur geostatistischen Visualisierung am Beispiel von „Personalized Pollen Profiling“ Daten**

Abbildung 2 dargestellt WPS-Framework von 52°north, welche deutlich spezialisierter sind und mehr Funktionalität bieten. Beispielhaft sei die Möglichkeit der Nutzung eines R-Script-Service oder von GRASS Funktionen genannt.

Es ist daher notwendig, eine Schnittstelle zwischen einer Geoserver-Funktion und einem WPS zu schaffen.

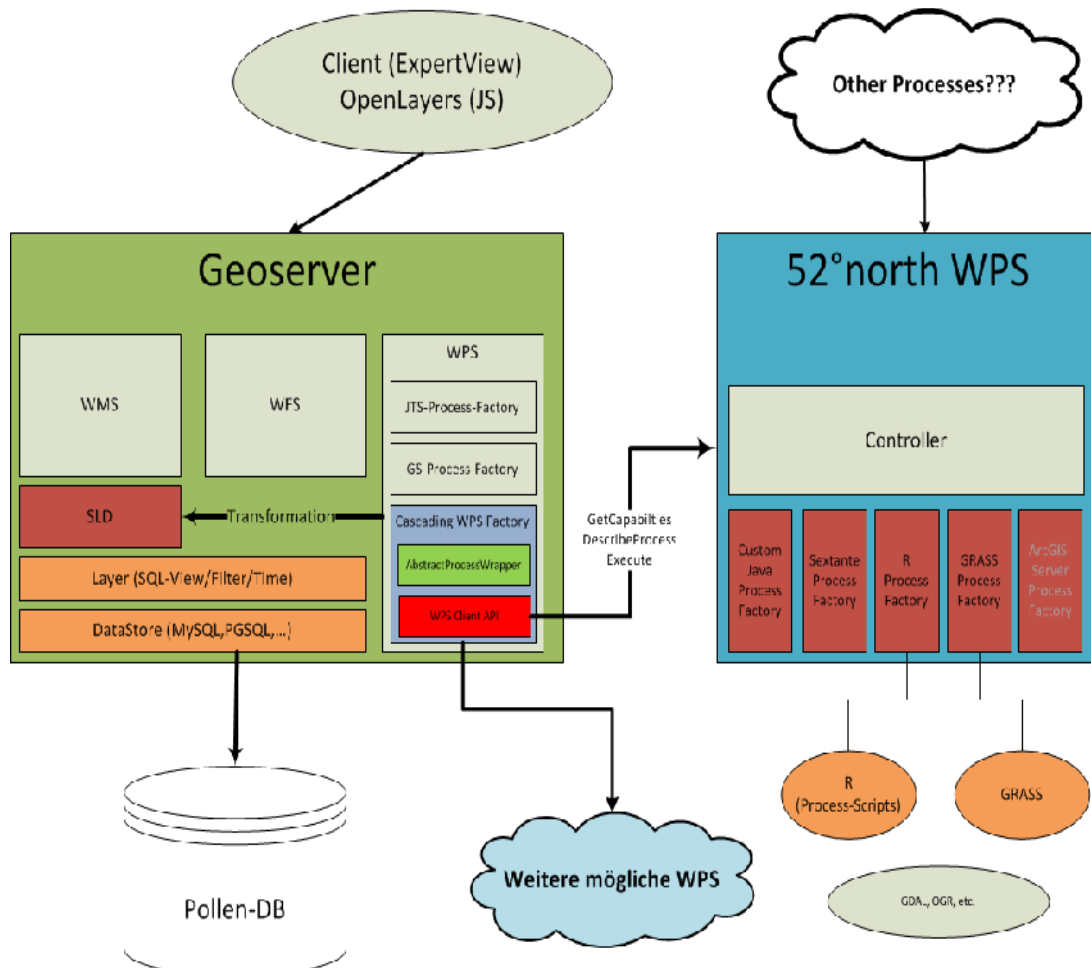


Abbildung 2: Modell für ein flexibles System eines Geostatistik-Service

Das Geoserver WPS Framework nutzt „Process-Factory“-Klassen um Prozesse zu implementieren. Dank der Nutzung von Spring [5] und Dependency Injection [6] lassen sich eigene Ableitungen geeignet injizieren. Die selbst injizierte Factory ist die „CascadingWPSFactory“. Diese Factory muss alle relevanten Daten über die Prozesse, sowie die Prozessobjekte selber erstellen, bzw. zur Verfügung stellen. Sie liest dafür zunächst die Metainformation (GetCapabilities / DescribeProcess) über die WPS-Schnittstelle des entfernten Dienstes aus. Aufgerufene Prozesse werden von der Factory als interne Prozesse verwaltet, welche den Execute-Aufruf an die Remote-Dienste initiieren. Eine Objektinstanz der „CascadingWPSFactory“ stellt einen entfernten WPS dar, so dass auch jeder Dienst über dieses Modell einen eindeutigen Namespace erhält.

Derzeit wird in der Entwicklung als Remote-WPS das Service-Framework von 52°north verwendet. Dank der Verwendung der WPS Standard Schnittstelle ist die Möglichkeit einer Anbindung sämtlicher WPS-Frameworks vorgesehen.

## Analyst-Client

Aus Anwendersicht dient die geostatistische Visualisierung dem Analysten. Dieser benötigt zunächst eine komplette Übersicht über die Pollenkonzentration in Abhängigkeit von der Pollenart und der Zeit als interpolierte Karte. Dank der zusätzlichen Metainformation sind zudem besonders personalisierte Darstellungsvariationen von großem Interesse und in dieser Form einmalig. Diese bieten Analysten / Ärzten die Möglichkeit, ein persönliches Belastungsprofil zu beobachten. Später sind räumlich-zeitliche Visualisierungen unter Berücksichtigung von weiteren Umgebungsinformationen denkbar.

Eine Evaluation des Systems ist zunächst mit unterschiedlichen Interpolationsalgorithmen, wie Kriging [7] (vgl. [8]), Inversive Distance Weighting (IDW) [9] oder dem Barnes Surface Algorithmus [10], welcher besonders bei Wetteranalysen Verwendung findet, geplant. Bei der personalisierten Auswertung sind trackbasierte Darstellungsvarianten angedacht.

## Fazit

Der Beitrag zeigt die Entwicklung eines flexiblen Systems zur geostatistischen Prozessierung und Visualisierung. Auf Grund der Nutzung in einem innovativen Umfeld ist es auf dynamisch wachsende Anforderungen ausgelegt. Hierzu trägt auch die Nutzung von ausschließlich freien Softwarekomponenten und möglichst vieler Standards bei.

Kontakt zu den Autoren:

Nikolai Bock  
i3mainz – Institut für Raumbezogene Informations- und Messtechnik  
Fachhochschule Mainz  
Lucy-Hillebrand-Str. 2, 55128 Mainz  
nikolai.bock@fh-mainz.de

Prof. Dr. Klaus Böhm  
i3mainz – Institut für Raumbezogene Informations- und Messtechnik  
Fachhochschule Mainz  
Lucy-Hillebrand-Str. 2, 55128 Mainz  
klaus.boehm@fh-mainz.de

## Literatur

- [1] Bundesministerium für Ernährung, Landwirtschaft und Verbraucherschutz: Mehr Lebensqualität – Aktionsplan gegen Allergien, S 3, 2007; [http://v1.bitv-test.de/dateien/pdf\\_test/794/AktionsplanAllergien.pdf](http://v1.bitv-test.de/dateien/pdf_test/794/AktionsplanAllergien.pdf) (Zugriff: 26.04.2013)
- [2] DYKEWICZ, M; HAMILOS, D.: Rhinitis and sinusitis, The Journal of allergy and clinical immunology. 125(2) pp. S103-S115, 2010.
- [3] LOZANO-VEGA, G.; BENEZETH, Y.; UHLER, M.; BOOCHS, F.; MARZANI, F.: Sketch of an automatic image based pollen detection system, DGPF Tagungsband 21 / 2012
- [4] <http://blog.opengeo.org/2012/08/22/surface-interpolation-in-geoserver/> (Zugriff: 24.4.2013)
- [5] <http://www.springsource.org> (Zugriff 26.04.2013)
- [6] [http://en.wikipedia.org/wiki/Dependency\\_injection](http://en.wikipedia.org/wiki/Dependency_injection) (Zugriff 26.04.2013)
- [7] BOHLING, G. Kriging, 2005, <http://people.ku.edu/~gbohling/cpe940/Kriging.pdf> (Zugriff 26.04.2013)

**Flexibles System zur geostatistischen Visualisierung am Beispiel von „Personalized Pollen Profiling“ Daten**

[8] ALBA, F., NIETO-LUGILDE, D., COMTOIS, P., DIAZ DE LA GUARDIA, C. DE LINARES, C., RUIZ, L.: Airborne-pollen map for *Lea europaea* L. In eastern Andalusia (Spain) using GIS: Estimation models, *Aerobiologia* 22(2) pp.107-116, Springer Science+Business Media B.V., 2006

[9] [http://gisbsc.gis-ma.org/GISBScL7/de/html/VL7a\\_V\\_lo7.html](http://gisbsc.gis-ma.org/GISBScL7/de/html/VL7a_V_lo7.html) (Zugriff 26.04.2013)

[10] BARNES, S. L.: A Technique for Maximizing Details in Memerial Weather Map Analysis, *J. Appl. Meteor.*, 3, 396–409, 1964

## FOSS-GIS im Kanton Basel-Landschaft

CLEMENS RUDERT

Die GIS-Fachstelle Basel-Landschaft begann im Jahr 2010 die Suche nach einer neuen GIS-Infrastruktur. Diese sollte sowohl ein neues Web-GIS wie auch Anbindungsmöglichkeiten für die im Kanton bestehenden Desktop-GIS beinhalten. Nach einem ausführlichen Evaluierungsprozess fiel der Entscheid zugunsten von OpenSource. Speziell auf das c2cgeoportal. Nach mehr als 2 Jahren und bereits einiger Zeit in produktivem Betrieb von GeoViewBL, kann die GIS-Fachstelle rück- und vorwärts-gewandt Einblick in dieses Projekt und in Erfahrungen geben. Es schweben uns 2 Vorträge vor. Der erste allgemein auf das Projekt und das Resultat bezogen. Der zweite dann speziell zu (Eigen-)Entwicklungen die auf der Basis vom c2cgeoportal möglich sind. Der 2. Vortrag soll eigentlich nur zeigen was möglich ist.

Interessante Aspekte für den 1. Vortrag sind: • OpenSource im Kanton/in öffentlichen Einrichtungen • der eigentliche Evaluationsprozess • Entwicklungsgemeinschaft auf breiter Basis – Die Zusammenarbeit mit anderen c2cgeoportal-Teilnehmern (Vor-/Nachteile) • Finanzielle Aspekte (Was kostet OpenSource wirklich?) • GeoViewBL und seine Bestandteile 1 (Infrastruktur, Leistungsfähigkeit) • GeoViewBL und seine Bestandteile 2 (Pakete und Bibliotheken, MapFish 2nd Level) Interessante Aspekte für den 2. Vortrag sind: • Anpassbarkeit des c2cgeoportals • Welche Entwicklungswerkzeuge setzen wir ein? • Wie entwickeln wir parallel am selben Projekt? • Eigenentwicklungen (Scan-Tool, CrimeMapping, Crisk-Tool, Abfragen-Tool)

## Frauen in OpenStreetMap. Oder: Vielfältigere Gruppen für OSM begeistern.

MANUELA SCHMIDT UND SILVIA KLETTNER (TECHNISCHE UNIVERSITÄT WIEN)

Aus bisherigen Studien zu OpenStreetMap (z.B. [1], [2]) geht hervor, dass nur 2–5% der OSM-Beitragenden weiblich sind. Dieser Anteil ist auch im Vergleich zu anderen Open Source- bzw. Open Content-Projekten sehr niedrig – bei Wikipedia wird beispielsweise von einem Anteil von ca. 9–13% ausgegangen [3]. Aber auch bei den teilnehmenden Männern scheint die Vielschichtigkeit der Gesellschaft nicht vertreten zu sein. Haklay und Budhathoki [4] beschreiben den prototypischen Teilnehmer als technologieversiert, gut ausgebildet und zwischen 20 und 40 Jahren. Für die OSM-Community ist dieses Thema durchaus relevant: Würden sich vielfältigere Gruppen von OSM angesprochen fühlen, gäbe es auch ein größeres Potential von Personen, die Daten erfassen und aktuell halten.

Um vielfältigere Gruppen für OSM zu begeistern, ist es wichtig zu verstehen, was diese Gruppen bisher von der Teilnahme abhält und welche Aspekte sie zur Beteiligung motivieren könnten. Fokus unserer Forschung ist die Gruppe der Frauen, die wie oben beschrieben in OSM derzeit stark unterrepräsentiert ist. Dabei wurden die folgenden Methoden angewandt: Eine Langzeitstudie mit neuen Mapperinnen, Telefon-Interviews mit aktiven Community-Mitgliedern, sowie ein Online-Fragebogen für Personen, die OSM kennen, aber nicht (mehr) aktiv beitragen. Im folgenden werden die Ergebnisse der Langzeitstudie, sowie erste Einzelergebnisse des Online-Fragebogens vorgestellt.

Die **Langzeitstudie** wurde im Frühjahr 2012 an der Technischen Universität Wien durchgeführt. In der qualitativen Studie wurden 11 neue Mapperinnen in ihrem Lernprozess zur aktiven Teilnahme in OpenStreetMap begleitet, um Motivatoren und Barrieren in der Auseinandersetzung mit OSM zu erheben. Die Teilnehmerinnen der Studie wurden gezielt ausgewählt, um dem typisch männlichen OSM-User zu entsprechen (technologieversiert, gut ausgebildet und zwischen 20 und 40 Jahren, siehe [4]). Zunächst wurden den Teilnehmerinnen in drei begleiteten Einheiten unterschiedliche OSM-Aktivitäten näher gebracht: 1) Mapping mit Walking Papers, 2) Mappen und Editieren auf Basis von Luftbildern mit JOSM, 3) Outdoor-Mapping mit GPS (Exkursion in den Wienerwald). Anschließend bekamen die Teilnehmerinnen die Aufgabe, zwei frei gewählte Gebiete selbstständig zu mappen und mit einem Vorher-/Nachher-Screenshot zu dokumentieren. Positive und negative Erfahrungen mit OSM wurden von den Teilnehmerinnen über die Laufzeit der Studie in einem Online-Mapping-Tagebuch festgehalten. Bei der Abschlussveranstaltung des Kurses stellten die Teilnehmerinnen ihre Mapping-Ergebnisse vor und diskutierten ihre Erfahrungen mit OSM, die im folgenden kurz zusammengefasst werden. Detaillierte Ergebnisse mit Originalkommentaren der Teilnehmerinnen finden sich in [5].

Zusammenfassung der berichteten *positiven Erfahrungen*:

- Erwerb neuer Fähigkeiten und neuen Wissens durch die Beschäftigung mit OSM
- Gemeinsames Erkunden der Umgebung bei der Exkursion
- Offener Zugang zu OSM und freie Nutzbarkeit der Daten
- Mapping als sinnvoller Beitrag für die Gesellschaft
- Unmittelbare Sichtbarkeit der eigenen Beiträge auf einer weltweit verfügbaren Karte
- Soziale Komponente des Mappens, z.B. Exkursion, gemeinsames Problemlösen

Zusammenfassung der berichteten *negativen Erfahrungen*:

- keine leicht ersichtliche Starthilfe bzw. Verwirrung durch eine Vielzahl OSM-bezogener Informationsseiten (OSM-Wiki, Foren, Tutorien, etc.)
- Bedienung der Editoren und Erlernen des Tagging-Schemas zunächst komplex
- fehlendes visuelles Feedback (z.B. durch Wartezeiten auf das Rendering oder weil bestimmte Daten nicht gerendert werden)
- Probleme/Fehlermeldungen beim Daten-Upload

## Frauen in OpenStreetMap. Oder: Vielfältigere Gruppen für OSM begeistern.

- hohe Zeitintensität

Folgende vier *Maßnahmen* wurden außerdem von den Teilnehmerinnen erarbeitet, um ihnen ähnliche Personengruppen zur Beteiligung in OSM zu motivieren: (1) Projektbasiertes Mapping: Mapping-Aktionen, die nicht nur der Vervollständigung der Karte, sondern auch zusätzlichen Zielen dienen (z.B. Crisis Mapping). (2) Vereinfachung des Einstiegs z.B. über ein einfach auffindbares, nutzungsfreundliches Tutorium. (3) Soziale Events: gemeinschaftliches Mapping und stärkere Einbindung sozialer Vernetzungsoptionen. (4) Verbesserung des visuellen und technischen Feedbacks: Konstruktive Fehlermeldungen und Lob/Belohnungen für erfolgreichen Datenupload.

In einem **Online-Fragebogen** wurden die in der Langzeitstudie erhobenen Motivatoren und Maßnahmen als Basis verwendet, um mehr darüber zu erfahren, wie Personen, die OSM zwar kennen, aber derzeit nicht aktiv sind, für OSM (re)aktiviert werden können. Die Umfrage wurde im Februar 2013 durchgeführt und hauptsächlich über Mailinglisten im Geobereich verbreitet. Insgesamt haben 516 Personen teilgenommen; davon fielen 242 in die Kategorie der Personen, die OSM kennen, aber derzeit nicht aktiv beitragen. Im folgenden findet sich eine Übersicht der häufigst gewählten Barrieren und Motivatoren.

Gründe für die Inaktivität bei OSM:

- Beitragen ist zu zeitaufwändig (62%)
- OSM scheint in meiner Region vollständig zu sein (50%)
- Ich habe vergessen teilzunehmen (43%)

Aspekte, die zur (Wieder)Beteiligung bei OSM führen könnten:

- Mapping für einen konkreten Zweck (z.B. Crisis Mapping) (48%)
- Weniger zeitaufwändige Mapping-Lösungen (z.B. Smartphone-App) (44%)
- Tutorial für OSM-Neulinge (39%)

Insbesondere die Motivatoren zur (Wieder)Beteiligung unterscheiden sich jedoch nach OSM-Erfahrung der Personen.

Umfrage-Teilnehmende, die sich als derzeit aktiv in der OSM-Community eingestuft haben, wurden zu ihrem Mapping-Werdegang, sowie zu ihrer Einschätzung der Gender-Thematik in OSM befragt. Auswertung und Analyse des Online-Fragebogens sind noch nicht vollständig abgeschlossen. Die Umfrageergebnisse werden voraussichtlich im Sommer 2013 auf der Projektwebsite verfügbar sein.

**Diskussion.** Die hier vorgestellten Ergebnisse der Langzeitstudie und des Online-Fragebogens lassen sich wahrscheinlich auch auf weitere, bislang nicht erreichte Gesellschaftsgruppen anwenden – die genannten Motivatoren und Maßnahmen müssen aber nicht notwendigerweise den Bedürfnissen anderer Zielgruppen entsprechen. Es bedarf daher einer Zielgruppen-orientierten Vorgehensweise und maßgeschneiderter Maßnahmen, um vielfältige Gruppen für die Teilnahme in OSM zu begeistern. In einem Projekt, das so stark Community-getrieben ist wie OSM, können derartige Änderungen jedoch nur aus der Community selbst entstehen. Die Ergebnisse der vorgestellten Studien können hierbei als Diskussionsgrundlage dienen.

Details und Updates zum Projekt unter <http://cartography.tuwien.ac.at/fem2map>

Kontakt zu den Autorinnen:

Manuela Schmidt  
eMail: manuela.schmidt@tuwien.ac.at

Silvia Klettner  
eMail: silvia.klettner@tuwien.ac.at

Technische Universität Wien, Department für Geodäsie und Geoinformation  
Erzherzog-Johann-Platz 1/E120-6, A-1040 Wien

## Frauen in OpenStreetMap. Oder: Vielfältigere Gruppen für OSM begeistern.

<http://cartography.tuwien.ac.at>

Die in diesem Artikel vorgestellten Ergebnisse stammen aus dem Projekt fem2map – einem vom österreichischen Bundesministerium für Verkehr, Innovation und Technologie (BMVIT) geförderten Projekt der Programmlinie FemTech. Das Projekt ist eine Zusammenarbeit der Technischen Universität Wien und der Salzburg Research Forschungsgesellschaft. Besten Dank an die Kolleginnen Renate Steinmann und Elisabeth Häusler (beide Salzburg Research) sowie Yuwei Lin (University of Salford).

### Literatur

- [1] Stark, Hans-Jörg: Empirische Untersuchung der Motivation von Teilnehmenden bei der freiwilligen Erfassung von Geodaten, Fachhochschule Nordwestschweiz, Präsentation, 2011.
- [2] Lechner, Marco: Nutzungspotentiale crowdsourcing-erhobener Geodaten auf verschiedenen Skalen, Dissertation, Freiburg, 2011.
- [3] Khanna, Ayush: Nine out of ten Wikipedians continue to be men: Editor Survey – <http://blog.wikimedia.org/2012/04/27/nine-out-of-ten-wikipedians-continue-to-be-men> Aufgerufen am 30.4.2013.
- [4] Haklay, Muki und Budhathoki, Nama: OpenStreetMap – Overview and Motivational Factors, Horizon Infrastructure Challenge Theme Day, The University of Nottingham, 2010.
- [5] Klettner, Silvia, Schmidt, Manuela and Steinmann, Renate: Barrieren und Motivatoren von VGI-Projekten aus Gender-Perspektive. In: Koch, A., Bill, R. and Donaubaue, A., Geoinformationssysteme 2013. VDE-Verlag, Berlin, 2013.

## GeoCouch

### Ein n-dimensionaler Index für Apache CouchDB und Couchbase

VOLKER MISCHÉ

Bei Apache CouchDB und Couchbase handelt es sich um sogenannte dokumentbasierte Datenbanken. Sie gehören somit in die Kategorie der nicht-relationalen Datenbanksysteme für die sich der Sammelbegriff „NoSQL“ eingebürgert hat.

Eine Stärke von Apache CouchDB ist die (Multi-Master) Datenreplikation. Das bedeutet dass der Datenbestand verschiedener Datenbankinstanzen synchron gehalten werden kann, und dennoch Änderungen an beliebiger Stelle vorgenommen werden können.

Die Replikation beschränkt sich nicht nur auf Apache CouchDB, sondern mittlerweile auf ein gesamtes Ökosystem. So ist es möglich Daten mittels HTML5-Technologie mit einem Browser zu synchronisieren. Diese stehen dem Anwender somit auch offline zur Verfügung, ohne dass eine Verbindung zu einem Server bestehen muss.

Couchbase hingegen hat seine Stärke bei der Skalierung des Systems. So werden die Daten automatisch auf mehrere Rechner verteilt. Das nachträgliche Hinzufügen oder Entfernen von Rechnern ist über eine Web-Oberfläche sehr leicht möglich. Im Falle des Ausfalls eines Rechners läuft das System ohne Unterbrechung weiter.

GeoCouch bietet sowohl für Apache CouchDB als auch Couchbase eine Lösung um n-dimensionale Anfragen machen zu können. Es ist also nicht nur möglich räumlich Anfragen zu stellen, sondern weitere Eigenschaften wie z.B. Zeit, Größen oder beliebige andere numerische Werte mit einzubeziehen.

Apache CouchDB, Couchbase und GeoCouch sind Open-Source und stehen unter der Apache License Version 2.0.

Links

<https://github.com/couchbase/geocouch/>

<http://couchdb.apache.org/>

<http://www.couchbase.com/>



## Geodaten absichern mit MapProxy

OLIVER TONNHOFER, OMNISCALE GMBH & Co. KG

MapProxy kann als Kartenproxy WMS- und Kachelanfragen nicht nur weiterleiten, verarbeiten und beschleunigen, sondern diese auch absichern. Zur Absicherung stellt MapProxy eine flexible Schnittstelle bereit, mit der auch bereits bestehende Benutzerdatenbanken eingebunden werden können.

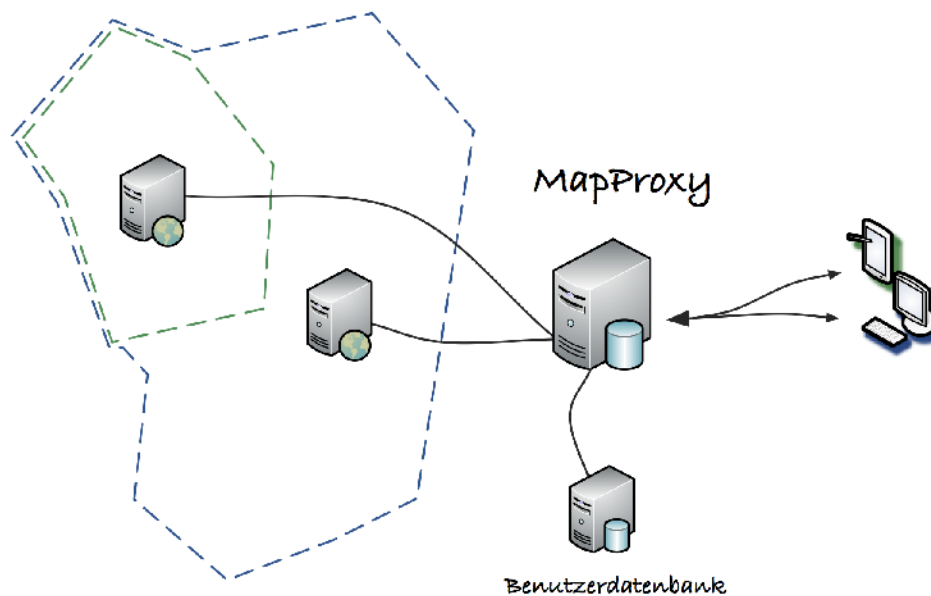


Abb. 1: MapProxy Absicherung

### Authentifizierung und Autorisierung

Bei der Absicherung von Diensten muss zwischen Authentifizierung und Autorisierung unterschieden werden. Unter Authentifizierung versteht man vereinfacht gesagt das Erkennen eines Benutzers. Dies kann über verschiedene Methoden erfolgen. Je nach Einsatzzweck und Sicherheitsanforderungen kann die Erkennung über die IP-Adresse des Benutzers, über den Vergleich von Benutzername und Passwort via HTTP Basic-Authentication oder mit komplexen Systemen wie RADIUS erfolgen.

Unter Autorisierung versteht man das Erlauben bestimmter Vorgänge. Im Kontext von MapProxy bedeutet dies z.B. den Zugriff auf einen bestimmten Layer per WMS. Die Autorisierung kann hierbei von einfachsten Anforderungen – alle authentifizierten Benutzer dürfen zugreifen – bis zu komplexen Lösungen – für jeden Benutzer liegen in einer Datenbank individuelle Zugangsbeschränkungen vor – gehen.

### Middleware

Die kurze Auflistung der möglichen Authentifizierungs- und Autorisierungsmöglichkeiten zeigt, dass hier eine Absicherungslösung für sämtliche Einsatzzwecke nur schwer zu realisieren ist. Daher wurde die Sicherheitsschicht von MapProxy als offene Schnittstelle entwickelt.

## Geodaten absichern mit MapProxy

Der Anwendungsentwickler implementiert hierbei die Authentifizierung und Teile der Autorisierung in einer individuellen WSGI-Middleware. Über die Sicherheitsschnittstelle wird MapProxy das Ergebnis der Autorisierung übermittelt, die anschließend durchgesetzt wird.

Entwickler können hierbei auf eine Vielzahl existierender Bibliotheken zurückgreifen, um beliebige Benutzerdatenbanken anzubinden und MapProxy in bestehende Sicherheitslösungen zu integrieren.

Die Autorisierungsschnittstelle von MapProxy kann hierbei für jede Anfrage gesondert unterscheiden, was erlaubt und was unterbunden werden soll:

- Freigabe ganzer Dienste
  - WMS
  - WMTS
  - TMS
  - KML
  - Demo
- Freigabe einzelner Layer
- Freigabe nach WMS Anfrage
  - GetCapabilities
  - GetMap
  - GetFeatureInfo
  - GetLegendGraphic
- Freigabe nach geographischem Bereich
  - Bounding Box
  - Polygon
  - Puffer um Geometrie

Die Freigaben können getrennt für jeden Layer angegeben werden. So können für Layer A GetMap- und GetFeatureInfo-Anfragen in einer BBOX und für Layer B GetMap-Anfragen innerhalb eines Polygons erlaubt werden. Die Capabilities-Dokumente werden entsprechend aktualisiert – Layer ohne Zugriffsrechte werden heraus gefiltert, Layerausdehnungen werden aktualisiert.

Bei der Freigabe von geographischen Bereichen werden die Kartenbereiche außerhalb der Geometrie transparent oder in der Hintergrundfarbe der Karte dargestellt.

## Geodaten absichern mit MapProxy

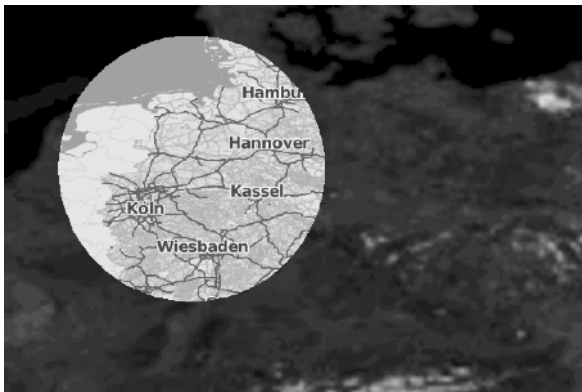


Abb. 2: Absicherung mit Puffer um Punkt



Abb. 3: Absicherung mit Polygon

### Beispielszenario

Abschließend sei hier noch eine Beispielanwendung skizziert, die in der Praxis eingesetzt wird.

Anwender melden sich in einer bestehenden Onlineanwendung über ein typisches HTML-Formular an. Die Autorisierung aller Anfragen in der Anwendung erfolgt über Cookies. Die MapProxy-Absicherungsschnittstelle liest diese Cookies ebenfalls aus und verwendet die daraus bezogenen Benutzerinformationen für die Autorisierung. Für Desktop-GIS Anwendungen, in denen diese Cookie-basierte Autorisierung nicht möglich ist, wird parallel hierzu eine Authentifizierung über HTTPS und Basic-Authentication angeboten.

Für die Autorisierung liegen für jeden Benutzer in der Datenbank Informationen über die Berechtigungen – wie Layernamen und Geometrie – vor, die von der Sicherheitsschicht ausgelesen und von MapProxy anschließend durchgesetzt werden.

### Kontakt zum Autor:

Oliver Tonnhofer  
Omniscale GmbH & Co. KG  
Nadorster Str. 60  
26123 Oldenburg  
tonnhofer@omniscale.de  
<http://omniscale.de/>

### Literatur

[1] MapProxy: <http://mapproxy.org/>

[2] MapProxy API Dokumentation: <http://mapproxy.org/docs/nightly/auth.html>

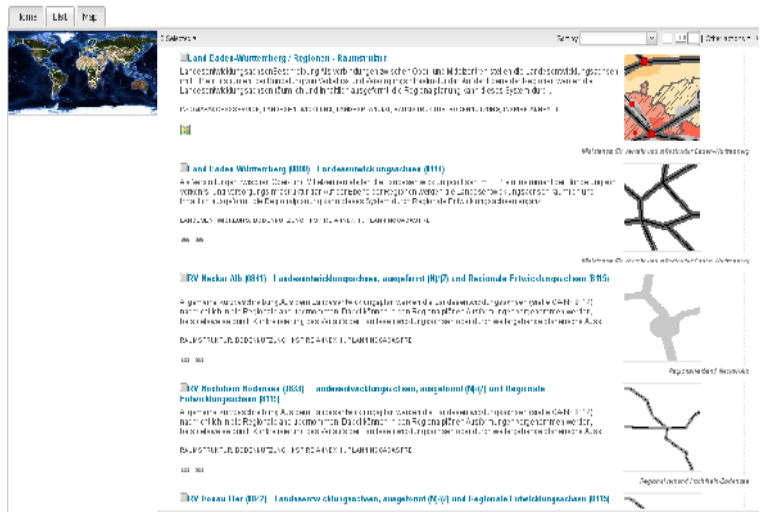
## GeoNetwork 2.8 - Neuerungen

DIPL.-GEOGR. DAVID ARNDT

**GeoNetwork opensource** ist ein interoperables Informations-Management-System, ausgelegt zum Zugriff auf georeferenzierte Daten, kartographische Produkte und die damit im Zusammenhang stehenden Metadaten. Dies ermöglicht Nutzern einen einfachen Zugang zu räumlichen Metadaten und Daten.

Das Hauptziel von **GeoNetwork opensource** ist es, die Zugänglichkeit breit gefächerter Daten, Geodaten und Geodienste, und damit in Zusammenhang stehender Informationen aus unterschiedlichsten Quellen zu organisieren, zu dokumentieren und dienstbasiert bereitzustellen. Dies soll auf einem standardisierten und konsistenten Weg geschehen.

**GeoNetwork opensource** implementiert sowohl eine Portal-Komponente, als auch eine Katalog-Datenbank. Die Software enthält Tools zum Erfassen, Verwalten und Veröffentlichen von räumlichen Metadaten und Services. **GeoNet-**

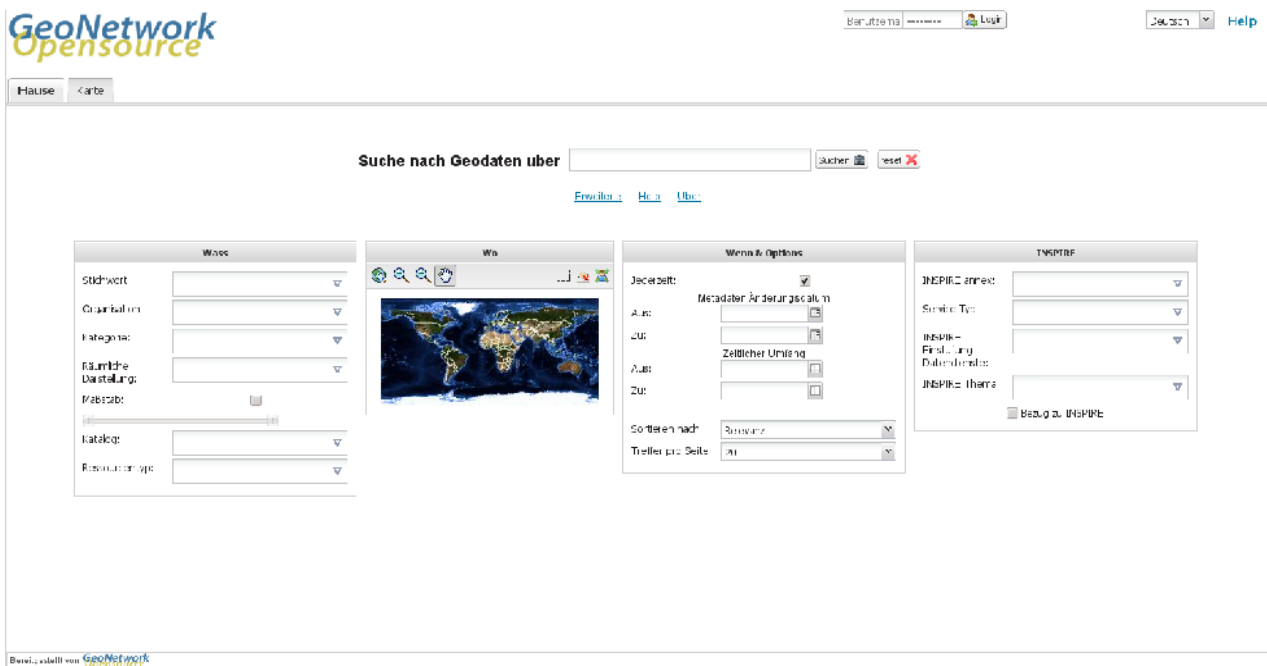


1. Anzeige von Suchergebnissen

**work opensource** erlaubt eine verteilte Suche nach Metadaten im eigenen, sowie eingebundenen externen Metadatenkatalogen verschiedener Anbieter, anhand unterschiedlicher Suchparameter. Zur Anzeige recherchierte Geodienste enthält **GeoNetwork opensource** einen Kartenviewer, der es jedem erlaubt dynamische Karten zu unterschiedlichsten Themen zu erstellen.

- **Portal Services** erlauben, sowohl den Zugriff auf räumliche Informationen, als auch das Verwalten und die Administration des Portals und der Benutzer. Eine Reihe von Authentifizierungs- und Zugriffsregeln regulieren den Zugriff auf die im Katalog enthaltenen Informationen und Services. Das Portal beinhaltet einen Metadateneditor, der in der Lage ist ISO-kompatible Metadateneinträge für geographische Daten nach ISO 19115 zu erstellen.
- **Catalog Services** erlauben das Sammeln, die Registrierung und die Wartung von Metadaten. Der Catalog Service implementiert eine Komponente zum Austausch mit anderen Catalog Services nach dem CSW-Protokoll. Damit kann **GeoNetwork opensource** auf andere Kataloge zugreifen und umgekehrt.
- **Daten Services** ermöglichen den Zugriff auf Inhalte von Quellen und Datenbanken und erlauben damit Datenbearbeitung über definierte Schnittstellen und Interfaces. Die Daten Services können über das Internet verteilt sein.

