

# Messen im freifunk freimap und mehr



Thomas Hirsch, EasterHegg 2008, Köln

# Übersicht

## Visualisierung – Ziele

- Stand der Dinge: Freifunk-Karten
- Zusammenführung von Datenquellen

## Messungen – Möglichkeiten und Ziele

### **Demo: freimap – Berlin && 24C3**

- Architektur
- Anwendungen

## Was wird? – Ein paar Ideen

- Längerfristige Messungen
- Anschließend: Diskussion / Workshop (ETA: 17:00 Uhr)

# Übersicht

...ist auch das Ziel von Visualisierung

- Visualisierung
- Messungen & Datenquellen

# Fragestellungen + Ziele

- Wie verhält sich eigentlich ein reales Mesh?
- Wie herum wird denn gefunkt?
  - Jeder nur mit seinem Uplink, oder kreuz + quer?
- Wie sollte Routing darin funktionieren?
- Warum klappt das denn jetzt wieder nicht?
- Phänomene und Fehler sichtbar machen
- Routing-Algorithmen testen ohne Testbed
  - Layer 2-3, Overlay, Streaming, etc...



# Vor- und Nachteile

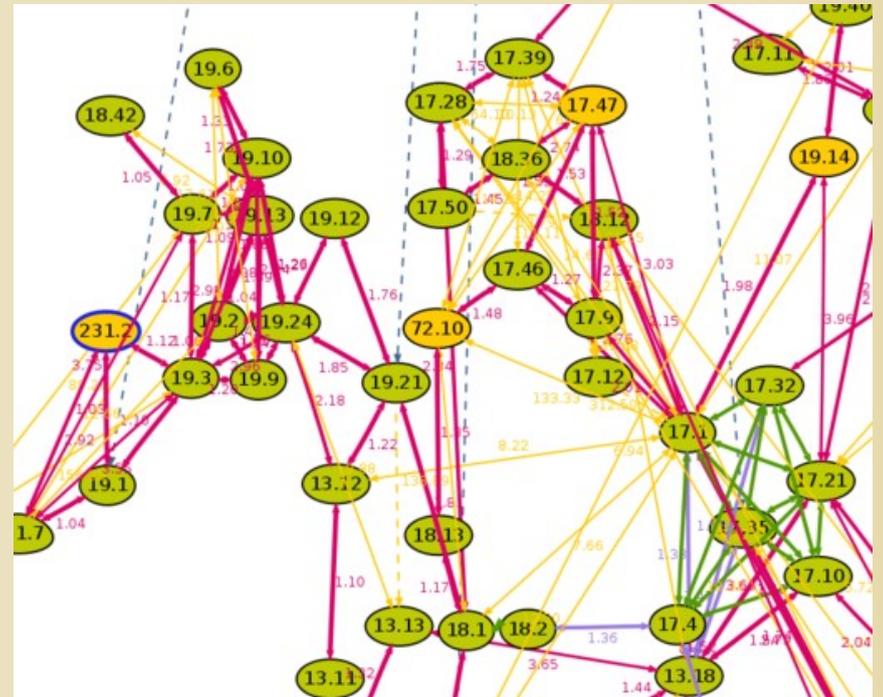
- + Benutzerfreundliches Web-Interface
- + Leichte Umsetzung
- Eine Karte stellt relativ statische Daten dar  
„Keine Zeit“, wie der Hase sagt.
- Auswertung der Daten schwierig

Dynamisches Verhalten schwer zu ermitteln.



# Datenquellen

- olsrd dot\_draw plugin ➡
- olsrd text\_info
- Selbst installierte Services
  - snmp
  - IPFIX/Netflow
- Andere bekannte Services
  - Dezentrale Map
  - Web-Interfaces / ffsomething
  - (Avahi, Obamp, ...)



