



# whoami

- Michael Maier <Michael.Maier@student.tugraz.at>
- Telematik-Student an der TU Graz seit 2003
- Linux-User (Debian/grml) seit 2004
- OpenStreetMap seit Juli 2010
  - OSM-Username: *species*
  - Mapping-Area Graz, Leoben
  - Mit dem Fahrrad, Motorrad und Öffis

# Was ist OpenStreetMap

- OpenStreetMap ist eine freie Weltkarte nach dem Wiki-Prinzip
- Entsteht aus der Arbeit von weltweit 600.000 Hobbykartografen
- Das komplette “planet file” ist inzwischen ca. 250 GB groß (xml)

# Wer steckt dahinter

- Menschen wie Du und ich ... "*Mapper*"
- Die OpenStreetMap Foundation
- Organisationen, die die Verwendung ihrer Daten erlauben  
zB Luftbilder von:
  - Yahoo
  - Bing
  - Geoimage.at

# Warum OpenStreetMap?

Nachteile kommerzieller Anbieter:

- Restriktive Lizenzen - only Free as in Beer
- Offline-Nutzung oft nicht erlaubt - Roaming!
- Absichtliche Fehler
- Änderungen/Richtigstellungen?

# Warum OpenStreetMap?

## Nachteile kommerzieller Anbieter:

- Restriktive Lizenzen - only Free as in Beer
- Offline-Nutzung oft nicht erlaubt - Roaming!
- Absichtliche Fehler
- Änderungen/Richtigstellungen?
- Bsp Google TOS: Durch die Nutzung schließen sie einen rechtsgültigen Vertrag mit Google - Dürfen unmündige Personen (unter 18?) Google Maps überhaupt nutzen?
- Kosten! Google verlangt ab 25K API-Zugriffen/Tag!

# Warum OpenStreetMap?

Vorteile von OpenStreetMap:

- Rohdaten sind frei verfügbar
- Jeder kann Dinge ändern
- Freie Karten für Navis
- Karten mit eigenem Stil rendern




Freiheit schafft Möglichkeiten!

# Wie funktioniert OpenStreetMap?




- Zentrale Datenbank
- Jeder kann Daten hinzufügen/ändern
- Qualitätskontrolle durch jeden!
- Jeder kann Daten/Extrakte runterladen






# Datenformat

- Punkte (Koordinaten), ⇒ “Node” 
- Flächen sind eine Reihe von Nodes, ⇒ “Way” 
- Gruppierungen von Ways ⇒ “Relations” 




# Datenformat

- Punkte (Koordinaten), ⇒ “Node” 
- Flächen sind eine Reihe von Nodes, ⇒ “Way” 
- Gruppierungen von Ways ⇒ “Relations” 




Jedes Element hat Eigenschaften ⇒ “Tags”, zB:

- amenity = Cafe 
- highway = footway 
- building = yes 
- landuse = farmland

# Datenformat




- Punkte (Koordinaten), ⇒ “Node” 
- Flächen sind eine Reihe von Nodes, ⇒ “Way” 
- Gruppierungen von Ways ⇒ “Relations” 

Jedes Element hat Eigenschaften ⇒ “Tags”, zB:




- amenity = Cafe 
- highway = footway 
- building = yes 
- landuse = farmland

Was taggen wir?

# Datenformat

- Punkte (Koordinaten), ⇒ “Node” 
- Flächen sind eine Reihe von Nodes, ⇒ “Way” 
- Gruppierungen von Ways ⇒ “Relations” 

Jedes Element hat Eigenschaften ⇒ “Tags”, zB:

- amenity = Cafe 
- highway = footway 
- building = yes 
- landuse = farmland

Was taggen wir?

Alles :-)

- highway=\*, landuse=\*, shop=\*, tourism=\*, ...
- ?=\*

Gebräuchliche Tags und Beschreibungen ⇒ Wiki!

# Quellen?

Alles aus dem Wiki:

- Howto: Minutely Mapnik [http://wiki.openstreetmap.org/wiki/DE:Minutely\\_Mapnik](http://wiki.openstreetmap.org/wiki/DE:Minutely_Mapnik)
- <http://wiki.osm.org/Mapnik>
- <http://wiki.osm.org/Tirex>
- [http://wiki.openstreetmap.org/wiki/Contours\\_on\\_the\\_Cycle\\_Map](http://wiki.openstreetmap.org/wiki/Contours_on_the_Cycle_Map)

# Übersicht

Wie kommt man zur Slippy Map?