Der Vortrag fokussiert insbesondere die Neuerungen der Version 2.8 die im Frühjahr 2013 erschienen ist.



## 2. Suchinterface

Dazu zählen unter anderem:

- Neue moderne Benutzeroberfläche
- Verbesserte Performanz und Stabilität
- Metadaten Versionierung
- Subtempates
- Verbesserte Suche und multilinguale Metadatenunterstützung
- Hinzufügen von Metadatenprofilen über die GUI
- Suchstatistiken
- Neuer Harvester
- Metadaten als PDF ausgeben
- Virtuelle CSW-Endpunkte

Der Vortrag fokussiert:

- Geo-Metadaten mit GeoNetwork opensource
- Neuerungen der Version 2.8

Kontakt zum Autor:

Dipl.-Geogr. David Arndt  
geoinformation+planung  
Gölzstraße 22  
72072 Tübingen  
arndt@geoinformation-planung.de

## GeoPackage, das Shapefile der Zukunft

# GeoPackage, das Shapefile der Zukunft

PIRMIN KALBERER

Der GeoPackage-Standard ist im Januar 2013 vom OGC als Draft veröffentlicht worden. Er vereint die Speicherung von Vektor- und Rasterdaten im verbreiteten SQLite DB-Dateiformat. Vektoren werden im Spatialite-Format und Rasterdaten wie MBTiles gespeichert.

Aufbauend auf der etablierten Embedded-Datenbank SQLite, steht seit März 2008 das Geodatenformat „Spatialite“ zur Verfügung. Damit können Vektor-Datensätze als Datenbank-Tabellen in einer einzigen Daten gespeichert und mit vollwertiger SQL-Funktionalität verarbeitet werden.

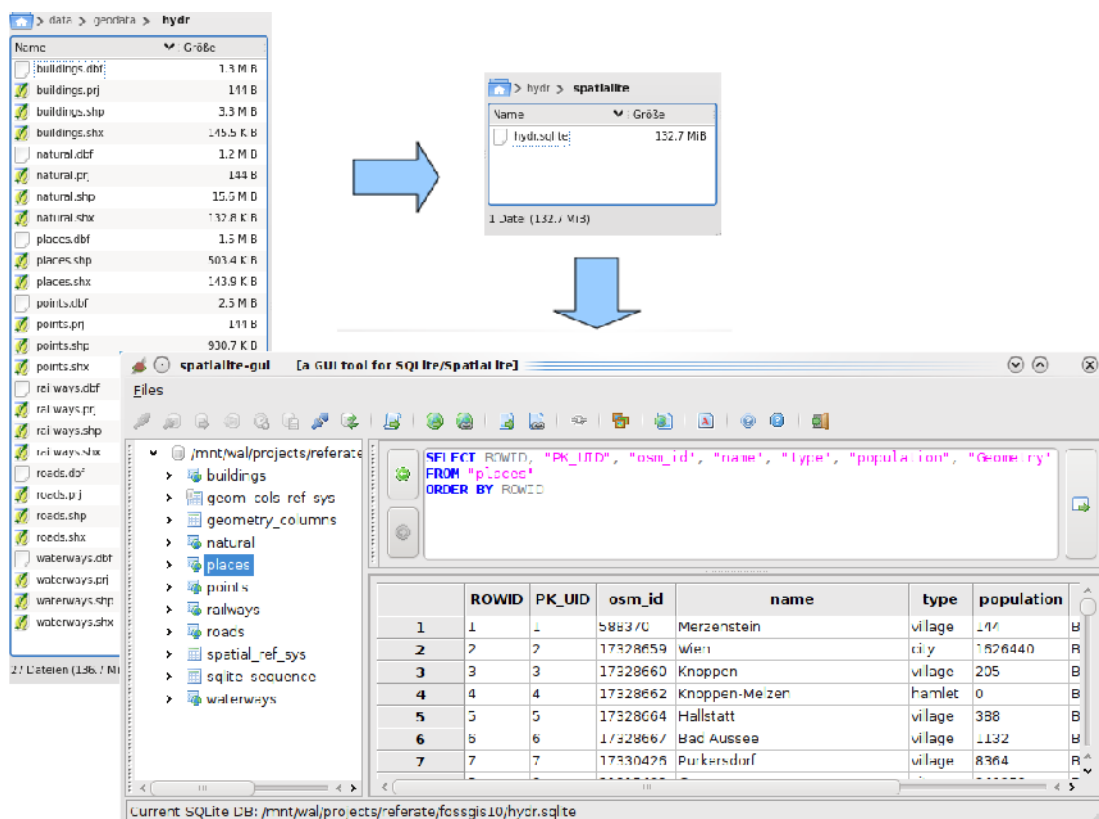


Abbildung 1: Shapefile-Import und Spatialite GUI

Der GeoPackage-Standard kombiniert die Vektordatenspeicherung mit der Speicherung von Raster-Kacheln ausgehend vom weitverbreiteten MBTiles-Format.

Diese Kombination eröffnet die Möglichkeit, Geodaten in einem praktischen Container-Format zu speichern, das auch von mobilen Geräten unterstützt wird und auf das sehr effizient zugegriffen werden kann.

## GeoPackage, das Shapefile der Zukunft

Kontakt zum Autor:

Pirmin Kalberer  
Sourcepole AG  
Weberstrasse 5  
8004 Zürich  
www.sourcepole.ch  
Twitter: @PirminKalberer

Links:

[1] GeoPackage Candidate Standard: <http://www.opengeospatial.org/standards/requests/95>

[2] SQLite: <http://www.sqlite.org/>

[3] SpatiaLite: <http://www.gaia-gis.it/gaia-sins/>

[4] MBTiles: <https://github.com/mapbox/mbtiles-spec>

## Geoprocessing mit OpenSource Desktop-GIS

### gvSIG CE, SEXTANTE und GRASS GIS

RUTH SCHÖNBUCHNER

Mit mehreren hundert Modulen zur Geoprozessierung gehört gvSIG CE zu den leistungsfähigsten Desktop-GIS auf dem Gebiet der Geodatenverarbeitung und -analyse. Diese umfassende Funktionalität ist zum Teil in gvSIG CE selbst implementiert, wird aber zum größten Teil über die Erweiterung SEXTANTE zur Verfügung gestellt, welche ihrerseits wiederum Konnektoren für externe Prozessoren wie GRASS GIS und SAGA GIS zur Verfügung stellt. Diese mehrschichtige Architektur führt einerseits zu enormer Vielseitigkeit und Flexibilität bei der Datenprozessierung, andererseits aber auch zu einer gewissen Überlappung zwischen Modulen, deren einzelne Vor- und Nachteile sich dem Benutzer nicht immer unmittelbar erschließen. Zwar werden die Besonderheiten der Datenhaltung von Systemen wie GRASS GIS so gut wie möglich vor dem Benutzer von gvSIG CE verborgen, doch sind Bruchstellen zwischen den beteiligten Komponenten nicht gänzlich vermeidbar. Der Vortrag widmet sich den unterschiedlichen Schichten der Geodatenprozessierung in gvSIG CE und zeigt auf, welche Komponenten sich für welche Anforderungen eignen, wobei besonderes Augenmerk auf die (semi-)automatisierte Prozessierung großer Datenmengen gerichtet wird. Ein weiterer Schwerpunkt der Ausführungen liegt auf besonderen Anforderungen bei der Qualitätssicherung, z.B. im wissenschaftlichen und unternehmenskritischen Bereich. Außerdem werden die wichtigsten Entwicklungslinien der Geoprozessierung unter gvSIG CE diskutiert und die Ziele, sowohl für die momentan in Vorbereitung befindliche, als auch für zukünftige Versionen von gvSIG CE, skizziert.



## Georeferenzierung alter Karten mit Grid Shift Files anhand der Tranchot-Karte

JAN HARTMANN

Eine neue Methode zur Überlagerung von alten und neuen Karten

Georeferenzierung von alten Karten mittels Grid-Shift Files in NTV2 format. Als Beispiel die Tranchot-Karte der Rheinlande von 1804, das Niederländische Kataster von 1832, und das Urkataster von Aachen

In 2007 hielt ich einen Vortrag an der FOSSGIS-Konferenz in Berlin über die globale Georeferenzierung der Tranchot-Karte:

[http://mapserver.sara.nl/conferences/2007/berlin\\_fossgis/](http://mapserver.sara.nl/conferences/2007/berlin_fossgis/)

In dem neuen Vortrag möchte ich diese Methode weiter verfeinern: wie kann man die Tranchot-Karte auch im Detail genau georeferenzieren, so dass Wege, Häuser, usw., genau über moderne Karten überlagert werden können. Es wird beschrieben wie man Grid Shift files mittels die GDAL Utilities produzieren kann, wie man die Vector Displacements mit PostGIS und QGIS herstellt, und wie man sie visualisiert mit MapServer Vector Field Rendering.

Als Beispiel wird wieder die Tranchot-Karte der Rheinlande verwendet, und ein grosses Projekt wobei innerhalb der nächsten Jahren die 17.000 Niederländische Urkatasterpläne von 1832 und die drie Millionen Hilfskarten von 1832 bis 1994 georeferenziert werden in einem massiven Cloud-Umgebung der Universität von Amsterdam.

## GeoServer

### the open source server for interoperable spatial data handling

SIMONE GIANNECCHINI

GeoServer is an open source geo-spatial server written in Java, following the common Java 2 Enterprise practices, allowing for the handling, distribution and analysis of geospatial data.

GeoServer allows to distribute, handle and analyses data using the most widely accepted OGC standards (WMS, WFS, WCS and WPS), without forgetting specific extensions for a transparent interacting with clients such as Google Earth and commercial software in general, and providing support for the now common protocols based on REST and GeoJSON for the distribution of simple vector based data.

The presentation will give the audience an exhaustive overview of GeoServer 2.2 functionalities for the creation of interoperable Spatial Data Infrastructures, with particular focus on the new functionalities introduced with the upcoming GeoServer 2.3 release as well as on the WPS 1.0 spatial data analysis capabilities.

Closing the presentation we'll see some recently added features and improvements, such as the ability to transform data on the fly while rendering, WFS 2.0 and WFS paging, optimizing PNG outputs, better WMS cascading and virtual services.

## Im- und Export von dxf-Daten mit OGR

HARALD SCHWENK

Eines der gebräuchlichsten Formate für den Austausch von räumlichen Inhalten ist das "drawing interchange format" kurz dxf-Format. Insbesondere wenn für die weitere Verwendung der Geodaten ein CAD vorgesehen ist, ist dxf das Format der Wahl. Der Beitrag behandelt die Möglichkeit, mittels der OGR Simple Feature Library, Geodaten aus einem PostgreSQL/PostGIS Datenbankmanagementsystem in ein dxf-Format zu exportieren.

In einem geringeren Umfang wird auch das gegenläufige Verfahren (Import von dxf-Daten in ein PostgreSQL/PostGIS Datenbankmanagementsystem) behandelt. Der geringere Umfang ist hier weniger einer fehlenden Relevanz geschuldet, sondern auf die Tatsache zurück zu führen, dass dxf-Daten mitunter deutlich überladen sind und in diesen Fällen kein befriedigendes Ergebnis zu erzielen ist. Hinzu kommt, dass dxf-Daten ganz erheblich über das Styling definiert und komponiert werden. Diese Kombination (räumliche Definition und Darstellung) wird in der klassischen Geodatenwelt über den attributiven Anhang abgebildet und die Darstellung erst im Rahmen der Visualisierung an die jeweilige Applikationen abgetreten. Die Strukturierung von Geodaten orientiert sich regelmäßig an den Simple Feature (Punkt, Linie, Fläche). Entsprechend werden die Daten dann modelliert. Das dxf-Format kann diese und viele andere Geometrietypen Verarbeiten. Eine strenge Modellierung der Daten nach den Geometrietypen ist dem dxf-Format allerdings fremd. Eine thematisch gebotene Verknüpfung erfolgt allerdings in beiden Kulturen über die Tabelle (GIS) bzw. den Layer (dxf).

Die Konvertierung dxf>PostgreSQL>dxf wird hier beispielhaft beschrieben. Grundsätzlich ist die OGR-Bibliothek eine Datendrehscheibe in die verschiedensten Formate und die Verfahrensweise ist abhängig vom Ausgangs- und Zielformat sehr ähnlich.

Der Beitrag zeigt die Konfiguration eines Ausgangsdatenbestandes mit einem PostGIS-View. Hierbei wird auf die Besonderheiten des Stylings von Symbolen, Texten und Linien eingegangen. Im Anschluss wird die Steuerung der OGR-Bibliothek dargestellt.

Das hier beschriebene Verfahren wird über Befehlszeilenkommandos gesteuert. Somit richtet sich der Beitrag an erfahrene Anwender und Menschen die Applikationsentwicklungen machen.

## Kollaboration um ein Open Source generisches Mapping System

MICHAEL KLABERMATTEN

Seit zwei Jahren sind mehrere Projekte um Open-Source Mapping in der Westschweiz entstanden. Anstatt einzelne Projekte zu entwickeln, wurde schnell klar, dass diese Kosten und Entwicklungen geteilt werden mussten, wodurch ein generisches und höchst konfigurierbares System gemeinsam entwickelt wurde. Sei es Kantonen, Städte, öffentliche Vereinigungen oder sogar private Unternehmen, die Partner müssen die Fähigkeit haben dieses System an ihre Bedürfnisse zu adaptieren. Seit zwei Jahren haben die verschiedenen Partner zwei Entwicklungszyklen in Auftrag gegeben. Die Server-Seite des Systems wird durch Pyramid (Python Webframework), MapFish, MapFish-print und MapServer angetrieben. Die Konfiguration der Layer und Themen des Geoportals ist in einer PostGIS Datenbank abgelegt und mit einem Administrationsystem konfigurierbar. Auf der Clientseite werden nur Javascript und HTML-Komponenten verwendet. Eine High-Level-Bibliothek wurde entwickelt namens CGXP. Die größte Herausforderung und einmalige Erfahrung ist die Gruppierung der verschiedenen Teilnehmer mit sehr unterschiedlichen Bedürfnissen.

Dieser Vortrag wird erläutern, wie diese User Gruppe zu Stande gekommen ist, wie die Zusammenarbeit innerhalb einer flexiblen Private-Public Partnerschaft organisiert ist, welche Vorteile einer solchen Organisation im Bereich Open Source Software Entwicklung bringen und welche Ergebnisse (WebGIS Projekt) erreicht wurden.

## Kort

## Kort

### Gamified Mobile App für die Verbesserung von OpenStreetMap

STEFAN ODERHOLZ



Da OpenStreetMap ein freies Projekt, welches es jedermann ermöglicht, Kartendaten zu nutzen und zu editieren, ist es nicht ausgeschlossen, dass fehlerhafte bzw. unvollständige Daten eingetragen werden. Es gibt verschiedene Tools, die es sich zum Ziel gesetzt haben, solche Fehler zu finden und aufzubereiten. Die so aufbereiteten Daten mussten bisher anschliessend manuell verglichen und korrigiert werden.

Zur Behebung dieser Fehler ist die cross-platform Web-App Kort entwickelt worden. Diese ist in JavaScript geschrieben und basiert auf dem Sencha Touch 2-Framework. Im Backend kommt eine PostgreSQL-Datenbank zum Einsatz. Die komplette Kommunikation ist mit REST-Schnittstellen realisiert. Die Fehlerdaten werden täglich von KeepRight[1] bezogen.

Mit Kort[2] soll das Verbessern von Karten-Daten auf unterhaltsame Weise ermöglicht werden. Dem Benutzer werden dazu die OpenStreetMap-Fehler auf einer Karte angezeigt. Falls er die Lösung für einen dieser Fehler kennt, kann er einen entsprechenden Lösungsvorschlag abgeben. Weitere Spieler der App überprüfen daraufhin den Vorschlag auf seine Gültigkeit. Um die Benutzer zu motivieren, die App regelmässig zu verwenden, wurden zahlreiche Spiel-Elemente eingesetzt. So erhält ein Benutzer für alle Aktionen Punkte (sogenannte Koins), die dann in einer Rangliste zum Vergleich dargestellt werden. Zudem können Auszeichnungen für besondere Leistungen gewonnen werden. Dieser Ansatz ist bekannt unter dem Begriff Gamification.

[1] <http://www.keepright.at> [2] <http://www.kort.ch>

Links:

<http://www.kort.ch/>

<http://play.kort.ch/>

Mapbender3 - nimmt Fahrt auf!

## Mapbender3 - nimmt Fahrt auf!

### Aufbau von WebGIS Anwendungen mit Mapbender3

ASTRID EMDE



Mapbender3 hat in den letzten Monaten Fahrt aufgenommen. Wir stellen nun die neue Version 3.0 auf der FOSSGIS 2013 vor.

Mapbender3 ist eine Software zur einfachen Erstellung von WebGIS Anwendungen. Basierend auf dem aktuellen Symfony2 Framework wurde eine moderne Webanwendung geschaffen, die durch das Baukastensystem der Bundles von Symfony2 einzeln als auch in andere Anwendungen integriert Verwendung finden kann. Mit der Integration von Doctrine DBAL und ORM als Datenbankabstraktionsebene stehen eine ganze Reihe von Datenbanksystemen zur Datenhaltung zur Verfügung, so z.B. PostgreSQL, MySQL, Oracle und SQLite.

Als moderne Web-Applikation basiert Mapbender3 auf den Javascript-Bibliotheken jQuery, OpenLayers und MapQuery als Brücke zwischen der OpenLayers- und jQuery-Welt.

Zusammen mit einem modernen Verwaltungsbackend für die Kartenanwendungen ist Mapbender3 ein komfortables Werkzeug für die Erstellung und Pflege von Kartenanwendungen.

Der Vortrag stellt zuerst das Frontend und seine Komponenten vor.

Derzeit liegen die Elemente Karte mit Maßstabsleiste, SRS-Auswahl und Navigation sowie Übersichtskarte vor. Außerdem verfügt Mapbender3 über einen Layerbaum, die Infoabfrage sowie Legende. Die Messfunktion wurde für Linien und Flächen implementiert. Ein Druckmodul wurde entwickelt sowie die Positionierung über GPS-Position. Über einen Such-Router können leicht SQL-Suchen definiert und in die Anwendung eingebaut werden.

Eine Übersicht der Elemente findet sich in der Mapbender3-Dokumentation unter:

<http://doc.mapbender3.org/en/bundles/Mapbender/CoreBundle/>

Der zweiten Teil des Vortrags geht auf das Administrations-Backend ein.

Über das browserbasierte Backend kann einfach über ein paar Klicks eine eigene Anwendung erstellt werden. Eigene Anwendungen können mit Diensten und diversen Elementen bestückt werden. Die einzelnen Elemente bieten wiederum elementspezifische Konfigurationsmöglichkeiten.

Das Administrations-Backend beinhaltet außerdem die Möglichkeit, ein Dienste-Repository aufzubauen. Derzeit werden WMS 1.1.1 und 1.3.0 als Datenquelle im Repository unterstützt. Weitere von OpenLayers unterstützte Datenquellen wie WMTS und WFS sollen bald hinzukommen.

Mapbender3 bietet weiterhin eine Benutzer- und Gruppenverwaltung mit der Möglichkeit Rechte zuzuweisen. Hier kommen Symfony2s Access Control Lists (ACLs) zum Einsatz.

Benutzer können auch über LDAP in Mapbender3 übernommen werden. Eine Sicherung von Diensten über OWSPROXY ist ebenfalls vorhanden.

## Mapbender3 - nimmt Fahrt auf!

Kontakt zur Autorin:

Astrid Emde  
WhereGroup GmbH & Co. KG  
Eifelstraße 7  
53119 Bonn  
+49 (0)228 909038-0  
astrid.emdewheregroup.com

Links

- <http://mapbender3.org>
- <http://doc.mapbender3.org>
- <http://demo.mapbender3.org>
- <http://github.com/mapbender/>
- <http://symfony.com/>

## MapFish

### Ein OSGeo WebGIS 2.0 Framework

EMANUEL BELO

MapFish ist ein Framework, welches die Bildung von individuell gestalteten und erweiterbaren Web-GIS Anwendungen vereinfacht. MapFish ist so gestaltet, dass es einfach zu nutzen ist, entweder als selbständige Anwendung oder als Bestandteil einer bestehenden Webseite. Als selbständige Anwendung ist MapFish via einige Parameter konfigurierbar und bietet schnell ein funktionales Web-GIS. MapFish kann auch in bestehenden Webseiten wie CMS oder Informations System orientierte Anwendungen ohne Aufwand integriert werden. Der Client Teil ermöglicht mittels OpenLayers2, ExtJS3 und GeoExt4, weiterführende WEB2.0 Funktionalitäten einzubauen. Der Server-Teil stellt in verschiedene Programmiersprachen flexible Prozesse zur Verfügung (Routing, Suchmaschine, thematische Kartografie, usw). Auf Serverseite sind momentan insbesondere Komponenten in Java (print), Ruby oder in Python verfügbar, betreffend Python ist seit letztem Jahr die Papyrus-Library, welches auf der Pyramid-Library basiert. Diverse Web2.0-Anwendungen wurden bereits auf Basis von Mapfish realisiert, und die Entwicklung wird speziell auch mit dem Aufkommen von Openlayers 3 weitergehen.

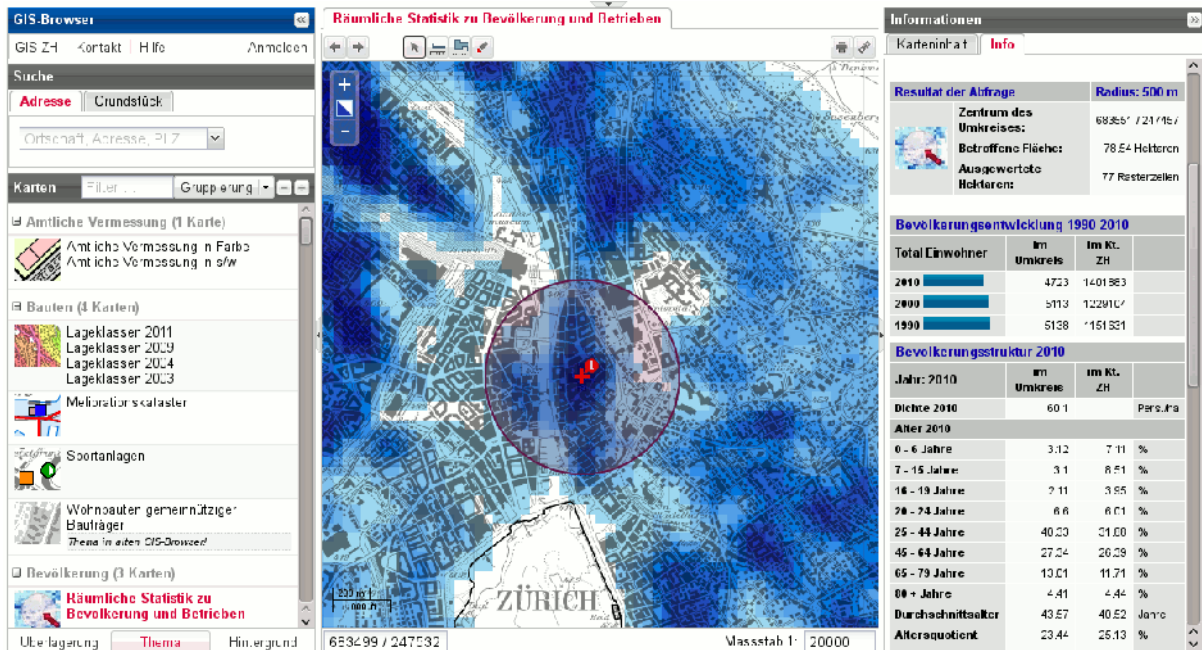
Dieser Vortrag stellt das MapFish Framework vor und zeigt auf in welche Richtung es sich weiter entwickeln wird.



## Mapfish Appserver - Funktionsreiches Web-GIS mit Standard-Protokollen

PIRMIN KALBERER

Der Mapfish Appserver baut auf OGC-Standards für die Kartendarstellung und dem Mapfish REST-Protokoll für die umfangreichen Editiermöglichkeiten.



Es können verschiedene Viewer vom minimalen Mobile- bis zum vollausgebauten GIS-Viewer parallel betrieben werden. Die Karten-Konfiguration erfolgt über das Einlesen von UMN Mapserver Mapfiles oder QGIS Projekten. Es werden automatische Legenden und Abfragetabellen erstellt, welche nach Belieben angepasst werden können. Weitere Eigenschaften sind die Multi-Site Fähigkeit, feingranuläre Zugriffskontrolle, erweiterbare Suchfunktionen und ein eingebautes Administrationsbackend.

Kontakt zum Autor:

Pirmin Kalberer  
Sourcepole AG  
Weberstrasse 5  
8004 Zürich  
www.sourcepole.ch  
Twitter: @PirminKalberer

Links:

- <http://maps.zh.ch>
- [https://github.com/pka/gb\\_mapfish\\_appserver](https://github.com/pka/gb_mapfish_appserver)

## MATSim: Open-Source-Verkehrssimulation mit offenen Daten

MICHAEL ZILSKE

Simulationen sind ein wichtiges Hilfsmittel für die Entscheidungsfindung in der Verkehrssystemplanung. Allerdings sind sie äußerst datenhungrig, und die Beschaffung und Nutzbarmachung proprietärer Daten ist zeitaufwändig und teuer, weswegen die Erforschung alternativer, insbesondere offener Datenquellen wichtig ist.

Daten werden auf zwei Seiten benötigt: Für die Angebotsseite (das Verkehrsnetz) und für die Nachfrageseite (das Nutzerverhalten). Während wir auf der Nachfrageseite noch immer hauptsächlich auf schwer zu beschaffende Haushaltsbefragungen und Verkehrszählungen angewiesen sind, sieht es auf der Angebotsseite besser aus. Zumindest in den urbanen Räumen der Industriestaaten scheint die Datenlage von OpenStreetMap für diese Art von Simulation ausreichend zu sein.

In Sachen offener Daten für den öffentlichen Verkehr ist die aktuelle Veröffentlichung erster Fahrplandaten im GTFS-Format ein Durchbruch. Wo diese verfügbar sind, gibt es in Kombination mit OSM jetzt erstmals die Möglichkeit, ein integriertes Simulationsnetz für motorisierten Individualverkehr und öffentlichen Verkehr quasi auf Knopfdruck an den Start zu bringen, ohne sich für ein neues Szenario erneut um Datenbeschaffung und -konvertierung kümmern zu müssen.

Nach dem Verkehrsverbund Berlin Brandenburg (VBB) haben die SWU Stadtwerke Ulm/Neu-Ulm nun den zweiten deutschen GTFS-Fahrplandatensatz zur Verfügung gestellt.

Das Verkehrssimulationspaket **MATSim** [1] ist eine gemeinsame Entwicklung der Technischen Universität Berlin, der Eidgenössischen Technischen Hochschule Zürich und der senozon AG. Im implementierten Simulationsmodell kommen Straßennetz- und Fahrplandaten an zwei unabhängigen Stellen zum Tragen: In der physikalischen Simulation und in der Planung.

In der **Planungsphase** planen die Nutzer des Verkehrssystems, repräsentiert durch Software-Agenten, ihre Fahrten. Sie haben im Laufe eines Tages eine Kette von Aktivitäten an bestimmten Orten zu absolvieren. Die Zeiteinteilung und die Art, wie sie die Wege zwischen den Aktivitätenorten zurücklegen, können sie planen. Dafür greifen sie auf einen Router zu, der das OSM-Straßennetz und den GTFS-Fahrplan verwendet.

In der **physikalischen Simulation** wird das Verkehrssystem in Interaktion mit seinen Nutzern simuliert. Das heißt: Die Fahrzeuge der öffentlichen Verkehrsbetriebe fahren ihre Routen ab, lassen Fahrgäste ein- und aussteigen, werden dadurch verzögert und verspäten sich, falls das Fahrgastaufkommen das Einhalten des Fahrplans nicht zulässt. Die Fahrgäste wiederum kommen eventuell später als erwartet ans Ziel, weil ihre Busse verspätet sind, verpassen Anschlüsse und erfahren, wie voll ihr Bus ist. Autofahrer teilen sich das kapazitätsbeschränkte Straßennetz und stehen gegebenenfalls miteinander im Stau. Busse, die keine eigene Fahrspur haben, beteiligen sich daran, indem sie sich durch Staus verspäten können, aber auch, indem sie, während sie Fahrgäste ein- und aussteigen lassen, nicht von Autos überholt werden können.

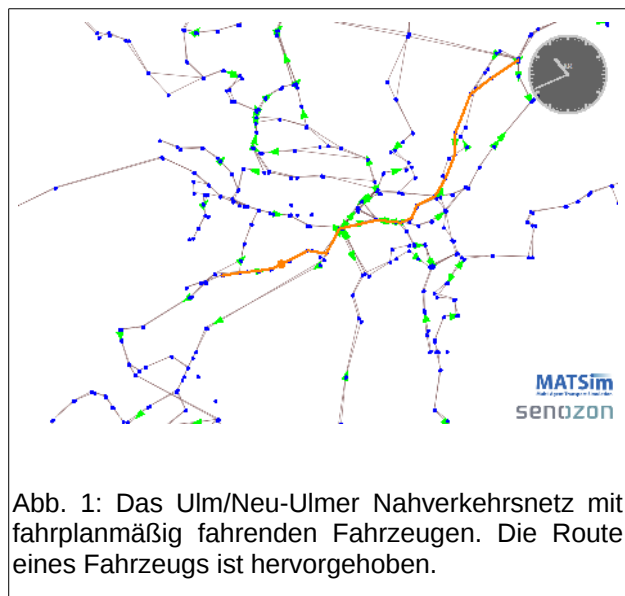


Abb. 1: Das Ulm/Neu-Ulmer Nahverkehrsnetz mit fahrplanmäßig fahrenden Fahrzeugen. Die Route eines Fahrzeugs ist hervorgehoben.

Diese beiden Schritte wechseln sich ab. Nachdem die Agenten geplant haben, führen sie ihre Pläne gemeinsam im Verkehrssystem aus und erfahren dabei, wieviel Zeit sie angesichts der beschränkten Kapazität des Systems und der Aktivitäten aller anderen Verkehrsteilnehmer mit Reisen beschäftigt

## MATSim: Open-Source-Verkehrssimulation mit offenen Daten

sind. Danach fasst jeder Agent den Ablauf des simulierten Tages in einem Punktwert für seinen **Nutzen** zusammen. Im einfachsten Fall mag dies einfach die (negative) mit Reisen verbrachte Zeit sein, aber auch individuelle Nutzenfunktionen, die die Nutzung unterschiedlicher Verkehrsmittel unterschiedlich gewichten, eine hohe Anzahl von Umsteigevorgängen bestrafen und die Zeitstruktur des Tages kontrollieren (um zu verhindern, dass etwa eine Hälfte der Bevölkerung unbeabsichtigt ihre Arbeitszeiten auf die Nacht verlegt, weil sie damit den Berufsverkehr vermeidet), sind möglich. Die Agenten haben ein Plangedächtnis. In der nächsten Iteration der Planungs- und Simulationsschleife wählen sie einen ihrer Pläne aus, und zwar einen vormals höher bewerteten Plan mit einer höheren Wahrscheinlichkeit als einen niedriger bewerteten. In jeder Iteration ist ein gewisser Anteil der Agenten kreativ und probiert eine bisher unversuchte Art, den Tag zu planen.

Dieser **koevolutionäre Prozess** entspannt sich nach einigen hundert Iterationen. Die Verkehrsteilnehmer verbessern dann ihre Punktwerte nicht mehr systematisch. In diesem Zustand ist das Modell dann so weit, mit weiteren Daten (etwa Verkehrszählungen) validiert zu werden und kann dann als Werkzeug verwendet werden, um Fragen zu untersuchen wie: Was passiert, wenn an dieser Stelle eine neue Straße gebaut wird? Wie würde sich eine Maut auswirken? Ein niedriger Fahrpreis für den öffentlichen Verkehr? Eine Lockerung von Arbeitszeitregelungen? Der agentenorientierte Ansatz erlaubt es, die durchschnittlichen Punktwerte von gewissen Agentengruppen (etwa den Einwohnern eines bestimmten Gebiets) vor und nach der Maßnahme zu vergleichen, und so deren Gewinner und Verlierer zu identifizieren.

Um diesen Prozess anzutreiben benötigt es Daten. Die Möglichkeit, Straßennetzmodelle aus OpenStreetMap und, in der Hoffnung, dass der durch VBB und SWU eingeläutete Trend sich fortsetzt, Fahrpläne aus veröffentlichten GTFS-Datensätzen gewinnen zu können, sind ein großer Fortschritt. Zur Validierung und Kalibrierung des Modells sind jedoch Verkehrszählungen (an der Straße und in öffentlichen Verkehrsmitteln) erforderlich. Diese Daten ebenfalls öffentlich zugänglich zu machen, ist sicherlich noch mit einigen organisatorischen Schwierigkeiten verbunden, aber durchaus vorstellbar. Die größte Schwierigkeit ist und bleibt das Nachfragemodell: Wie viele Personen wollen an welchen Orten welche Art von Aktivität ausüben und haben dabei welche Zeit- und Verkehrsmittelpräferenzen? Ein solches Modell aus offenen oder ohne Datenschutzbedenken zu öffnenden Daten zu gewinnen, ist die größte Herausforderung.

Kontakt zum Autor:

Dipl.-Inform. Michael Zilske  
Technische Universität Berlin  
Salzufer 17-19  
10587 Berlin  
michael.zilske@tu-berlin.de

Literatur

[1] <http://www.matsim>

## Metadatenmanagement mit CKAN

### API, OGC CSW Support und Integration in bestehende Systeme.

DANIEL KÜRNER

In diesem Beitrag soll die Verwendung der Daten-Portal-Software CKAN für die Verspeicherung von (Geo-)Daten und korrespondierenden Metadaten eines universitären Forschungsprojekts vorgestellt werden. CKAN gilt als de-facto Standard für Open-Data Portale und kommt auf regionaler (z.B. daten.berlin.de), nationaler (z.B. data.gov.uk oder die Open Government Plattform für Deutschland) und internationaler Ebene (z.B. publicdata.eu) zum Einsatz. CKAN verfügt über eine modulare Plugin-Architektur, was die Anpassung des Systems über eigene Erweiterungen ermöglicht. Die gesamte Logik der Anwendung ist Bestandteil der API und über Remote-Procedure-Calls aus Drittanwendungen aufrufbar. Über die API wird Drittsystemen die Möglichkeit gegeben die Kernfunktionalität von CKAN programmatisch zu nutzen. Darüber hinaus ist lesende Funktionalität über eine REST-API verfügbar. Das Domänenmodell von CKAN besteht aus einem modularen System von Entitäten, das es ermöglicht, eine Vielzahl bestehender Datenstrukturen abzubilden. Die optionale Erweiterung ckanext-spatial ermöglicht die Assoziation von Datensätzen mit Geometrien im GeoJSON-Format und darüber hinaus vollständige CSW Funktionalität durch die Integration des OGC-konformen CSW-Servers pyCSW (py-csw.org).

Die Integration von CKAN in bestehende Systeme soll am Beispiel des interdisziplinären Sonderforschungsbereichs 806 (SFB806) vorgestellt werden. Der SFB806 ist ein, von der Deutschen Forschungsgemeinschaft (DFG) gefördertes, interdisziplinäres Forschungsprojekt an den Universitäten Köln, Bonn und Aachen, welches sich mit der Ausbreitung des modernen Menschen (Homo Sapiens) von Afrika nach Mitteleuropa befasst. Insgesamt sind über 100 Forscher am Projekt beteiligt. Diese produzieren Daten, welche sicher archiviert und der Forschungscommunity zugänglich gemacht werden müssen.

Links:

<http://crc806db.uni-koeln.de/data>

## Mobile Karten erstellen mit OSM, OpenLayers und Overpass Api

ROLAND OLBRICHT

Mit OpenStreetMap stehen nicht nur freie Daten zur Verfügung, sondern wegen der hervorragenden Community decken diese auch ein sehr breites thematisches Spektrum ab, und sie erreichen dabei eine Aktualität, die sich eher in Tagen und Stunden als Monaten und Jahren bemisst.