# Korrelation von Daten

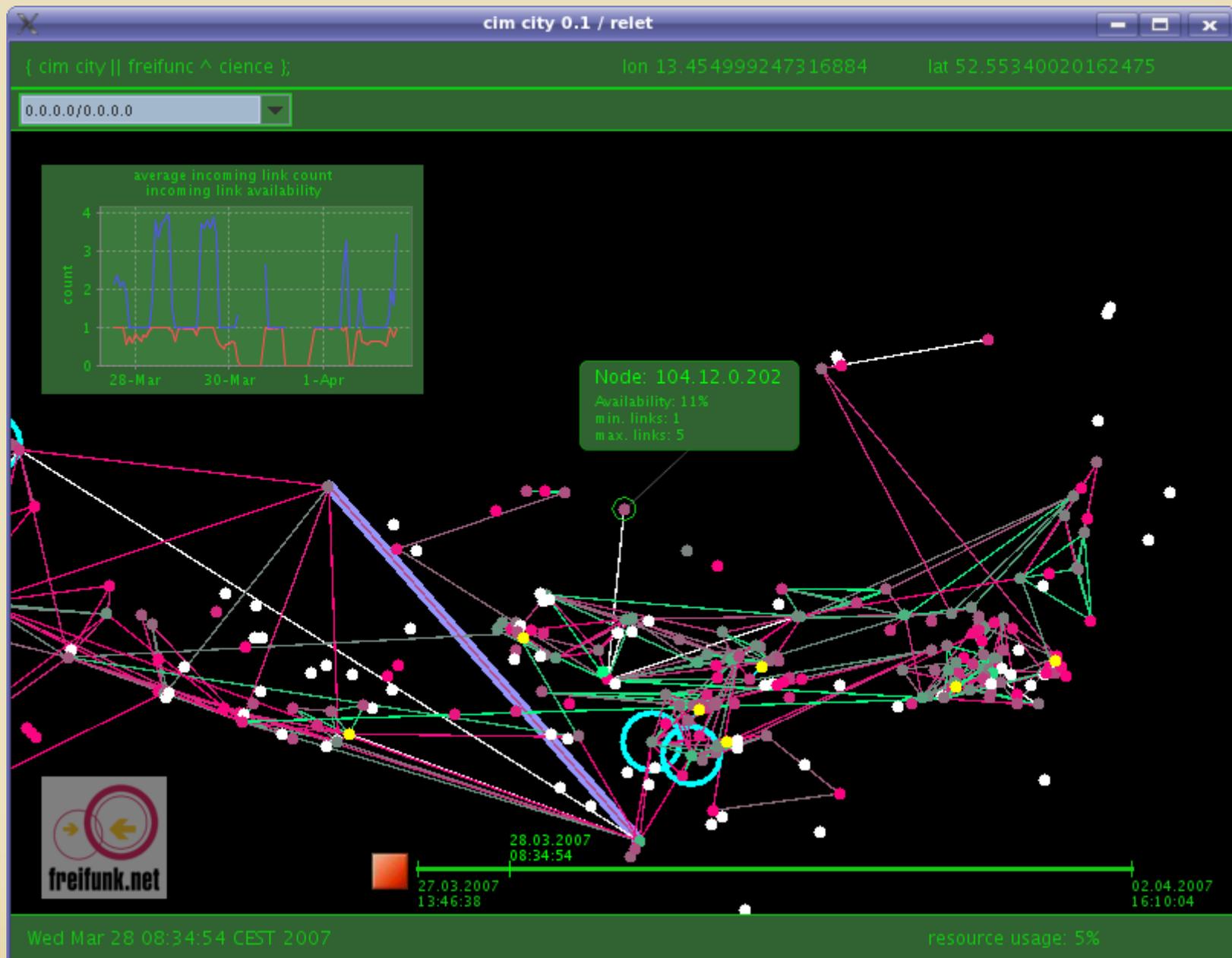
zum Beispiel: Topologie + Koordinaten



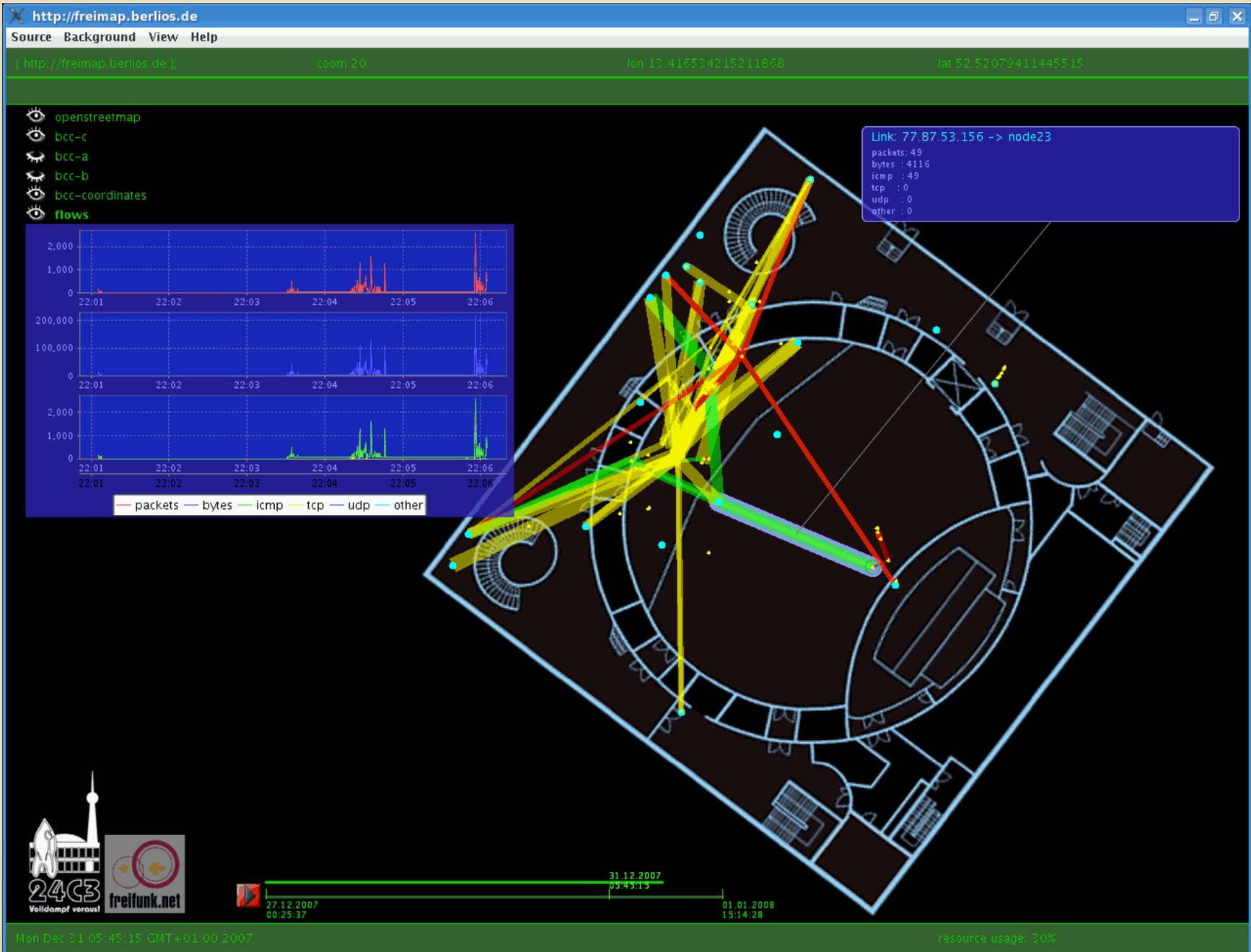
# Aufzeichnung von Daten

- Visualisierung: Bewegung auf der Zeit-Achse
- Direkte Anfragen an die Datenbank
- Erstellen von Referenz-Traces
- Hallo Vorratsdatenspeicherung?
  - Freifunk ist nicht „sicher“
  - Freifunk ist verortet, nicht „anonym“
  - Datenform, nicht Dateninhalt messen – Flows
  - ...

