

# MapProxy

Kartendienste beschleunigen, absichern und verteilen

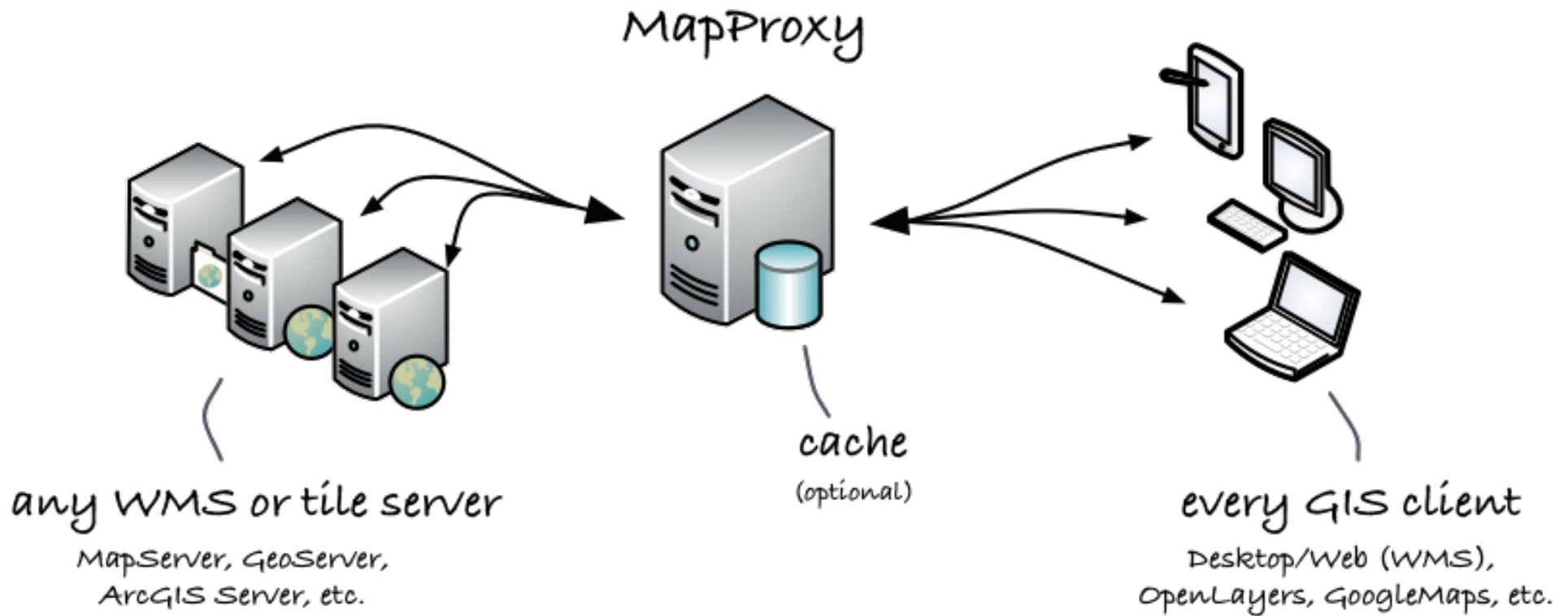
Thorsten Hildebrand  
WhereGroup GmbH und Co. KG

Bonn, 14.12.2017

# Überblick

- Was ist MapProxy?
- Dienste beschleunigen
- Cacheformate
- Dienste manipulieren und absichern
- HQ / Retina Kacheln
- sonstige Funktionalitäten

# MapProxy



# MapProxy

- Aktuelle Version: 1.11.0 (20.11.2017)
- Python Software
- Läuft auf allen gängigen Serversystemen (z.B. Linux, Windows)

# Überschaubare Konfiguration

```
...
layers:
  - name: osm
    title: WhereGroup Demo WMS
    sources: [osm_cache]
caches:
  osm_cache:
    grids: [webmercator]
    sources: [osm_wms]
sources:
  osm_wms:
    type: wms
    req:
      url: http://osm-demo.wheregroup.com/service?
      layers: osm
grids:
  webmercator:
    base: GLOBAL_WEBMERCATOR
...
```

# Überschaubare Konfiguration

```
seeds:  
  coverage_seed:  
    caches: [osm_cache]  
    levels:  
      to: 11  
  
coverages:  
  deutschland:  
    datasource: /daten/mapproxy/GM.txt  
    srs: 'EPSG:3857'
```