Mit der Kombination der freien Werkzeuge OpenLayers und der Overpass API lassen sich diese leicht in einer Karte für thematische Overlays visualisieren, die sowohl gleichermaßen auf dem Desktop wie auf mobilen Endgeräten funktioniert, aber trotzdem nicht mehr Infrastruktur als eine einfache HTML-Seite braucht. Dank Overpass API ist sie stets minutenaktuell und auch spontan im HTML-Code oder sogar zur Laufzeit konfigurierbar.

Anhand einer Beispielkarte werden die Möglichkeiten erläutert: Es wird gezeigt, wie durch ein durchdacht einfaches Bedienkonzept und Nutzung der Geolokalisierung eine Smartphone-freundliche Karte entsteht. Wir werden eigene Kategorien von Points of Interest spontan als Kartenoverlay hinzufügen. Und es wird gezeigt, wie durch Kombination von Tag-Verarbeitung auf dem Client und Rückgriff auf die Nominatim-API zu jedem POI automatisch eine Adresse ermittelt werden kann.

## Möglichkeiten und Risiken der unternehmerischen Nutzung von Crowdsourcingprojekten am Beispiel von Open Street Map

FALK ZSCHEILE

### Einleitung

Mit Social Communitys und Crowdsourcing haben wir ein Internetphänomen vor uns, von dem sich Unternehmen wirtschaftliche Vorteile erhoffen. Da die Crowdworker ihre Arbeit freiwillig und unentgeltlich verrichten, stellt sich für sie die Frage, ob es möglich ist, das schier endlos erscheinende Potential von kostenlosen Arbeitsstunden, die sich in Communitys versammeln, für die eigenen Unternehmensziele nutzbar zu machen.

Gerade im Bereich von Geoinformationen kann es verlockend erscheinen, die Datenerfassung und Qualitätssicherung zumindest teilweise durch Crowdsourcing erledigen zu lassen, anstatt die Daten von Drittanbietern teuer einzukaufen. Bevor ein Unternehmen aber beschließt, diese Möglichkeit zu nutzen, sollte es sorgfältig die Vor- und Nachteile analysieren.

Im Folgenden wird auf einige der wichtigen Kriterien am Beispiel von Open Street Map näher eingegangen.

### Crowdsourcing als Phänomen

Aus den Besonderheiten von Crowdsourcing ergibt sich bereits ein erstes Gerüst von Nutzungskriterien.

Mit dem Begriff Crowdsourcing wird das Zusammenwirken vieler Menschen beschrieben, wodurch mit kleinen Beiträgen gemeinsam etwas generiert wird, dass außerhalb der Möglichkeiten des Einzelnen liegt[vgl. 8]. Der Begriff geht auf Howe zurück[8] und wird im Deutschen auch als Schwarmintelligenz oder Weisheit der Vielen übersetzt. Hier sollte man sich davor hüten, diese Begriffe wörtlich zu nehmen. Allein aus dem Zusammentreffen vieler Personen entsteht nicht zwangsläufig etwas Vernünftiges, es bedeutet nicht die Summierung der Intelligenz der Einzelnen[9]. Entscheidend ist, dass es sich um eine Aufgabe handelt, die sich für Zusammenarbeit eignet, keine (zu) hohen Anforderungen an die Kenntnisse des Einzelnen stellt und eine Form der (Selbst-) Steuerung ermöglicht.

Da es viele Aufgaben gibt, die sich nur oder zumindest besser gemeinsam erledigen lassen, dafür gibt es bereits aus der Ära vor Web 2.0 einige und seitdem viele Beispiele. So ist die effektive Gefahrenabwehr durch die Verwaltung nur Dank der Notrufnummer und der Mitwirkungsmöglichkeit durch die Bevölkerung denkbar[3, S. 234]. Aus der digital vernetzten Welt sind die Beispiele ungleich vielfältiger und reichen von intermediären Plattformen über die gemeinsame Entwicklung von freien Lösungen (Software, Wissen, Daten) und öffentlichen Initiativen (Open Government) bis hin zu unternehmenseigenen Plattformen[6, S. 14 .]. Dabei sind die hinter einem Projekt stehenden finanziellen Interessen zunächst kein Kriterium bei der Beurteilung, ob die Aufgabe Crowdsourcingfähig ist. Dies spielt vielmehr erst im Rahmen des Selbstverständnisses der Community eine Rolle. Dabei ist immer zu beachten, dass nicht überall, wo der Begriff Community verwendet wird, dahinter auch eine Personenmehrheit im Sinne von Crowdsourcing steht.

## Eigensinnproblem in unstrukturierten Organisationen

Eingangs wurde bereits geschildert, dass ein wesentlicher Vorteil von Crowdsourcing Projekten darin liegt, dass sie durch kleine Beiträge vieler ein wertvolles (Wirtschafts-) Gut schaffen.

Ein organisatorisches Grundproblem in arbeitsteiligen Organisationen ist, dass die effektive Arbeitsleistung des Einzelnen mit zunehmender Arbeitsteilung abnimmt, weil zunehmend Reibungsverluste durch Abstimmungsprobleme und ähnlichem zwischen den einzelnen Einheiten auftreten. Diese sogenannten Interdependenzen[vgl. 13, S. 4 .] halten sich anders als in klassischen arbeitsteiligen Organisationen in Grenzen. Crowdsourcingprojekte funktionieren nur, wenn es egal ist, von wem eine Aufgabe wahrgenommen wird. Der Gedanke dahinter ist einfach -- hat Mitwirkender A keinen Spaß mehr an der Tätigkeit, so werden über kurz oder lang ein oder mehrere andere Mitwirkende an die Stelle des Mitwirkenden A treten und dessen Tätigkeit in etwa gleichwertig weiterführen. Ausfälle verursachen also keine spürbaren Lücken.

Eine notwendige Voraussetzung für ein Crowdsourcingprojekt, dass freie Lösungen anstrebt, ist daher, dass die individuellen Anforderungen der Aufgabe an die einzelnen Mitwirkenden nicht zu hoch sind. Nur so kann jene kritische Masse zusammen kommen, die für ein Funktionieren unerlässlich ist.

Aber auch die kleinteilige Aufgabenverteilung in Crowdsourcingprojekten macht diese nicht frei von Interdependenzen und damit von Abstimmungsprozessen. So einfach die einzelne Aufgabe auch ist, so muss doch geklärt werden, wie sie zu erledigen ist. Unter den Mitwirkenden muss ein Konsens hergestellt werden, wie eine Aufgabe zu erfüllen ist. Wenn der einer Aufgabe zu Grunde liegende Sachverhalt von verschiedenen Mitwirkenden unterschiedlich wahrgenommen wird, dann verfehlt das Projekt sein Ziel, weil kein nutzbares Ergebnis erzielt werden kann. Bei den deshalb notwendigen Abstimmungsprozessen tritt das so genannte Eigensinnproblem[vgl. 13, S. 9 .] auf: Die einzelnen Mitwirkenden folgen bei der Aufgabenerfüllung einer Eigenlogik oder Eigeninteressen. Den hieraus entstehenden Problemen ist am Beispiel von Open Street Map auf dieser Konferenz ein eigener Beitrag gewidmet: "Open Street Map in Freiheit erstarrt?" An dieser Stelle sei nur kurz angedeutet, das hier erhebliches Konfliktpotential lauert, das die Entwicklung und Vereinheitlichung des Datenschemas bei Open Street Map verzögern kann.

Hier kann ein Unternehmen allenfalls behutsam steuernd einwirken, in dem es mit Hilfe von Software bestimmte Entwicklungen im Datenschema unterstützt und so Abstimmungsprozesse in eine bestimmte Richtung lenkt. Ein Beeinflussung gegen Überzeugungen der Community ist aber in keinem Fall möglich (und auch nicht ratsam). Denkbar wäre auch ein eigenes Datenschema vorzuschlagen, das mit dem vorhandenen nicht kollidiert und sich möglicherweise mit der Zeit als das bessere in der Datenbank durchsetzt.

## Integrationsmechanismen in unstrukturierten Organisationen

Je arbeitsteiliger eine Organisation strukturiert ist, um so notwendiger sind Integrationsmechanismen, die dafür sorgen, dass Eigeninteressen und Eigenlogik der einzelnen Einheiten nicht zum Versagen der Organisation führen.

Einer der stärksten Integrationsmechanismen bei freien Projekten sind gemeinsame Werte und Ziele[10, S. 222]. Diese sorgen für den notwendigen Rahmen der Organisation.

Verbindendes Element bei Open Street Map ist die Schaffung einer frei zugänglichen Datenbank für Geoinformationen, die auch eine kommerzielle Verwertung zulässt. Wie die Lizenzdiskussion beim Wechsel von der CC BY-SA (Namensnennung Weitergabe unter gleichen Bedingungen)[1] zur Open Database Licence (ODbL)[2] zeigt, sind aber auch hier Grenzen vorhanden.

Die geteilten Werte und Vorstellungen spiegeln sich also auch in der Lizenz eines Projektes wider. Die Lizenzfrage ist für ein Unternehmen insbesondere unter dem Gesichtspunkt interessant, welche Nut-

## Mobile Karten erstellen mit OSM, OpenLayers und Overpass Api

zungen und Verwendungen zugelassen sind. Hier muss geprüft werden, ob die Lizenz mit den Unternehmenszielen vereinbar ist. Dabei sollte sich die Überprüfung nicht nur auf den rechtlichen Aspekt beschränken, sondern auch erforscht werden, ob und welche ideellen oder moralische Erwartungen an eine unternehmerische Nutzung geknüpft werden. Auch wenn Daten lizenzkonform genutzt werden, sollte dieser Aspekt nicht ignoriert werden. Die Communitymitglieder müssen das Gefühl haben, dass es sich um ein Geben und Nehmen handelt, nicht um eine Einbahnstraße[vgl. 14, S. 5].

Eine Möglichkeit, sich geteilte Werte und Einstellungen zunutze zu machen, ist die Kombination von Communitys, soweit sie zumindest in Teilen übereinstimmen. Besitzt beispielsweise ein Unternehmen eine eigene Community, möchte die von dieser gepflegte Datenbasis aber vergrößern, so ist die (partielle) Kombination der Daten mit einer anderen Community denkbar. Bei Open Street Map ist eine solche Kooperationen im Bereich der Restaurantdaten[7] sowie bei nautischen Daten[4] unternommen worden. Hier ergibt sich für die beteiligten Projekte eine win-win-Situation. Beide Projekte vergrößern ihre Datenbasis und die Community, die an den Daten arbeitet. Eine möglichst große Community ist ein entscheidendes Kriterium für funktionierendes Crowdsourcing und sorgt für eine gute Pflege der Datenbasis. Dabei kann die Datenbasis durch gegenseitigen Im- bzw. Export getrennt gehalten werden.

## (Selbst-) Steuerung unstrukturierter Organisationen

Der für ein Unternehmen am schwierigsten zu kalkulierende Aspekt ist die Organisation und damit einhergehende Selbststeuerung des Crowdsourcingprojekts.

Eine (Selbst-)Steuerung in unstrukturierten Organisationen ist auf verschiedene Arten möglich, sie kann auf Charisma, Reputation[vgl. 12, S. 85] oder gemeinsamen Werten basieren. Insbesondere bei Open Source Software sind alle diese Steuerungsmechanismen anzutreffen[10, S. 223]. Je breiter jedoch ein Projekt angelegt ist und je kleinteiliger die Beiträge sein können, um so schwerer ist es, Reputation aufzubauen und Charisma in Abstimmungsprozessen zur Geltung zu bringen.

### Reputation und Charisma

Im Rahmen von Open Street Map lässt sich Reputation noch am ehesten auf Programmiererseite mit der Entwicklung von Werkzeugen zur Datenbearbeitung oder Qualitätskontrolle

gewinnen. Diese Art der Steuerung klingt auch in der Bezeichnung "Do-ocracy" oder "Meritokratie" an wer sich einer Aufgabe annimmt, der darf (in einem gewissen Rahmen) auch das Wohin bestimmen[vgl. 12, S. 51 f.]. Für den einzelnen Mapper ist es dagegen gerade im gut erfassten Deutschland relativ schwer, durch das besonders gute Erfassen größerer Gebiete noch Ruhm und Ehre zu gewinnen. Aber zumindest regional bei Stammtischen sind solche Mapper anzutreffen, deren Argumente aufgrund bisheriger Leistungen mehr Gehör finden als andere. Wobei neben der Reputation auch die Fähigkeit und der Wille vorhanden sein muss, in Diskussionen steuernd einzugreifen.

Unter diesen Voraussetzungen sind Steuerungsmöglichkeiten für Unternehmen, die bereits etablierte Crowdsourcingprojekt nutzen wollen, von vornherein begrenzt. Am vielversprechendsten ist es, das Projekt mit Software zu unterstützen, die den Mitwirkenden die eigentliche Arbeit erleichtert. Durch Unterstützung bestimmter vom Unternehmen gewünschter Entwicklungen kann hier Einfluss genommen werden. Dabei muss aber sensibel vorgegangen werden, denn nichts ist kontraproduktiver für ein Unternehmen als ein schlechtes Image innerhalb des Projektes.



## Mobile Karten erstellen mit OSM, OpenLayers und Overpass Api

### Geteilte Werte

Wie bereits dargelegt, besteht bei unstrukturierten Organisationen ein wesentlicher Integrationsfaktor in den geteilten Werten. Daraus können sich auch Rahmenregeln bzw. feste Entscheidungsprozesse und damit ein Steuerungsmechanismus entwickeln.

Trotz gemeinsamer Werte hat sich bei Open Street Map bisher in keinerlei Hinsicht ein klarer Entscheidungsfindungsmechanismus etabliert. Weder ist ein "Wohltuender Diktator" vorhanden, noch existiert eine "gewählte Regierung". Am ehesten lassen sich noch Elemente einer "Meritokratie" erkennen[vgl. zu den einzelnen Organisationsformen in der Open Source Softwareentwicklung 12, S. 49 .]. Insgesamt handelt es sich aber eher um eine basisdemokratische bis anarchische Organisationsstruktur.

In Bezug auf das Tagging-Schema ist der verbreitetste Weg, eine Lösung zu finden, die Diskussion im Forum oder auf Mailinglisten. Die Diskussionsseiten im OSM-Wiki werden nach dem Eindruck des Autors demgegenüber verhältnismäßig wenig genutzt. Auch der Proposalprozess[11], welcher ein Abstimmungsverfahren für ein Tag im Wiki vorsieht, wird von der Mehrheit der Mapper nicht genutzt und genießt auch in Bezug auf das Abstimmungsergebnis keine Verbindlichkeit und wenig Autorität.

Das Wiki eignet sich aber gut, um eigene Vorstellungen über richtiges Tagging zu dokumentieren, dann bleibt jedem allerdings nur die Hoffnung, dass der Schwarm die Idee aufgreift und weiter trägt. Und nicht selten kommt es zu Konflikten, wie und wo etwas zu dokumentieren ist.

So kann es passieren, dass mehrere Tags mit ähnlicher Bedeutung nebeneinander existieren. Eine Entwicklung in eine bestimmte Richtung ergibt sich bestenfalls mit der Zeit. Das Problem lässt sich aber in aller Regel durch Bearbeitung der Daten mittels Software außerhalb der OSM-Datenbank, also lokal beim Datennutzer, in den Griff bekommen. Eine Vereinheitlichung in der OSM-Datenbank selbst mittels Bot, stellt hingegen ein in der Community nicht akzeptiertes Vorgehen dar und würde sofort nach Entdeckung rückgängig gemacht.

### Geteilte Einstellungen

Mit den Wertvorstellungen einher gehen auch die Faktoren, die zur Mitarbeit in einem Projekt motivieren. Typisch für alle freien Projekte ist dabei, dass fast ausschließlich aus der Mitwirkung selbst die Bedürfnisbefriedigung gezogen wird. Man spricht hier von sogenannter intrinsischer Motivation[vgl. 5, S. 24 f.]. Im Detail geht es dabei um Motive wie: Die Tätigkeit selbst bringt Erfüllung, ein selbstgestecktes Ziel wird erreicht aber auch die Befolgung von Normen um deren selbst willen[5, S. 24, 10, S. 224]. Es fehlt also an klassischen Hierarchien genau so, wie meist auch keine ökonomischen Notwendigkeiten bei der Mitarbeit vorhanden sind[12, S. 42 f.].

Für Unternehmen ist die Nutzung von Projekten, die sich zu einem Großteil auf Personen stützen, welche intrinsisch motiviert sind, eine Herausforderung in Bezug auf Steuerung. Denn die freiwilligen und unentgeltlich Mitwirkenden reagieren in der Regel besonders sensibel auf tatsächliche oder scheinbare Einflussnahme von außen. Fühlen sie sich nicht mehr hinreichend im Projekt respektiert oder präsentiert, ziehen sie sich (meist unwiderruflich) zurück.

Unter diesen Rahmenbedingungen verbieten sich all zu offensive Teilnahmen an Diskussionen durch Unternehmen in der Community von selbst. Das Tagging lässt sich aber, wie oben beschrieben, mittelbar durch entsprechende Software beeinflussen. Da aber damit kaum große Veränderungen, sondern allenfalls Verbesserungen durchsetzbar sind, hat man im Wesentlichen nur die Wahl ein Projekt zu nutzen, wie man es vorfindet oder sich nach Lösungen außerhalb von Crowdsourcing umzusehen.

## Kostenfaktor Crowdsourcing

Kommt nach dem bisher Gesagten eine Kooperation oder Nutzung in Betracht, so muss sich das Unternehmen auch mit den sonstigen wirtschaftlichen Kosten auseinandersetzen. Was vom Unterneh-

## Mobile Karten erstellen mit OSM, OpenLayers und Overpass Api

men an Investitionen in die Community erwartet wird, hängt im Wesentlichen von deren Selbstverständnis ab. Allgemein lässt sich aber sagen, je stärker eine Nutzung ist, um so eher wird von der (Open Street Map) Community auch erwartet, dass vom Unternehmen etwas in ideeller oder materieller Art an das Projekt zurück fließt. Das können ganz verschiedene Gesten der Wertschätzung sein, die von der hier schon oft genannten Software, die Organisation von Events, die Bereitstellung von Servern bis hin zu Flyern, Postern oder einfach einer grafisch ansprechend gestalteten Karte unter einer freien Lizenz reichen kann.

Darüber hinaus kann es notwendig sein, Personal für die Entwicklung von Software für die Community, den Kontakt zu dieser oder für Recherchen im Projekt frei zu haben. Bei Open Street Map kann es beispielsweise einen nicht unerheblichen Rechercheaufwand bedeuten, um beim Tagging Bedeutungsunterschiede und Überschneidungen aufzudecken, um diese später bei der Datennutzung berücksichtigen zu können.

Gerade diese Unschärfen im Datenbestand machen es auch notwendig, die Daten vor der Weiterverarbeitung aufzubereiten. Dies kann im Unternehmen selbst geschehen oder muss gegebenenfalls als Dienstleistung eingekauft werden. In jedem Fall verbergen sich hier, je nach Anforderungen an den Datensatz, Kosten, die ein Spezifikum der Datensammlung mittels Crowdsourcing darstellen und so bei amtlichen oder kommerziell angebotenen Daten nicht auftreten. Als Faustformel kann man sagen, dass dieser Aufwand umso größer wird, je weiter man sich vom allgemein genutzten und damit etablierten Daten entfernt.

## Fazit

Im Ergebnis ist festzuhalten, dass die Zusammenarbeit mit Crowdsourcing-Projekten nicht absolut planbar ist. Das Selbstverständnis und die Selbstorganisation machen es notwendig Chancen und Risiken im Einzelfall gegeneinander abzuwägen. Es ist zu beachten, dass Crowdsourcing entgegen dem ersten Anschein nicht zum Nulltarif zu bekommen ist, sondern eigene Kosten verursacht.

Aufgrund der kaum ausgeprägten Entscheidungsstrukturen bei Open Street Map muss man den vorgefundenen Datenbankinhalt akzeptieren wie er ist und einer kritischen Analyse unterziehen, ob er für den Unternehmenszweck geeignet ist. Kleinere Richtungskorrekturen im Datenschema erscheinen über die Schaffung entsprechender Software zum Editieren der Daten möglich, bleiben aber von der Akzeptanz in der Community abhängig. Daneben bedarf der Datenbestand zunächst einer Aufbereitung, bevor er genutzt werden kann.

Kontakt zum Autor:

Falk Zscheile  
August-Bebel-Straße 4, 09224 Gröna  
falk.zscheile@gmail.com

## Literatur

[1] Creative Commons. Attribution-ShareAlike 3.0. url: <http://creativecommons.org/licenses/by-sa/3.0/legalcode> (besucht am 25. 04. 2013).

[2] Open Data Commons. Open Database License (ODbL) 1.0. url: <http://opendatacommons.org/licenses/odbl/1.0/> (besucht am 25. 04. 2013).

[3] Christian Forsterleitner und Thomas Gegenhuber. Lasst die Daten frei! Open Government als kommunale Herausforderung und Chance. In: Freiheit vor Ort. Handbuch kommunale Netzpolitik. Hrsg. von Leonhard Dobusch, Christian Forsterleitner und Manuela Hiesmair. München 2011, S. 233-246.

[4] Freie Tonne. Theorie OSM-Import. url: <http://www.freietonne.de/index.php?site=99&infotyp=1&PHPSESSID=tut8kgsrqk6vf0jh54daah59o7> (besucht am 29. 04. 2013).

## Mobile Karten erstellen mit OSM, OpenLayers und Overpass Api

- [5] Bruno S. Frey und Margit Osterloh, Hrsg. Managing Motivation. Wie Sie die neue Motivationsforschung für Ihr Unternehmen nutzen können. 2. Aufl. Wiesbaden 2002.
- [6] Oliver Gassmann. Crowdsourcing. Innovationsmanagement mit Schwarmintelligenz. München 2010.
- [7] Gonam. Über Gonam Offene Daten. url: <http://gonam.de/main/benefits> (besucht am 29. 04. 2013).
- [8] Jeff Howe. The Rise of Crowdsourcing. 2006. url: <http://www.wired.com/wired/archive/14.06/crowds.html> (besucht am 08. 07. 2012).
- [9] Sascha Lobo. Kollaborationen im Netz: Den Schwarm interessiert nur das Ergebnis . In: Spiegel Online (2. Apr. 2013). url: <http://www.spiegel.de/netzwelt/web/0,1518,891986,00.html> (besucht am 02. 04. 2013).
- [10] Michèle Morner. Das Open-Source-Software-Phänomen organisatorisch betrachtet. In: zfo 71 (2002), S. 219-225.
- [11] Open Street Map. Proposed features. 15. Apr. 2013. url: [http://wiki.openstreetmap.org/wiki/Proposed\\_features](http://wiki.openstreetmap.org/wiki/Proposed_features) (besucht am 29. 04. 2013).
- [12] Michael Prokop. Open Source Projektmanagement. Softwareentwicklung von der Idee zur Marktreife. München 2010.
- [13] Max J. Ringlstetter. Organisation von Unternehmen und Unternehmensverbindungen. Einführung in die Gestaltung der Organisationsstruktur. München 1997.
- [14] Matias Roskos. Community – die Deppen, die für mich arbeiten? In: HMD 281 (2011), S. 4-5.

## Netzwerkanalysen mit GRASS GIS

### Ein praktischer Überblick

OTTO DASSAU

Nahezu jede GIS-Anwendung bietet mittlerweile Möglichkeiten der Netzwerkanalyse auf Basis von Vektordaten.

In GRASS GIS ist das Thema Netzwerkanalyse bereits seit 2006 auf der Agenda und wird seitdem ständig weiterentwickelt. Mittlerweile sind es mehr als 10 Algorithmen, um Basisdaten aufzubereiten und unterschiedlichste Fragestellungen zu lösen. Beispiele können Laufzeit-/Kostenanalysen durch die Berechnung von Isodistanzen in Kanalsystemen sein, Berechnungen von Zuständigkeitsbereiche von Feuerwehren oder Polizeistationen innerhalb einer Region, das Planen/Optimieren von Rundreisen für Handlungsreisende oder touristische Ziele oder die optimale Verknüpfung von Haushalten bei der Neuverlegung von Strom- oder Gasleitungen entlang eines Strassennetzes.

Der Vortrag soll anhand von Beispielen einen Einblick in die Möglichkeiten der Netzwerkanalyse mit GRASS GIS geben. Dabei wird auch darauf eingegangen, worauf bei der Datenaufbereitung geachtet werden sollte und wie man Parameter richtungsabhängig zuweisen kann. Als Datengrundlage werden OpenStreetMap Strassendaten verwendet.

## Neuentwicklungen in Quantum-GIS

MARCO HUGENTOBLER

Das QGIS Projekt hat letzten Sommer die Version 1.8 veröffentlicht. Da QGIS eine äusserst aktive Entwicklergemeinschaft hat, hat sich seit 1.8 bereits viel getan. Dieser Vortrag zeigt die wichtigsten Neuerungen und gibt einen Ausblick auf die zukünftige Version 2.0.

## Offene Standards und frei Software – Zusammenspiel, Entwicklungen, Unterschiede

ATHINA TRAKAS

Ähnlich wie die immer wiederkehrenden Fragen nach „Was ist Freie Software?“, „Was ist Open Source Software?“, „Was ist die OSGeo?“ gibt es immer wieder Fragen wie „Was ist das OGC?“, „Wie entsteht ein Standard?“, „Wie kann ich mich beim OGC einbringen?“. Daher ist der Vortrag als Grundlagenvortrag gedacht, der aber auch Ausblick auf aktuelle Entwicklungen im OGC gibt.

Offene Standards bilden die Grundlage für viele Anwendungen in einer immer komplexer werdenden IT-Welt. Geodaten und daraus resultierende Geoinformationen finden immer mehr Anwendung und werden von immer mehr Nicht-Fachleuten genutzt. Ein standardisierter Zugriff und Interoperabilität zwischen verschiedenen Systemen und Plattformen ist eine wichtige Voraussetzung, um die große Menge Daten und Informationen sinnvoll zu nutzen und miteinander verknüpfen zu können. In der Geo-Branche sind die Standards des Open Geospatial Consortium (OGC) in Bereichen wie 3D-Anwendungen, dem „Internet der Dinge“, aber auch bei „klassische“ Geoportalanwendungen nicht mehr wegzudenken.

Die zentrale Aufgabe des OGC ist die globale Entwicklung, Förderung und Harmonisierung von offenen Standards und Architekturen, die die Integration von raumbezogenen Daten und Diensten für Benutzeranwendungen ermöglichen und entsprechendes Marktpotenzial schaffen.

Ähnlich wie die immer wiederkehrenden Fragen nach „Was ist Freie Software?“, „Was ist Open Source Software?“, „Was ist die OSGeo?“ gibt es immer wieder Fragen wie „Was ist das OGC?“, „Wie entsteht ein Standard?“, „Wie kann ich mich beim OGC einbringen?“.

Daher beleuchtet dieser (als Grundlagenvortrag gedachte) Vortrag u.a. folgende Aspekte:

- Grundlagen des OGC: Was arbeitet das OGC und wie ist sie aufgebaut? Wer sind die Mitglieder? Was sind aktuelle Themen?
- Wie entsteht ein Standard? Vom Interoperability Program zum Standards Program (z.B. 3D Portrayal Interoperability Experiment) [1, 2, 3]
- Welche Rolle spielt Freie Software im OGC (Beispiele: OGC Referenzimplementationen, das OGC Compliance Program)
- Neuigkeiten aus den einzelnen Arbeitsgruppen, wie z.B. welche aktuellen Entwicklungen gibt es im Bereich Linked Data? Welche Arbeitsgruppen wurden in den vergangenen 12 Monaten in Leben gerufen?, Wie ist der Standard der Dinge der OGC Web Services Testbeds (OWS-9 und OWS-10).
- Ausblick auf Aktivitäten / Meetings in 2013/2014

Links:

[1] Interoperability Program: <http://www.opengeospatial.org/ogc/programs/ip>

[2] Standards Program: <http://www.opengeospatial.org/ogc/programs/spec>

[3] 3D Portrayal Interoperability Experiment: <http://www.opengeospatial.org/projects/initiatives/3dpie>

## Open Standards, Open Source und Open Data: Zuviel des Guten?

### Wie die verschiedenen Aspekte von Offenheit zusammenpassen.

ARNULF CHRISTL



Der Vortrag beleuchtet ausnahmsweise mal die Schattenseiten dieser drei Gesellen, denn: Ja, es gibt sie, z.B.

- behindern Standards Innovation,
- zerstört Open Source bewährte Geschäftsmodelle und
- Open Data fördert das Chaos.

Eine konstruktive Herangehensweise zeigt, dass es lediglich gilt diese Schattenseiten auszuleuchten, um das volle Potential expliziter Offenheit ausschöpfen zu können.

### Open Standards

Ein Blick auf das Alter einiger Standards und deren Praktikabilität in der heutigen IT-Landschaft zeigt, wie schwer es ist Innovation einzuführen. Und das ist auch gut so, denn nicht jede Innovation ist auch gleich gut. Oft sind Innovationen auch einfach nur neu und vollkommen überflüssig oder müssen Bewährtes gar nicht zwingend ersetzen, sondern können es ergänzen. Der Beitrag stellt Möglichkeiten vor, Innovation in geregelter Form einzubringen.

### Open Source

Open Source zerstört proprietäre Geschäftsmodelle und ist deshalb ein Verlust für herkömmliche Softwarehersteller die sich weigern ihr Geschäftsmodell an veränderte Bedingungen anzupassen. Open Source führt aber auch zu Innovation, belebt den Markt und fördert den Wettbewerb. Lediglich 10% des Umsatzes in IT Projekten basiert auf proprietären Geschäftsmodellen.

### Open Data

Der Begriff Open Data ist noch nicht klar definiert und bedarf zusätzlicher Erklärungen. Die wachsende Verfügbarkeit von Open Data wirft eine ganze Reihe neuer und alter Fragen auf, die hier vorgestellt werden. Es gilt transparente Antworten auf diese Fragen zu finden um die Daten und den Umgang damit kontinuierlich verbessern zu können.

Innovation fordert kontinuierliche Investition. Das geht besser, wenn alle Akteure (Stakeholder) an den Prozessen teilnehmen können und nicht, wie es bisher oft der Fall war, alles geheim und hinter verschlossenen Türen entschieden wird.

Im Grunde stimmt es also doch: Alles offen, alles gut. Man muss nur wissen wie.

### Links

- [http://arnulf.us/Open\\_Geodata](http://arnulf.us/Open_Geodata)
- [http://arnulf.us/Open\\_Source\\_and\\_Open\\_Standards\\_White\\_Paper](http://arnulf.us/Open_Source_and_Open_Standards_White_Paper)

## Open Street Map in Freiheit erstarrt?

### – Gedanken zur Fortentwicklung durch die Community –

FALK ZSCHEILE

#### 1 Einleitung

Open Street Map ist als auf Crowdsourcing[vgl. 5] basierendes Projekt auf die Mitarbeit vieler Personen angewiesen. Diese Grundkonzeption bringt das Zusammentreffen von sehr unterschiedlichen Persönlichkeiten und Akteuren mit sich. Das Zusammenwirken in einem mehr oder weniger inhomogen zusammengesetzten Projekt ist nicht einfach.

Fast jeder Mapper hat es auf der Mailingliste oder im Forum schon einmal erlebt, dass zu einer Fragestellung unterschiedliche Meinungen zum Vorschein kamen, die einen Thread von mehr als fünfzig Beiträgen zur Folge hatten. Irgendwann zwischen dem hundertsten und hundertfünfzigsten Beitrag ist die Diskussion dann eingeschlafen, ohne dass man das Ausgangsproblem auch nur ansatzweise lösen konnte. Dafür hat man unterwegs aber unzählige Nebenschauplätze besucht und behandelt. Spätestens wenn man merkt, dass bestimmte Schlagwörter immer und immer wieder unendlich lange Threads verursachen, ohne dass sich auch nur annähernd eine Lösung abzeichnet, kann selbst der überzeugteste Mapper Anflüge von Frustration kaum noch unterdrücken.

Und wer hat sich nicht schon darüber geärgert, dass die eigenen Ergänzungen im Wiki, mit denen man sein gesammeltes Wissen systematisiert hat, um es mit anderen zu teilen, von selbsternannten "Tagwächtern" zurückgesetzt wurde, ohne dass es danach bei der Dokumentation des Tags auch nur einen Schritt voran ging?

Ganz ähnlich kann es passieren, wenn man unverhofft taggend in einem Gebiet erwischt wird, in dem ein anderer Mapper seine Duftmarken gesetzt hat und nun eifersüchtig über dieses Gebiet und sein wacht.

Dies alles erweckt bei so manchem Mitwirkenden den Eindruck, dass Open Street Map auf der Stelle tritt und sich in Bezug auf das Taggingschema und Wiki nicht bewegt.

Warum dieser Eindruck entsteht und welche Lösungsstrategien es gibt, soll nachfolgend behandelt werden.

#### 2 Inkrementelle Arbeitsweise von Open Street Map

Ähnlich wie Softwareprojekte, die auf Open Source Basis entwickelt werden, lässt sich das Vorgehen bei Open Street Map sehr gut als inkrementelles oder iteratives Vorgehen beschreiben.

Dabei gibt es aber wesentliche Unterschiede zu Softwareprojekten, aber auch gegenüber Wikipedia.

##### 2.1 Inkrementelles Vorgehen bei der Softwareentwicklung

Eine inkrementelle Entwicklung nach den Vorstellungen aus der Open Source Softwareentwicklung[vgl. 3, S. 314] würde für ein Taggingschema etwa folgendermaßen ablaufen: Aus einer Idee oder einem Bedürfnis heraus wird ein Tag entwickelt, das in die Datenbank Eingang findet und schließlich auch von einer Anwendung, beispielsweise einem Kartenrenderer, ausgewertet und dargestellt wird. Durch die Anwendung des Tags zeigen sich dessen Vor- und Nachteile nicht nur für die eigentlichen Initiatoren des Tags, sondern einer Vielzahl von Beteiligten (Mappern). Es beginnt ein Nachdenken darüber, wie das Tag modifiziert oder verfeinert werden muss, damit es sich zu weiteren Anwendungen, die über den ursprünglichen Zweck hinaus gehen, eignet. Ein so verfeinertes Schema findet dann wiederum Eingang in die Datenbank und wird seinerseits anhand von Anwendungen überprüft und im Bedarfsfall weiterentwickelt. Soweit die Idealvorstellung eines ständig sich zum Besseren hin entwickelnden Taggingschemas aus der Informatik. Dabei kann bei der Softwareentwicklung regel-

## Open Street Map in Freiheit erstarrt?

mäßig unterstellt werden, dass das eigentliche Ziel der Software, also die zu erfüllende Aufgabe bei Beginn der Entwicklung klar ist und die Diskussionen sich im Wesentlichen auf die beste Umsetzung und der geeigneten Mittel hierfür konzentrieren. Selbst dabei kommt es häufig zu Differenzen, wovon die Vielfalt der Distributionen in der Linuxwelt ein beredetes Zeugnis ablegt.

Auch bei Wikipedia ist das Ziel, wofür man arbeitet, klar umrissen. Es geht um die Zusammenstellung enzyklopädischen Wissens.[9] Etwas anders verhält es sich bei Open Street Map. Ziel des Projektes ist es, eine freie Datenbank für geografische Objekte zu schaffen. Dabei bleibt aber bewusst offen, zu welchem Zweck die Daten eingesetzt werden sollen: „OpenStreetMap is a database of mapping information which can be used for many different purposes [...]“.[8] Eine solche ziellose Datensammlung wird von einem Handlungsmodell aus der Politikwissenschaft möglicherweise besser beschrieben als vom eben angerissenen Modell der Softwareentwicklung.

### 2.2 Inkrementelles Handeln in der Politikwissenschaft

Ein nicht so strukturiertes Bild wie in der Softwareentwicklung hinterlässt das inkrementelle (Verwaltungs-) Handeln als Beschreibungsmodell für die Entscheidungsfindung bei den Beteiligten oder Betroffenen bei komplexen Entscheidungsprozessen. Die Vertreter dieses Handlungsmodells sprechen deshalb auch selbstironisch von “the science of muddling through“[vgl. 6]. Ein Blick auf diese Form des Entscheidungsprozesses erscheint lohnenswert, wenn er sich auch erst auf den zweiten Blick erschließt. Wie bereits im Zusammenhang mit der inkrementellen Softwareentwicklung angedeutet, darf man dort von einem relativ konkret umrissenen Ziel, das erreicht werden soll, ausgehen. Anders verhält es sich beim inkrementellen Handlungsvorgang, wie er von Lindblom beschrieben wird. Hier hat die Verwaltung zwar auch ein Ziel vor Augen, dieses ist aber, was die Form seiner Umsetzung angeht, wenig konkret. Es wird davon ausgegangen, dass eine rationale Planung des Ablaufs nicht möglich und auch nicht erstrebenswert sei.[1, S. 167] Stattdessen sei eine Strategie der unkoordinierten kleinen Schritte bei der Entscheidungsfindung sinnvoll.[1, S. 167]

Ähnlich verhält es sich mit Open Street Map. Gemeinsames Ziel aller Mitwirkenden sind freie Geodaten. Was aber genau mit diesen Daten gemacht werden soll, ist nicht von vornherein klar. Und die Intention bei der Erfassung kann völlig verschieden sein. Dem einen genügt es, wenn das Tagging zum Rendern einer Landkarte ausreicht. Das nächste Projekt will routingfähige Karten erstellen und wieder ein anderes Projekt legt besonderen Wert auf eine gute Darstellung in drei Dimensionen. Ähnlich wie im politischen Prozess ist das große Ziel klar, die dahinter stehenden Interessen der Protagonisten sind im Einzelnen unbekannt, zumindest aber verschieden und manchmal sogar konträr. Gleichsam liegt das Fehlen von festen Entscheidungsstrukturen im Wesen von Crowdsourcing als auch von politischen Prozessen.