# Übersicht

## Wie kommt man zur Slippy Map?

- Javascript-API im Browser
- Tiles werden vom Renderer erzeugt
- Renderjobs werden von Queue-Manager verwaltet
- Queue-Manager bekommt Tasks vom Webserver-Modul
- Webserver liefert Tiles aus
- Hintergrund:
  - GIS-Datenbank Postgres/PostGIS
  - DB-Fill mit Osmosis/osm2pgsql
  - Minutely Updates

# Userseitig

Wie funktioniert die Kartenanzeige im Browser?



# Userseitig

Wie funktioniert die Kartenanzeige im Browser?

- Javascript-Framework lädt on Demand Kacheln (Tiles) vom Server

zB

- OpenLayers
- Leaflet by Cloudmade

# Tiles?

- Ein Tile ist ein 256x256px PNG.
- Tiles werden zB folgendermaßen referenziert:  
`http://tile.openstreetmap.org/7/63/42.png`  
`$Hostname/$tiledirs/$zoom/$x/$y.png`
- Serverseitig werden sog. *Metatiles* verwendet, die aus 8x8 Tiles bestehen. (Weniger Rendraufwand)

# Renderer:Mapnik

Portal siehe <http://wiki.osm.org/Mapnik>

- Python
- Stile in XML
- Daten von Postgres oder Files

# Queue-Manager: Tired

Sprich: T-Rex siehe <http://wiki.osm.org/Tired>

- Multi-CPU unterstützung
- Job-Verteilung auf mehrere Rechner
- Steuerung durch:
  - wird vom Apache *mod-tile* gesteuert
  - batch-jobs per skript
- Munin/Nagios-Überwachung

## Webserver: Apache mit *mod-tile*

Liefert aus gerenderten Metatiles ausgeschnittene Tiles aus.

## Webserver: Apache mit *mod\_tile*

Liefert aus gerenderten Metatiles ausgeschnittene Tiles aus. Sehr simples Modul...

- Zoomlevel hardcoded  $\Rightarrow$  recompile for z19
- metatiles muessen in `/var/lib/mod_tile/` liegen! (hardcoded)

# Postgres/PostGIS

## Postgresql-DB mit PostGIS-Erweiterung

- PostGIS noch selbst zu kompilieren!
- Empfehlung: hstore-Extension
- Version: 8.3 oder 9.x, nicht 8.4!
- Wichtig: Tunen der DB, siehe postgres-Hilfe

# Hardware

Hardware:

- DB entweder in RAM
- oder auf SSD



# Hardware

## Hardware:

- DB entweder in RAM
- oder auf SSD
- Alles andere Sinnlos...

## Selbst für Mitteleuropa:

- 6-Core Phenom 2.8GHz
- 16GB Ram
- SSD mit 500 MB/sec und 85000 IOPS
- tiles und Rest auf 3 anderen Platten

# Füllen der Datenbank

## OSM-Daten:

- Runterladen des ganzen Planet oder Auszügen von Geofabrik.de
- ev. ausschneiden eines gebiets mit Osmosis
- einspielen der Daten mit osm2pgsql

## Höhenschichtlinien für Contour-Maps:

- Daten fuer AT von der NASA downloaden
- Anleitung zum Importieren mittels scripts: [http://wiki.openstreetmap.org/wiki/Contours\\_on\\_the\\_Cycle\\_Map](http://wiki.openstreetmap.org/wiki/Contours_on_the_Cycle_Map)

# Minutely Updates

- Es gibt minutely diffs des Planeten (.osc) am OSM-Server
- cron-job downloaded sie
- diff-applien mit osm2pgsql
- bounding box bei osm2pgsql angeben.

# Help

- Erste Station sollte das Wiki sein: [wiki.openstreetmap.org](http://wiki.openstreetmap.org)
- Immer noch etwas Unklar? ⇒ Mailingliste [talk-at](mailto:talk-at)

# Help

- Erste Station sollte das Wiki sein: [wiki.openstreetmap.org](http://wiki.openstreetmap.org)
- Immer noch etwas Unklar? ⇒ Mailingliste talk-at
- Stammtisch! In Graz alle 2 Monate - der nächste Ende Mai im Brot & Spiele