# - Demo -

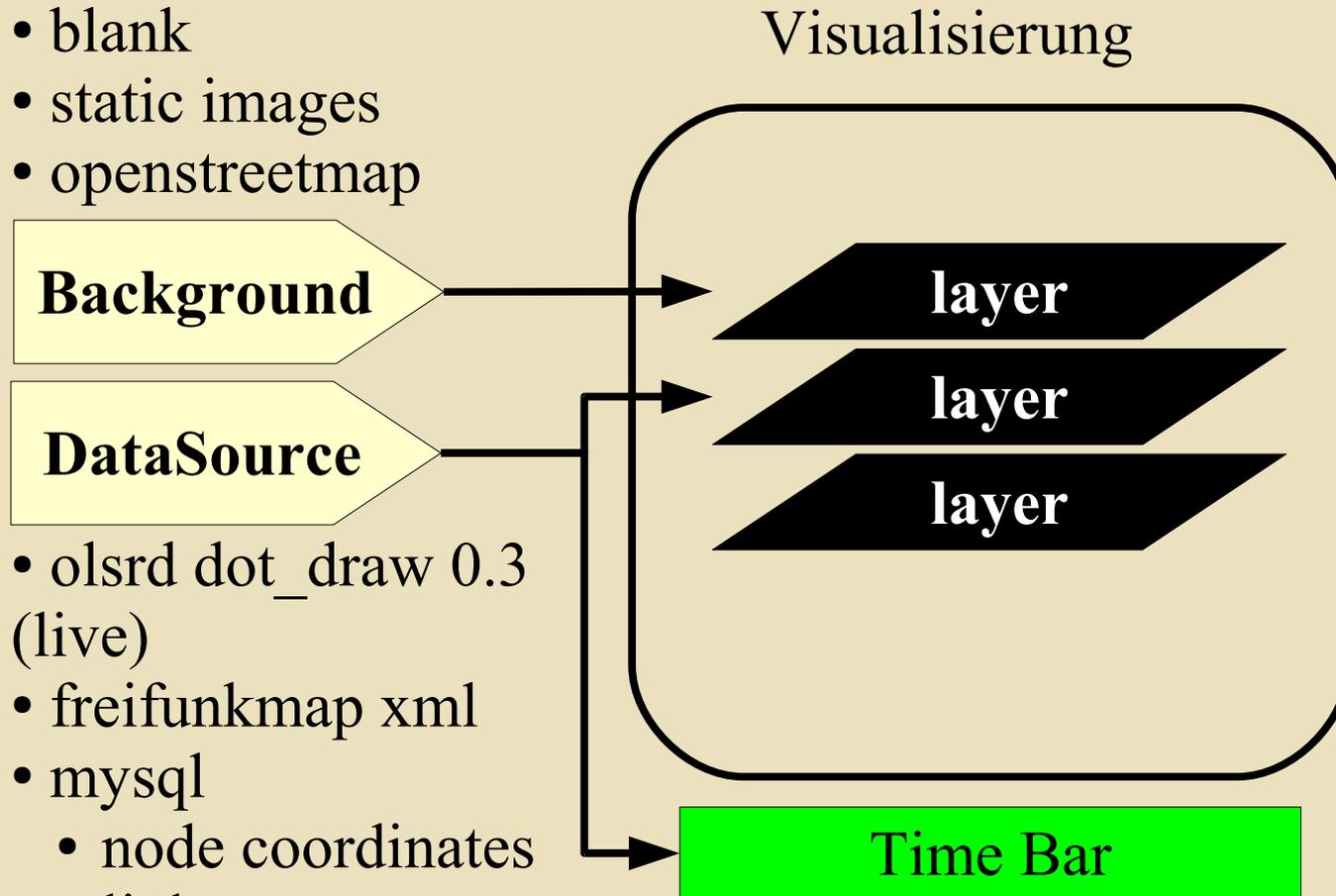


# - Demo: Flows -



# Architektur

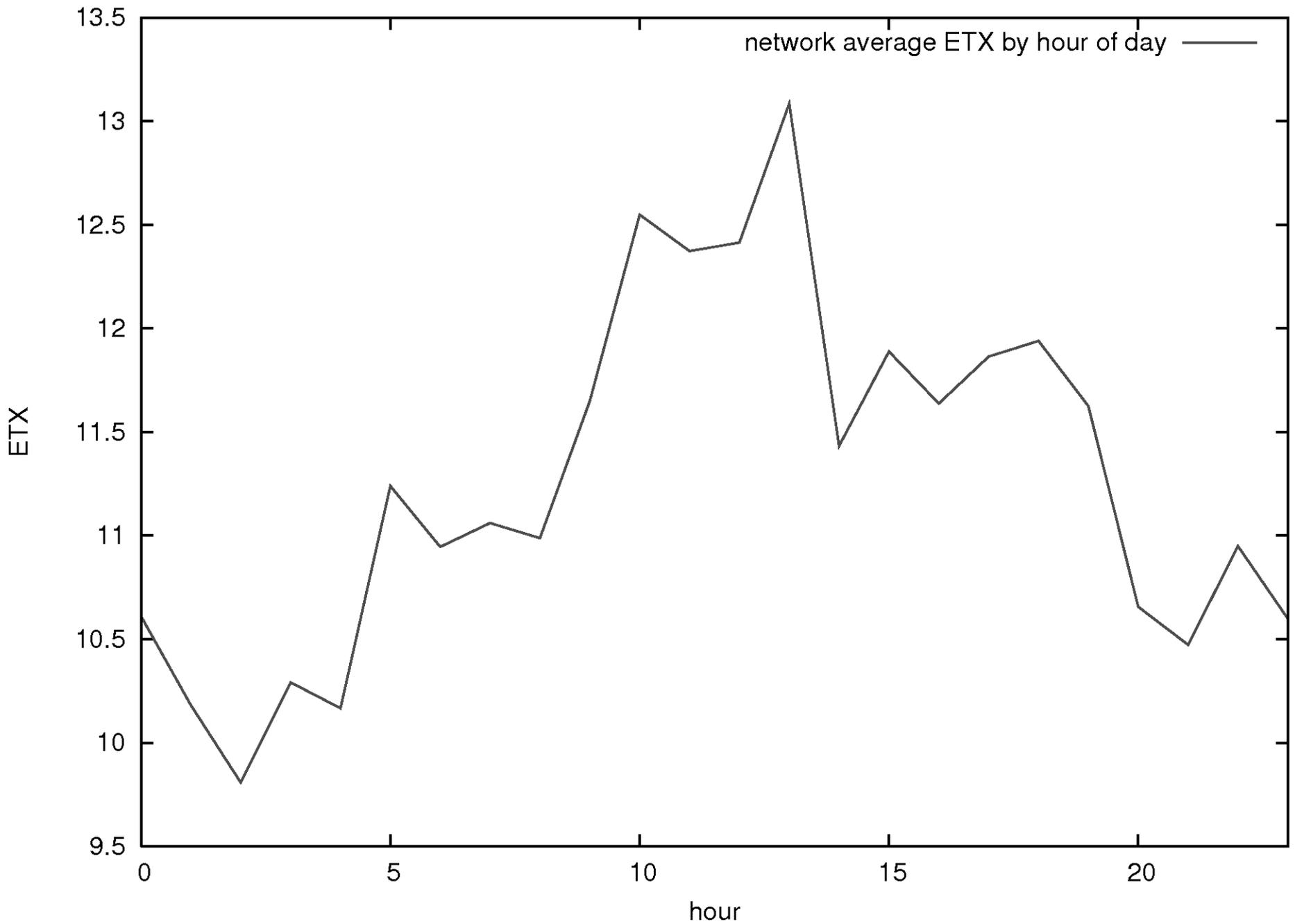
- blank
- static images
- openstreetmap
- olsrc dot\_draw 0.3 (live)
- freifunkmap xml
- mysql
  - node coordinates
  - links
  - netflow
  - snmp (in progress)



# DataSource Interface

- `init ( )`
- `addDataSourceListener ( ... )`
- `getNodeList ( )`
- `getFirst/LastUpdateTime ( )`
- `getClosestUpdateTime (long time)`
- `getLinks (long time)`

# Ein paar Ergebnisse



# Anwendungen

- Bei den Wissenschaftlern
  - MOME / CRAWDAD
  - Anchor-Free Localization
  - Simulationen
    - Routing
    - Streaming
    - Delay/Disruption Tolerant Networking
    - Wenn Implementierung oder Deployment schwierig ist.

# Idee: Referenztrace

- Korrelation von mehreren Datenquellen
- Sammelpunkte an den Uplinks
  - OLSR Topologie
  - Traffic Flows
  - Router Status / SNMP
- Dauer: 1 Monat
- Creative Commons Datensatz für CRAWDAD+MOME
- Öffentliche Ankündigung im Netzwerk
- Einfache Unkenntlichmachung der Teilnehmer
  - IP „1.2.3.4“ nach „node1“
  - Zeit „12:34:56“ nach „t0 + 123456 Sekunden“
  - Aber: Eine Topologie kann man nicht verstecken

# Ideen im wiki.freifunk.net

- Freifunk wandelt sich vom Access zum Service Net
  - Service Discovery / Avahi ++
  - Community Services, Streaming, Search, Sharing
- Könnte freimap eine Oberfläche sein?
  - Layer gegen die Informationsflut
  - Services in der Karte
  - Aktive Komponenten
    - Aktives Messen (Traceroute von-nach)
    - Eigene Inhalte veröffentlichen

# Diskussion

In Folge: „Workshop“ - freimap Installation

```
$> svn checkout ...
```

```
$> ./compile
```

```
$> vim config.yaml
```

```
$> ./run
```

+ Fragen: Eigene Datenquellen / Datenbanklayout?