Dabei findet die Entscheidungsfindung nach dem inkrementellen Modell nicht hierarchisch gesteuert, sondern unkoordiniert statt. Die notwendigen Informationen werden von verschiedenen Seiten beige-steuert und berücksichtigt. Der Prozess findet durch gegenseitige Verhandlung und Anpassung statt, wobei kein Beteiligter den anderen dominieren oder unterdrücken kann.[Zum Ganzen vgl. 1, S. 168]

Die gleichen Voraussetzungen finden sich auch bei Open Street Map. Es existiert in der Community keine echte Hierarchie. Gerade Organisationsmodelle, die auf Reputation oder Autorität des Initiators aufbauen, wie man sie beispielsweise bei Open Source Softwareprojekten finden kann[vgl. 7, S. 223], sind hier nicht anzutreffen.

Bei der Schaffung eines neuen Tags lässt sich keine Reputation für den Einzelnen gewinnen. Auch für ein besonders gelungenes Tag verliert sich der Ursprung recht schnell und fällt nicht auf seinen Urheber zurück, anders als in der Softwareentwicklung für eine besonders gute Problemlösung. Auch Steve Coast als Gründer von Open Street Map hat eher den Stellenwert eines “Gründungsmaskottchens“, auf das man mit einem Lächeln verweist besitzt aber anders als Linus Torvalds bei Linux keinerlei Autorität, was Entscheidungen anbelangt.

Welche Schlussfolgerungen gibt uns die Lehre vom inkrementellen Handeln mit auf den Weg?



## **Open Street Map in Freiheit erstarrt?**

Zunächst orientieren sich die Entscheidungen am status quo und streben jeweils nur kleine Verbesserungen an. Es wird somit immer nur eine schrittweise Problemlösung erreicht, die nicht endgültig sein kann und will, sondern sich nur als Fortschritt in eine vermutlich erfolgversprechende Richtung versteht. Die getroffenen Entscheidungen orientieren sich nicht an übergeordneten Zielen, sondern am aktuellen Missstand.[1, S. 168]

### **2.3 Lösungsstrategien für Open Street Map**

Welche Konsequenzen ergeben sich daraus nun für das Vorgehen bei Open Street Map?

Weil Probleme nur dann angepackt werden, wenn sie akut sind, darf man nicht verzagen, wenn man auf Nachfrage keine oder keine befriedigende Antwort auf ein Problem erhält.

#### **2.3.1 Taggingschema**

Handelt es sich um eine Frage zum (richtigen) Tagging einfach weitermachen und die eigene Lösung möglichst nachvollziehbar dokumentieren. Dabei aber immer im Hinterkopf haben, dass jederzeit eine neue Diskussion mit offenem Ergebnis entstehen kann. Dann keinesfalls auf der eigenen Lösung beharren, sondern darlegen, warum man einen bestimmten Lösungsansatz gewählt hat.

Außerdem sollte man bei späteren Diskussionen möglichst auf Argumentationsmuster, die ins Grundsätzliche gehen, verzichten und das Ziel, nicht die Systematik in den Vordergrund stellen. Die meisten schätzen praktikable Lösungen für den konkreten Fall.

Dabei unbedingt beachten, bei der eigenen Problemlösung ein Schema zu nutzen, das in die Bedeutung bestehender Tags nicht eingreift, um keinen unnötigen Widerstand bei der Problemlösung zu verursachen. Hierfür bieten sich Subtags oder eigene Namensräume an. Damit setzt man im günstigsten Fall einen eigenen inkrementellen Prozess in Gang, der Überschneidungen und Unklarheiten[10, S. 38.] aufdeckt und eine Weiterentwicklung anstößt. Ein weiterer Vorteil möglichst widerspruchsfrei erweiterbarer Taggingschemen ist die Möglichkeit unterschiedlichen Interessens- und Wissenshintergründen gerecht zu werden.

## Open Street Map in Freiheit erstarrt?

### 2.3.2 Featurerequest

Hat das Problem einen Featurerequest im weitesten Sinne zum Gegenstand und ist man selbst nicht in der Lage hierzu etwas beizusteuern, dann bleibt nichts anderes übrig, als sich in Gleichmut zu üben und auf eine Lösung zu warten. Wichtig ist, dass man verdeutlicht, warum man einen Bedarf sieht. Weckt man auch bei anderen Interesse, dann erhöht sich die Wahrscheinlichkeit, dass eine Lösung gefunden wird ungemein. Demgegenüber wirkt eine vorgetragene reine Anspruchshaltung "die Anderen haben das auch, also schafft das jetzt auch für OSM" eher kontraproduktiv. Hier hängt letztlich viel davon ab, ob man den richtigen Ton trifft. Erfolgt keine Reaktion, so bleibt als einzige Chance der Versuch, das Thema nach ein paar Monaten (modifiziert) erneut hervorzuholen.

### 2.3.3 Die Kunst der kleinen Schritte

Für alle bisher geschilderten Fälle gilt zumindest ein Fixpunkt: Die Lösung sollte immer nur in kleinen Schritten vollzogen werden. In den maßgeblichen Kommunikationsmedien (Forum, Mailingliste, Wiki etc.) zählt Kürze und Prägnanz. Die wenigsten nehmen sich oder haben die Zeit, vollständig ausgearbeitete Konzepte nachzuvollziehen. Sie werden sich immer nur Teilaspekte heraussuchen, zu denen man etwas zu sagen hat. Hierbei verliert man sich in einzelnen Details, und das eigentliche Ziel fällt aus dem Fokus.

Bei der Diskussion um das richtige Tagging ist es erfolgversprechender, im Einzelfall Vorschläge zu unterbreiten, sich mit der Darlegung des eigenen Gedankengebäudes (so vorhanden) aber zurückzuhalten. Fremde Vorschläge, von denen man den Eindruck hat, dass sie in eine Richtung führen könnten, sollte man soweit notwendig modifizieren, aber dabei nie unerwähnt lassen, dass man den Ansatz gut findet.

Im Wiki hat man eher Erfolg, wenn man im Abstand von mehreren Monaten kleine Änderungen vornimmt, als eine Neukonzeption der Seite zu wagen. Größere Änderungen stoßen quasi zwangsläufig auf Widerstand, weil in aller Regel mehr Leute dem status quo anhängen als solche, die sich mit Problemen und ihrer Lösung befassen. Die Zeit, ein Problem zu durchdenken, haben die wenigsten, aber die meisten freuen sich, wenn sie eine Lösung entdecken, die ihr eigenes Problem bereits löst. Man muss neue Ansätze also gewissermaßen zwischen das vorhandene einschieben.

Bei Featurerequests fragt man besser nach kleinen Verbesserungen an bestehenden Dingen, anstatt völlig Neues zu fordern.

## 3 Open Street Map als unstrukturierte Organisation

Open Street Map ist nicht nur eine Organisation, die (fast) keine Hierarchie aufweist, sie ist darüber hinaus auch noch wesentlich heterogener aufgebaut als eine Community, die sich mit der Entwicklung von Open Source Software beschäftigt. Für die Softwareentwicklung sind zumindest Programmierkenntnisse nötig, die zwangsläufig für einen gemeinsamen Hintergrund bei den Entwicklern bei notwendigen Problemlösungen sorgen. Bei Open Street Map fehlt es dagegen an einem solchen verbindenden Element. Das Datenschema ist bewusst so gewählt, dass Einstiegshürden möglichst gering bleiben und jeder der gewisse Grundkenntnisse für Computer mitbringt auch mitarbeiten kann. Entsprechend treffen bei Open Street Map ein Vielzahl von Menschen mit den unterschiedlichsten persönlichen, sozialen und beruflichen Hintergründen aufeinander.

### 3.1 Das "Diversityproblem"

Die Organisationssoziologie beschäftigt sich unter dem Stichwort Diversity Management mit den Folgerungen aus diesen Erkenntnissen. Dabei spielen die direkt wahrnehmbaren Aspekte (surface-level-diversity)[vgl. 4, S. 990], wie Alter, Geschlecht, Hautfarbe oder Nationalität bei einem Internetprojekt wie Open Street Map eine eher untergeordnete Rolle. Ein Ausnahme bildet hier allenfalls die Sprache. Demgegenüber bereiten die nur mittelbar wahrnehmbaren Aspekte (deep-level-diversity)[vgl. 4, S. 990] wie Erfahrungen und Persönlichkeit die weit größeren Probleme. Unproblematisch sind bei Open

## Open Street Map in Freiheit erstarrt?

Street Map demgegenüber die inneren Werte. Hier findet sich vielmehr das verbindende Element der Mitwirkenden die Freude geografische Daten zu sammeln, um sie der Allgemeinheit zur Nutzung zur Verfügung zu stellen.

Diese Heterogenität ist zugleich Segen und Fluch[vgl. 4, S. 988] von Open Street Map: Einerseits sorgt die Heterogenität für eine unglaubliche Akkumulation von Wissen aus den unterschiedlichsten Bereichen und ermöglicht dem Projekt hierdurch, für verschiedenste Fragestellungen eine Lösung zu finden. Andererseits kann es aufgrund der Heterogenität geschehen, dass sich die einzelnen Akteure bei Lösungsversuchen schlicht nicht verstehen, weil sich jeder wegen eines anderen persönlichen, sozialen oder Bildungshintergrunds unter dem scheinbar gleichen Sachverhalts etwas anderes versteht oder vorstellt.

Bei Open Street Map fehlen die Voraussetzungen, um ein echtes Diversity Management zu betreiben. Deshalb soll an dieser Stelle nur auf einfache Mittel der Kommunikation hingewiesen werden, mit denen die größten Missverständnisse im Umgang miteinander ausgeräumt werden können. Oft sind es die im Hintergrund verborgenen Auffassungen, die eine schnelle Lösung verhindern und zu quälend langen und dennoch unergiebig Threads führen.

### 3.2 Problemvermeidung durch Kommunikation

So bietet es sich an, vergewissernd zu kommunizieren daher vor der Kundgabe der eigenen Meinung den Inhalt des Vorredners kurz zusammenzufassen. So kann leicht aufgedeckt werden, wenn etwas falsch verstanden wurde. Eine andere Möglichkeit ist bei (vermuteten) Unklarheiten noch einmal gezielt nachzufragen, damit die Hintergedanken des Kommunikationspartners klarer hervortreten. Immer sollte versucht werden, nicht nur unmittelbar den Ausführungen des Gegenübers argumentativ entgegenzutreten, sondern man sollte auch analysieren, welcher Leitgedanke hinter den einzelnen Argumenten steht.

Ein Musterbeispiel für die fehlende Analyse von Zielen der einzelnen Kommunikationspartner ist das Dauerthema Rad-/Fußwege in ihrer unterschiedlichen Ausprägung. Für den normorientierten Juristen ist ein Radweg primär das, was die StVO und ein Verkehrsschild dazu macht. Dem Vielfahrer in der Stadt ist vor allem die Information wichtig, wo er schnell und sicher voran kommt. Dem Tourenfahrer auf dem Land ist es besonders wichtig, dass er Radwege mit einer vernünftigen Oberfläche vorfindet. Dem GPS-Liebhaber kommt es hingegen aufs gute Routing an, sein Augenmerk liegt dann auf der baulichen Trennung. Machen sich die Akteure hier nicht ihre unterschiedlichen Ziele und Auffassungen klar und gehen außerdem davon aus, dass es nur eine richtige Lösung geben kann, ist ein Endlosthread garantiert, weil man sich meist lediglich darauf beschränkt, dem Gegenüber zu zeigen, dass sein Argument falsch ist und das eigene richtig, anstatt zu versuchen, die unterschiedlichen Anforderungen herauszuarbeiten und dann mit dem Taggingsschema abzubilden.

## 4 Leben mit der Unvorhersehbarkeit von Entwicklungen

Trotz verbessertem Verständnis für die Abläufe bleiben Entwicklungen bei Open Street Map weitestgehend unkalkulierbar. Welche Themen aufgegriffen werden und welche weiter ein Nischendasein führen, ist kaum vorherzusehen.

Auch dieses Phänomen wurde (in anderem Zusammenhang) als Modell in der Politikwissenschaft beschrieben. Damit lässt sich das Wahrgenommene zumindest erklären: die sogenannte Cabbage-Can-Theorie.[Vgl. 1, S. 177 .] Sie besagt, dass komplexe Entscheidungen aus vier unterschiedlichen Elementen bestehen[2, S. 333]: 1. Lösungen, die nach Problemen suchen, auf die sie angewendet werden können, 2. Teilnehmer, die versuchen, in Entscheidungsprozessen eine relevante Rolle zu spielen, 3. Situationen, die es erlauben, Entscheidungen zu treffen, und schließlich 4. Probleme, die unab-

## Open Street Map in Freiheit erstarrt?

hängig von vorhandenen Lösungen, Aktivisten und Gelegenheiten darauf warten, bearbeitet zu werden.

Schaut man sich insbesondere den ersten, zweiten und den letzten Punkt an, so kann man diese leicht auch bei Open Street Map wiederentdecken. Zu denken ist nur an manche User, die in Form von Beiträgen im Forum oder auf der Mailingliste zu fast jedem Thema etwas zu sagen haben, oder die vermeintliche Vielzahl von Unklarheiten im Taggingschema und bei der Umsetzung von Anwendungen.

Dies wird von vielen Mappern oft als unbefriedigender Stillstand empfunden. Je abseitiger das Thema und je größer der ins Blickfeld genommene Kartenabschnitt, um so größer die Wahrscheinlichkeit, dass man über kurz oder lang enttäuscht wird. Hier kann nur geraten werden, sich über die eigene Leistung zu freuen, auf die schon jetzt möglichen Anwendungen stolz zu sein und anderen anzubieten, sie bei ihrer Arbeit zu unterstützen. Demgegenüber sollte man sich vor Anspruchshaltungen bezüglich Qualität und Fortschritt hüten, sonst sind Enttäuschungen vorprogrammiert, gegen die kein Kraut gewachsen ist, weil sie in der Funktionsweise von Open Street Map selbst liegen.

Kontakt zum Autor:

Falk Zscheile  
August-Bebel-Straße 4  
09224 Grüna  
falk.zscheile@gmail.com

## Literatur

- [1] Jörg Bogumil und Werner Jann. Verwaltung und Verwaltungswissenschaft in Deutschland. Einführung in die Verwaltungswissenschaft. 2. Aufl. Wiesbaden 2009.
- [2] Michael D. Cohen, James G. March und Johann P. Olsen. Ein Papierkorb-Modell für organisatorisches Wahlverhalten. In: Entscheidung und Organisation. Kritische und konstruktive Beiträge, Entwicklungen und Perspektiven. Hrsg. von James March und Thomas von Lingen. Wiesbaden 1990, S. 329--372.
- [3] Christian Horn, Peter Forbrig und Immo O. Kerner, Hrsg. Lehr- und Übungsbuch der Informatik. Grundlagen und Überblick. 3. Au. Bd. 1. Leipzig 2003.
- [4] Sujin K. Horwitz und Irwin B. Horwitz. The Effect of Team Diversity on Team Outcomes: A Meta-Analytic Review of Team Demography. In: Journal of Management 33 (2007), S. 987--1015.
- [5] Jeff Howe. The Rise of Crowdsourcing. 2006. url:  
<http://www.wired.com/wired/archive/14.06/crowds.html> (besucht am 08. 07. 2012).
- [6] Charles E. Lindblom. The Science of Muddling-Through. In: Public Administration Review 19 (1959), S. 79--88.
- [7] Michèle Morner. Das Open-Source-Software-Phänomen organisatorisch betrachtet. In: zfo 71 (2002), S. 219--225.
- [8] Open Street Map. Using OpenStreetMap. 13. Apr. 2013. url:  
[http://wiki.openstreetmap.org/wiki/Using\\_OpenStreetMap](http://wiki.openstreetmap.org/wiki/Using_OpenStreetMap) (besucht am 20. 04. 2013).
- [9] Wikipedia. Was ist die Wikipedia? 12. Apr. 2013. url:  
[http://de.wikipedia.org/wiki/Wikipedia:Über\\_Wikipedia](http://de.wikipedia.org/wiki/Wikipedia:Über_Wikipedia) (besucht am 20. 04. 2013).
- [10] Falk Zscheile. Open Street Map ein offenes Datenschema und seine Probleme. In: GeoForum MV 2010 Vernetzte Geodaten: vom Sensor zum Web. Tagungsband zum 6. GeoForum MV. Hrsg. von Ralf Bill u. a. Berlin 2010, S. 37--40.

## OpenStreetMap- und ALK-Daten gemeinsam nutzen – Eine Karte für Mecklenburg-Vorpommern

DOMINIK HELLE, OMNISCALE GMBH & Co. KG

### Einleitung

Daten aus der Verwaltung und Daten aus dem OpenStreetMap-Projekt in einer gemeinsamen Karte darzustellen, das war das Ziel in dem Projekt „Stadtplandienst für Mecklenburg-Vorpommern“. Es sollte das Beste aus beiden Datenquellen in einer gemeinsamen Karte vereint und diese dem Bürger zur Verfügung gestellt werden.

Neben den unterschiedlichen Datenquellen lag ein weiteres Augenmerk auf der harmonischen Darstellung und der performanten Bereitstellung der Karte. Zudem wurde bei dem System darauf geachtet, dass die Karte mit wenig Aufwand möglichst aktuell gehalten werden kann.

Vereinfacht wurde die Durchführung des Projekts durch die neue OpenStreetMap-Lizenz. So konnten die beiden unterschiedlichen Datenquellen in einer gemeinsamen Karte zusammengefasst werden, ohne dass die Daten der Verwaltung einem Lizenzwechsel unterzogen werden mussten.

Im Projekt kam ein Open-Source-Software Stack zum Einsatz: Die Datenhaltung erfolgte in PostGIS. Für den Import der OpenStreetMap-Daten wurde Imposm verwendet. Das Zeichnen der Karte wurde mit dem Kartenserver MapServer durchgeführt. Anschließend wurde die Karte als Kacheldienst und Web Map Service (WMS) mittels MapProxy zur Verfügung gestellt.

### Datenquellen

Bevor die Daten aus Verwaltung und OpenStreetMap gemeinsam in einer Karte dargestellt werden konnten, mussten diese jeweils für den aktuellen Anwendungszweck aufbereitet und vereinheitlicht werden. Als Basis wurde hierfür eine PostGIS-Datenbank gewählt in dem die Daten eingespielt und in das Koordinatensystem UTM32 transformiert wurden.

Die verwendeten ALK-Daten lagen als Shapedatei vor. Aus ihnen wurden die Gebäude mit Hausnummern, die Flächennutzung sowie die administrativen Grenzen übernommen. Neben dem Import in die Datenbank wurden die ALK-Daten für den aktuellen Anwendungszweck optimiert.

Die Optimierung erfolgte vor allem im Hinblick auf die performante Bereitstellung der Karten. Hierfür wurde die Anzahl der Flächen reduziert. Flächen mit der selben Nutzungsart – in dem Stadtplan einheitlich dargestellt – wurden zu gemeinsamen Flächen zusammengefasst. Die Anzahl der Objekte, die gezeichnet werden musste, konnte somit deutlich verringert werden (siehe Abb. 1).

Die Aktualisierung der ALK-Daten wurde so konzipiert, dass lediglich ein Austausch der Shapedateien vorgenommen werden muss. Durch ein automatisiertes Import-Skript werden die Daten eingespielt und die oben angesprochenen Optimierungen durchgeführt. Die Aktualisierung der Daten kann im Produktiv-Betrieb durchgeführt werden.

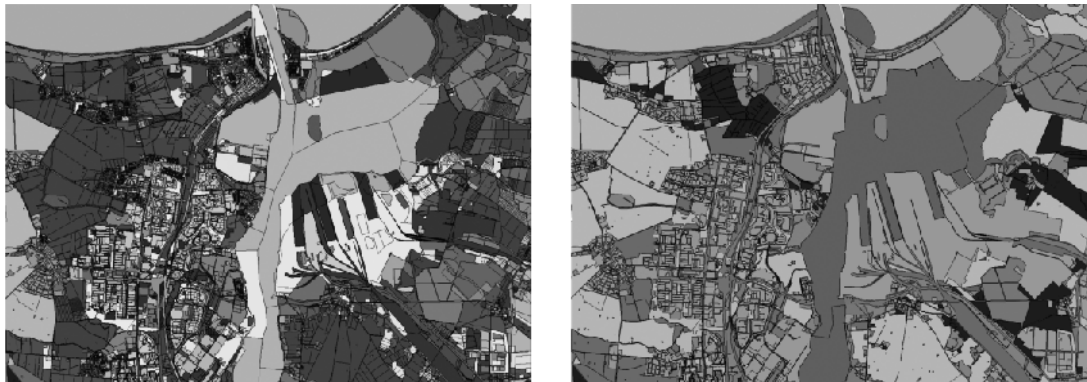


Abb. 1: Zusammenfassen der einzelnen Flurstücke

Für die Aufbereitung der OpenStreetMap-Daten wurde die Open-Source-Software Imposm verwendet. Imposm ermöglicht es, dass nur die OpenStreetMap-Daten in die Datenbank importiert werden, die für den Anwendungszweck benötigt werden. Aus dem OpenStreetMap-Projekt wurden sämtliche Straßen-daten (vom Pfad bis zur Autobahn), alle Flächen- und Gebäudebeschriftungen und jegliche Art von Wasserflächen (vom Graben bis zum See) übernommen.

Im Zuge der Umsetzung des Stadtplandienstes wurde Imposm auch um ein Polygon-Clipping erweitert. Diese Funktion ermöglicht es, dass nur die Daten in die Datenbank importiert werden, die in einem definierten Bereich liegen: in diesem Fall Mecklenburg-Vorpommern. Alle Geometrien die außerhalb des definierten Bereiches liegen werden abgeschnitten oder gelöscht.

Neben der umfangreichen Importfunktion erstellt Imposm ein eigenes Datenbankschema und unterschiedliche Tabellen für die verschiedenen Datentypen z.B. für Straßen oder Wasserflächen. Dies ermöglicht einen schnelleren Zugriff und die Karte kann einfacher gestaltet und anschließend schneller gezeichnet werden.

Ein weiterer Vorteil von Imposm ist das automatische Erstellen von generalisierten Daten. So werden zum Beispiel Straßen mit Imposm für kleinere Zoomstufen vereinfacht, damit große Datenmengen in kleinen Maßstäben schneller abgerufen werden können.

Für die Aktualisierung der OpenStreetMap-Daten wurde die Open-Source-Software Osmosis in Verbindung mit OSM-Diff-Dateien verwendet. In den OSM-Diff-Dateien sind die jeweiligen Änderungen aus dem abgefragten Zeitraum gespeichert. Osmosis aktualisiert nun mittels OSM-Diff-Dateien den einmalig erstellen OpenStreetMap-Ausschnitt und hält diesen laufend auf dem neusten Stand. Anschließend wird der Ausschnitt erneut mit Imposm in die Datenbank eingespielt.

Neben den OpenStreetMap Rohdaten werden auch die zwischengespeicherten Karten der Anwendung aktualisiert. Bereiche, in denen sich Änderungen ergeben haben, werden vom System als GeoJSON-Datei gespeichert und an MapProxy übergeben. Die MapProxy-Seeding Schnittstelle generiert anhand dieser Datei die entsprechenden Bereiche der Karte neu.

Änderungen an den OpenStreetMap-Daten – die auch von den Behörden durchgeführt werden – sind somit innerhalb von 24 Stunden in der Karte zu sehen.

### Kartendienst

Neben den Datenquellen lag ein weiteres Augenmerk auf der Darstellung der Karten. Die Karte wurde an einen vorgegebenen Farbkatalog angepasst. Zudem wurde das Layout so gewählt, dass möglichst viele Straßennamen dargestellt werden. Hierfür wurden weitere Optimierungen, wie zum Beispiel die Abkürzung von Straße zu Str., durchgeführt.

## OpenStreetMap- und ALK-Daten gemeinsam nutzen – Eine Karte für Mecklenburg-Vorpommern

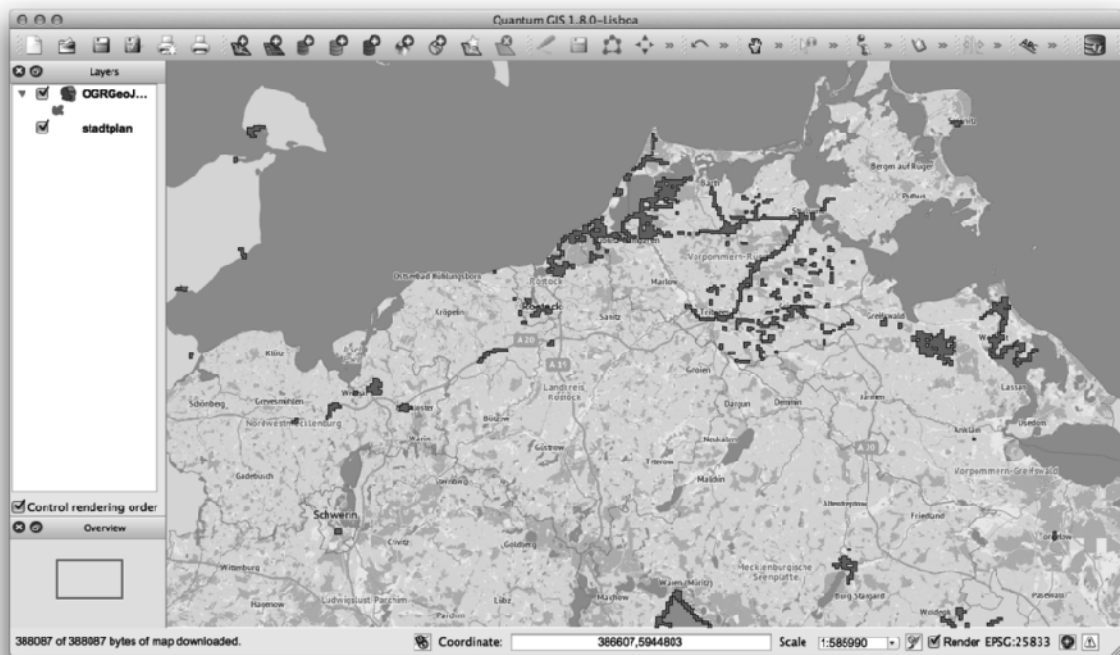


Abb. 2: Bereiche zum Neugenerieren der Karte von MapProxy

Mittels MapProxy wird die Karte in 20 verschiedenen Maßstäben bereitgestellt. MapProxy ist ein WMS- & Kachelproxy, mit dem Kartendienste durch Vorgenerieren und Zwischenspeichern beschleunigt werden können. MapProxy bietet den Dienst als WMS für Desktop-GIS und als Kacheldienst für Web-GIS an. Durch die Verwendung von MapProxy konnte das Ziel der hohen Geschwindigkeit erreicht werden, welches heute vom Benutzer erwartet wird. Zudem ist MapProxy für das Aktualisieren der Karten zuständig.

Als Endprodukt steht die Karte allen interessierten Bürgern unter der CC-BY-SA 3.0 Lizenz zur Verfügung.

Kontakt zum Autor:

Dominik Helle  
Omniscale GmbH & Co. KG  
Nadorster Straße 60  
26123 Oldenburg  
helle@omniscale.de  
<http://omniscale.de>

Literatur

- [1] MapProxy: <http://mapproxy.org>
- [2] Imposm: <http://imposm.org>
- [3] MapServer: <http://mapserver.org>
- [4] Stadtplandienst-Demo: <https://geo.sv.rostock.de/stadtplan.html>

## OpenTopoMap

### topographische Karten aus OpenStreetMap

PHILIPP HOCHREUTHER



#### OpenTopoMap

Das Projekt OpenTopoMap stellt eine Karte aus den Daten der OpenStreetMap zur Verfügung, welche sich am Stil der amtlichen topographischen Karten der deutschen Landesvermessung orientiert. Eine Verfügbarkeit für ganz Europa ist angestrebt. OSM-Daten werden gefiltert und in eine PostgreSQL-Datenbank eingespeist. Mit Mapnik 2.1 werden drei Ebenen einzeln gerendert und mittels ImageMagick zu einer Kachel vereinigt – der zweite Layer, die Höhenlinien, werden dabei einmalig vorgerechnet, wodurch der Renderdurchsatz verzehnfacht wird.

Ziel des Projektes ist es, eine Karte hervorzubringen, welche wie das amtliche Kartenmaterial für universelle Zwecke eingesetzt werden kann, beispielsweise sowohl zur Orientierung im Innenstadtbereich wie auch als Wanderkarte. Hierzu wird zwischen Detailreichtum und Lesbarkeit für jede Zoomstufe in aufwändiger Kleinarbeit die passendste Lösung gesucht und umgesetzt. Zugleich sind erste Ansätze zur Generalisierung enthalten. OpenTopoMap arbeitet eng mit der Universität Erlangen-Nürnberg zusammen: ein Großrechner ist am Rechenzentrum der Universität angesiedelt und steht auch für mit dem Projekt verknüpften Forschungsarbeiten zur Verfügung, welche sich aus der Kooperation zwischen den Instituten für Geographie und Informatik ergeben. Das Projekt ist über Github veröffentlicht, und die freiwilligen Entwickler freuen sich weiterhin auf Verbesserungsvorschläge anderer Nutzer. Im Vortrag wird das Projekt selbst vorgestellt sowie auf die technische Umsetzung kartographischer Details eingegangen.

#### Links

<http://opentopomap.org>



## Ort-Suche mit PostgreSQL/PostGIS

STEPHAN WAGNER

Ein Ort lässt sich über dessen Lagekoordinaten und den Ortsnamen beschreiben und speichern. Doch wie lässt sich der Ort in einem Haufen anderer Orte gezielt und effizient auffinden? Aktuelle PostgreSQL/PostGIS - Installationen bieten leistungsfähige Werkzeuge zur räumlichen wie textbasierten, unscharfen Ort-Suche.

Es wird in einem ersten Schritt auf die Analogie zwischen koordinatenbasierter, geographischer Suche und der Textsuche wie Ortsnamen eingegangen. Der Einsatz eines Suchradiuses mithilfe des kNN-GiST Index wird erläutert und an einem Beispiel die Leistungsfähigkeit demonstriert. Die Methoden und Werkzeuge zur unscharfen Textsuche wie Regex, Soundex, Metaphone/Demetaphone, fuzzystrmatch/Levenshtein, Trigramm, FTS werden anhand von Fallbeispielen erläutert und Stärken und Schwächen der verschiedenen Vorgehensweisen gegeneinander abgewogen. Es wird auf die Problematik mehrsprachiger Ortsnamen eingegangen und gezeigt, wie man auch mehrsprachige Suchanfragen mit der in PostgreSQL standardmässig enthaltenen Volltextsuche (FTS) treffsicher beantworten kann. Zum Abschluss wird anhand einer massgeschneiderten Webapplikation auf Basis von GeoDjango illustriert, wie die Möglichkeiten der Ort-Suche mit PostgreSQL/PostGIS dem Endanwender zur Verfügung gestellt werden können.

OSGeo-Live rocks!

## OSGeo-Live rocks!

### Einblick in ein erfolgreiches Community-Projekt

ASTRID EMDE

Das Projekt OSGeo-Live gewährt einen umfassenden Überblick über freie und offene GIS-Software und GeoDaten

OSGeo-Live ist ein OpenSource-Projekt, das mehr als 50 Softwareprojekte aus dem Bereich FOSS+GIS bündelt. Sie finden Projekte aus den Bereichen Web Mapping Clients und Server, DesktopGIS, Datenbanken, Krisenmanagement, Navigation und Karten sowie räumliche Tools.

Die OSGeo-Live ist somit ein sehr gutes Beispiel für ein erfolgreiches OpenSource-Projekt. Dabei handelt es sich nicht um ein klassisches Software-Projekt, sondern um eine Zusammenstellung verschiedenster Anwendungen und Informationen, die dem interessierten Publikum als gut sortierte Werkzeugkiste angeboten wird.

Projektübergreifend wurden viele Freiwillige gefunden, die regelmäßig die Inhalte aktualisieren. Damit ist ein Produkt entstanden, was als globale Visitenkarte nicht nur der OSGeo-Projekte dient. OSGeo-Live kann in Workshops und eigenen Veranstaltungen verwendet werden. Alle FOSS- und GIS-relevante Software wird mehrsprachig und mit Dokumentation zur Verfügung gestellt. So liegen beispielsweise Übersetzungen ins Deutsche, Italienische, Polnische, Griechische, Japanische, Französische, Catalanische, Chinesische, Koreanische vor.

Dieser Vortrag liefert Einblicke in die Entstehung und den Aufbau dieses Projektes. Die ehrgeizigen Ziele wie mehrsprachige Dokumentation, Benutzung von Beispieldatensätzen und Support in der Community stellen ganz unterschiedliche Anforderungen. Es gilt nicht nur technische Probleme zu lösen. Hinzu kommen terminliche Absprachen, damit zu bestimmten Anlässen wie wichtigen Konferenzen und Workshops aktuelle Versionen bereit stehen.

Auch auf der **FOSSGIS Konferenz 2013** kommt die OSGeo-Live in den Workshops zum Einsatz und wird als DVD an die Teilnehmer der Konferenz verteilt.



## OSM Buildings

### 3D Gebäudevisualisierung mit JavaScript

JAN MARSCH



OSM Buildings ist ein junges Open Source Projekt zur Visualisierung von abstrakten Gebäudegeometrien, basierend auf OpenStreetMaps Daten.

Die Darstellung erfolgt im Webbrowser mit gängigen Kartensystemen, wie z.B. LeafletJS oder OpenLayers, jedoch unabhängig von Plugins oder WebGL.

Im Detail dargestellt werden die Datengewinnung aus OSM-XML Dateien, Post-GIS Daten oder CartoDB Diensten. Der Aufbau des Datenservices wird beschrieben mit den Funktionen zur Aggregation und Datenkompression.

Es erfolgt eine Einführung in Funktionen der Client Bibliothek, insbesondere in 3D Projektion und Performanceoptimierung.

OSM Buildings wird mit ähnlichen Projekten verglichen, Gemeinsamkeiten und Unterschiede herausgestellt.

Anschließend wird ein Ausblick geboten auf in Arbeit befindliche Erweiterungen, wie z.B. 3D Floorplan Visualisierung.

## OSM Daten und MariaDB / MySQL

### Verbesserungen in aktuellen Releases

H<sub>H</sub>HOLZGRA

Aktuelle MariaDB- und MySQL-Versionen bieten deutliche Verbesserungen gegenüber der bisherigen GIS Implementation die bis einschließlich MySQL 5.5 im Einsatz war. Erstmals sind so zum Beispiel echte räumliche Relationen wie "ist enthalten in" oder "überlappt mit" möglich statt nur vereinfacht auf die Bounding Boxen (MBR - Minimal Bounding Rectangle) der einzelnen Geometrien zurückzugreifen.

MariaDB und MySQL werden damit zum ersten mal ernsthaft für GIS Anwendungen interessant, auch wenn der Funktionsumfang nach wie vor nicht mit dem von zB. PostGIS aufschließen kann.

Der Vortrag beschreibt zunächst den aktuellen Funktionsumfang der GIS Implementationen in den beiden Entwicklungszweigen, einschließlich der neuen Features und der Detailunterschieden zwischen aktuellen MariaDB und MySQL Versionen.

Im Anschluß wird eine für das MySQL API angepasste Version des OSM Import-Tools "osm2pgsql" vorgestellt und der vorhandene Funktionsumfang an Hand von Beispielen auf importierten OSM-Daten demonstriert.

## overpass turbo – Einfache Analyse von OpenStreetMap Daten

MARTIN RAIFER

overpass turbo (<http://overpass-turbo.eu>) ist eine grafische Benutzerumgebung für die Overpass API. Diese ermöglicht es auf eine einfache Art und Weise beliebige Abfragen auf OpenStreetMap Daten auszuführen und die Ergebnisse auf einer interaktiven Karte zu betrachten. Damit können auf sehr einfache Art und Weise beliebige Aspekte von OpenStreetMap beleuchtet werden.



