



FOSSGIS Konferenz 2010
Freitag, 5. März 2010

Sven Geggus <sven@gegg.us>



- Motivation
- Nutzung von Rasterdaten (Luftbilder)
- Nutzung von Vektordaten (GIS-Formate)
- Ausblick



Motivation

- Fragen zum Umgang mit „Datenspenden“ werden auf der Mailingliste häufig gestellt
- Importscripte schlecht dokumentiert
- GIS-Formate passen schlecht zum OSM Datenmodell
- Eigene Erfahrung weitergeben
- Diskussion über verfügbare (freie) Tools



Nutzung von Rasterdaten

- Einbindung in OSM Editoren über WMS (JOSM, Merkaartor)
- Einbindung als TMS (Potlatch)
- Datenaufbereitung relativ einfach



Datenaufbereitung von Rasterdaten

- Umprojizieren in EPSG:4326 (lat/long)
- Ausgangsdaten in Deutschland meist in Gauss-Krüger Projektion
- Freie Software verfügbar (GDAL-Tools)
- Hohe Positionsgenauigkeit durch Verwendung von amtlicher Korrekturdatei BETA2007 erreichbar

<http://wms.openstreetmap.de>

- Einbindung beliebiger Bilder möglich
- Technik
 - UMN-Mapserver/Mapscript
 - Apache mod_rewrite



TMS-Beispiel (Potlatch):

<http://wms.openstreetmap.de/tms/Lauf/18/139286/89455.png>

WMS-Beispiel (JOSM, Merkaartor):

<http://wms.openstreetmap.de/wms?layers=Lauf&>



Datenaufbereitung von Rasterdaten (Beispiel)

Ausgangsdaten:

10000x10000 Pixel Rasterkacheln in GK4-Projektion

Zieldaten:

Bild für wms.openstreetmap.de (HFA Format .img) mit
Auflösungspyramide in lat/lon (EPSG:4326)



Datenaufbereitung von Rasterdaten (Beispiel)

1. Aus Einzelbildern ein großes Bild erzeugen:

```
$ gdal_merge.py -v -of HFA -o grossesbild.img *.tif
```

2. Großes Bild in lat/long WGS84 (EPSG:4326) umprojizieren:

```
$ gdalwarp -r cubicspline -multi -of HFA \  
-s_srs '+init=epsg:31468 +nadgrids=<path-to>/BETA2007.gsb +wktext' \  
-t_srs EPSG:4326 grossesbild.img grossesbild-4326.img
```

3. Auflösungsrampe erzeugen:

```
$ gdaladdo --config USE_RRD YES grossesbild.img 2 4 8 16 32 64 128 256
```



Nutzung von Vektordaten

- Einbindung in Editoren wie bei Rasterdaten (WMS/TMS)
 - Methode wie bei Rasterdaten (einfach)
 - <http://wms.openstreetmap.de>
- Direkte Konvertierung in OSM-Format und upload zu openstreetmap.org
 - Konvertierung aufwendiger
 - keine/geringe manuelle Nachbearbeitung notwendig



Konvertierung von Vektordaten

- „Datenspenden“ meist in GIS-(z.B. Shapefiles) oder CAD-Formaten (z.B. DXF-Format)
- Eingangsformate ohne Topologie
- OSM Format mit Topologie (Routing)



Datenformate

OSM

- **Nodes**
- **Ways**
Verbindung mehrerer Nodes
- **Relations**
Liste von Nodes und Ways

GIS

- **Point**
- **Linestring**
- **Polygon**



Datenkonvertierung verfügbare (freie) Software

- GDAL-tools
 - keine Ausgabe im OSM Format möglich
 - sehr gut geeignet für Umrechnung von Projektionen geeignet



Datenkonvertierung verfügbare (freie) Software

- Scripts von svn.openstreetmap.org
 - `polyshp2osm.py`
 - Sehr gut geeignet für Polygone
 - Manuelle Anpassung an Use-Case notwendig
 - `shp2osm.pl`
 - Sehr rudimentär



Beispiel 1: Import von Häusergeometrien (shapefile) Gemeinde Neunkirchen am Sand

- Koordinatenkonvertierung mit gdal:

```
ogr2ogr -s_srs '+init=epsg:31468 +nadgrids=path/to/BETA2007.gsb +wktext' \  
-t_srs epsg:4326 -f 'ESRI Shapefile' outputfile.shp inputfile.shp
```

- Konvertierung in OSM-Format mit polyshp2osm.py
- Entfernen von duplicate Nodes mit dedupe-nodes.pl oder JOSM Validator plugin
- Upload der Daten mit JOSM oder bulk-upload script



Beispiel 2: Import der zugehörigen Hausnummern (DXF-Datei)

- Konvertierung in Shapefile
(kein mir bekanntes freies Tool verfügbar)
- Koordinatenkonvertierung mit GDAL
- Hausnummern in Postgis (shp2pgsql)
- Häusergeometrien in Postgis (osm2pgsql)
- Verschneiden mit Postgis (Adresse an Haus, nicht an Node)
- Erzeugung neuer OSM Datei mit Spezialscript



Ausblick

- Nutzung von Rasterdaten problemlos möglich
- Nutzung von Vektordaten teilweise problematisch
 - Oft angepasste Scripte erforderlich
 - Vielleicht schreibt mal jemand:
 - Einen DXF Eingang für gdal
 - Einen OSM Ausgang für GDAL
 - Ein universelleres Script shp2osm