# Berechnung Anzahl Kacheln

```
mapproxy-util grids mapproxy.yaml --seed seed.yaml --coverage deutschland
```

```
webmercator:
```

```
  Configuration:
```

```
    bbox*: [-20037508.342789244, -20037508.342789244, 20037508.342789244, 20037508.342789244]
```

```
    origin: 'nw'
```

```
    srs: 'EPSG:3857'
```

```
    tile_size: [256, 256]
```

```
Coverage: deutschland covers approx. 0.0645% of the grid BBOX
```

```
Levels: Resolutions, # x * y = total tiles (approx. tiles within coverage)
```

00:	156543.03392804097,	#	1 * 1	=	1 (0)
01:	78271.51696402048,	#	2 * 2	=	4 (0)
02:	39135.75848201024,	#	4 * 4	=	16 (0)
03:	19567.87924100512,	#	8 * 8	=	64 (0)
04:	9783.93962050256,	#	16 * 16	=	256 (0)
05:	4891.96981025128,	#	32 * 32	=	1024 (0)
06:	2445.98490512564,	#	64 * 64	=	4096 (2)
07:	1222.99245256282,	#	128 * 128	=	16384 (10)
08:	611.49622628141,	#	256 * 256	=	65536 (42)
09:	305.748113140705,	#	512 * 512	=	262144 (168)
10:	152.8740565703525,	#	1024 * 1024	=	1.05M (675)
11:	76.43702828517625,	#	2048 * 2048	=	4.19M (2703)
12:	38.21851414258813,	#	4096 * 4096	=	16.78M (10813)
13:	19.109257071294063,	#	8192 * 8192	=	67.11M (43255)
14:	9.554628535647032,	#	16384 * 16384	=	268.44M (173022)
15:	4.777314267823516,	#	32768 * 32768	=	1073.74M (692091)
16:	2.388657133911758,	#	65536 * 65536	=	4294.97M ( 2.77M)
17:	1.194328566955879,	#	131072 * 131072	=	17179.87M ( 11.07M)
18:	0.5971642834779395,	#	262144 * 262144	=	68719.48M ( 44.29M)
19:	0.29858214173896974,	#	524288 * 524288	=	274877.91M ( 177.18M)

# Cacheformate

- **File**
  - diverse Verzeichnisstrukturen
    - TileCache (zz/xxx/xxx/xxx/yyy/yyy/yyy.format)
    - MapProxy (zz/xxxx/xxxx/yyyy/yyyy.format)
    - TMS (zz/xxxx/yyyy.format)
    - quadkey
    - arcgis
    - eigene Strukturen



# Cacheformate

- **MBTiles** (eine große Datei pro Cache)
- **sqlite** (je eine große Datei pro Zoomlevel im Cache)
- **couchdb**
- **riak**
- **redis** (in-memory Datenbank)
- **geopackage** (ähnlich MBTiles und sqlite, aber neuerer, bekannterer Standard)
- **S3** (Amazonn Simple Storage Service)
- **compact** (ArcGIS) Version 1 und 2 (~16.000 Kacheln pro Datei)

# Farbkanal tauschen

caches:

my\_cache:

```
sources: [wms_source]
```

caches:

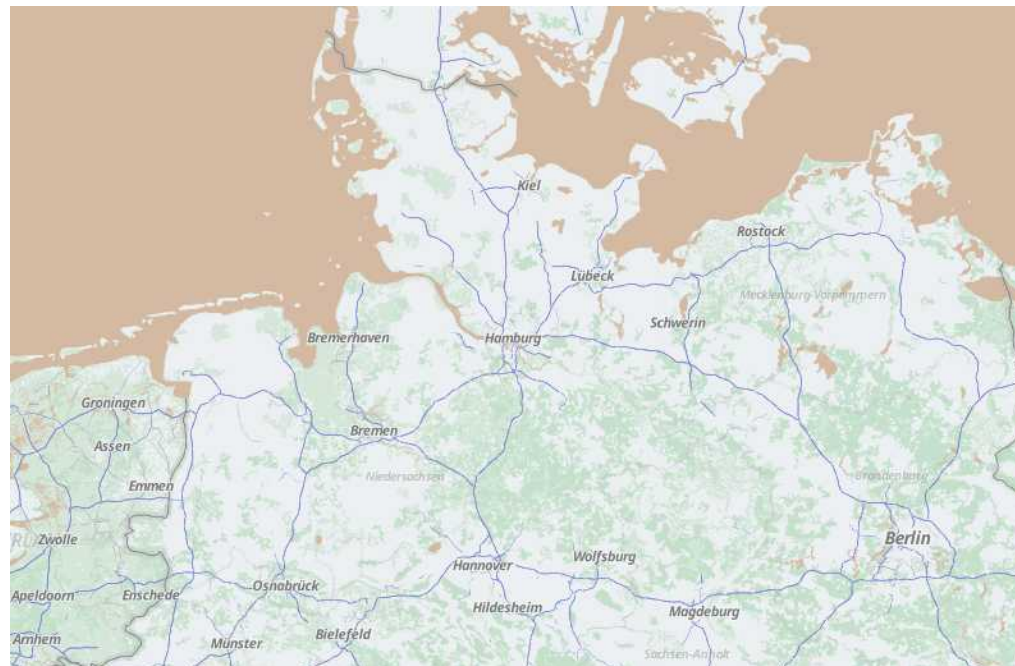
my\_cache:

```
sources:
```

```
  r: [{source: wms_source, band: 0}]
```

```
  g: [{source: wms_source, band: 1}]
```

```
  b: [{source: wms_source, band: 2}]
```



# Falschfarbenbild



cache:

my\_cache:

sources:

r: [{source: wms\_source, band: 0}]

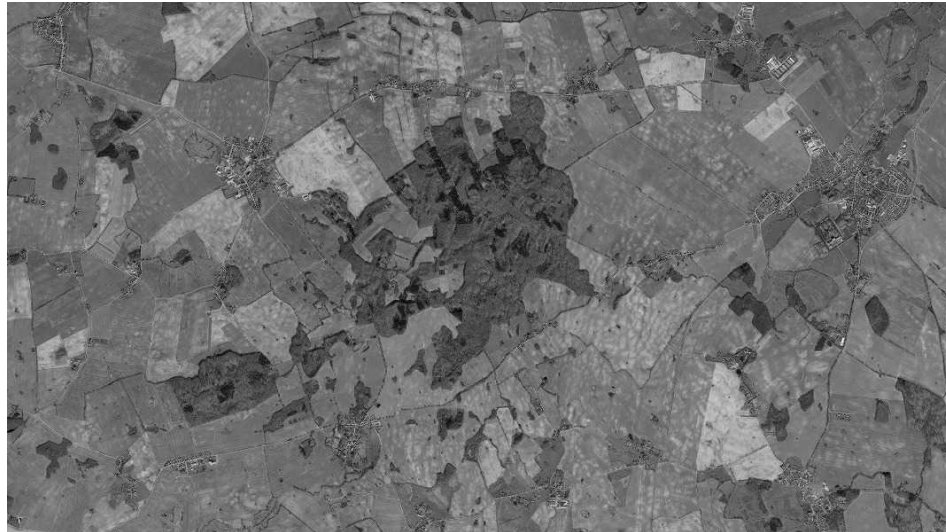
g: [{source: wms\_source, band: 1}]

b: [{source: wms\_source, band: 2}]



# Graustufen Bild

```
cache:  
  grayscale_cache:  
    sources:  
      1: [  
        {source: dop_rgb, band: 0, factor: 0.25},  
        {source: dop_rgb, band: 1, factor: 0.7},  
        {source: dop_rgb, band: 2, factor: 0.05},  
      ]
```



# Dienste ausstanzen

sources:

osm\_wms:

type: wms

req:

url: <http://osm-demo.wheregroup.com/service?>

layers: osm

coverage:

clip: true

#bbox: [5, 50, 10, 55]

#srs: 'EPSG:4326'

datasource: /daten/mapproxy/GM.txt

srs: 'EPSG:3857'



# Dienste absichern

- MapProxy Security API
  - eigener Python Code
  - Benutzer und Gruppen (Authentisierung)
  - Rechteauswertung (Autorisierung)
- Wirkt sich auch auf GetCapabilities aus :-)

```
def authorize(service, environ, layers=[], **kw):
    if 'mylayer' in layers and service.startswith('intern'):
        return {'authorized': 'none'}
    else:
        return {'authorized': 'full'}
```

# Dienste absichern

- Dienste für einzelne Benutzer begrenzen

```
def authorize(service, environ, layers=[], **kw):
    if environ['REMOTE_USER'] == 'bob':
        return {
            'authorized': 'partial',
            'layers': {
                'sat': {'map': True},
                'osm': {
                    'map': True,
                    'limited_to': {
                        'geometry': Point(8, 53).buffer(2),
                        'srs': 'EPSG:4326',
                    }
                }
            }
        }
    ...
```



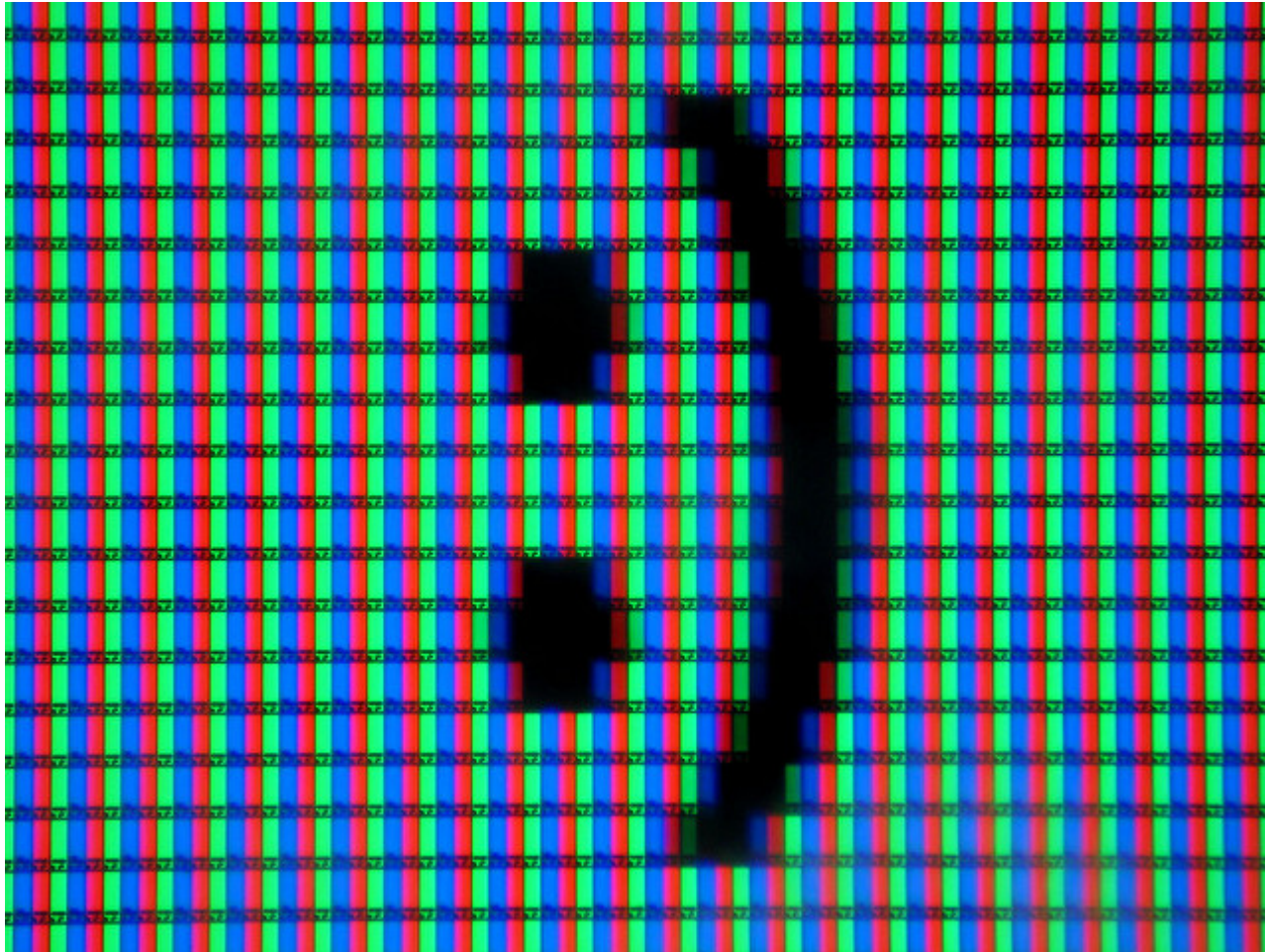
# Dienste absichern

- Dienste an komplexen Geometrien begrenzen
- Sicherheitsschicht in Python → viele Möglichkeiten
- reines Clipping seit 1.10.0 auch schon direkt in der Quelle möglich

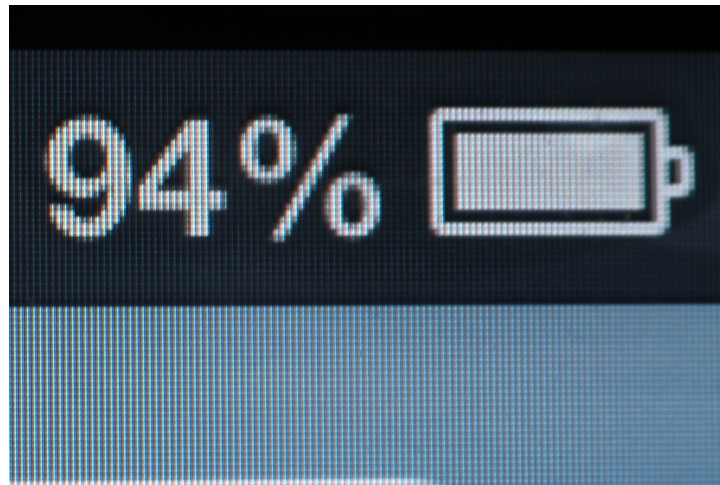
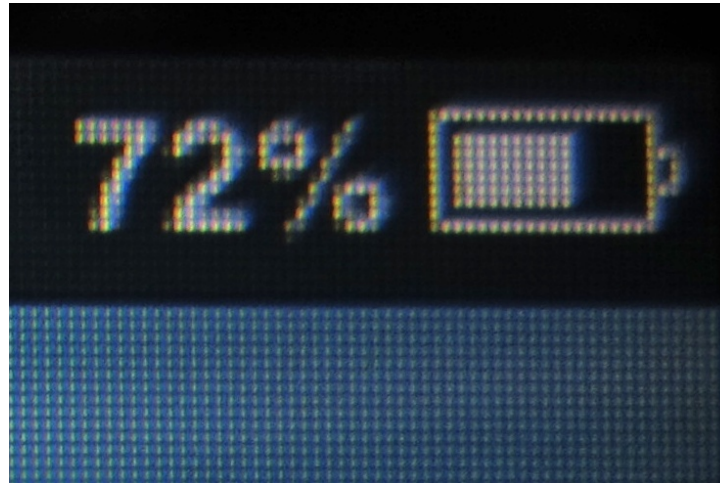




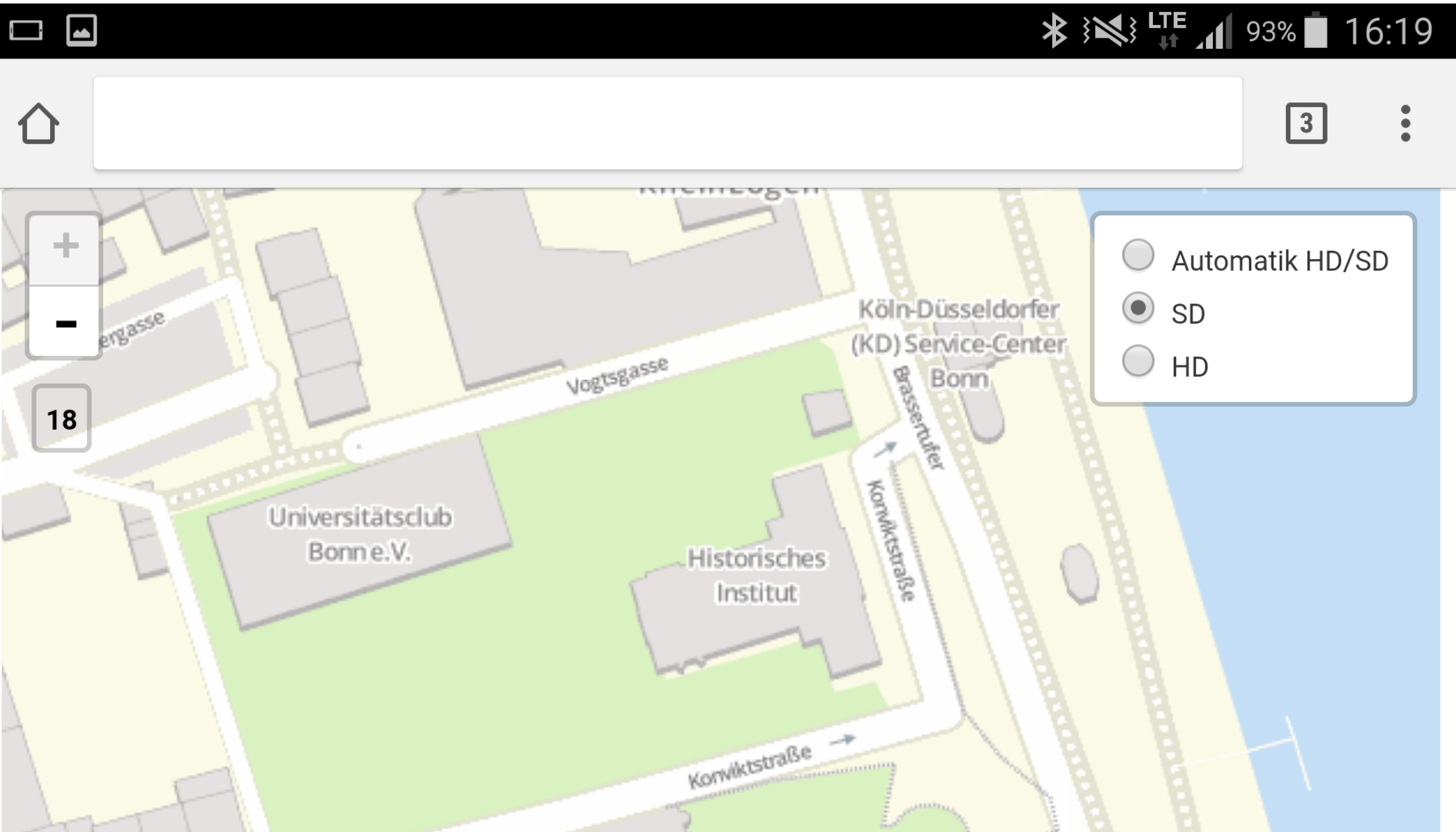
# Image pixel = display pixel ?



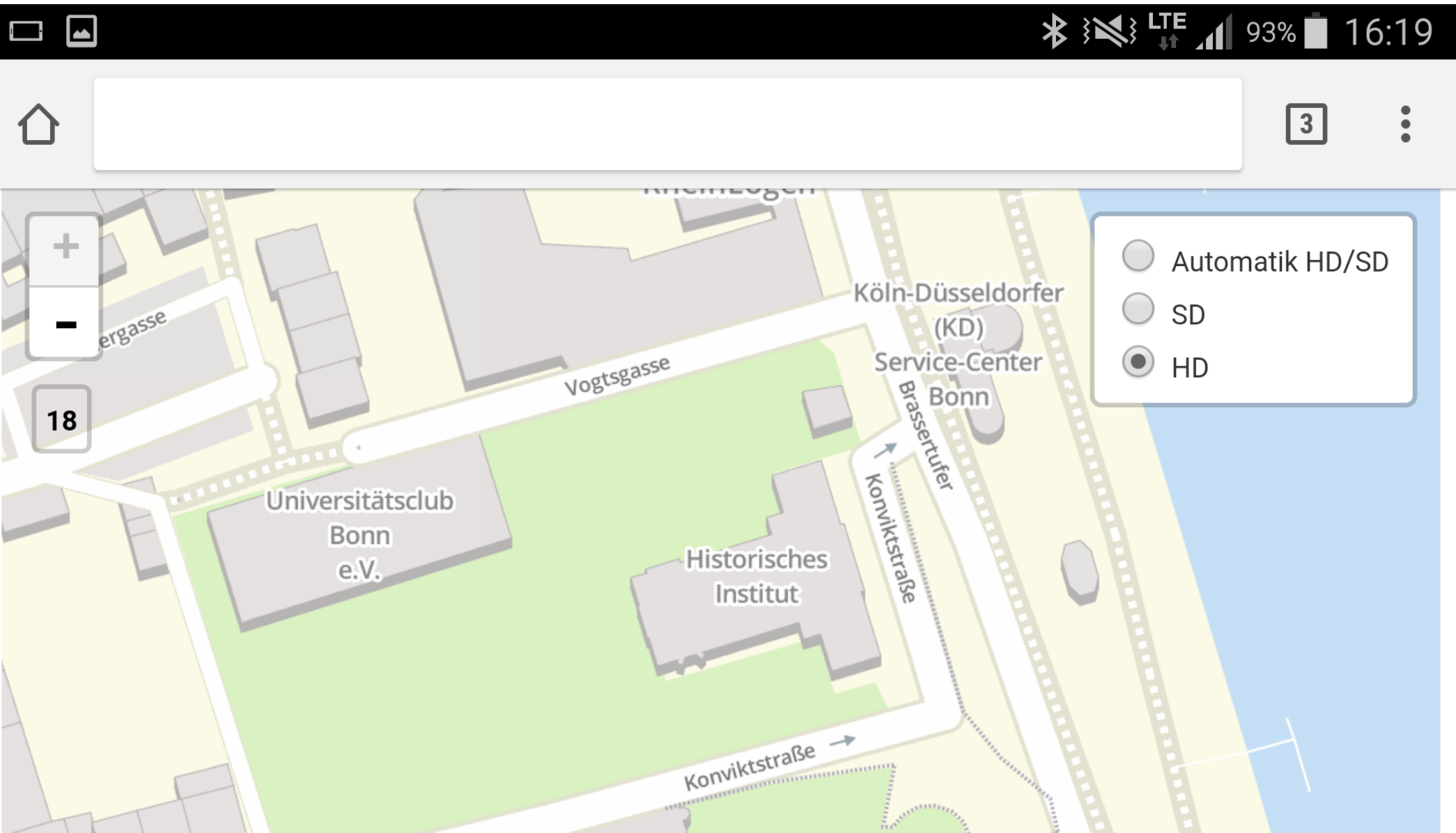
# Retina-, HQ-, HighDPI- Displays



# Mapnik SD



# Mapnik HQ



# Grid mit 512x512 Pixeln

```
grids:  
  webmercator:  
    srs: "EPSG:3857"  
    origin: nw  
    min_res: 156543.03392804097  
  webmercator_hq:  
    srs: "EPSG:3857"  
    origin: nw  
    min_res: 78271.51696402048  
    tile_size: [512, 512]
```

# Mapnik und MapServer

```
sources:  
  map_hq_source:  
    type: mapnik  
    mapfile: ./mapnik.xml  
    scale_factor: 2  
  map_hq_source:  
    type: wms  
    req:  
      url: http://example.org/mapserv  
      layers: basemap  
      map_resolution: 144 # 2 * DEFAULTRESOLUTION
```

# Scaled HQ images

- 2 × Größe mit 1/2 Auflösung → 0/0/0.png)



# Scaled HQ images

- 2 × Größe mit 1/2 Auflösung → 18/136246/88065.png)





# Sonstige Funktionalitäten (Auswahl)

- Wasserzeichen setzen (Achtung bei Tiled-Diensten)
- Wasserzeichen entfernen (Achtung TOS beachten)
- Layer zusammenfügen
- Farben verändern oder Transparenz einfügen
- Proxy zur Bereitstellung interner Dienste (in beide Richtungen)
- Durchleitung von FeatureInfo (mit optionalem XSLT)
- Bereitstellung von Maßstäben außerhalb des originalen Dienstes
- Caches im MixedMode

# Sonstige Funktionalitäten (Auswahl)

- Grid erstellen
- Speichert gleiche Kacheln nach Möglichkeit nur einmal
- Umgehung von Größenbeschränkungen (z.B. Druck)
- Verschiedene Protokollierungsmöglichkeiten
  - Mapproxy.log und source-request.log
- Seeding in definierten Bereichen, jetzt auch intersections, unions und differences
- Hintergrund Seeding
- mapproxy-util autoconfig
  
- ... → <https://mapproxy.org/docs/latest/>

**Vielen Dank für Ihre Aufmerksamkeit ...**

**Welche Fragen o. Anregungen haben Sie?**

**Thorsten Hildebrand**

thorsten.hildebrand@wherogroup.com  
Tel.: +49 (0)30 513 02 78 75

WhereGroup GmbH & Co. KG  
Gillweg 3 | 14193 Berlin

Zentrale:  
WhereGroup GmbH & Co. KG  
Eifelstraße 7 | 53121 Bonn

Tel.: +49 (0)228 909038-0  
Fax: +49 (0)228 909038-11

info@wherogroup.com  
www.wherogroup.com