Abb. 1: overpass turbo im Einsatz

### Motivation

Durch OpenStreetMap ist eine neue Gruppe von technisch interessierten Personen an die Welt der Geoinformatik herangebracht worden. Viele würden mit den Daten, die sie und Andere zu OSM beitragen gerne mehr anfangen als diese „nur“ auf einer Online-Karte zu betrachten.

Die direkte Nutzung der Daten war zwar von Anfang an bei OSM vorgesehen, und ist auch einer der Grundprinzipien hinter dessen Open-Data Modell, jedoch stand für die meisten Anwendungsfälle eine recht hohe technische Barriere im Weg, die es zu Überwinden galt.

Der traditionelle Ansatz, nämlich lokal eine eigene Geo-Datenbank zu betreiben, stellt meiner Meinung nach für die meisten Fälle eine zu große Hürde dar – vor allem dann wenn es darum geht, neue interessierte Benutzer und Entwickler an OpenStreetMap heranzuführen.

Genau dieses Problem löste die sogenannte „Overpass API“, eine Geo-Datenbank die speziell für OpenStreetMap Daten entwickelt wurde und welche öffentlich über das Internet zugreifbar ist. Die Overpass API besitzt eine eigene (stark an OSM-Daten angelehnte) Abfragesprache, die relativ einfach erlernbar ist, wenn bereits mit dem Datenmodell von OSM vertraut ist. Das heißt, man benötigt dank Overpass API keine lokale Geo-Datenbank mehr, und kann trotzdem globale, minuten-aktuelle Abfragen über die enormen Datenmengen von OpenStreetMap schnell durchführen.

Das Arbeiten mit der Overpass API erforderte aber weiterhin „Programmierkenntnisse“ und war in vielerlei Hinsicht alles andere als einfach zu benutzen. Hier kommt *overpass turbo* ins Spiel. Das Ziel hinter *overpass turbo* ist, damit viele Problemstellungen der „Analyse von OSM-Daten“ in einem Bruchteil der Zeit, mit überschaubarem Aufwand und praktisch ohne technische Voraussetzungen zu lösen.

## overpass turbo – Einfache Analyse von OpenStreetMap Daten

### Anwendungsfälle

Nachfolgend ein paar typische Anwendungsfälle, bei denen man mit Hilfe von overpass turbo sehr komfortabel an sein Ziel kommt. Alle Beispiele können unter [1] selbst ausprobiert werden.

- Zeige mir alle Trinkbrunnen der Umgebung
- Erstelle eine Liste von Apotheken der Stadt
- Gibt es ein Restaurant der Umgebung, welches italienische Küche anbietet
- Gibt es in der Stadt Straßen die nach einer bestimmten Person benannt sind?
- Erstelle eine einfache ÖPNV-Karte
- Zeige mir von allen Apotheken in der Stadt nur jene, die sich innerhalb von x Meter von einer Bushaltestelle befinden.



Abb. 2: verschiedene OSM-Objekte

Eine der größten Stärken der Overpass API und damit auch von turbo ist die Datenfilterung (sowohl nach geographischen als auch nicht-geographischen Eigenschaften). Im Prinzip kann aber jede Fragestellung beantwortet werden, welche von der Overpass API unterstützt wird.

### „Erweiterte“ Overpass Abfragen

Um das Arbeiten mit der Overpass API zu erleichtern verwendet overpass turbo eine „Erweiterung“ der Overpass Abfragesprache. So ist es beispielsweise möglich mit dem Kürzel „`{{bbox}}`“ das langwierige Heraussuchen und einfüllen der Koordinaten eines gewünschten Gebietes vollkommen automatisiert zu erledigen. Analog ist es möglich, eigene Variablen zu definieren: nach „`{{variable=irgendetwas}}`“ wird „`{{variable}}`“ durch „`irgendetwas`“ ersetzt.

Eine weitere Anwendung dieser erweiterten Syntax ist die Möglichkeit der Verwendung von MapCSS-Stylesheets: Durch „`{{style: ... }}`“ wird entsprechender MapCSS-Code zur Änderung des Designs der Ergebnisdaten eingebunden:

### MapCSS Stylesheets

In vielen Fällen ist es sehr hilfreich, die Ergebnisse einer Overpass API Anfrage nochmals optisch aufzuarbeiten. Beispielsweise könnte hier die Darstellung Restaurant-POIs mit unterschiedlicher Farbe je nach Küche sein, oder die Darstellung der Schwierigkeit von Wanderwegen als verschiedene Linienstile.

overpass turbo verwendet dafür MapCSS [2], eine an CSS angelehnte Sprache für Karten-Designs, die im OSM-Umfeld verbreitet ist. Damit können Farbe, Breite, Linien-Stil, Icons und Symbole von allen dargestellten Objekten beliebig angepasst werden; und das abhängig von den Tags, die die einzelnen Objekte tragen. Durch die MapCSS-Anweisung „`{{style: way[highway=path] { color:red; } }}`“ können beispielsweise alle Pfade rot eingefärbt werden. Ein paar Beispiele können unter [1] selbst ausprobiert werden.

## overpass turbo – Einfache Analyse von OpenStreetMap Daten



Abb. 3: verschiedene Linienstile

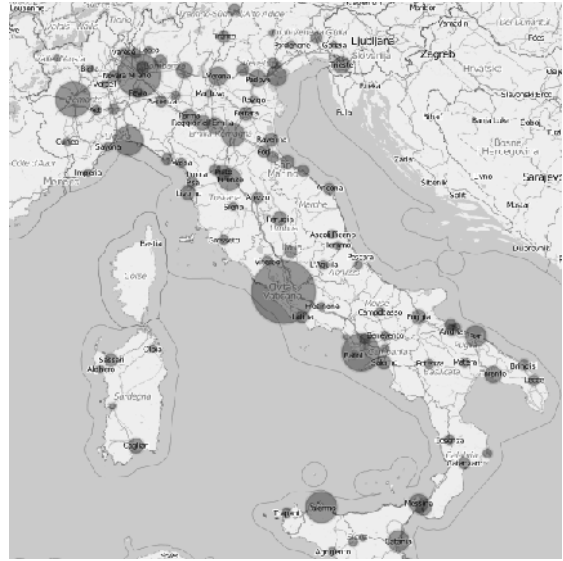


Abb. 4: Einfaches Kartogramm der Einwohnerzahl einiger italienischer Städte

### Export

Eine weiteres Anwendungsgebiet für overpass turbo ist es OpenStreetMap-Daten zu exportieren. Die Möglichkeiten reichen hier von PNG-Bildern des Kartenausschnitts mit den geladenen Daten, über verschiedene offene Datenformate wie GeoJSON [3] und GPX bis hin zur Möglichkeit, eine interaktive Karte als IFrame in eine Webseite einzubetten. Über GeoJSON können beispielsweise ausgewählte OSM Daten zur weiteren Verwendung in anderen Geo-Analyse-Werkzeugen erzeugt werden.

### Technisches

Das besondere an overpass turbo in technischer Hinsicht ist, dass die Anwendung komplett als Javascript im Browser des jeweiligen Benutzers ausgeführt wird.

Hierbei profitiere ich stark davon, dass es bereits viele sehr gute Komponenten gibt, die verschiedene Arbeiten übernehmen. Zu aller erst nenne ich hier Leaflet [4], ein großartiges Modul zum Darstellen von Karten um Web. Daneben verwende ich CodeMirror [5], ein Code-Editor für das Web. Die Map-CSS-Implementierung stammt von Richard Fairhurst. Weitere Bibliotheken sind für verschiedene andere Features im Einsatz.

overpass turbo stellt darauf aufbauend alles zur Verfügung, was mit Overpass/OSM zusammenhängt. Hauptsächlich:

- Parsen von OSM Daten
- Umwandlung in GeoJSON
- Styling von GeoJSON Objekten mit MapCSS
- Export der Daten

Der Ablauf der beim Ausführen einer Overpass Query von Statten geht ist vereinfacht dargestellt folgender:

1. Zusammenbauen der aktuellen Query (ausfüllen der Kürzel, usw.)
2. Absenden der Anfrage an den Overpass API Server
3. Resultat interpretieren: Fehlermeldung oder gültige XML/JSON Daten?
4. OSM-Daten parsen und in GeoJSON konvertieren

## overpass turbo – Einfache Analyse von OpenStreetMap Daten

5. Vektordaten mit Hilfe von Leaflet darstellen
6. und zugleich mit MapCSS stylen

Im Laufe der Entwicklung ergaben sich ein paar interessante Problemstellungen, die es zu Lösen galt: z.B. muss für die Umwandlung in GeoJSON bekannt sein, ob es sich bei einem Objekt um eine Linie oder eine Fläche handelt, was im OSM Datenmodell a priori nicht bekannt ist. Im Zuge dessen habe ich mit Hilfe des „Taginfo“ Dienstes von Jochen Topf eine Heuristik [6] entwickelt um diese Frage zu beantworten.

### Ausblick

Neben der immer-währenden und immer-notwendigen Verbesserung der Usability / Benutzerfreundlichkeit, sowie der Erweiterung der MapCSS-Unterstützung stehen noch ein paar größere Ideen im Raum, die ich gerne für overpass turbo implementieren würde:

- Alternative Datenquellen (lokale OSM-Dateien, die OSM „main API“, etc.)
- Eine einfache Art, um Overpass Querys zu erstellen (ohne die Overpass Query Language zu kennen)
- Scripting: Es wäre naheliegend, (erfahrenen) Benutzern die Möglichkeit zu geben, eigenen Javascript-Code ausführen zu lassen, und damit das Verhalten von overpass turbo zu beeinflussen. So könnten z.B. OSM-Daten mit anderen Datenquellen (z.B. statistischen Daten) verschnitten werden.

Kontakt zum Autor:

Martin Raifer  
martin.raifer@gmail.com

### Literatur

- [1] [http://wiki.openstreetmap.org/wiki/Overpass\\_turbo/FOSSGIS2013](http://wiki.openstreetmap.org/wiki/Overpass_turbo/FOSSGIS2013)
- [2] <http://wiki.openstreetmap.org/wiki/MapCSS>
- [3] <http://geojson.org>
- [4] <http://leafletjs.com>
- [5] <http://codemirror.net>
- [6] [http://wiki.openstreetmap.org/wiki/Overpass\\_turbo/Polygon\\_Features](http://wiki.openstreetmap.org/wiki/Overpass_turbo/Polygon_Features)



## Performanceoptimierte WMS-Dienste mit QGIS Server

DR. MARCO HUGENTOBLER, SOURCEPOLE AG

### 1. Einleitung

Web Mapping Services (WMS) generieren häufig eine grosse Last auf dem Server, da jede Karte neu gerendert werden muss. Eine hohe Leistungsfähigkeit des WMS Servers ist daher für Webkarten, die direkt auf den WMS zugreifen, absolut zentral. Aber auch wenn sich ein Tilecache zwischen Webkarte und WMS befindet, ist es wichtig, dass der Cache innert nützlicher Frist erneuert werden kann. Dieser Beitrag befasst sich zuerst mit der Definition der Leistungsfähigkeit und deren Messung. Es werden praktische Werkzeuge gezeigt, mit denen Flaschenhälse entdeckt und verbessert werden können. Es folgen einige allgemeine Performancetips für Serveradministratoren, um Webkarten mit QGIS Server performant zu generieren. Zum Schluss wird, anhand des WMS der amtlichen Vermessung der Schweiz, die Performance von QGIS Server mit dem UMN Mapserver verglichen.

### 2. WMS Leistungsfähigkeit und Messung

Yang et al. (2011) definiert vier Perspektiven von WebGIS Performance, die wichtigste davon die Zeit: „Time complexity is the the most important factor in performance measurement. It refers to the time spent between the submission of a request by an end user and the response finalized at the client side“. Die akzeptable Wartezeit für eine Antwort wird, basierend auf Corner (2010), mit 8 Sekunden angegeben. Das mag für Webbasierte GIS-Analysen angemessen sein, für WMS Abfragen hingegen dürfte die von den Benutzern tolerierte Antwortzeit wesentlich tiefer liegen.

Für Kartenservices ist nicht nur die Antwortzeit für eine Abfrage interessant. Ebenfalls wichtig ist, wie gut das System bei parallelen Abfragen von mehreren Benutzern skaliert (Durchsatz pro Zeiteinheit).

Die Beantwortung einer WMS-Abfrage setzt sich, vereinfacht aus verschiedenen Schritten zusammen:

- Daten von der Datenquelle lesen
- Rendern des Bildes im Speicher
- Umwandeln des Bildes in ein Webformat (png8/24/32, jpeg, ...)
- Transfer des Bildes vom Server zum Client

#### 2.1 Werkzeuge zur Diagnose

In der Linux-Welt gibt es einige nützliche Werkzeuge, um Informationen über Leistung und mögliche Verbesserungen zu gewinnen:

- Firebug misst die Zeit für Webanfragen. Ausserdem ist ersichtlich, welchen Anteil der Datentransfer ausmacht.
- vmstat zeigt Informationen zum virtuellen Speicher und zur Prozessorauslastung. Es ist ersichtlich, ob Disk oder Prozessor der limitierende Faktor ist.
- jmeter erlaubt die Analyse der Performance bei parallelen Anfragen (Durchsatz). Eine solche Analyse wird in Kapitel 4 durchgeführt.
- valgrind hilft beim Analysieren, in welchen Funktionen der WMS Server viel Zeit verbraucht. Wichtig ist, beim Profilen auch einen zweiten Durchlauf zu simulieren, da QGIS Server beim Aufstarten vieles cacht (Projektfile, Layer, CRS, gerasterte SVG Symbole).

### 3. Tipps für performante Webkarten mit QGIS Server

- Anzahl Objekte und Rendervorgänge mittels massstabsabhängiger Symbolisierung begrenzen.
- Räumliche Indizes für Vektoren und Pyramiden für Raster verwenden.
- Datenbankzugriff über Local socket anstatt TCP, falls die Datenbank auf der gleichen Maschine läuft wie QGIS Server.
- Besonders zeitintensiv beim Rendern sind gestrichelte Linien, Polygonumrandungen, Labelbuffers.
- Für Raster jpeg als Format verwenden, um so wenig Daten als möglich zwischen Server und Client zu transportieren.
- Für Vektoren png8 verwenden falls von der Qualität her möglich. Sowohl das Umwandeln als auch der Transfer zum Client sind schneller als bei png24 oder 32.
- Mit der Anzahl der gleichzeitig laufenden QGIS Server Instanzen experimentieren (bei Apache fcgi-Variablen 'FcgidMinProcessesPerClass' und 'FcgidMaxProcessesPerClass').
- Die Antwortzeit mit progressivem Rendering verkürzen („Progressive WMS“). Dabei wird zuerst ein niedrig aufgelöstes Bild angefordert, das vom WMS schnell geliefert werden kann und als Vorschau dient. Erst dann wird das Bild in der richtigen Auflösung angefordert. Dieses Verfahren ist z.B. im WMS-Dienst 'QGIS cloud' realisiert.
- Einen Reverse Proxy Cache, z.B. Varnish, vorschalten und die Karte in Kacheln abfragen. Der Cache beschleunigt nicht nur das Laden der Webseite (Javascript, Grafiken, CSS), sondern bietet auch eine gute Alternative zu einem filebasierten Tilecache[1].
- Wenn die Webkarte immer noch nicht schnell genug ist, kann als letztes Mittel ein filebasierter Tilecache vor den WMS geschaltet werden. Ein Tilecache ist allerdings immer mit erhöhtem Speicherverbrauch, mehr Wartung und weniger Flexibilität verbunden.

### 4. Benchmark WMS der amtlichen Vermessung

Der folgende Benchmark wurde mit den Daten der amtlichen Vermessung des Kantons Glarus durchgeführt. Es wurden zwei WMS mit möglichst identischen Symboldefinitionen gemäss KKVA (2010) aufgesetzt, einer mit QGIS Server (Version QGIS Enterprise 13.01, basierend auf 1.9) und einer mit UMN Mapserver (Version 6.2.1). Als Datenquelle kam eine PostGIS Datenbank zum Einsatz. Mit einem jmeter Skript wurde der Durchsatz (Anfragen pro Sekunde) gemessen, mit 1,2,4,8,16,32,64 parallelen Anfragen. Angefragt wurden zufällig ausgewählte Ausschnitte in den Massstäben 1:2'000 – 1:5'000 (in der AV sind viele Ebenen erst in grossen Massstäben sichtbar). Ausgabeformat war png8. WMS Server, Datenbank und jmeter wurden auf derselben Maschine laufengelassen (Laptop Quadcore 2.7 GHz). Somit ist der Netzwerktransfer im Test nicht berücksichtigt, sondern nur Lesen von Daten, Rendern sowie Konvertierung in Grafikformat.

## Performanceoptimierte WMS-Dienste mit QGIS Server



Abbildung 4: Ausschnitt AV-WMS, gerendert mit QGIS Server

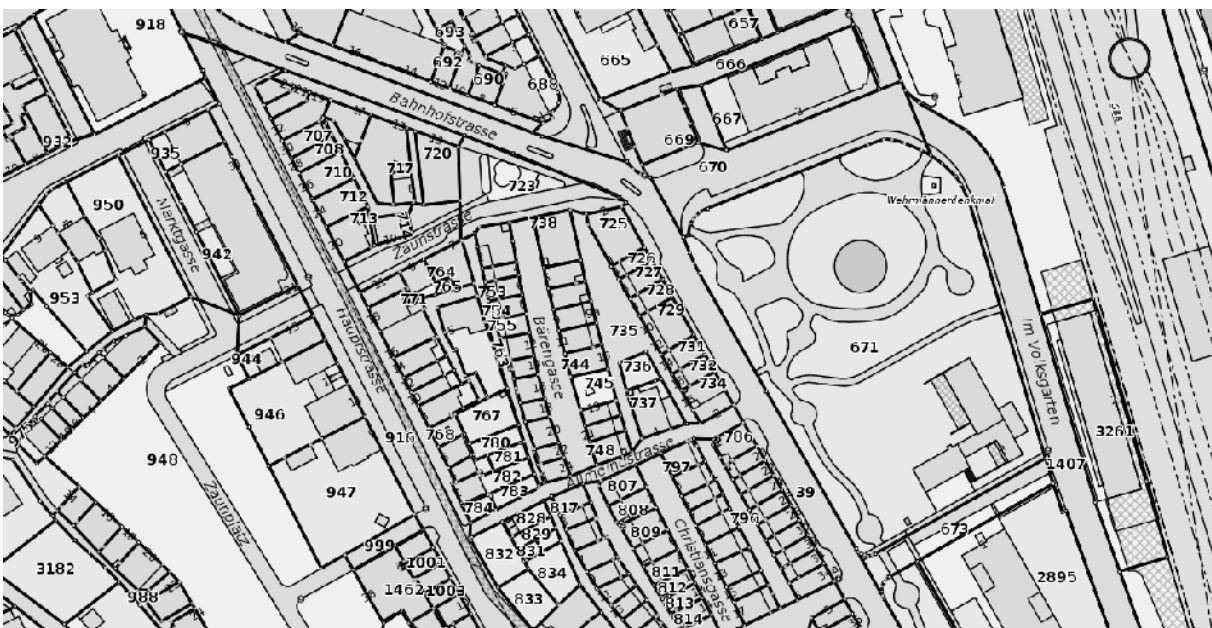


Abbildung 5: Ausschnitt AV-WMS, gerendert mit UMN Mapserver

### 4.1 Erste Resultate

Die Anwendung ist sehr CPU-intensiv, da nach kurzer Einlaufphase die Daten von Postgres gecacht sind. Sowohl UMN mapserver als auch QGIS Server erreichen sehr gute Durchsatzraten. Bei 32 oder sogar 64 gleichzeitige Anfragen können allerdings die maximale Antwortzeit beträchtlich ansteigen.

## Performanceoptimierte WMS-Dienste mit QGIS Server

N Requests	Count	Avg (ms)	Min (ms)	Max (ms)	Throughput (Req/s)
1	100	128	52	712	7.7
2	100	101	34	457	16.4
4	200	130	41	744	27.4
8	200	246	44	2322	26.7
16	400	517	47	3960	25
32	384	761	46	9491	29.7
64	768	1474	63	25124	27.1

Tabelle 1: Resultate QGIS Server

N Requests	Count	Avg (ms)	Min (ms)	Max (ms)	Throughput (Req/s)
1	100	151	86	357	6.5
2	100	112	57	198	15.1
4	200	166	84	440	21.9
8	200	313	65	935	22.3
16	400	644	83	3232	22.8
32	384	965	60	12649	23.2
64	768	1901	100	28496	22.7

Tabelle 2: Resultate UMN mapserver

### 4.4 Analyse mit Valgrind

Mit dem Profiler Valgrind kann analysiert werden, wieviel Zeit der WMS Server in welchen Funktionen braucht. Von Valgrind nicht erfasst wird die Zeit, in welcher der Prozess auf I/O wartet. Eine erste Analyse einer Anfrage des AV-WMS liefert folgendes Bild:

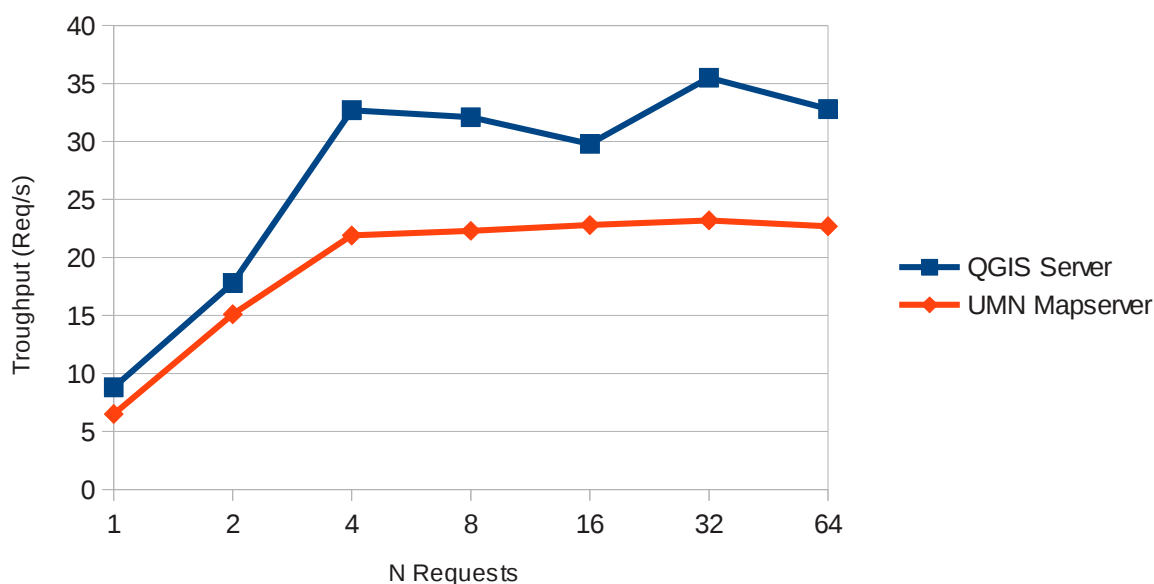
- 50% der Zeit wird für das Umwandeln des Bildes nach 8-bit und png verbraucht
- Das Rendern braucht die anderen 50% der Zeit
- Das Rendern der wenigen Labels braucht 25% der Zeit. Hier scheint es noch viel Optimierungspotenzial zu geben
- Polygonkoordinaten werden transformiert, obwohl dies nicht nötig ist. Dies kann direkt behoben werden und spart 4% der Zeit.
- Für die Konvertierung des Bildes nach 8-Bit wird häufig die Funktion QImage::pixel aufgerufen, was 5% der Zeit braucht. Dies kann optimiert werden, indem stattdessen die Funktion QImage::scanLine verwendet wird.

Nach Verbesserungen der beiden letzten Punkte wird der QGIS Server WMS noch schneller. Selbstverständlich werden die Verbesserungen (neben vielen anderen) in QGIS Enterprise 13.02 sowie QGIS 2.0 enthalten sein.

## Performanceoptimierte WMS-Dienste mit QGIS Server

N Requests	Count	Avg (ms)	Min (ms)	Max (ms)	Throughput (Req/s)
1	100	112	39	644	8.8
2	100	86	27	407	17.8
4	200	107	36	523	32.7
8	200	206	38	2415	32.1
16	400	444	43	3718	29.8
32	384	611	42	7268	35.5
64	768	1273	53	20593	32.8

Tabelle 3: Resultate QGIS Server nach erster Optimierung



### 4.3 Fazit

Das Beispiel des AV-WMS ist in hohem Masse praxisrelevant, da fast alle Kantone einen solchen WMS anbieten. Da die Anwendung sehr CPU-intensiv ist, lohnen sich Optimierungen an der WMS Serversoftware besonders.

Performancetests sind mit vielen Annahmen verbunden. Es braucht weitere Test, um die Sensitivität bezüglich Testausschnitten, Format und FastCGI Einstellungen zu untersuchen. Um gezielt Performanceschwächen in den Servern zu finden, ist es weiter interessant, einzelne Ebenen separat zu testen oder besonders langsame Anfragen mit dem Profiler Valgrind zu untersuchen.

Kontakt zum Autor:

Dr. Marco Hugentobler  
Sourcepole AG

## Performanceoptimierte WMS-Dienste mit QGIS Server

Weberstr. 5, 8004 Zürich  
+41 44 440 77 11  
marco.hugentobler@sourcepole.ch

### Literatur

- [0] Corner, S. 2010: the 8-second rule, <http://www.submitcorner.com/Guide/Bandwidth/001.shtml>. Last accessed: 10th May 2013.
- [1] Loechel, Alexander; Schmid, Stephan 2012: Caching techniques for high-performance Web Map Services. Proceedings of the AGILE'2012 International Conference on Geographic Information Science, Avignon.
- [2] Konferenz der Kantonalen Vermessungsämter (KKVA) 2010 : AV -WMS, Web Map Service mit den Daten der amtlichen Vermessung. Empfehlungen für die Realisierung.
- [3] Open Geospatial Consortium: Open GIS Web Map Server Implementation Specification, Version 1.3, 2006.
- [4] QGIS cloud 2013: <http://www.qgiscloud.com>. Last accessed: 10th May 2013.
- [5] Yang, *Chaowei* ; Wu, *Huayi et al* 2011: WebGIS performance issues and solutions. In: Advances in Web-based GIS, Mapping Services and Applications. Taylor & Francis Group, London.

## QGIS-Eingabemasken für PostGIS-Layer leicht gemacht

BERNHARD STRÖBL

### Einleitung

In relationalen Datenbanken können komplexe Sachverhalte konsistent und redundanzfrei gespeichert werden. Da sie auch Geometrien speichern können, bieten sie sich als zentrale Datenspeicher für Geodateninfrastrukturen an. Sie ermöglichen es auch, komplexere Sachverhalte und Abläufe zu modellieren, weshalb zu einzelnen Geometrieobjekten häufig eine Vielzahl von Sachdaten gespeichert wird. Selbst bei an sich einfachen Sachverhalten sind als Folge der Normalisierung meistens mehrere Tabellen mit Beziehungen unterschiedlicher Kardinalität beteiligt.

PostgreSQL/PostGIS ist als relationale Datenbank im FOSSGIS-Bereich weit verbreitet. Damit Nutzer Daten eingeben und pflegen können, sind entsprechende Eingabemasken im Front-End nötig. QGIS 2.0 bietet bereits viele Möglichkeiten der Anpassung eines Layers an die Datenmodellierung an. Das Bearbeitungselement *Wertabbildung* z.B. erzeugt eine Combobox, mit der Werte einer 1:n-Verknüpfung in den Layer eingetragen werden können. Ein Layer kann auch direkt mit anderen Layern verknüpft werden (Kardinalität 1:1 bzw. n:1), die verknüpften Felder lassen sich aber nicht bearbeiten. Insbesondere für n:m-Beziehungen und Konsistenzprüfungen (z.B. *NotNull*) sind bisher eigene Masken oder Plugins zu programmieren.

### Datengetriebene Eingabemasken

Das QGIS-Plugin *DataDrivenInputMask* erzeugt Eingabemasken für PostgreSQL-Layer zur Laufzeit. Mit ihnen können die Daten des Layers selbst, aber auch die aller verbundenen Relationen – egal welcher Kardinalität – bearbeitet werden. Das Plugin fragt dazu Systemkataloge der Datenbank [1] ab und setzt Informationen über Datentypen und Constraints automatisch in entsprechende Bearbeitungselemente um. Es entsteht kein Aufwand durch die Einrichtung des Layers oder die Programmierung eines Plugins. Insbesondere entspricht die Datenmaske stets dem Stand der Datenmodellierung in der Datenbank.

### Unterstützte Datentypen

Unterstützt werden z. Zt. die Datentypen *int2*, *int4*, *int8*, *char*, *varchar*, *text*, *bool* und *date*. Für jeden Datentyp wird ein passendes QWidget [2] erzeugt und angeboten (siehe Tabelle). Der Hauptunterschied zur Standardeingabemaske ist, dass Falscheingaben ausgeschlossen werden: Zahlenfelder akzeptieren keinen Text und ist ein Feld als *NotNull* definiert, kann die Eingabemaske nicht geschlossen werden, wenn das Feld leer ist. Defaultwerte werden bei neuen Datensätzen automatisch eingefügt.

### Unterstützte Beziehungen

Es werden alle denkbaren Beziehungen (1:1, 1:n und n:m) zwischen Tabellen unterstützt, sofern sie über Fremdschlüssel definiert sind.



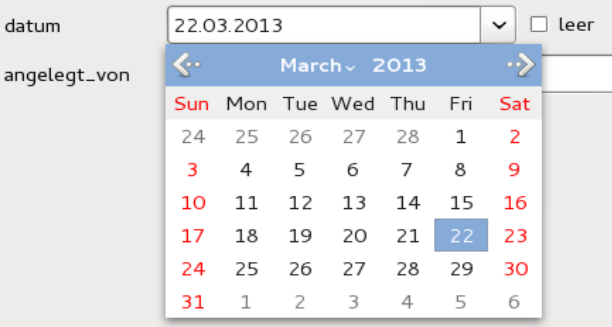


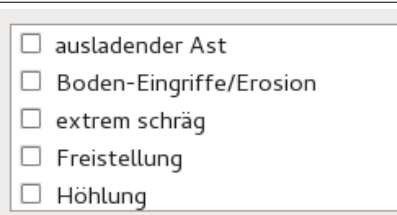
Eins-zu-eins-Beziehungen (1:1) sind eher selten. Hier ist einem Datensatz einer Tabelle genau ein Datensatz einer anderen Tabelle zugeordnet. Sinnvoll erscheint nur folgender Anwendungsfall: Mehrere Tabellen sollen eine Reihe gemeinsamer Felder haben. Anstatt nun diese Felder in beiden Tabellen anzulegen (und Constraints zu definieren), werden sie nur in einer „Eltern“-tabelle angelegt. Der Primärschlüssel der „Kind“-tabelle ist Fremdschlüssel auf den Primärschlüssel der Elterntabelle. In der Eingabemaske der Kindtabelle tauchen alle Elterntabellen als eigene Tabs auf. Dies funktioniert auch über mehrere „Generationen“ und ermöglicht damit eine effektive Gestaltung der Eingabemasken über das Datenbankdesign.

## QGIS-Eingabemasken für PostGIS-Layer leicht gemacht

Eins-zu-viele-Beziehungen (1:n) treten häufig mit sogenannten *Lookup*-Tabellen auf. Beispiel Biotoptypen: ein Biotoptyp pro Fläche, wobei ein und derselbe Biotoptyp auf mehreren Flächen vorkommen kann. *Lookup*-Tabellen bestehen in der Regel nur aus zwei Feldern, nämlich dem Primärschlüssel (z.B. Biotoptyp) und einem Notnull-varchar-Feld (Biotop). Bezieht sich ein Fremdschlüssel auf eine derartige Tabelle, wird dem Nutzer eine QComboBox angeboten, aus der er den zutreffenden Texteintrag auswählen kann. Wird jedoch die Eingabemaske der verknüpften Tabelle geöffnet (n:1), wird dem Nutzer ein QTableWidgetItem angeboten.

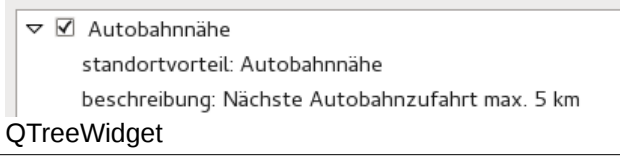
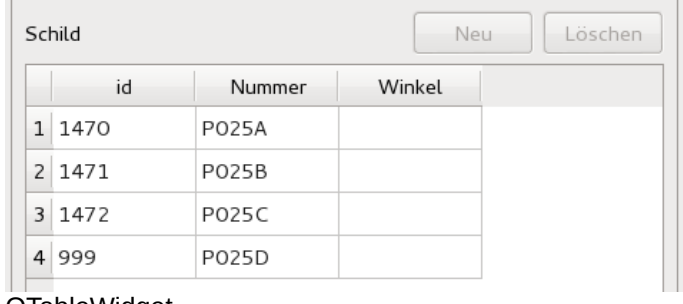
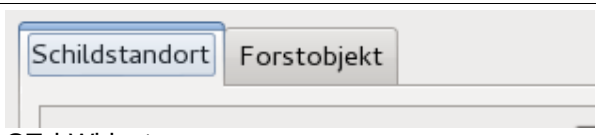
Viele-zu-viele-Beziehungen (n:m) werden üblicherweise über eine Zwischentabelle realisiert, wobei die Primärschlüssel der beteiligten Tabellen gemeinsam den Primärschlüssel der Zwischentabelle bilden, Beispiel: mehrere Eigentümer pro Objekt, wobei ein Eigentümer auch mehrere Objekte besitzen kann. Für n:m-Beziehungen werden, je nach Anzahl der Felder unterschiedliche QWidgets eingesetzt (siehe Tabelle).

Tabelle: mögliche Eingabeelemente und ihre Verwendung

Abbildung/QWidget	Verwendung, Hinweise
 <p>integer_value 42</p> <p>QLineEdit</p>	Zahlen- und char/varchar-Felder, sofern keine Fremdschlüssel
 <p>Busanbindung <input checked="" type="checkbox"/></p> <p>QCheckBox</p>	Ja/Nein-Felder
 <p>datum 22.03.2013 <input type="checkbox"/> leer</p> <p>angelegt_von</p> <p>QDateEdit</p>	Datumfelder; mit <i>leer</i> kann ein NULL-Datum gespeichert werden, diese Möglichkeit fällt weg, wenn ein Notnull-Constraint für das Feld definiert ist
 <p>text_value</p> <p>QTextEdit</p>	Textfelder beliebiger Länge
 <p>VSP_Massnahme_id (Massnahme) Fällung</p> <p>QComboBox</p>	1:n-Beziehungen; angezeigt wird stets das erste Notnull-varchar Feld, ist keines vorhanden, der Primärschlüssel
 <p><input type="checkbox"/> ausladender Ast</p> <p><input type="checkbox"/> Boden-Eingriffe/Erosion</p> <p><input type="checkbox"/> extrem schräg</p> <p><input type="checkbox"/> Freistellung</p> <p><input type="checkbox"/> Höhlung</p> <p>QListWidget</p>	n:m-Beziehungen mit nur einem weiteren Feld in der verbundenen Tabelle; bereits ausgewählte Elemente werden beim Öffnen nach oben verschoben



## QGIS-Eingabemasken für PostGIS-Layer leicht gemacht

Abbildung/QWidget	Verwendung, Hinweise
 <p>QTreeWidget</p>	n:m-Beziehungen mit mehr als einem weiteren Feld in der verbundenen Tabelle
 <p>QTableWidget</p>	<p>1:1-Beziehungen (Eingabemaske der „Eltern“-tabelle ist geöffnet); es kann jeweils nur ein Element angelegt werden</p> <p>n:1-Beziehungen (Abb. links)</p> <p>n:m-Beziehungen mit mehr als zwei Feldern in der Zwischentabelle</p>
 <p>QTabWidget</p>	1:1-Beziehungen (Eingabemaske der „Kind“-tabelle ist geöffnet)

### Hinweise zum Datenbankdesign

Das Datenbankdesign sollte mit der nötigen Sorgfalt vorgenommen werden. Folgende Punkte verdienen besondere Beachtung:

- Primärschlüssel sollten aus einem Feld bestehen.
- Zwischentabellen (n:m) werden nicht geladen, sondern direkt über SQL-Befehle manipuliert, deshalb kann hier der Primärschlüssel – wie üblich – zusammengesetzt sein.
- Die referenzielle Integrität muss stets auf der Datenbank sichergestellt werden, ist ein Feld ein Fremdschlüssel, muss ein entsprechender Constraint definiert werden.
- Datensichten können mit diesem Plugin nicht bearbeitet werden, weil für sie keine Constraints definiert werden können.
- Kommentare zu Feldern oder verbundenen Tabellen werden als Tooltip angezeigt.

### Schlussbemerkungen

Das Plugin *DataDrivenInputMask* bietet sich für alle QGIS-Nutzer an, die komplexere Datenmodelle in PostgreSQL/PostGIS realisieren und dafür keine eigenen Eingabemasken programmieren können oder wollen. Es ist mit den QGIS-Versionen 1.8 und 2.0 lauffähig. Einige Funktionen der Klasse *Dd-Manager* sind als API-Funktionen gekennzeichnet, die es ermöglichen, die Funktionalität in eigene Plugins einzubinden. Interessant ist in diesem Zusammenhang die Funktion *initLayer*, die die Eingabemaske erzeugt. Über sie können einzelne Felder aus der Maske ausgeblendet oder ihre Beschriftung verändert werden. Auch die Reihenfolge der Felder in der Maske kann darüber geändert werden, normalerweise entspricht sie der Reihenfolge, mit der die Felder in der Tabelle angelegt wurden.

Für QGIS 2.1 gibt es Pläne, Verknüpfungen jedweder Kardinalität zu ermöglichen und durch Bearbeitungselemente zu unterstützen. Sollte dieser Plan realisiert werden und die Funktionalität über die Python API zur Verfügung stehen, wird das Plugin so angepasst, dass es Layer entsprechend der ausgelesenen Datenbankinformationen mit anderen Layern verknüpft und ihre Eingabemaske verändert.

## QGIS-Eingabemasken für PostGIS-Layer leicht gemacht

Diese Beschreibung entspricht dem Stand April 2013, das Plugin wird permanent weiterentwickelt.

Kontakt zum Autor:

Bernhard Ströbl  
Kommunale Immobilien Jena  
Am Anger 26 D-07743 Jena  
03641/49-5190  
bernhard.stroebl@jena.de

Literatur

[1] <http://www.postgresql.org/docs/manuals/> abgerufen am 12.4.2013

[2] <http://qt-project.org/doc/qt-4.8> abgerufen am 12.4.2013

## Qualitätssicherung von Geodaten auf der Basis von Web Processing Services

HORST DÜSTER

Bevor neue Daten in eine Geodaten-Infrastruktur überführt werden, müssen sie Qualitätssicherungsverfahren durchlaufen. Diese Prozesse können sowohl generisch, als auch speziell für ein bestimmtes Datum konfektioniert sein. Um sicher zu stellen, dass diese Prozesse langlebig und verfügbar sind, bietet es sich an OGC-Konforme Services zu diesem Zweck zu nutzen. Deshalb werden im Vortrag die Vor- und Nachteile OGC-Konformer Web Processing Services (WPS) als Basis der Qualitätssicherung von Geodaten diskutiert.

## Rendering und Bereitstellung massiver Geodaten unter Verwendung von OpenWebGlobe und MapCache in der Cloud

ROBERT WÜEST, MARTIN CHRISTEN, BENJAMIN LOESCH

Fachhochschule Nordwestschweiz Institut Vermessung und Geoinformation

### Einführung

Generierung und Nutzung von interaktiven geographischen 3D Daten über das World Wide Web (WWW) ist seit der Entstehung des WWW vor 20 Jahren eine Bestrebung. Abgesehen von diversen frühentwickelten Formaten wie VRML [1] und GeoVRML [2] waren die meisten Ansätze auf den Einsatz von proprietärer Software sowie sehr speicheraufwändige Hardware angewiesen, so dass der Betrieb eines solchen Systems mit erheblichem Aufwand und Kosten verbunden war.

Daten von globaler Ausdehnung für einen virtuellen 3D Globus stammen hauptsächlich von Satelliten oder durch werden durch Bildflüge generiert. Weiter kommen Daten von Laserscannern und mobiler-fasste Daten von fahrenden Messsystemen hinzu.

Vor allem die Speicherung und Bereitstellung von grossen prozessierten Datensätzen globaler Ausdehnung ist auch heute noch eine Herausforderung, die Genauigkeit der Daten und somit der Speicherverbrauch stetig zunimmt. Moderne Erfassungssysteme liefern mehrere Terrabytes an Daten pro Stunde.

Dieser Beitrag zeigt einen möglichen Ansatz um globale Geodaten speichereffizient und umfassend zu visualisieren. Zum einen wird die OpenWebGlobe ([www.openwebglobe.org](http://www.openwebglobe.org)) Technologie, die am Institut Vermessung und Geoinformation der Fachhochschule Nordwestschweiz entwickelt wird, zur Visualisierung von einem virtuellen 3D Globus verwendet und zum anderen wird unter Verwendung von MapCache eine Lösung zur selektiven und effizienten Datenaufbereitung vorgestellt.

### Stand der Forschung

Virtuelle Globen, manchmal auch 3D-Geobrowser genannt, sind in der Lage 3D-Umgebungen interaktiv darzustellen. Dabei lassen sich sehr grosse Mengen von georeferenzierten räumlichen Inhalten über das Internet streamen.

Virtuelle Globen gewannen in den letzten Jahren stark an Aufmerksamkeit in Medien und der Geo-Community, welche vor allem dem kommerziellen Produkte Google Earth (<http://earth.google.com>) zu verdanken ist.

Durch den HTML5 Standard ergeben sich neue Möglichkeiten virtuelle Globen direkt in einem Internet Browser einzusetzen unter Verwendung von WebGL [3]. Neben der hier verwendeten OpenWebGlobe (<http://www.openwebglobe.org>) Technologie existieren noch weitere WebGL basierte virtuelle Globen, wie Cesium (<http://cesium.agi.com/>), ReadyMap (<http://readymap.com/>) und WebGL Earth (<http://www.webglearth.org/>).

Für die beschleunigte Bereitstellung von Tile-basierten Bilddaten existieren mehrere Cache-basierte Lösungen: Mit TileCache (<http://tilecache.org>) gibt es eine OpenSource Implementierung einer WMS-C konformen Serverlösung unter der BSD Lizenz. Ebenfalls frei verfügbar sind die Lösungen MapProxy (<http://www.mapproxy.com>) unter der Apache 2.0 Lizenz und GeoMapCache (<http://geowebcache.org>) unter LGPL. Auch MapCache (<http://mapserver.org/mapcache>) eine Serverimplementierung zur Beschleunigung von WMS Layern geschrieben als Apache Modul in C, wird oft verwendet.

### Virtueller Globus OpenWebGlobe

## Rendering und Bereitstellung massiver Geodaten unter Verwendung von OpenWebGlobe und MapCache in der Cloud

Der Virtuelle Globus "OpenWebGlobe" ist Open Source unter MIT Lizenz verfügbar (<http://www.openwebglobe.org>). Der Source-Code des Viewers ist komplett in JavaScript geschrieben und benutzt die WebGL Technologie, welche 3D-Darstellung ohne Installation eines Plugins ermöglicht.

Der Technologie zugrundeliegend ist ein geodätisches Referenzellipsoid was der Minimierung von Fehlern und Erhöhung der Genauigkeit dienlich ist [4]. Verwendet wird das WGS84 Referenzellipsoid das unter WebGL direkt in einem herkömmlichen Web-Browser gerendert wird.

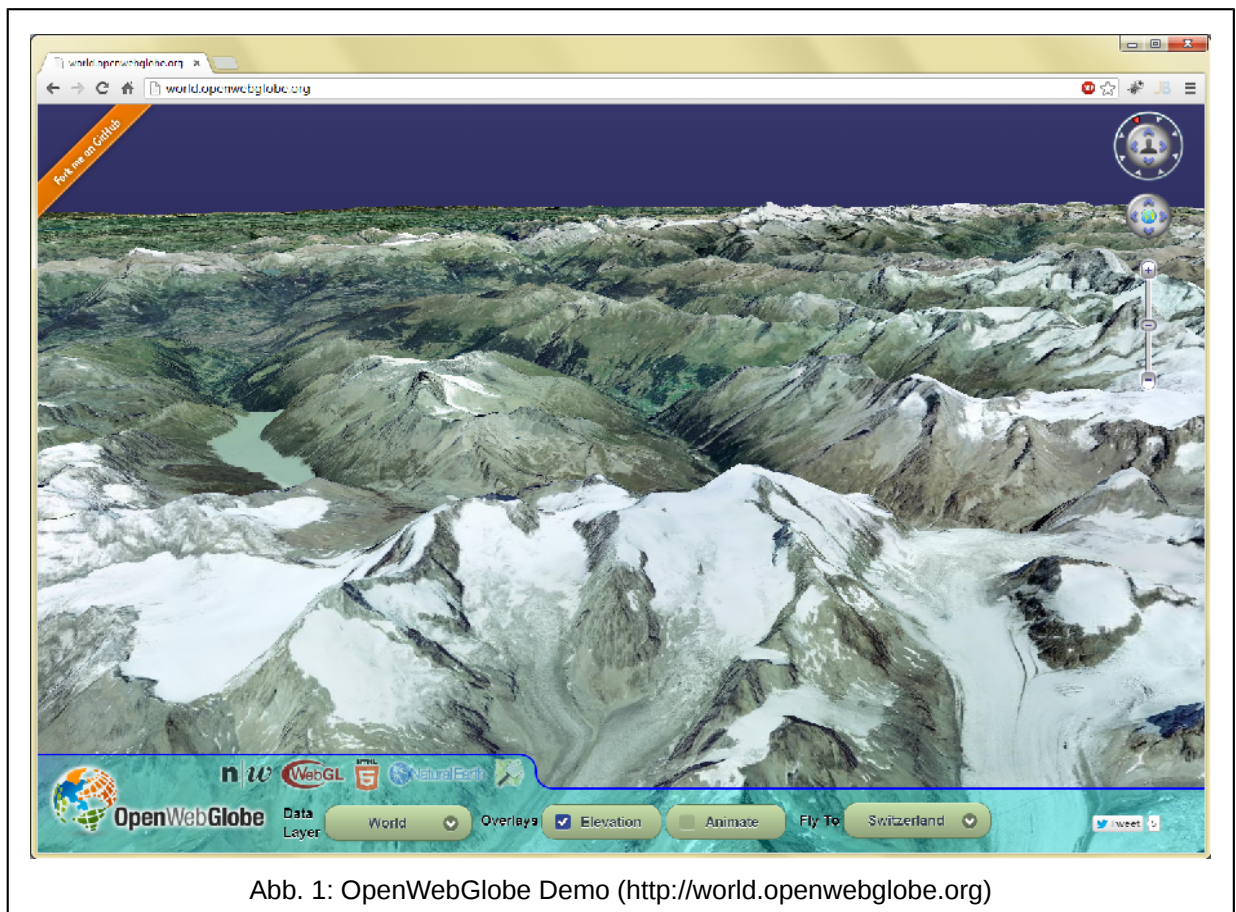


Abb. 1: OpenWebGlobe Demo (<http://world.openwebglobe.org>)

Auf dem Screenshot Abb. 1 ist eine mit dem OpenWebGlobe SDK erstellte Szene zu sehen die diverse Bild- und Höhendaten visualisiert. Alle Daten werden als Kacheln aus einer entsprechenden Quadtree Struktur geladen [6]. Höhendaten werden im Javascript Object Notation Format (JSON) gestreamet, das analog zu den Bildformate PNG und JPG zur Laufzeit in den Webbrowser geladen werden kann.

### Skalierbares Caching Architektur mit MapCache

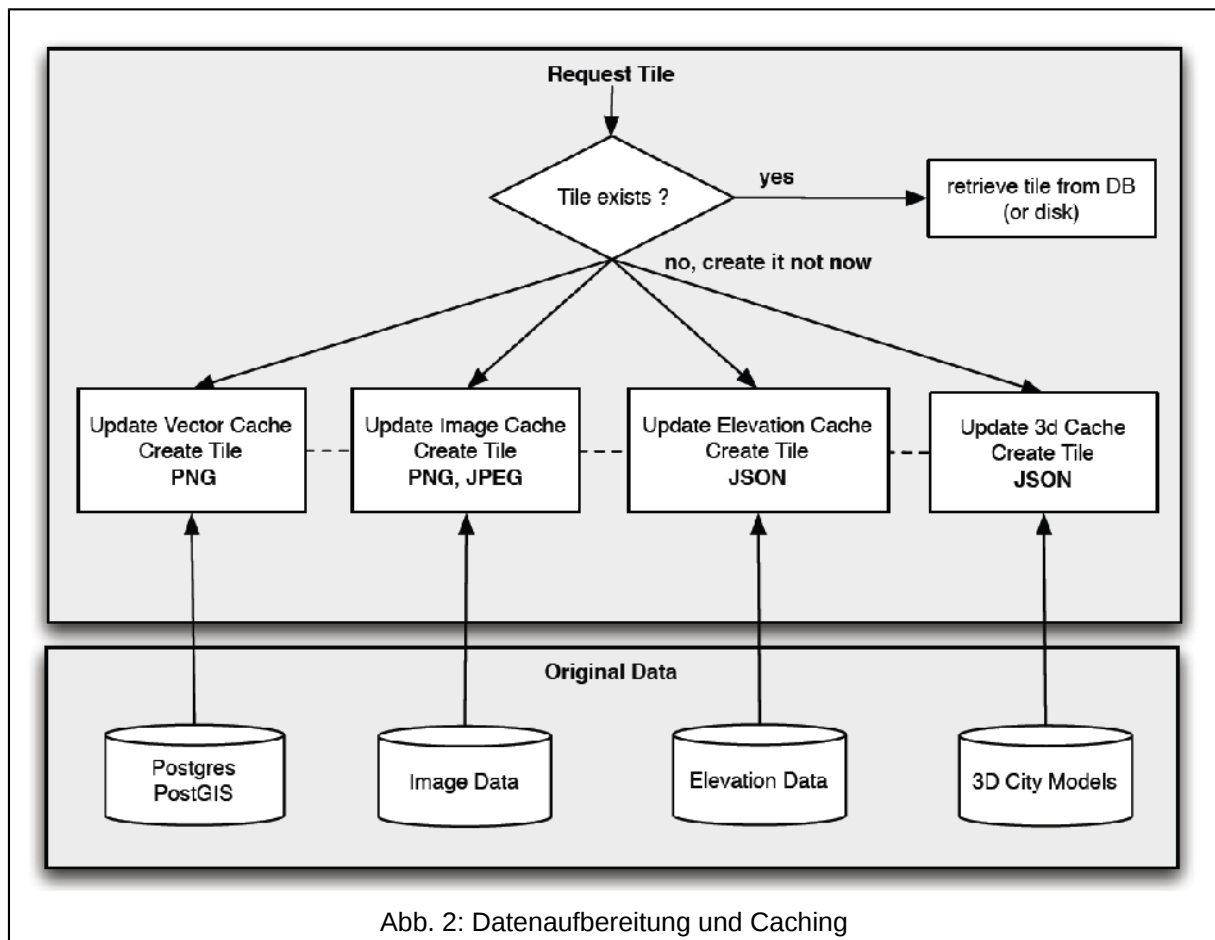


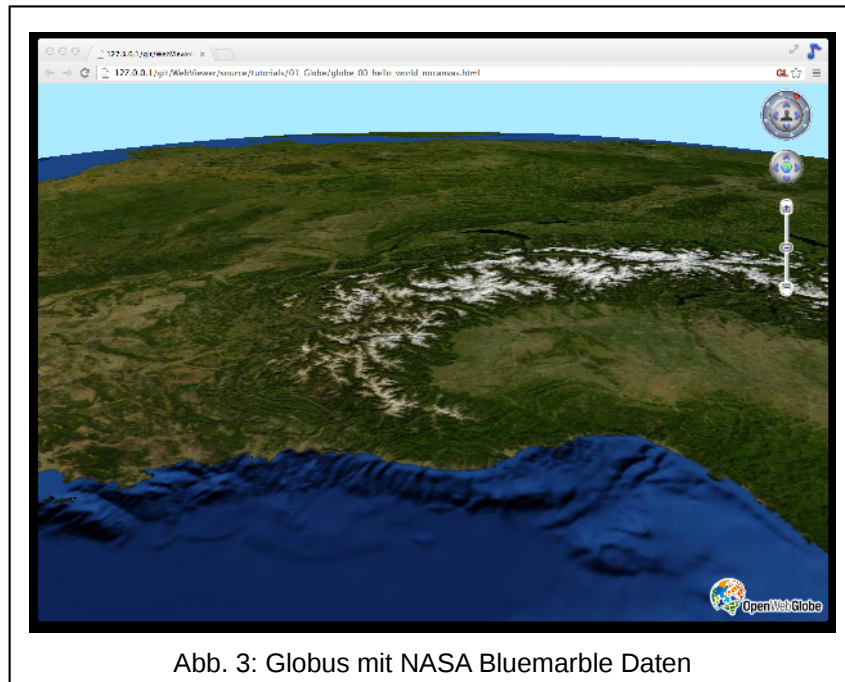
Abb. 2: Datenaufbereitung und Caching

Die Detaillierung von globalen Datensätzen stellt auch mit moderner Hardware immer noch ein viel bestimmendes Kriterium dar. Speicherung und Verbreitung von solch grossen Datensätzen ist mit der Entstehung von Cloud-Based Services wie Amazon Simple Storage (S3) [7] zwar stark vereinfacht worden, Datenmengen von mehreren Terrabytes sind dennoch problematisch. Weil aufbereitete globale Datenkacheln selten vollumfänglich genutzt werden, lohnt es sich mittels Caching hoch frequentierte Datenkacheln bereitzustellen während nicht so stark genutzte Daten mittels „On-The-Fly“ Processing auf Abruf bereitstehen.. Abb. 2 zeigt den vorgestellten Workflow für globale Daten verschiedenster Art.

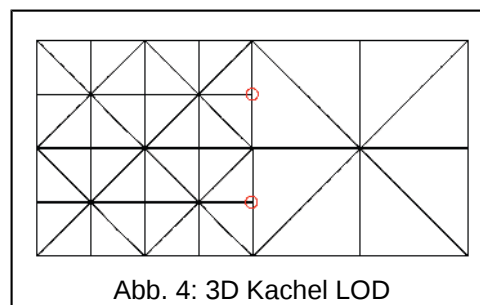
Zwecks Caching wurde eine Erweiterung für MapCache implementiert die die Aufbereitung von 3D Kacheln erlaubt.

## Rendering und Bereitstellung massiver Geodaten unter Verwendung von OpenWebGlobe und MapCache in der Cloud

### Generierung von 3D Kacheln



Mit entsprechenden Geometriedaten kann ein Globus der mit Bilddaten ausgestattet wurde mit Höhendaten erweitert werden vgl. Abb. 3.



Um diese Geometriedaten effizient rendern zu können, müssen die 3D Daten ähnlich den Bildkacheln in eine räumliche Datenstruktur wie dem Quadtree überführt werden um Level-Of-Detail (LOD) zu ermöglichen vgl. Abb. 4.

### Resultate

Die implementierte Lösung wird bereits eingesetzt (<http://world.openwebglobe.org>) und liefert dank der effizienten cachebasierten Aufbereitung ohne vorhergehende langwierige Datenprozessierung globale Datensätze bis zur Visualisierung. Zusätzlich zur Aufbereitung von Bilddaten können nun auch mit MapCache Kacheln mit 3D Geometrie gerechnet und gecacht werden.

Die Performance bei der Aufbereitung von 3D Geometrie ist vergleichbar mit der von Bilddaten. Das Precaching von 12 LOD Stufen (ca. 22 Millionen Kacheln) von ASTER Global Digital Elevation betrug 60'418 Sekunden auf einer 4 CPU-Kerne Rechnerinstanz von Amazon Webservices (<http://aws.amazon.com>) mit 8 GB Arbeitsspeicher. Daraus ergibt sich eine Schreibrate von etwa 370 Kacheln pro Se-

## Rendering und Bereitstellung massiver Geodaten unter Verwendung von OpenWebGlobe und MapCache in der Cloud

kunde. Aufgrund des hohen Anteils an „leeren“ Kacheln durch Ozeanflächen, müssen diese Zahlen mit Vorbehalt betrachtet werden.

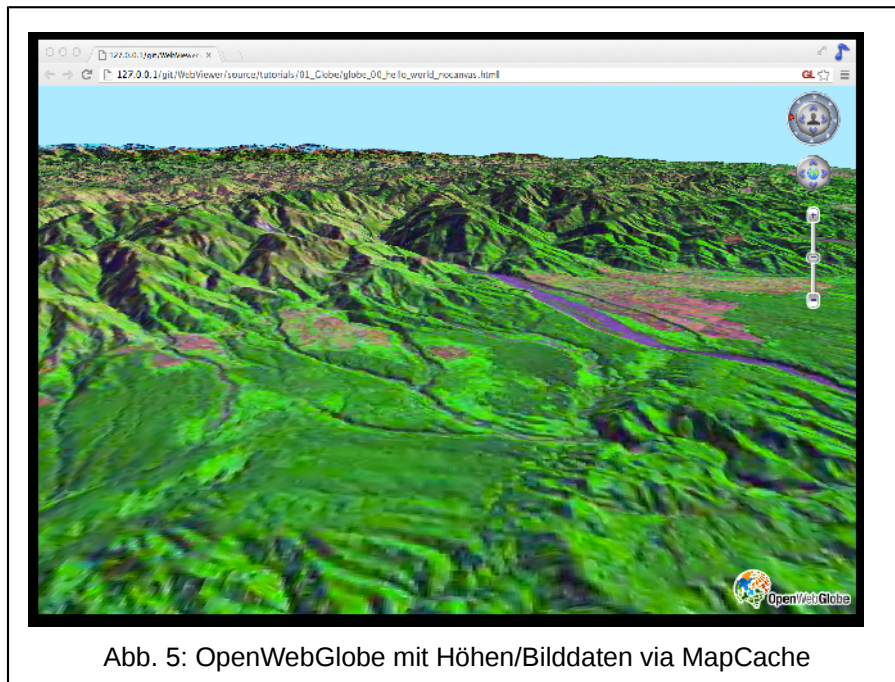


Abb. 5: OpenWebGlobe mit Höhen/Bilddaten via MapCache

Eine Erweiterung des Workflows mit dem OpenStreetMap (OSM) Rendering Workflow in Python [9] erlaubt es direkt aus einer PostgreSQL Datenbank Datenkacheln zu rendern und zu cachen. Eine kombinierte Darstellung von OSM Daten transparent über die Bildkacheln überlagert ist in Abb. 6 zu sehen.

### Fazit und Ausblick

OpenWebGlobe ist eine auf räumliche Daten fokussierte 3D Technologie und profitiert im Wesentlichen von seiner Flexibilität durch WebGL und bietet so eine plattformunabhängige Alternative zu anderen virtuellen Globen.

Durch einen verteilten Workflow zur Datenaufbereitung mit angeschlossenem Caching können massive Datensätze einfacher bereitgestellt und genutzt werden. Der Ansatz mit MapCache bietet die Möglichkeit neben den Bildkacheln auch 3D Daten in einer räumlichen Datenstruktur zu benutzen. Dies eröffnet neue Möglichkeiten für Streamingansätze von 3D Daten auf einem virtuellen Globus.

In Zukunft liegt der Fokus auf der Optimierung der Aufbereitung und dem Reduzieren von Redundanzen wie den homogenen Flächen die durch Ozeanen oder nicht kartiertem Gebiet entstehen.

Weiter wäre eine Integration von 3D Kacheln gemäss dem recht neuen Standard W3DS (Web 3D Service), der bei der OGC in Bearbeitung ist, denkbar.



## Rendering und Bereitstellung massiver Geodaten unter Verwendung von OpenWebGlobe und MapCache in der Cloud

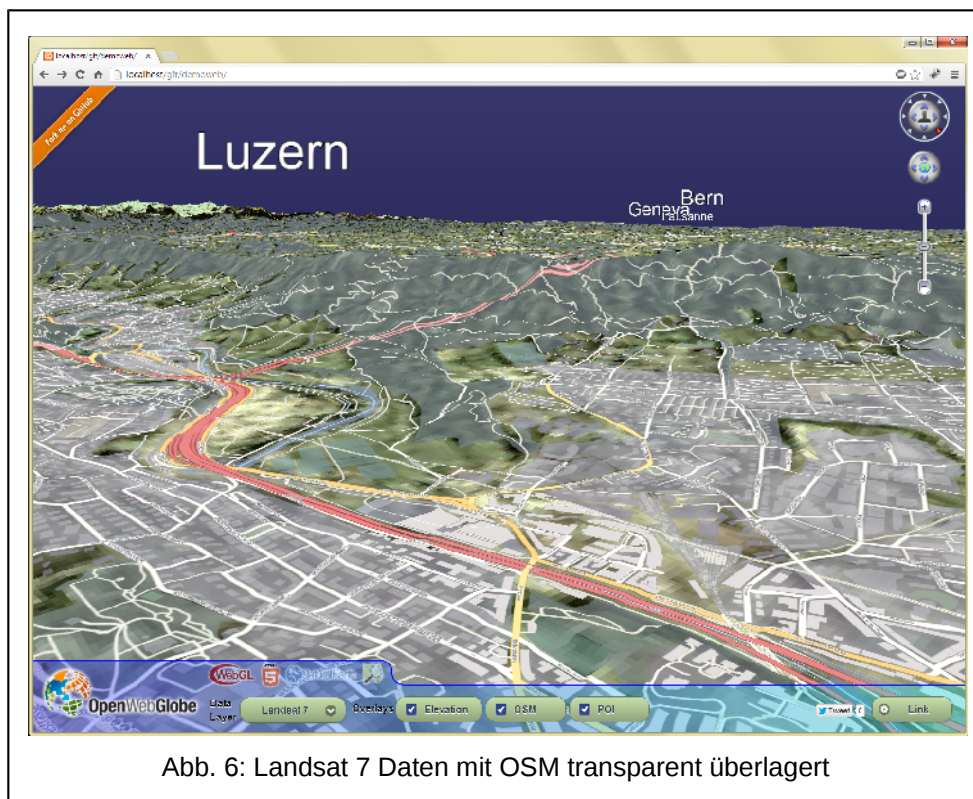


Abb. 6: Landsat 7 Daten mit OSM transparent überlagert

Kontakt zu den Autoren:

Robert Wüest, Martin Christen, Benjamin Loesch  
Fachhochschule Nordwestschweiz Institut Vermessung und Geoinformation  
Gründenstrasse 40  
4132 Muttenz  
(robert.wueest, martin.christen, benjamin.loesch)@fhnw.ch

### Literatur

- [1] G. Bell, A. Parisi, und M. Pesce, *The Virtual Reality Modeling Language*. 1995.
- [2] M. Reddy, L. Iverson, und Y. G. Leclerc, „Under the hood of GeoVRML 1.0“, in *Web3DVRML 2000 fifth symposium on Virtual reality modeling language*, 2000, S. 23–28.
- [3] C. Marrin, *WebGL Specification, Version 1.0*. 2011.
- [4] M. Christen und S. Nebiker, „i3D OpenGlobe SDK“, Jan-2011.
- [5] M. Christen und S. Nebiker, „OpenWebGlobe SDK - An Open Source High-Performance Virtual Globe SDK for Open Maps“, gehalten auf der State of the map 2011, 2011.
- [6] R. Pajarola, M. Antonijuan, und R. Lario, „{QuadTIN:} quadtree based triangulated irregular networks“, in *{VIS} '02: Proceedings of the conference on Visualization '02*, Washington, {DC}, {USA}, 2002, S. 395–402.
- [7] Amazon, „Amazon Simple Storage Service (Amazon S3)“, 2013. [Online]. Verfügbar unter: <http://aws.amazon.com/de/s3/>. [Zugegriffen: 23-Apr-2013].
- [8] M. Christen, *Processing Geodata using the OpenWebGlobe Tools*. 2012.
- [9] R. Wüest, *Paralleles Pre-processing und optimiertes Rendering globaler Openstreetmap-Daten in OpenWebGlobe*. FHNW - University of Applied Sciences Northwestern Switzerland, 2012.

## Schweizmobil Geodatawarehouse

### open source GIS und ERP

EMMANUEL BELO

Die Stiftung Schweizmobil bietet Informationen rund um Langsamverkehr in der Schweiz auf ihrer Webseite [www.schweizmobil.ch](http://www.schweizmobil.ch) an. Da fast sämtliche Informationen einen starken Raumbezug haben, werden die Daten neu in einer räumlichen Datenbank verwaltet., besteht bereits seit einiger Zeit eine enge Verknüpfung zur Webkarte, z.B. über ein API oder Direktlinks. Die Daten hinter der Webseite wurden aber bis jetzt über eine nicht-räumliche Datenbank verwaltet. Mit dem Projekt WebDB hat die Stiftung SchweizMobil das Ziel, die Datenbank-Infrastruktur komplett zu erneuern und die Vorteile einer räumlichen Datenbank verknüpft mit einem komplexen Datenverwaltungssystem zu nutzen. Die Daten werden in einer Postgres / PostGIS Datenbank abgelegt und über ein OpenObjects-basiertes Interface verwaltet. OpenObject bietet durch ihren Einsatz in BI-Software (OpenERP) weitgehende Möglichkeiten für die Verwaltung der SchweizMobil-Daten, insbesondere mit dem GeoEngine, welches die Visualisierung der geographischen Informationen in Karten ermöglicht. Das System ist seit Februar produktiv und erlaubt, die Daten auf der Schweizmobil-Webseite auf einfache Art und Weise aktuell zu halten.

Dieser Vortrag zeigt auf, wie Open Source Komponente aus der Geo- und ERP-Welt optimal zusammen eingesetzt werden können um das ganze Potenzial von Open Source Software zu nutzen.

## Semi-automatische Adressdatenerfassung

### Erfassung von georeferenzierten Adressdaten aus Fotos von Smartphones unter Einbezug von OSM

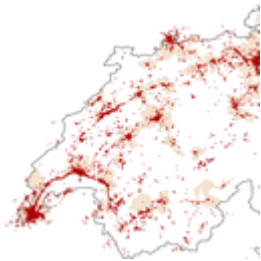
HANS-JÖRG STARK

Das Projekt OpenAddresses (OA) wurde im Jahr 2007 konzipiert, gestartet und seither permanent weiterentwickelt. OA ermöglicht das Erfassen von geokodierten Adressen über eine Web-Anwendung. Geokodierte Adressen sind ein wichtiger Bestandteil einer Geodateninfrastruktur und dienen zahlreichen Anwendungen als wichtige Ortsinformation. OA erlaubt die erfassten Daten kostenlos herunterzuladen und sie zu verwenden. OA verwendet als Hintergrunddaten OpenStreetMap (OSM) Daten und ist seit einiger Zeit bestrebt, eine möglichst aktuelle Synchronisation zur OSM-Datenbank herzustellen. Dieser Beitrag stellt eine Browseranwendung vor, die erstellt wurde, um Fotos von Smartphones für die semi-automatische Erfassung von geokodierten Adressen zu verarbeiten. Das Vorgehen ist dabei so, dass ein Anwender mit einem Smartphone oder einer Kamera mit entsprechender Sensorik Fotos von Gebäuden erstellt. Diese digitalen Bilder, werden anschliessend über die erwähnte Browseranwendung auf einen Server geladen und dort prozessiert. Dabei werden die erwähnten Informationen ausgelesen. Über Nominatim wird die Kameraposition zur Zeit der Aufnahme abgefragt und mithilfe der Information des Richtungssensors wird die geschätzte Position des aufgenommenen Gebäudes ermittelt. Die so erhaltenen Informationen werden in der Browseranwendung dargestellt: zum einen wird der Kamerastandort und die geschätzte Gebäudeposition in einer Karte dargestellt, zum andern werden die aus Nominatim erhaltenen Adressinformationen in einer Tabelle aufgeführt. Der Anwender kann nun einerseits im Kartenfenster die korrekte Adress- bzw. Gebäudeposition festlegen und andererseits die erhaltenen Adressinformationen falls nötig ändern, ergänzen oder bestätigen. Sind die Angaben vollständig und korrekt, werden die so erhobenen geokodierten Adressinformationen direkt in die Datenbank von OA eingetragen. Nebst der Browseranwendung wird im Beitrag auch auf die Qualität dieses Ansatzes und Risiken und Chancen eingegangen.

## Spatial Data Mining: Wenn, dann Open Source

### Einsatz von SpatiaLite, QGIS, R und D3.js zur Voranalyse und Visualisierung

TIMO GROSSENBACHER



Im Rahmen meiner Masterarbeit am Geographischen Institut der UZH bin ich damit beschäftigt, grosse Mengen an Standortdaten aus sozialen Netzwerken auf räumliche Muster zu untersuchen. Das Ziel wäre, **Mobilitätsmuster**, wie sie durch den Verkehrs-Mikrozensus regelmässig erhoben werden, in sozialen Netzwerken wiederzufinden.

Um dies zu ermöglichen, müssen die Punktdaten zuerst vorverarbeitet werden - von der Konvertierung in ein geeignetes Format über die Erkennung von Ausreissern bis zum raum-zeitlichen Clustering, verschiedenste Schritte mit unterschiedlichen Anforderungen sind involviert. Konventionelle GIS bieten für einen

solchen Prozess jedoch keine Allzwecklösung an, deshalb müssen verschiedene Technologien kombiniert werden.

**SpatiaLite** erlaubt mit seinen erweiterten SQL-Funktionen einfache deklarative Statistiken wie z.B. die Durchschnittsdistanz, die eine Person täglich zurücklegt. **QGIS** ermöglicht eine schnelle Sichtung und Kontrolle der Daten, während mit den räumlichen Packages in **R** erweiterte Methoden des Data Mining, wie z.B. Clustering, angewendet werden können. Dank den entsprechenden Python APIs können die Daten problemlos weitergereicht werden. Schliesslich bietet **D3.js** das Rüstzeug, raumzeitliche Punktdaten performant zu animieren.

Im Rahmen des Vortrags wird zuerst ein kleiner Einblick in meine Masterarbeit gegeben. Danach werden anhand von Beispielen die Einsatzgebiete der oben erwähnten Technologien erläutert. Hier wird insbesondere bei D3.js, das sich wachsender Popularität erfreut, auf die vielfältigen Möglichkeiten der Geovisualisierung eingegangen.

Der Vortrag hat schliesslich das Ziel, aufzuzeigen, wie offene Standards und offene Tools es überhaupt erst ermöglichen, hoch-spezifische Datenanalysen durchzuführen. Open Source füllt hier einen Nischenplatz, für den proprietäre Systeme schlichtweg nicht konzipiert sind.

Links:

<http://d3js.org/>

<http://www.gaia-gis.it/gaia-sins/>

<http://qgis.org/>

<http://www.r-project.org/>

## Stadtplan aus OSM-Daten

### Erstellung eines analogen grossformatigen Stadtplan aus OSM Daten am Beispiel von Basel

HANS-JÖRG STARK

Der eingereichte Beitrag hat zum Ziel, aus OpenStreetMap (OSM) Daten einen grossformatigen, analogen Stadtplan zu erstellen, der mit kommerziellen Printprodukten vergleichbar ist bzw. eine im Hinblick auf Datenlizenzen kostenlose Alternative darstellt. Diese Aufgabe wurde bereits in früheren Jahren von Frederik Ramm und anderen gelöst. Da unterschiedliche Werkzeuge für die Erstellung eines solchen Stadtplans verfügbar sind und diese sich weiterentwickeln bzw. teilweise auch stagnieren oder verschwinden, hat diese Arbeit zum Ziel, eine Empfehlung für den gegenwärtigen Zeitpunkt zu geben. In diesem Zusammenhang stellten sich unterschiedliche Forschungsfragen, die im Wesentlichen in zwei Teilen bearbeitet wurden und so präsentiert werden: In einem ersten Teil werden mit einer Vergleichsstudie verschiedene Opensource Werkzeuge für das Rendern von OSM Daten getestet. So wurde das Werkzeug ermittelt, welches sich am besten für das Rendern der Daten für einen analogen Stadtplan eignet. In einem zweiten Teil wurde schliesslich ein analoger Stadtplan von Basel mithilfe des eruierten Rendertools erstellt. Dabei wurde auch ein Strassenverzeichnis auf der Grundlage von OpenStreetMap Daten generiert und auf dem Stadtplan platziert. Am Schluss des Beitrags folgt ein Fazit und Empfehlungen.

## Tile based map publishing with WMTS TileServer, MapTiler an TileMill

PETR PRIDAL-KLOKAN

Fast online maps for web, mobile and desktop applications from a traditional Apache/PHP/MySQL web hosting or from the Amazon S3 cloud

Almost all mapping applications available today on the web and in mobile devices are in fact using tiles as the core for online distribution of maps. This presentation shortly explains the tiling basics and shows the open-source software typically used for generating and distributing tiled maps in the most popular Spherical Mercator system (EPSG:3857 / EPSG:900913) visible in Google Maps, OpenStreetMap, Bing Maps and other online maps. The tools for rendering maps such as the MapTiler (powered by GDAL/GDAL2Tiles) and TileMill (powered by Mapnik) will be demonstrated. The presentation also introduces a new open-source project called TileServer which allows to distribute rendered map tiles (directory or MBTiles) from any ordinary web server (Apache/MySQL) in the standardised OpenGIS WMTS format without a need to install, configure or maintain any additional dynamic software or web server extension. Any traditional web hosting can be therefore used for online distribution of maps. It is a rapid, extremely high-performance and scalable way how to publish maps. Hosting of the rendered maps is also possible from the cloud, such as Amazon S3/CloudFront or RackSpace Files. The published custom maps can be used in HTML5/JavaScript applications (OpenLayers/Leaflet), on mobile devices (iOS: MapKit/RouteMe, Android: OSMDroid), or even opened in desktop GIS systems such as QGIS, UDig and ESRI ArcGIS Desktop. This presentation contains a practical demonstration of map tile rendering and online publishing.

Links:

<http://www.maptiler.org/>

<http://www.maptiler.org/google-maps-coordinates-tile-bounds-projection/>

<http://mapbox.com/tilemill/docs/crashcourse/introduction/>

<https://github.com/klokantech/tileserver-php/>

## Umstellung auf bw. Einsatz von FOSSGIS in der Voralberger Landesverwaltung: Technische Betrachtung

### Eine Kombination aus Einsatz und Erfolg

NIKOLAUS BATLOGG

Die Umstellung auf FOSSGIS für zentrale Bereiche im GIS der Vorarlberger Landesverwaltung (Vo-GIS) erfolgte zwischen 2009 bis 2012. Die Notwendigkeit technische Rahmenbedingungen zu verbessern, Kenntnisse und Erfahrungen im Bereich OpenSource sowie eine neutrale Einstellung gegenüber freier Software, hat den Einsatz von FOSSGIS ermöglicht. Umgestellt wurden die Bereiche Webdienste, Datendownload und Standard Desktop GIS.

Es können folgende Eckpunkte zusammengefasst werden:

- Von zentraler Bedeutung ist ein offenes Datenformat.
- FOSSGIS war nicht vorgegeben, hat sich aber bewährt.
- CGI-Mapserver mit Linux als Betriebssystem ist performant und stabil. Linux als Serversystem kann zudem weitere Aufgaben übernehmen (z.B. SFTP Download, Reverse Proxy, Bandbreitenkontrolle).
- QGIS als Desktop GIS hat sich als richtige Wahl erwiesen. Derzeit gibt es ca. 400 Installation innerhalb der Landesverwaltung. QGIS ist stabil, sehr gut skalierbar und verfügt bereits über einen großen Funktionsumfang.
- Kostenersparnis durch Wegfallen der Lizenzkosten.
- FOSSGIS bedeutet auch Transparenz. Umfangreiche Hilfe kann im Internet zu praktisch fast allen Problemen gefunden werden.
- Freier Informationsaustausch: Unabhängige Datenformate und Software sind zukunftssträchtige Standards.
- FOSSGIS fördert den Mehrwert von Geoinformation. Durch die freie Verfügbarkeit der Anwendungen werden Anforderungen an die Verwaltung wie z.B. Open Government Data oder die Zusammenarbeit mit Dienstleistern ideal unterstützt.

## Von Vektordaten zum Rasterkartenwerk mit QGIS Server

ANDREAS SCHMID

Die Schweizer Kantone erstellen den sogenannten Basisplan der amtlichen Vermessung, ein Rasterkartenwerk in den Massstäben 1:5000 und 1:10'000, das automatisch aus den Vektordaten der amtlichen Vermessung generiert und regelmässig neu erstellt wird. Der Kanton Solothurn setzt hierzu QGIS Server und GDAL ein.

### Ausgangslage

Die Darstellung des Basisplans der amtlichen Vermessung ist vorgegeben, damit er über alle Kantone hinweg einheitlich ist. Die zu verwendenden Strichstärken, Symbolgrössen, Schriftgrössen usw. sind in Millimetern definiert.

Das Amt für Geoinformation des Kantons Solothurn setzt bereits FOSSGIS-Software ein, u.a. PostGIS und QGIS. Die Ausgangsdaten der amtlichen Vermessung, welche die Grundlagen für den Basisplan bilden, sind bereits in einer PostGIS-Datenbank vorhanden.

### Gewählte Lösung

Für das Rendering der Karten wurde QGIS Server gewählt, weil es so möglich ist, die Symbolisierung direkt in der Einheit, wie sie auf der fertigen Karte dargestellt werden soll (Millimeter), zu definieren. Eine Umrechnung in Pixel oder Meter ist damit nicht notwendig. Zudem ist das GUI von QGIS Desktop für die Erarbeitung der Symbolisierung sehr übersichtlich und praktisch in der Anwendung, besonders auch beim Aufbauen von komplexen Symbolisierungen.

Für die Endverarbeitung der von QGIS Server erzeugten Kartenbestandteile wird GDAL eingesetzt, weil es die notwendigen Werkzeuge bietet um die Daten in die gewünschte Form zu bringen. Mit diversen GDAL-Befehlen werden die Bestandteile der Karten so zusammengesetzt, einander überlagert, zugeschnitten und schlussendlich als GeoTIFF abgespeichert.

Der gesamte Ablauf der Erzeugung des Basisplans der amtlichen Vermessung ist in Abbildung 1 dargestellt.

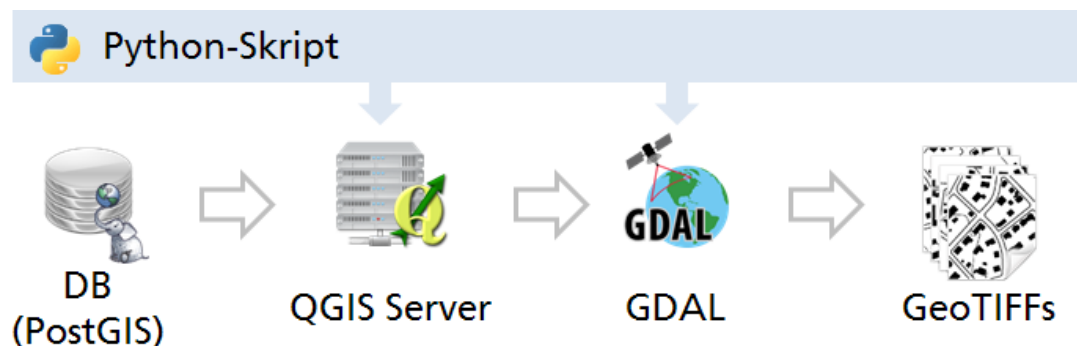


Abb. 1: Ablauf der Erzeugung des Basisplans der amtlichen Vermessung: QGIS Server generiert aus den Daten aus der PostGIS-Datenbank die Grundlagen der künftigen Kartenblätter. Mit GDAL werden diese weiterverarbeitet und als GeoTIFF abgespeichert. Den gesamten Ablauf steuert ein Python-Skript.

### Symbolisierung und Beschriftungen

Die Symbolisierung wurde in QGIS Desktop erarbeitet und als QGIS-Projekt abgespeichert. Auch komplexe Signaturen konnten mit dem GUI von QGIS sehr einfach aufgebaut werden. Die meisten



## Von Vektordaten zum Rasterkartenwerk mit QGIS Server

Strichstärken usw. konnten direkt in der Einheit, wie sie auf der fertigen Karte dargestellt werden sollen (Millimeter), eingegeben werden. Für einige Symbolisierungen mussten die Vorgaben dennoch in Meter umgerechnet werden, z.B. dort, wo Schriftmarkierungen zum Einsatz kommen.

Der Plan enthält Beschriftungen, die an den Blatträndern präzise zusammenpassen müssen. Um dies zu erreichen, musste bei Beschriftungen von Flächen die in der gegenwärtigen Entwicklerversion von QGIS neu verfügbare Option „Platzierung im Zentroid vom ganzen Polygon“ gewählt werden. So wird die Beschriftung immer am gleichen Ort platziert, auch wenn das Polygon am Rand des Kartenausschnitts liegt und nur teilweise sichtbar ist.

Für die Beschriftungen von Punkten wurde eine View erstellt, die auch die X- und Y-Koordinaten des Punktes als Attribute umfasst. Diese wurden bei den Beschriftungsoptionen als Positionierungskordinaten angegeben, womit die Beschriftungen ebenfalls immer am gleichen Ort platziert werden. (In der Version 2.0 von QGIS wird es auch für Punkte möglich sein, die Beschriftungen immer genau über dem Punkt zu platzieren, womit nicht mehr die datendefinierten Einstellungen verwendet werden müssen.)

Damit Beschriftungen, die auf Kartenblattränder fallen, vollständig gerendert werden, werden die Beschriftungslayer etwas grösser als die eigentlichen Kartenblätter erstellt und danach auf die Kartenblattgrösse zugeschnitten.

Die Anzahl der Gewässerbeschriftungen muss reduziert werden, weil zu viele Beschriftungspunkte in den Daten vorliegen. Dies wird realisiert, indem mit der PostGIS-Funktion `ST_SnapToGrid` ein Punktraster mit 500m Abstand erstellt wird und nur diejenigen Punkte beschriftet werden, die jeweils am nächsten bei einem dieser Rasterpunkte liegen. So wird eine einfache Generalisierung der Beschriftungen erreicht.

### Produktion von 1-Bit-PNGs

Der Basisplan wird auch in einer schwarz-weissen Variante erstellt, die keine Graustufen umfassen darf. Hierzu musste im Code von QGIS Server eine Anpassung vorgenommen werden, damit auch 1-Bit-PNG-Bilder erzeugt werden können. Die Umsetzung dieser Anpassung war erstaunlich einfach zu realisieren, weil sie analog zu den bereits bestehenden 8-Bit-PNGs gelöst werden konnte. Es mussten lediglich 7 Zeilen Code hinzugefügt und 2 Zeilen geändert werden.

### Python-Skript und GDAL-Befehle

Das Python-Skript ruft zunächst für jedes Kartenblatt verschiedene WMS-URLs (GetMap-Requests) auf, mit welchen von QGIS Server die Kartenbilder angefordert werden. QGIS Server rendert die Karte gemäss den Darstellungsdefinitionen im QGIS-Projekt und liefert sie im PNG-Format aus. Für jedes Kartenblatt, dessen Ausdehnungen das Skript einer Textdatei entnimmt, werden die Situation (in drei Teilen, damit die Dateien nicht allzu gross werden) und die Beschriftungen separat angefordert.

QGIS Server erlaubt im GetMap-Request die Angabe eines DPI-Werts. Davon wird Gebrauch gemacht, um damit indirekt den Massstab, in welchem die Karte gerendert werden soll, anzugeben. Mit folgender Formel wird aus der Kartenblattgrösse und dem gewählten DPI-Wert ermittelt, in welcher Grösse in Pixeln die Karte erstellt werden muss:

$$\text{Kartengrösse [Pixel]} = \text{Kartenblattgrösse [mm]} * (\text{DPI-Wert} / 25.4)$$

Mit diesen Angaben werden die Kartensignaturen exakt in der richtigen Grösse gezeichnet.

Nachdem die Basis-Bestandteile des Kartenblattes erstellt worden sind, ruft das Python-Skript nacheinander folgende GDAL-Befehle auf, um sie zum fertigen Kartenblatt zusammenzufügen:

- `gdalbuildvrt`: Zusammensetzen der drei Teilbilder der Situation zu einem Bild
- `gdalwarp`: Beschriftungen überlagern und gleichzeitig den Bildinhalt auf die Kantonsfläche clippen, als GeoTIFF abspeichern
- `gdal_translate`: Bild komprimieren
- `gdaladdo`: Overviews (verkleinerte Ansichten für kleinere Massstäbe) zum Bild hinzufügen

## Resultat

Mit dem beschriebenen Vorgehen konnte das Planwerk erfolgreich realisiert werden. Ein fertiges Kartenblatt ist im Massstab 1:5000 in Farbe im Durchschnitt rund 80MB gross, schwarz-weiss rund 4MB. Ein Beispiel, wie das Resultat aussieht, zeigt Abbildung 2.

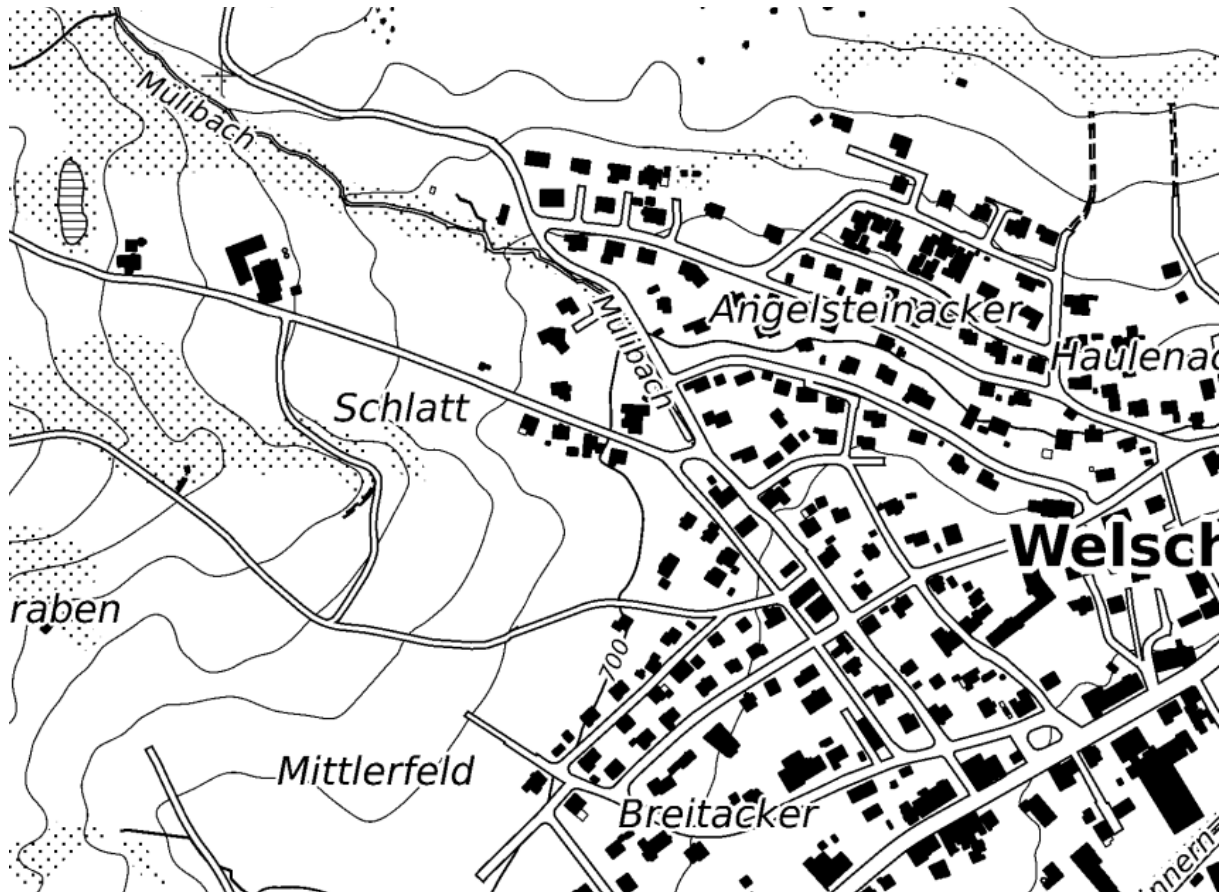


Abb. 2: Ausschnitt aus dem Basisplan der amtlichen Vermessung 1:10'000 schwarz-weiss

Kontakt zum Autor:

Andreas Schmid  
Kanton Solothurn, Amt für Geoinformation  
Rötistrasse 4, 4501 Solothurn, Schweiz  
+41 32 627 75 92  
andreas.schmid@bd.so.ch

## WebGL für die Geomatik

### Perspektiven im Geomatik Bereich

EMMANUEL BELO

Die Verbreitung der HTML5-Norm erlaubt es, neue Technologien zu einzusetzen. WebGL, als Teil der HTML5-Norm, ermöglicht es die Rechenstärke der Graphikkarte zu benutzen. Hoch performante 2D-Webmapping Anwendungen sollen dieses Prinzip nutzen. Als Beispiel bietet Google Maps/WebGL neue Funktionalitäten sowie eine verbesserte Ergonomie im Vergleich zur klassischen Anwendung. Im 3D Bereich, wo Berechnungen sehr aufwändig sind, kann man jetzt mit dieser neuen Technologie Web-Anwendungen entwickeln, die ohne Plugin in einem Browser laufen.

Diese Präsentation wird die WebGL Technologie vorstellen sowie ihre potentielle Einsatzmöglichkeit im WebGIS Bereich. Ausserdem werden Synergien zwischen 2D und 3D Webmapping-Anwendungen angedeutet.

## Wer ist der Boss bei OpenStreetMap?

### Meinungsbildung und Entscheidungsfindung in einem offenen Projekt

FERDERIK RAMM

Für Aussenstehende - und selbst für viele aktive Mapper - ist es nicht immer ganz klar, wohin sich das Schiff OpenStreetMap bewegt und wie man vielleicht ein kleines bisschen Einfluss auf den Kurs nehmen kann. Es gibt eine OpenStreetMap Foundation, Working Groups, Local Chapters, und eine riesige Community, die von Organisation relativ wenig wissen will. Es gibt Entwickler, die tun was sie wollen, und andere Entwickler, die im Auftrag von kommerziellen oder nichtkommerziellen Organisationen an OSM mitarbeiten.

Dieser Vortrag soll einen Überblick über die vorhandenen und fehlenden Entscheidungs- und Machtstrukturen im Projekt geben und wird auch ein paar Prognosen für die Zukunft wagen.

## Werkzeuge zur Nutzung von OpenStreetMap-Daten in GIS

STEFAN KELLER

Das Wikipedia-ähnliche OpenStreetMap-Projekt (OSM) liefert Geodaten mit grossem Detailreichtum. Die OSM-Daten ergänzen damit behördliche Geodaten u.a. mit Restaurants, Rastplätzen oder Aussichtspunkte. Doch viele wissen nicht, wie das OSM-Objektmodell aufgebaut ist, wie man OSM-Daten durchsucht oder können mit dem XML-Rohdatenformat von OpenStreetMap nichts anfangen.

Zunächst soll also demonstriert werden, wie man Daten findet. Dies geschieht v.a. mit dem OSM Wiki [1] und mit Taginfo [2].

Dann wird aufgezeigt, wie man ohne zu programmieren, ohne lokale Installation (und auch ohne Kommandozeilen-Werkzeuge) auf freie Geodaten-Quellen zugreifen oder sie herunterladen kann. Es sind dies u.a. der Desktop-Konverter OGR [3] und der freie Onlinedienst „GeoConverter“ [4], der selber auf OGR basiert. Beide konvertieren eine OSM-Datei in ein GIS-Format, so dass die Geodaten auch für diese Systeme zugänglich werden.

Ein spezielles Werkzeug ist OpenPOIMap [5], das (Schweizer) Points-of-Interest anbietet, die auch als WFS-Service bezogen werden können. WFS bedeutet Web Feature Service und ist ein internationaler Standard für den Zugriff auf Vektordaten [6]. Mit SpatiaLite-Tools [7] kann man u.a. Strassen- und Bahnnetzwerke importieren und analysieren.

Sämtliche gezeigten Software-Werkzeuge sind Open Source.

Am Schluss kommen auch frei zugängliche Webdienste wie das Overpass API [8] zur Sprache. Diese und weitere Infos werden laufend auf diesem Wiki gesammelt [9].

## Werkzeuge zur Nutzung von OpenStreetMap-Daten in GIS

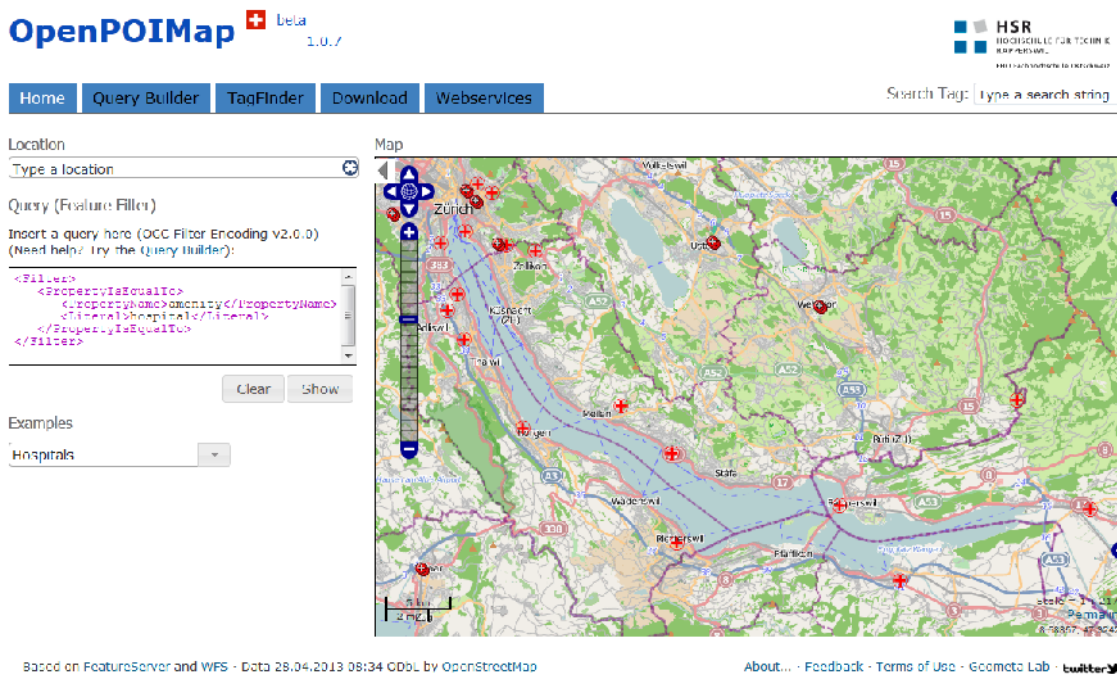


Abb. 1: OpenPOIMap – Datenbank für den freien Zugang zu POIs [5].

### Kontakt zum Autor:

Prof. Stefan Keller  
HSR Hochschule für Technik Rapperswil (FHO)  
Oberseestrasse 10, CH-8640 Rapperswil  
+41 55 222 47 46  
sfkeller@hsr.ch

### Literatur

- [1] OpenStreetMap Wiki: <http://wiki.openstreetmap.org/wiki/> . Zuletzt aufgerufen am 28.6.2013.
- [2] Taginfo: <http://taginfo.openstreetmap.org/> . Zuletzt aufgerufen am 28.6.2013.
- [3] OGR: <http://www.gdal.org/ogr/> . Zuletzt aufgerufen am 28.6.2013.
- [4] GeoConverter: <http://geoconverter.hsr.ch/> . Zuletzt aufgerufen am 28.6.2013.
- [5] OpenPOIMap: <http://openpoimap.ch/> . Zuletzt aufgerufen am 28.6.2013.
- [6] Web Feature Service (WFS) Spezifikation: [http://de.wikipedia.org/wiki/Web\\_Feature\\_Service](http://de.wikipedia.org/wiki/Web_Feature_Service) . Zuletzt aufgerufen am 28.6.2013.
- [7] SpatiaLite/SQLite-Datenbank: <http://www.gaia-gis.it/gaia-sins/> . Zuletzt aufgerufen am 28.6.2013.
- [8] Overpass API: [http://giswiki.hsr.ch/Overpass\\_API](http://giswiki.hsr.ch/Overpass_API) . Zuletzt aufgerufen am 28.6.2013.
- [9] „Nutzen von OpenStreetMap-Daten in GIS“: [http://giswiki.hsr.ch/Nutzen\\_von\\_OpenStreetMap-Daten\\_in\\_GIS](http://giswiki.hsr.ch/Nutzen_von_OpenStreetMap-Daten_in_GIS). Zuletzt aufgerufen am 28.6.2013.

## Wird das Geoportal der Verwaltung genutzt?: Metrik der Dienste von geo.admin.ch

### Spagat zwischen Feature Creep und Nutzererwartungen

DAVID OESCH

In den drei Jahren Betrieb haben sich die Dienste von geo.admin.ch, dem Geoportal des Bundes, bei weiten Teilen der Bevölkerung etabliert. Erwartungen der Endnutzer als auch die Anforderungen an das verwendete OpenSourceFramework und der Lizenzbedingungen der Daten waren und sind die Hauptherausforderung:

- Welche Funktionen/Dienste werden genutzt – und wo beginnt der feature creep?
- Welche Daten finden grösste Resonanz – und bei welche Geoinformationen besteht ein Bedarf?
- Fazit: FOSS im eGov Geoportal Bereich – Stärken und Schwächen

Mit echten, aktuellen Daten aus Nutzungsstatistiken und –umfragen von geo.admin.ch sind die quantitativen und qualitativen Grundlagen gegeben, um eine sinnvolle Diskussion bezüglich dem ROI von webmapping Diensten zu führen. Ein faszinierender Fundus an Daten erlaubt es, einen Blick darauf zu werfen, wie die Öffentlichkeit Geodaten im Web nutzt – und wo die Trends der FOSSGIS Gemeinschaft hingehen sollen:

- Webkarten sind für eine Mehrzahl der Web-Anwender auch 2013 eine Herausforderung. Bei der Informationsvermittlung (Suche und Resultatedarstellung) ist ein Paradigamawechsel gefordert: Weg vom Expertentool hin zur Suchmaschine
- Kartenportale der Verwaltung, eine Kurzvisite: Laden – betrachten – Seite verlassen: Der Umgang mit Daten der Verwaltung ist für viele Bürger/Endnutzer immer (noch) zu komplex. Einfache Lösungen, die konkrete Fragestellungen beantworten, sind zu bevorzugen.
- Von Daten zu Karten: Thematisch und kartographisch aufgearbeitete Karten generieren mehr Besucherverkehr als Rohdaten – sie eignen sich auch besser für die Indexierung in Suchmaschinen. Kommunikation der (Geo)daten soll Endnutzer in einfachen aggregierter Form geliefert werden.
- Auto-complete services garantieren ein positives User Experience: Vorschläge von Suchresultaten durch ein einziges Suchfenster sind ein Muss.
- Attributinfo: Nutzer klicken tatsächlich auf icons - Wenn das icon so aussieht wie ein Google Ballon Icon.
- Standardeinstellungen werden selten durch Nutzer geändert.
- Mobile – responsive design bei Kartenanwendungen ist gefragt
- Datendienste anstatt Datendownload: Geodaten haben meist kurze Nachführungszyklen. Offene Dienste, basierend auf offenen Standards vereinfachen updates.
- Datenlizenzstruktur: Herausforderung auf Seite Verwaltung ist die „Institutional jealousy“ : Die Tendenz von Datenherren, Informationen zurückzuhalten und die Zusammenarbeit mit anderen Organisation zu unterdrücken – um die volle alleinige Kontrolle über die Verwendung der Daten zu behalten. Die Lösungsansatz ist die Betroffenen zum data sharing / Kooperation zu motivieren

Auch wenn Technologie und die gesetzliche Grundlage noch so solide sind, individuellen Aspekte Anwender und Datenherr sind schwer greifbar. Nun hängt aber der Erfolg einer effektiven Nutzung von Geoinformation gerade von diesen Aspekten ab. Es geht also darum, institutionelle Widersprüche zu bereinigen, unter der Berücksichtigung, dass jede Institution seine Eigenheiten und Selbstständigkeit bewahren kann, ohne dabei das Ziel der Vernetzung der Datendienste aus den Augen zu verlieren.

### **Wird das Geoportal der Verwaltung genutzt?: Metrik der Dienste von geo.admin.ch**

Was das WebMapping für Verwaltungen benötigt: Weniger GIS, mehr nutzerfreundliche auto-complete Funktionen und Suchmaschinenoptimierung: Warum? Der Nutzer will, dass Suche und Repräsentation der Resultate für Kartenmaterial wie für den Rest des Web funktioniert.

Links: [www.geo.admin.ch](http://www.geo.admin.ch) /[map.geo.admin.ch/](http://map.geo.admin.ch/)

## Geo-Metadaten in OpenData Portalen

ARMIN RETTERATH



### Einleitung

In den letzten Jahren nahm die Verbreitung von OpenData Portalen auf Basis der Software CKAN immer mehr zu. Die von der OKF initiierte Software hat sich dabei zu einem De-facto-Standard entwickelt. Sowohl das OpenData Portal der Europäischen Union, als auch viele OpenData Portale der EU-Mitgliedstaaten und kleinerer Institutionen basieren auf dem in Python entwickelten Softwarepaket CKAN. Als RDBM-Backend hat sich in diesem Umfeld Postgres etabliert. Bei CKAN handelt es sich um ein Metainformationssystem mit einem speziell auf OpenData ausgerichteten Informationsmodell, das über eine "spatial" Extension auch als Geo-Metadatenkatalog genutzt werden kann.

Als das Land Rheinland-Pfalz Ende 2012 beschloss, ebenfalls ein eigenes OpenData Portal einzurichten, wurde zunächst nach möglichen Datenquellen recherchiert. Ziel war es, vorhandene Metadatenbestände - soweit möglich - automatisiert an das neue Portal anzuschließen. Als Datenquellen boten sich hier PortalU.rlp sowie das vollständig auf FOSS basierende GeoPortal.rlp an.

Das OpenData Portal des Landes sollte bis zur CeBIT 2013 frei geschaltet werden. Zunächst wurde versucht, das CSW 2.0.2 Interface von Geonetwork zu nutzen, um die Geo-Metadaten über die CKAN harvest-extension zu "ernten". Während des Versuchs zeigten sich aber starke konzeptionelle wie auch technische Schwächen dieser Vorgehensweise. Aus diesem Grund wurde der Ansatz zugunsten eines vom GeoPortal.rlp angestoßenen "push"-Verfahrens verworfen. Bei diesem Verfahren werden die Geo-Metadaten der im GeoPortal registrierten Webservices in das CKAN spezifische json Format transformiert und dann über die CKAN json API direkt in den OpenData Katalog überführt.

Mit der Umsetzung der Schnittstelle im März 2013 konnte in Deutschland erstmalig die Übertragung der Inhalte eines umfangreichen Geo-Metadatenkatalogs in ein OpenData Portal realisiert werden. Die Daten aus dem GeoPortal.rlp sind derzeit, sowohl im rheinland-pfälzischen **Open Government Data Portal** (<http://www.daten.rlp.de>) als auch unter <http://www.govdata.de> recherchierbar. Ein Anschluss ans Portal der Europäischen Kommission befindet sich in Vorbereitung.

### Modellierung von Geo-Metadaten in CKAN

Im Prinzip handelt es sich bei der Open Source Software CKAN um ein webbasiertes Metainformationssystem, das darauf ausgerichtet ist, Daten im Internet verfügbar zu machen. Mit CKAN sollen *freie* Datensätze veröffentlicht, geteilt und auffindbar gemacht werden. Besonders wichtig ist dabei die einfache Modellierung von Lizenzen. Der Nutzer eines OpenData Katalogs soll auf den ersten Blick sehen, welche Datensätze über freie Lizenzen - im Sinne von OpenData<sup>1</sup> - zur Verfügung stehen. Daher wurde beim Entwurf des Datenmodells besonderer Wert darauf gelegt, Lizenzen einheitlich abzubilden und auch die *Offenheit* der Lizenz über ein Attribut zu klassifizieren (*isopen*).

Das Datenmodell<sup>2</sup> selbst ähnelt im Grunde dem eines typischen Geo-Metainformationssystem wie beispielsweise geonetwork-opensource.

---

1 <http://opendefinition.org/okd/deutsch/>

2 Die den Untersuchungen zugrundeliegende Software basiert auf der Version 1.8.1b von CKAN. Änderungen des Datenmodells der neuen Versionen 2.0 bzw. 2.1 sind noch nicht berücksichtigt.



## Geo-Metadaten in OpenData Portalen

Es beinhaltet neben den Datensätzen (*packages*), auch Ressourcen (*resources*), Tags (*tag*), Nutzer (*user*) und Gruppen (*group*) Elemente.

Eine Besonderheit ist, dass es neben den Datensätzen die s.g. Ressourcen gibt. Diese sind unabhängige Verweise auf Zugriffspunkte, die den jeweiligen Datensatz in verschiedenen Formaten zur Verfügung stellen. Ein und derselbe Datensatz könnte damit z.B. sowohl als CSV wie auch als XML bereitstehen. Um diesen Zusammenhang abbilden zu können, werden die Relationen zwischen *packages* und *resources* in der Datenbank verwaltet.

Das Datenmodell ist dabei sehr flexibel und lässt sich über ein Extension-System (js *extras* Objekt) leicht an unterschiedliche Anforderungen anpassen.

Auszug aus einem CKAN Datensatz mit angegebener WMS Ressource (rheinland-pfälzisches OGD<sup>3</sup>):

```
{
  "maintainer": "VerbandsgemeindeGerolstein",
  "point_of_contact": "VerbandsgemeindeGerolstein",
  "id": "1e7454dc-8ca0-444b-be6e-db8c3a41ff7f",
  "metadata_created": "2013-03-01T12: 29: 32.204641",
  ...
  "resources": [
    {
      ...
      "package_id": "1e7454dc-8ca0-444b-be6e-db8c3a41ff7f",
      ...
      "id": "b2d4fa8d-8349-410b-8671-d1e06af2a7db",
      ...
      "description": "WMS Capabilities Link zur Integration in GIS oder Webapplikationen",
      "format": "WMS",
      "created": "2013-05-02T14:56:29.739776",
      "url": "http://www.geoportal.rlp.de/mapbender/php/wms.php?layer_id=35236&REQUEST=GetCapabilities&VERSION=1.1.1&SERVICE=WMS",
      ...
      "resource_type": "visualization"
    }
  ],
  "tags": [
    "Bauleitplan",
    ...
  ],
  "point_of_contact_address": {
    "email": "rathaus@gerolstein.de",
    ...
  },
  "groups": [
    "gdi-rp",
    ...
  ],
}
```

---

3 **OpenGovernmentDataPortal** – <http://www.daten.rlp.de> - <http://www.daten.rlp.de/api/rest/dataset/00158f6b-9ab8-44c4-58f3-c64d98a5c8e3>

## Geo-Metadaten in OpenData Portalen

```
"name": "00158f6b-9ab8-44c4-58f3-c64d98a5c8e3",
"isopen": false,
"url": null,
"ckan_url": "http://daten.rlp.de/dataset/00158f6b-9ab8-44c4-58f3-c64d98a5c8e3",
"notes": "Bebauungspl\u00e4ner der Stadt Gerolstein\r\nBebauungsplan\r\nKombinierte Satzung Gerolstein-Michelbach\r\nder Ortsgemeinde Gerolstein",
"title": "Gerolstein-Ebene: Kombinierte Satzung Gerolstein-Michelbach",
"maintainer_address": {
...
},
"terms_of_use": {
  "license_id": "dl-de-by-nc-1.0",
...
},
"extras": {
  "temporal_coverage_to": "2012-09-06 00:00:00",
...
},
"revision_id": "ca4f8aa5-c36e-4f4f-a5a7-09fb6db1c052"
}
```

Es gibt f\u00fcr CKAN eine Erweiterung<sup>4</sup>, die die zus\u00e4tzliche Verwaltung von Geometrien erlaubt. Dabei wird das Postgres Datenbankschema um die PostGIS Funktionen erweitert und CKAN erh\u00e4lt Plugins, die eine r\u00e4umliche Suche, das Anzeigen von WMS im Frontend (einfacher OpenLayers-Client) sowie eine CSW Schnittstelle implementieren.

### Weitere Besonderheiten von CKAN:

- Es gibt sowohl ein **name** als auch **id** Attribut, wobei **id** von der jeweiligen CKAN Instanz beim INSERT vergeben wird.
- CKAN verwaltet die Daten grunds\u00e4tzlich mit einer **Vollhistorie** – jede \u00c4nderung wird aufgezeichnet.
- F\u00fcr die Suche verwendet CKAN den **Apache Solr**<sup>5</sup> Index. Es gibt eine mit Konfigurationsdatei, die das Datenmodell auf den Index mapped.
- Das **group** Element werden sowohl f\u00fcr die Klassifizierung von Datens\u00e4tzen, als auch als Autorisierungselement (Gruppe/Organisation) verwendet.
- CKAN verf\u00fcgt \u00fcber drei verschiedene json **API's**<sup>6</sup>, die genutzt werden k\u00f6nnen, um den Katalog in beliebige CMS zu integrieren oder automatisch befüllen zu lassen.
- \u00dcber die Erweiterung **ckanext-harvest**<sup>7</sup> lassen sich grunds\u00e4tzlich auch Daten von CSW 2.0.2 API ISO 1.0 Interfaces (ISO19139 Metadaten) automatisiert in eine CKAN Instanz integrieren.

Zusammenfassend kann festgestellt werden, dass die Software CKAN *prinzipiell* dazu geeignet ist, um Geo-Metadaten zu verwalten. Ihre St\u00e4rken hat sie bei den einfachen API's, der Erweiterungsm\u00f6glichkeit des Datenmodells sowie der stark wachsenden Community im OpenData Umfeld. Auf die konzeptionellen und technischen Probleme, die sich bei genauerer Untersuchung gezeigt haben, wird sp\u00e4ter n\u00e4her eingegangen.

4 ckanext-spatial: <https://github.com/okfn/ckanext-spatial>

5 <http://lucene.apache.org/solr/>

6 <http://docs.ckan.org/en/ckan-1.8/api-v2.html#model-api>, <http://docs.ckan.org/en/ckan-1.8/apiv3.html>

7 <https://github.com/okfn/ckanext-harvest>

## Modellierung von standardisierten Geo-Metadaten

Im Gegensatz zum (Meta-)Datenmodell von CKAN geht es bei standardisierten Metadaten aus dem Geo-Umfeld um die Beschreibung von Geodaten und Geodiensten.

Hier hat sich mit der Einführung der OGC Catalogue Service IS 2.0.2<sup>8</sup> im Sommer 2007 und der Übernahme der Vorgabe des zugehörigen ISO Applikationsprofils<sup>9</sup> in das Metadaten Guidance Paper<sup>10</sup> für die Umsetzung der INSPIRE-Richtlinie einiges getan. Gerade die Vorgaben der EU haben dazu geführt, dass die Modellierung sukzessive vereinheitlicht wird. Durch die Anpassung der Softwaresysteme entsteht hier gerade ein De-facto-Standard, der sich momentan auch weltweit verbreitet.

Das Modell selbst ist in sich etwas komplex<sup>11</sup>. Es basiert grundsätzlich auf den Normen ISO19115, ISO19119 sowie ISO19139. Ein Datensatz und ein zugehöriger Dienst werden dabei mit zwei unterschiedlichen xml-codierten Metadatenätzen<sup>12</sup> beschrieben. Die Relation, die beim CKAN Modell im json Objekt implizit vorhanden ist, ist im Falle der Geo-Metadaten nur über gleiche Identifikatoren bzw. Links in den XML Dateien realisiert. Abbildung 6 verdeutlicht die Beziehungen der in Geodateninfrastrukturen verwendeten XML Dokumente untereinander.

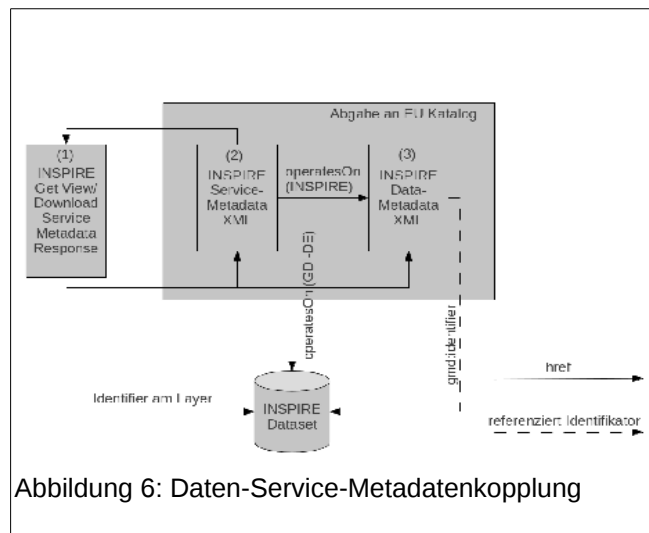


Abbildung 6: Daten-Service-Metadatenkopplung

Die konkrete Relation über den eindeutigen Datensatzidentifikator kann den folgenden Auszügen aus zwei gekoppelten INSPIRE Metadatenätzen entnommen werden<sup>13</sup>:

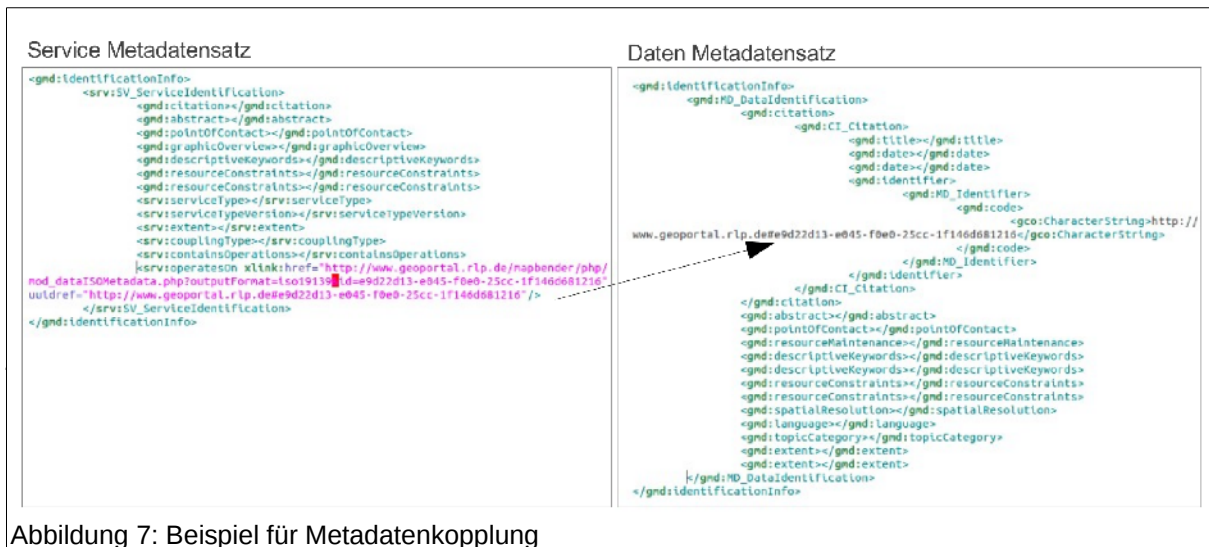


Abbildung 7: Beispiel für Metadatenkopplung

12 Beispiel für Naturschutzgebiete in RP:

**Service:** [http://www.geoportal.rlp.de/mapbender/php/mod\\_layerISOMetadata.php?SERVICE=WMS&outputFormat=iso19139&id=30825](http://www.geoportal.rlp.de/mapbender/php/mod_layerISOMetadata.php?SERVICE=WMS&outputFormat=iso19139&id=30825)

**SERVICE=WMS&outputFormat=iso19139&id=30825**

**Data:** [http://www.geoportal.rlp.de/mapbender/php/mod\\_dataISOMetadata.php?outputFormat=iso19139&id=e9d22d13-e045-f0e0-25cc-1f146d681216](http://www.geoportal.rlp.de/mapbender/php/mod_dataISOMetadata.php?outputFormat=iso19139&id=e9d22d13-e045-f0e0-25cc-1f146d681216)

**outputFormat=iso19139&id=e9d22d13-e045-f0e0-25cc-1f146d681216**

13 Hinweis: Das Beispiel beschreibt nicht die vom deutschen AK Metadaten empfohlene Verknüpfung. Dort wird empfohlen den Datensatzidentifikator im Attribut xlink:href abzulegen.

## Geo-Metadaten in OpenData Portalen

An dieser Stelle ist anzumerken, dass bei der Modellierung von Metadaten im Geo-Bereich beide Instanzen - sowohl der Datensatz als auch der Dienst – über eigene Angaben zu Nutzungsbedingungen und Zugriffsbeschränkungen verfügen. Außerdem sind die Lizenzen nicht harmonisiert. Im Umfeld von INSPIRE spricht man hier von **Constraints related to access and use**. Für das Mapping von INSPIRE konformen Geo-Metadaten in das CKAN Datenmodell sind insbesondere die s.g. **Conditions for applying for access and use** relevant. Diese werden aber grundsätzlich in Form von Freitext angegeben. Der XPath im INSPIRE Metadatensatz ist `identificationInfo[1]/*/resourceConstraints/*/useLimitation` und kann mehrfach vorkommen.

Zusammenfassend gilt hier:

- Geo-Metadaten liegen grundsätzlich in verschiedenen XML Dateien vor.
- Die Beziehung untereinander ist nur über einen Identifikator, bzw. bei INSPIRE eine zusätzliche URL vom Service-Metadatensatz auf den Daten-Metadatensatz realisiert.
- Die Nutzungsbedingungen liegen i.A. nur als Freitext vor.
- Die Nutzungsbedingungen für Daten und Dienste sind i.d.R. unterschiedlich.

## Mapping der Datenmodelle und Synchronisierung

### Erste Versuche

Um nun ausgewählte Geo-Metadaten der GDI-RP in das OpenData Portal des Landes zu überführen, wurde zunächst ein CKAN in der Version 1.8.1b, wie er auch für das Produktivsystem verwendet werden sollte, lokal installiert. Man ging davon aus, dass sich über die für das <http://data.gov.uk> Projekt entwickelte CSW-Schnittstelle (Teil von *ckanext-harvest*), ISO19139 Metadatensätze automatisiert in den CKAN überführen lassen würden.

Die GDI-RP betreibt für die Abgabe der Metadaten an den Geodatenkatalog.DE und damit auch an INSPIRE eine *geonetwork-opensource* Instanz<sup>14</sup>, die über eine CSW-Schnittstelle verfügt und über die schon seit mehreren Jahren INSPIRE-konforme Metadaten abgegeben werden.

Beim Versuch auf diese produktive Schnittstelle zuzugreifen, zeigte sich, dass das CSW-Harvestingverfahren von CKAN prinzipiell funktionsfähig ist. Leider waren die im CKAN Quelltext verankerten Funktionen zur Überführung der Datenmodelle zu sehr an das britische Metadatenprofil angepasst, was zu Fehlern beim Mapping führte. Der Harvestingprozess dauerte für 3500 Datensätze mehrere Stunden. Die Daten wurden aufgrund des beschriebenen fehlerhaften *fetching-process* (Validierung & Modelltransformation) nicht in der CKAN Oberfläche angezeigt.

Um die Tests zu beschleunigen, wurde eine lokale *geonetwork-opensource* Instanz installiert und verschiedene Beispieldateien eingespielt. Dieser Katalog konnte in einem vernünftigen Zeitrahmen vom CKAN Harvester ausgelesen werden, jedoch kam es beim Mapping der Datenmodelle immer noch zu Fehlern. Die Idee war ursprünglich, die ISO19139 Daten sukzessive so anzupassen, dass der CKAN Harvester diese anstandslos übernehmen würde. Da aber im Github zu erkennen war, dass das CSW Harvesting wohl noch *work in progress* ist und man außerdem das Problem mit den Freitext-Lizenzfeldern mit einem von CKAN ausgelösten Harvesting-Prozess nicht in den Griff bekommt, musste nach einer anderen Lösung gesucht werden.

---

<sup>14</sup> <http://www.gdi-rp-dienste.rlp.de/geonetwork/>

### Umsetzungskonzept für Rheinland-Pfalz

#### Grundlagen

In Rheinland-Pfalz basiert die Architektur der GDI, im Gegensatz zu den meisten anderen deutschen Bundesländern, nicht auf der Verwendung eines herkömmlichen Geo-Metadatenkatalogs. Das Konzept für die GDI-RP wurde im Jahr 2005 zu einem Zeitpunkt erstellt, zu dem es noch keine funktionsfähigen Standards für den Austausch von Geo-Metadaten zu Daten und Diensten gab. Es basiert auf dem Prinzip der Extraktion und des Caching von Dienst- und Daten-Metadaten aus OGC Capabilities Dokumenten und erlaubt Anbietern von OWS, diese Metadaten in einem zentralen *Metadatenproxy* zu editieren sowie mit weiteren Informationen anzureichern. Die für die INSPIRE Umsetzung und den Aufbau der GDI-DE benötigten ISO19139 Metadaten werden dabei grundsätzlich *on the fly* bzw. *on demand* aus dem internen Datenmodell erzeugt.

#### Harmonisierung der Lizenzen

Das zentrale Informationsmodell wurde, seit der Inbetriebnahme 2007, immer wieder an die Anforderungen der Praxis angepasst. Eine dieser Anforderungen war es, die veröffentlichten Dienste im Geo-Portal.rlp mit vordefinierten Lizenzbedingungen derart anzureichern, dass ein Nutzer nach *freien* Datenquellen recherchieren kann<sup>15</sup>. Dieses Verfahren selbst wurde jedoch bis Anfang 2013 nur sporadisch genutzt; die meisten Anbieter blieben bei der Bereitstellung eigener Nutzungsbedingungen über die schon zuvor erläuterten Freitext-Felder.

---

<sup>15</sup> OpenData war 2008 noch nicht im Fokus der Politik.

## Geo-Metadaten in OpenData Portalen

Im Zusammenhang mit dem Austausch von Metadaten zwischen Geodateninfrastrukturen und Open-Data Portalen, war das o.g. Konzept jedoch zielführend. Hier besteht der Bedarf, eindeutig lizenzierte Daten automatisiert an andere Plattformen zu übertragen. Die Verantwortung für die Angabe von Nutzungsbedingungen liegt beim Datenanbieter selbst. Das Datenmodell der GDI-RP wurde im Februar 2013 dahingehend erweitert, dass – ähnlich wie beim CKAN – die im Portal hinterlegten Lizenzen zusätzlich ein *isopen* Attribut erhielten. Nach der Umsetzung der neuen Möglichkeit im Backend wurden die Anbieter von Diensten über die Änderungen informiert und darauf hingewiesen, dass im Falle der Zuordnung einer von vier abgestimmten Lizenzen, ihre Daten dann auch im OpenData Portal des Landes auffindbar seien. Diese Möglichkeit wurde dankend angenommen und viele Anbieter stellten daraufhin ihre Daten unter eine weniger restriktive Lizenz.

## Übertragung der Metadaten

Für die Übertragung der Metadaten aus dem GeoPortal.rlp in das rheinland-pfälzische OpenData Portal wurde der Weg über die CKAN API gewählt. Das Verfahren basiert darauf, dass die Geo-Metadaten des Informationsmodells der GDI-RP zunächst in das CKAN Datenmodell (json - UTF8) transformiert<sup>16</sup> und dann über insert, update und delete Funktionen der CKAN API verwaltet werden. Die CKAN verfügt über drei verschiedene API's<sup>17</sup>, die alle auf den einfachen http Methoden GET, POST, PUSH basieren und deren Parameter und Rückgabewerte in den meisten Fällen json-Objekte sind.

- API 1 – Ansprechen von Objekten über *id*, *RESTful*
- API 2 – Ansprechen von Objekten über *name*, *RESTful*
- API 3 (action API), auch Rückgabe von *results*, *errors* und *help* Informationen

Für diese API gibt es Clients in verschiedenen Programmiersprachen. Da die dem GeoPortal.rlp zugrundeliegende Software *mapbender* auf php basiert, wurde ein unter MIT Lizenz stehende Client<sup>18</sup> adaptiert und als Klasse in den mapbender Funktionsumfang integriert.

Grundsätzlich war der Aufwand bei der Implementierung der API überschaubar. Probleme können jedoch auftauchen, wenn im OpenData Portal Anpassungen am CKAN Quelltext vorgenommen werden. Leider ist das in Rheinland-Pfalz passiert und hat dazu geführt, dass zur Synchronisierung der Datensätze alle 3 API's verwendet werden müssen.

Der Synchronisierungsprozess aktualisiert momentan alle Datensätze des GeoPortal.rlp, gleichgültig ob diese sich geändert haben oder nicht. In Zukunft ist es vorgesehen, dass hier vor einem *update* ein Vergleich von timestamps erfolgt.

Die Erzeugung eines CKAN Datensatzes aus dem GeoPortal.rlp Informationsmodell kann grundsätzlich über zwei verschiedene Ansätze erreicht werden:

16 [http://www.geoportal.rlp.de/mapbender/php/mod\\_exportMapbenderLayer2CkanObjects.php](http://www.geoportal.rlp.de/mapbender/php/mod_exportMapbenderLayer2CkanObjects.php)

17 <http://docs.ckan.org/en/ckan-1.8/api.html>

18 [https://github.com/jeffreybarke/Ckan\\_client-PHP](https://github.com/jeffreybarke/Ckan_client-PHP)

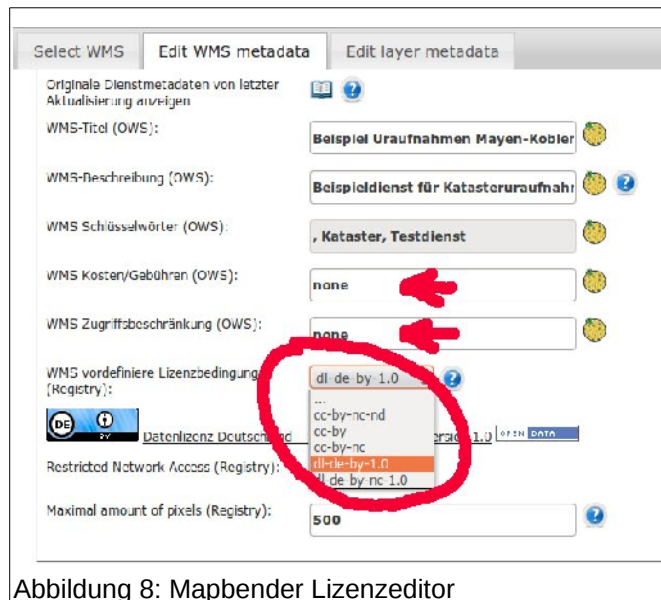


Abbildung 8: Mapbender Lizenzeditor

## Geo-Metadaten in OpenData Portalen

1. Erzeugung eines CKAN *packages* aus einem Geo-Daten-Metadatenatz. Anschliessend müssen alle verknüpften Geo-Service-Metadatenätze in jeweils ein CKAN *resource* Objekt überführt und an das *package* angehängt werden.
2. Erstellung eines CKAN *packages* für ein WMS Layer Objekt und Anreicherung von Ressourcen für verschiedene Zugriffsoptionen auf dieses Layer Objekt.

Grundsätzlich ist der erste Weg zu bevorzugen. Es gibt aber insbesondere drei Gründe die das verhindern:

- Die Daten-Service-Metadatenkopplung ist nicht für alle im GeoPortal.rlp registrierten Dienste vorhanden. Nur für die INSPIRE Daten ist sie verpflichtend (75 von 4200 Datensätze!). Für die anderen Ressourcen müsste sie erst durch die Datenanbieter generiert werden.
- Das Geo-Metadatenmodell sieht unterschiedliche Nutzungsbedingungen auf Datensatz- und Diensteebene vor. Diese lassen sich nicht in das CKAN Datenmodell abbilden.
- Eine saubere Bereitstellung von Nutzungsbedingungen auf Ebene des Datensatzes ist derzeit nicht möglich. Hier fehlt noch eine Abstimmung, wie man klassifizierte Lizenzbedingungen in ISO19139 Dateien abbildet (das sollte dann aber weltweit einheitlich erfolgen).

## Ergebnisse

Mit der beschriebenen Vorgehensweise konnte in Deutschland *erstmalig* eine automatisierte Übertragung von Geo-Metadaten aus einem Länderkatalog in ein OpenData Portal realisiert werden. Damit war keine redundante Neuerfassung von Metadaten über HTML-Frontends oder mit Hilfe von Excel-Tabellen mehr notwendig.

Der Weg, der dabei beschritten wurde, erlaubt es zusätzlich auch weitere Nutzungsmöglichkeiten (*resources*) im OpenData Portal zur generieren, z.B.:

- Direkter Aufruf eines Kartenviewers mit Anzeige der jeweiligen Datenquelle
- Verlinkung der speziellen Geo-Metadatenanzeige des GeoPortal.rlp:



Außerdem wurde ein großer Schritt in Richtung mehr Transparenz bezüglich der Nutzungsbedingungen von Geodatendiensten geschaffen. Es wird demnächst möglich sein auch die Frage „*zeige mir nur alle frei zugänglichen Datenquellen an*“ zu beantworten.

Die aktive Synchronisierung des CKAN über die API ermöglicht hochaktuelle Metadaten – im Falle von Änderungen könnten update-Prozesse getriggert werden.

Aktuell stammen 788 von insgesamt 1503 Datensätzen im OpenData Portal Rheinland-Pfalz aus dem GeoPortal.rlp!

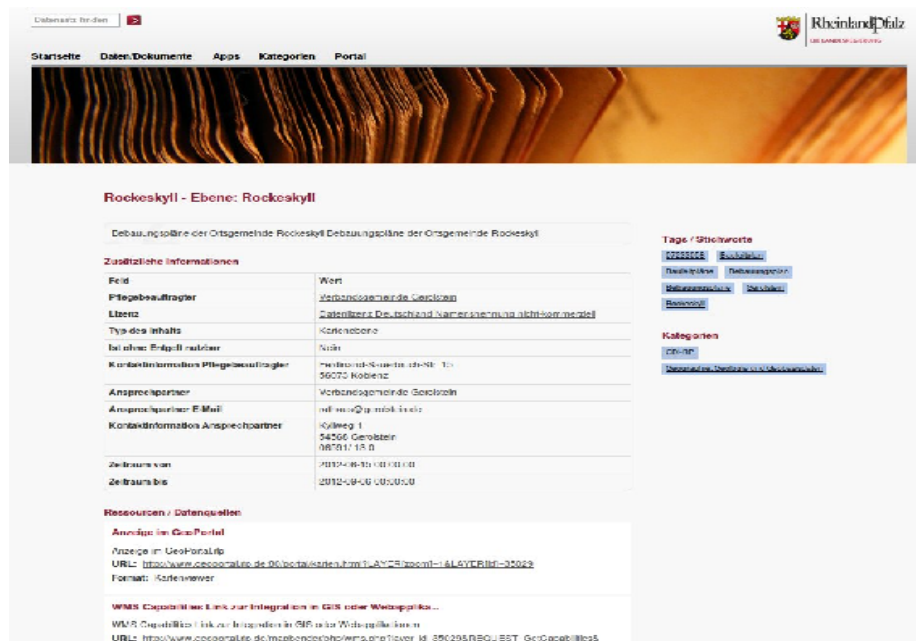


Abbildung 9: Metadaten zu einem WMS Layer Bebauungsplan im OGD RP



### Lessons learned / Pitfalls

#### Meta-Datenmodelle

Eine saubere Überführung des standardisierten Geo-Metadatenmodells, wie es z.B. von INSPIRE gefordert wird, ist in das CKAN Modell aus folgenden Gründen derzeit **nicht** ohne weiteres möglich:

- Es gibt unterschiedliche Geo-Metadatenprofile, die einer Entwicklung eines einheitlichen Harvesting-Verfahrens im Wege stehen.
- Das Vorhandensein unterschiedlicher, nicht maschinell auswertbarer Nutzungsbedingungen auf Daten- und Diensteebene der Geo-Metadaten.
- Fehlerhafte und/oder nicht vorhandene Relationen zwischen Daten- und Dienst-Metadaten auf Seite der Geodatenkataloge.
- Fehlendes einheitliches Mapping zwischen ISO Kategorien und CKAN *group* Elementen.

#### Technik

- Das automatisierte Update schon kleiner Datenmengen über die CKAN API (ein paar hundert Datensätze) kann durch inkrementelle Fortführung des *apache-solr* zu Speicherüberläufen führen.<sup>19</sup>
- Es ist wichtig, bei der Synchronisierung eindeutige Datensatzidentifikatoren auf der Seite des führenden Systems zu haben, die dann als CKAN *name* genutzt werden.
- CKAN Objekte können nicht gelöscht werden, sondern bekommen den *status*: „*deleted*“. Danach ist es nicht mehr änderbar! Das bedeutet, dass man um ein CKAN Objekt aus der Suche zu entfernen und später wieder mit gleichem *name* anzulegen, dessen *name* ändern muss (bspw.: *new\_name: old\_name\_timestamp*).

#### Sonstiges

Die OpenData Informationen gelangen auf vielen Wegen in die CKAN Kataloge. Teils werden sie von Hand eingestellt, teils über automatisierte Schnittstellen generiert. Es kann z.B. vorkommen, dass eine Stelle Metadaten für ein Online-Auskunftssystem, das auf einer Datenbank arbeitet über ein HTML-Frontend erfasst und auf der anderen Seite Metadaten für einen WMS Dienst, der auf dieselbe Datenbank zugreift, über eine automatisierte Schnittstelle in das OpenData Portal übertragen werden. Aus Sicht des OpenData Portals gibt es jetzt zwei *package* Objekte mit unterschiedlichen *id* und *name* Werten. Das ist natürlich unsinnig. Eine Lösung könnte sein, dass Geo-Metadaten ein „*genuine\_name*“ Feld bekommen, in das dann eine UUID eingetragen werden kann. Diese kann von einer ganz anderen Stelle vergeben worden sein. Das Synchronisierungsverfahren könnte aber dann bei Bedarf nach einem Datensatz mit diesem *name* suchen und die verknüpften Dienste dort als *resource* Objekte anhängen.

---

<sup>19</sup> Hängt ggf. auch mit der Vollhistorie des CKAN zusammen.

### Fazit

Man muss feststellen, dass es beim Versuch des automatischen Abgleichs von Geo-Metadaten mit OpenData Katalogen auf CKAN Basis einige semantische, inhaltliche und technische Probleme gibt, die bisher noch nicht gelöst sind. Manche der Probleme lassen sich durch pragmatische Ansätze umgehen, andere müssen wahrscheinlich auf der Ebene der Standardisierung angegangen werden.

Um den aktuellen politischen OpenData Prozess voranzubringen und damit mehr und mehr Daten *frei* zugänglich zu machen, ist es notwendig, die Datenmodelle bestehender Metainformationssysteme und OpenData Kataloge abzugleichen und deren Inhalte automatisiert zu synchronisieren. Das gilt natürlich nicht nur für Geo-Metadaten sondern auch für allgemeine Metadatenkataloge, wie z.B. das Umweltinformationssystem PortalU<sup>20</sup>.

Bei den Anbietern von Daten muss weiterhin das Bewusstsein geschaffen werden, dass man einen wirklichen volkswirtschaftlichen Mehrwert nur dann erhält, wenn die Hürden für die Nutzung der Daten so niedrig wie möglich sind.

Die Integration bestehender Geodateninfrastrukturen in den OpenData Prozess bietet hier Chancen, die genutzt werden sollten.

Kontakt zum Autor:

Armin Retterath  
Zentrale Stelle Geodateninfrastruktur Rheinland-Pfalz  
LVerGeo RP  
Ferdinand-Sauerbruch-Straße 15  
56073 Koblenz  
0261/492-466  
armin.retterath@lvermgeo.rlp.de

Weitere Informationen im Netz

- [1] <http://www.geoportal.rlp.de>
- [2] <http://ckan.org/>
- [3] <http://www.daten.rlp.de>
- [3] <http://www.govdata.de>
- [4] <http://publicdata.eu>
- [5] <http://inspire.jrc.ec.europa.eu>
- [6] <http://lucene.apache.org/solr/>
- [7] <http://www.opengeospatial.org/>
- [8] <http://okfn.org/>
- [9] [http://www.mapbender.org/Handbuch\\_Geoportal](http://www.mapbender.org/Handbuch_Geoportal)

---

<sup>20</sup> <http://www.portalu.de>

OSGeo-Live rocks!

## OSGeo-Live rocks!

ASTRID EMDE



OSGeo-Live ist ein OpenSource-Projekt, das mehr als 50 Softwareprojekte aus dem Bereich FOSS+GIS bündelt. Hier findet sich eine Sammlung aus den Bereichen Web Mapping Clients und Server, DesktopGIS, Datenbanken, Krisenmanagement, räumliche Tools und Bibliotheken sowie Daten.

Jedes halbe Jahr wird eine neue Version von OSGeo-Live erstellt, die dem Nutzer die verschiedenen Anwendungen, Daten und Informationen als sortierte Sammlung anbietet. Der Einstieg wird leicht gemacht, da die Installation und Konfiguration der Software entfällt. Alles liegt bereits vor und kann direkt getestet werden. Dokumentationen und Quickstart-Dokumente helfen dabei beim Einstieg.

OSGeo-Live soll als gutes Beispiel für ein erfolgreiches OpenSource-Projekt vorgestellt werden. Projektübergreifend wurden viele Freiwillige gefunden, die regelmäßig die Inhalte aktualisieren. So ist ein Produkt entstanden, das als globale Visitenkarte nicht nur der OSGeo-Projekte dient.



OSGeo-Live kann in Workshops und eigenen Veranstaltungen verwendet werden. Die FOSS- und GIS-relevante Software wird mehrsprachig und mit Dokumentation zur Verfügung gestellt. Auch zu den verschiedenen OGC Standards gibt es Informationen. Die Dokumentation liegt mittlerweile in zwölf Sprachen vor (englisch, deutsch, italienisch, polnisch, griechisch, japanisch, französisch, catalanisch, chine-

## OSGeo-Live rocks!

sich, koreanisch, russisch). Mit jeder neuen Version kommt neue Software und neue Übersetzungen hinzu.

Gerade für FOSS-GIS-Neueinsteiger bietet OSGeo-Live einen guten Überblick. Wenn Ihnen keine Geodaten zum Testen zur Verfügung stehen, können die enthaltenen Beispieldaten von Natural Earth und OpenStreetMap benutzt werden.

Dieser Vortrag liefert Einblicke in die Entstehung und den Aufbau des Projektes. Die ehrgeizigen Ziele wie mehrsprachige Dokumentation, Benutzung von Beispieldatensätzen und Support in der Community stellen ganz unterschiedliche Anforderungen dar. Es gilt nicht nur technische Probleme zu lösen. Hinzu kommen terminliche Absprachen, damit zu bestimmten Anlässen wie wichtigen Konferenzen und Workshops aktuelle Versionen bereit stehen.

### Aufbau

Durch das OSGeo-Live Projekt werden bootfähige ISO-Images als auch virtuelle Maschinen zur Verfügung gestellt. Das gesamte System basiert auf dem Ubuntu-Derivat Xubuntu. Darin enthalten sind vollständig installierte und konfigurierte Anwendungen aus der gesamten GIS- Welt. Es muss nichts lokal installiert werden, der benutzte Rechner wird in keiner Weise verändert. Die ISO-Images können entweder auf DVD-Rohlinge gebrannt, als virtuelles DVD-Laufwerk für virtuelle Maschinen oder per USB-Stick benutzt werden. Mit Hilfe eines USB- Sticks können auch Daten geladen bzw. gespeichert werden, um sie in einer folgenden Sitzung weiter benutzen zu können.

Jedes Projekt bietet eine Dokumentation. Dazu gehört eine Übersicht, ein Quickstart und ggf. die Originaldokumentation des Projektes. Die Übersicht enthält eine kurze Beschreibung des Projektes, Angaben zu den Features, benutzte Lizenz sowie Bezugsquellen und Links zur Projektseite. Mit Hilfe der Quickstarts werden dem Benutzer die ersten Schritte mit dem jeweiligen Projekt vereinfacht.

Folgende Komponenten finden sich auf OSGeo-Live 6.5:

### Desktop GIS

- Quantum GIS
- GRASS GIS
- gvSIG Desktop
- User-friendly Desktop Internet GIS (uDig)
- Kosmo Desktop
- OpenJUMP GIS
- Saga

### Browser Clients

- OpenLayers - Free Maps for the Web
- Geomajas - GIS framework
- Mapbender - Geo-Portal-Lösung
- MapFish - framework for building rich web-mapping applications
- Cartaro- Geospatial CMS

### Internet Dienste

## OSGeo-Live rocks!

- GeoServer
- MapServer
- deegree
- EOxServer - Web Coverage Service
- GeoNetwork - Metadata Katalog und Katalogdienst für das Web
- pycsw - Metadata Katalog
- MapProxy - Proxy für WMS und Tile-Dienste
- QGIS Server -
- 52°North WSS - Web Security Service
- 52°North WPS - Web Processing Service
- 52°North SOS - Sensor Observation Service
- TinyOWS - WFS-T Service
- ZOO-Project - Web Processing Service

## Datenbanken

- PostGIS - Räumliche Datenbank
- SpatiaLite - Leichtgewichtige Datenbank
- Rasdaman - Multi-Dimensionale Datenbank für Rasterdaten
- pgRouting - Routing für PostGIS

## Navigation und Karten

- GpsDrive - GPS Navigation
- GpsPrune - Anzeige, Bearbeitung und Konvertierung von GPS Tracks
- OpenCPN - Darstellung von Seekarten und GPS
- OpenStreetMap - Open Street Map Werkzeuge
- Viking - GPS Datenanalyse und -anzeige

## Spezielle GIS Software

- GeoKettle - ETL Tool (Extrahieren, Transformieren und Laden)
- GMT - Kartographisches Rendering
- Mapnik - Kartographisches Rendering
- TileMill - Styling and Publishing
- MapTiler - Erzeuge Bildkacheln (Tiles)
- OSSIM Image Processing
- OTB - Bildprozessierung
- R Spatial Task View - Statistische Berechnungen

## OSGeo-Live rocks!

### GIS Werkzeuge

- Sahana Eden - Katastrophenmanagement
- Ushahidi - Kartendarstellung und Zeitachsen für Ereignisse
- osgEarth - 3D Terrain Rendering
- MB-System - Sea Floor Mapping
- Marble - Spinning Globe
- zyGrib - Wettervorhersagekarten

### Daten

- Natural Earth - Globale Daten
- OSGeo North Carolina, USA Schulungsdatensatz
- OpenStreetMap - Beispiel-Extrakt von OpenStreetMap

### GIS Bibliotheken

- GDAL/OGR - Geospatial Data Translation Tools
- JTS Topology Suite (JTS) - Java Topology Suite
- GeoTools - Java GIS Toolkit
- GEOS - C/C++ Spatial Library
- MetaCRS - Coordinate Reference System Transformations
- libLAS - LiDAR Data Access

### Geschichte von OSGeo-Live

Schon 2007 entstand auf der FOSS4G die Idee, eine DVD mit OSGeo Software zu erstellen. Im Jahr 2009 wurde die Version 2.0 auf der FOSS4G mit 21 Projekten vorgestellt und war ein großer Erfolg. Die DVD wurde von LISASoft zusammengestellt und unter dem Namen Arramagong veröffentlicht.

2010 folgte dann die Version 3.0 mit 34 Projekten auf der FIG (International Surveyors Conference). Die DVD kam hier bereits in Workshops zum Einsatz.

Im September 2010 kam zur FOSS4G in Barcelona die Version 4.0 heraus. Diese Version trug dabei erstmals den Namen OSGeo-Live. Auch mit dieser Version ist die Anzahl der Projekte gestiegen. Mit 43 Projekten und einer englischen Dokumentation zu den Projekten und zu OGC Standards wurde die DVD auf der Konferenz verteilt und in den Workshops eingesetzt. Einen Monat später wurde die DVD auch auf der INTERGEO in Köln ausgegeben.

Im Jahr 2012 ist der Einsatz von OSGeo-Live beeindruckend. OSGeo-Live wurde weltweit auf mehr als 40 Veranstaltungen vorgestellt oder verwendet.

Auch auf der FOSSGIS Konferenz 2013 wird OSGeo-Live (in der Version 6.5) in den Workshops genutzt und als DVD an die Teilnehmer der Konferenz ausgegeben.

Es gibt ein Kernteam aus 4 Personen, das das Projekt koordiniert. Das Packen der Projekte übernehmen über 80 Beteiligte, ähnlich viele erstellen die Übersetzungen und Tausende schreiben die Software oder erfassen die Daten.

OSGeo-Live macht OSGeo Software und die OSGeo Foundation greifbar. Das Projekt spiegelt die hohe Aktivität und zeigt anschaulich, in welchem rasantem Tempo sich die OpenSource Software und Daten entwickeln.

## **OSGeo-Live rocks!**

Herzlichen Dank an alle Beteiligten und viel Spaß mit OSGeo-Live!

Kontakt zur Autorin:

Astrid Emde  
WhereGroup GmbH & Co. KG  
Eifelstraße 7  
53119 Bonn  
+49 (0)228 909038-0  
astrid.emdewheregroup.com

## Links

- Webseite <http://live.osgeo.org/>
- Wiki: [http://wiki.osgeo.org/wiki/Live\\_GIS\\_Disc](http://wiki.osgeo.org/wiki/Live_GIS_Disc)
- IRC: <irc://irc.freenode.net/#osgeolive>



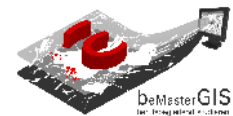
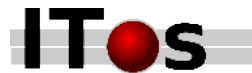
Goldsponsor:



Silbersponsoren:



Bronzesponsoren:



Mediasponsoren:



Medienpartner: