

Virtual Raster Tables und andere GDAL-Tools

Performante und einfache Rasterdatenprozessierung mit dem GDAL-Werkzeugkasten

FOSSGIS 2011, Heidelberg

Agenda

- GDAL/OGR
 - Der Werkzeugkasten im Querschnitt
 - Gdalbuildvrt, gdalinfo, gdal_translate,
gdaladdo
- Einsatz der Virtual Raster Tables (VRT)
 - Erstellen und manipulieren
 - Optimieren
 - Einbinden in eine Prozesskette
- Praxisbeispiel Geo4:
 - Nutzung der VRTs in der Produktion

- Stephan Holl
 - Projektleiter und Senior technical consultant bei Intevation GmbH im Bereich MapServer/PostGIS
 - Langjährige Erfahrung im Bereich Open Source GIS-Lösungen
 - MapServer/PostGIS-Experte
 - Mitarbeit im FOSSGIS e.V.
 - Mitarbeit am QGIS-Benutzerhandbuch

- **Geospatial Data Abstraction Layer**
 - Das Schweizer Messer der Geodatenverarbeitung
 - Satz aus Kommandozeilenwerkzeugen für Raster- und Vektorbearbeitung
 - Lizenz: X11/MIT
 - <http://www.gdal.org>
-
- Mehr Details: Workshop von Andreas Fischer morgen auf der FOSSGIS besuchen!

- `gdalbuildvrt`
 - Erstellen eines VRTs von Eingabedaten
- `gdal_translate`:
 - Ausschneiden, transformieren in unterschiedliche Formate, Umprojizieren etc.
- `gdalinfo`: Basisinfos über Rasterdaten
- `gdalwarp`: transformieren von Rasterdaten
- `gdal_retile.py`: Imagepyramiden erstellen
- `gdal_rasterize`: Vektordaten aufrastern
- `etc.`

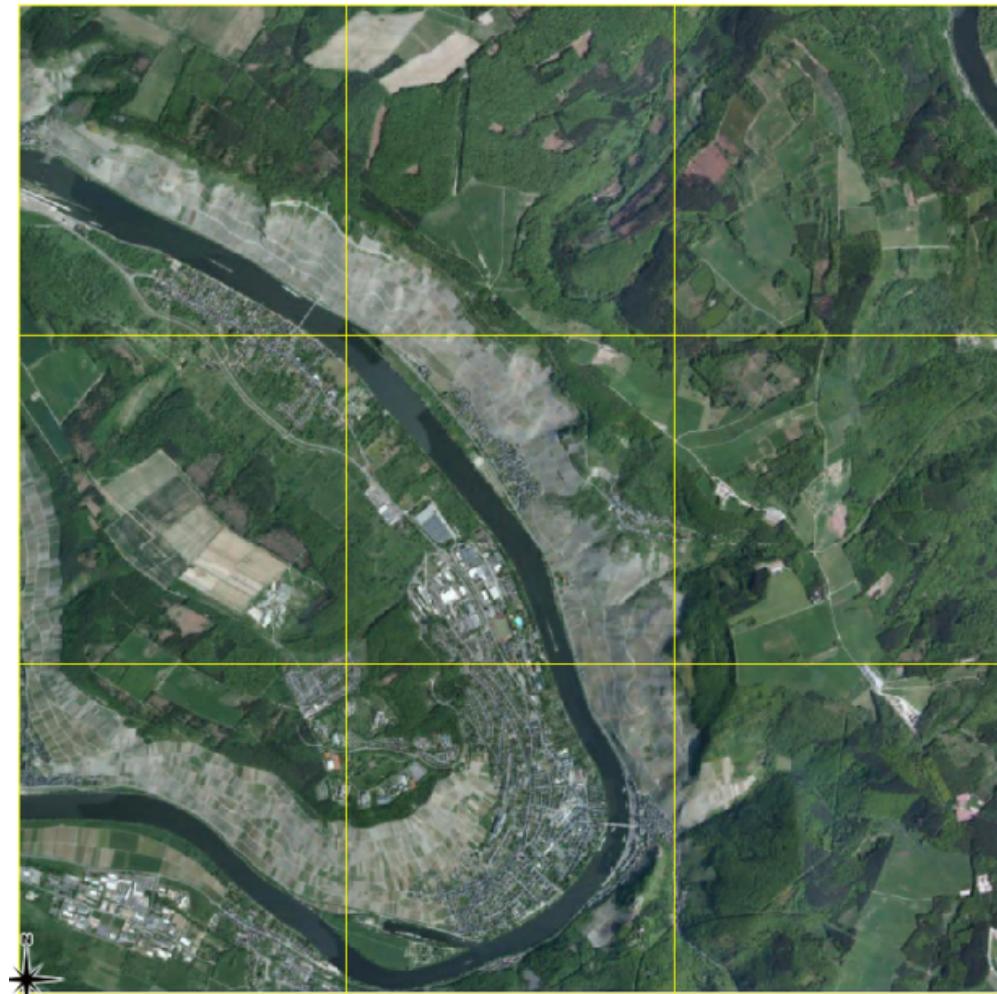
Virtual Raster Table

- XML-Dateistruktur
- Virtuelles GDAL-Format zum Kombinieren anderer GDAL-Formate zu einem neuen Datensatz
- Operationen z.B. mit `gdal_translate`, `gdaladdo`, `gdalwarp` etc. sind möglich
- programmatisch, über Tools oder manuell erzeugbar

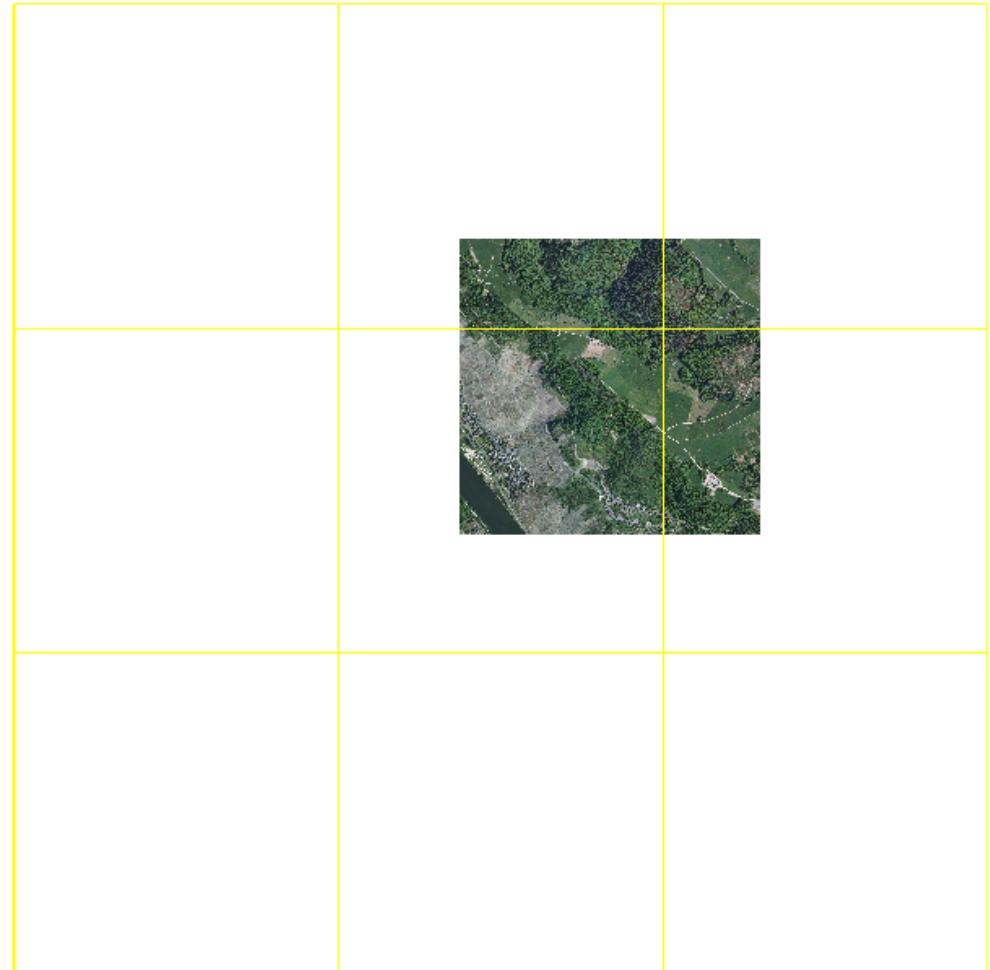
```
<VRTDataset rasterXSize="30000" rasterYSize="30000">
  <GeoTransform> 3.58000000000000e+05, 1.999999999999998e-01, 0.000000000000000e+00, 5.53600000000000e+06,
  0.000000000000000e+00, -1.999999999999998e-01</GeoTransform>
  <VRTRasterBand dataType="Byte" band="1">
    <ColorInterp>Red</ColorInterp>
    <SimpleSource>
      <SourceFilename relativeToVRT="1">3585530.tif</SourceFilename>
      <SourceBand>1</SourceBand>
      <SourceProperties RasterXSize="10000" RasterYSize="10000" DataType="Byte" BlockXSize="10000" BlockYSize="32"/>
      <SrcRect xOff="0" yOff="0" xSize="10000" ySize="10000"/>
      <DstRect xOff="0" yOff="20000" xSize="10000" ySize="10000"/>
    </SimpleSource>
  [...]
  </VRTRasterBand>
  <VRTRasterBand dataType="Byte" band="2">
    <ColorInterp>Green</ColorInterp>
    <SimpleSource>
      <SourceFilename relativeToVRT="1">3585532.tif</SourceFilename>
      <SourceBand>2</SourceBand>
      <SourceProperties RasterXSize="10000" RasterYSize="10000" DataType="Byte" BlockXSize="10000" BlockYSize="32"/>
      <SrcRect xOff="0" yOff="0" xSize="10000" ySize="10000"/>
      <DstRect xOff="0" yOff="10000" xSize="10000" ySize="10000"/>
    </SimpleSource>
  [...]
  </VRTRasterBand>
  <VRTRasterBand dataType="Byte" band="3">
    <ColorInterp>Blue</ColorInterp>
    <SimpleSource>
      <SourceFilename relativeToVRT="1">3585534.tif</SourceFilename>
      <SourceBand>3</SourceBand>
      <SourceProperties RasterXSize="10000" RasterYSize="10000" DataType="Byte" BlockXSize="10000" BlockYSize="32"/>
      <SrcRect xOff="0" yOff="0" xSize="10000" ySize="10000"/>
      <DstRect xOff="0" yOff="20000" xSize="10000" ySize="10000"/>
    </SimpleSource>
  </VRTRasterBand>
</VRTDataset>
```

- Vorteile:
 - Image-Katalog einfach erzeugbar
 - als ein Datensatz (in QGIS, MS) handhabbar
 - dynamisch
 - Maskierung von Bereichen
- Nachteile:
 - etwas kryptisch in der manuellen Editierung
 - Spezialfunktionen (z.B. Moving-Window-Filter)
müssen händisch eingepflegt werden
 - ?!

Extrakt von Teilbereichen



gesamt.vrt



part.tif

```
gdal_translate -projwin 360747 5534547 362603 5532720 gesamt.vrt part.tif
```

Farben 'vertauschen'



RGB

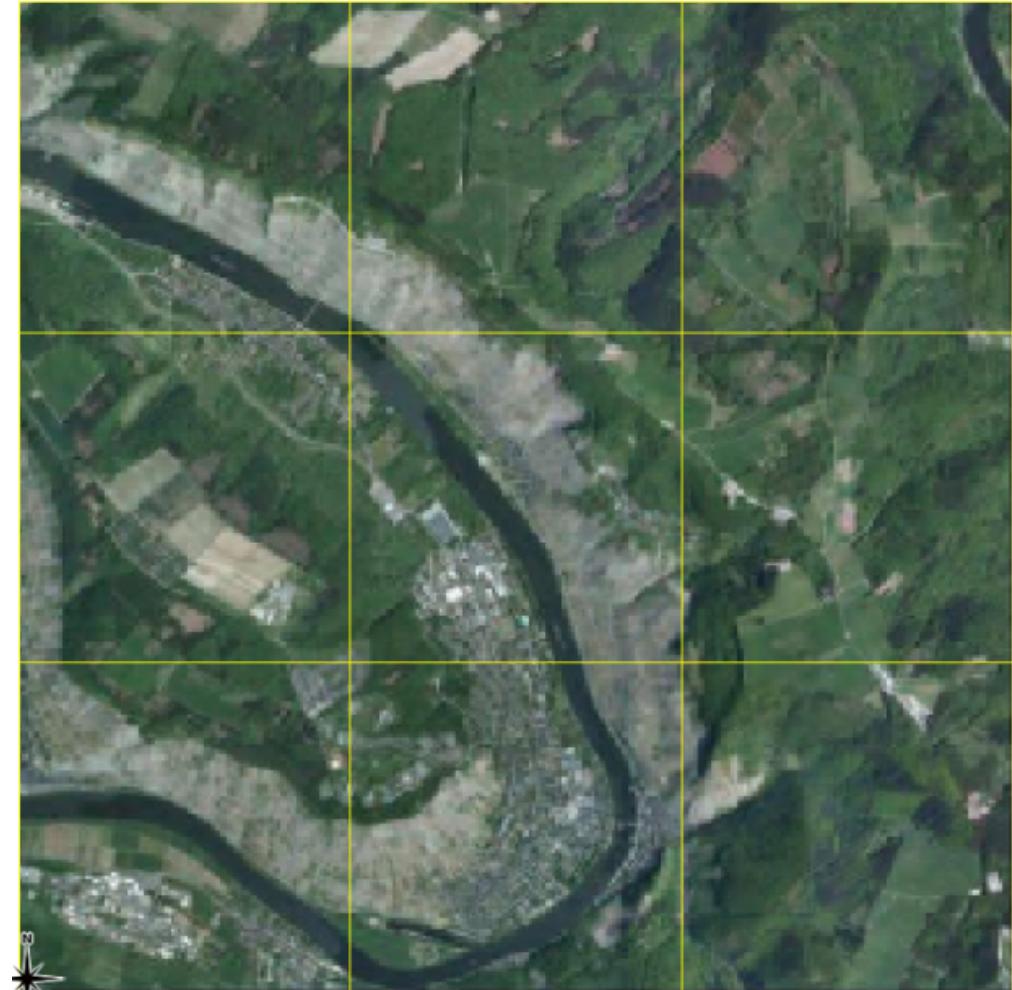


BGR

Generalisieren



gesamt.vrt (30.000x30.000 Pixel)



generalize.vrt (300x300 Pixel)

```
gdal_translate -outsize 1% 1% gesamt.vrt generalize.tif
```

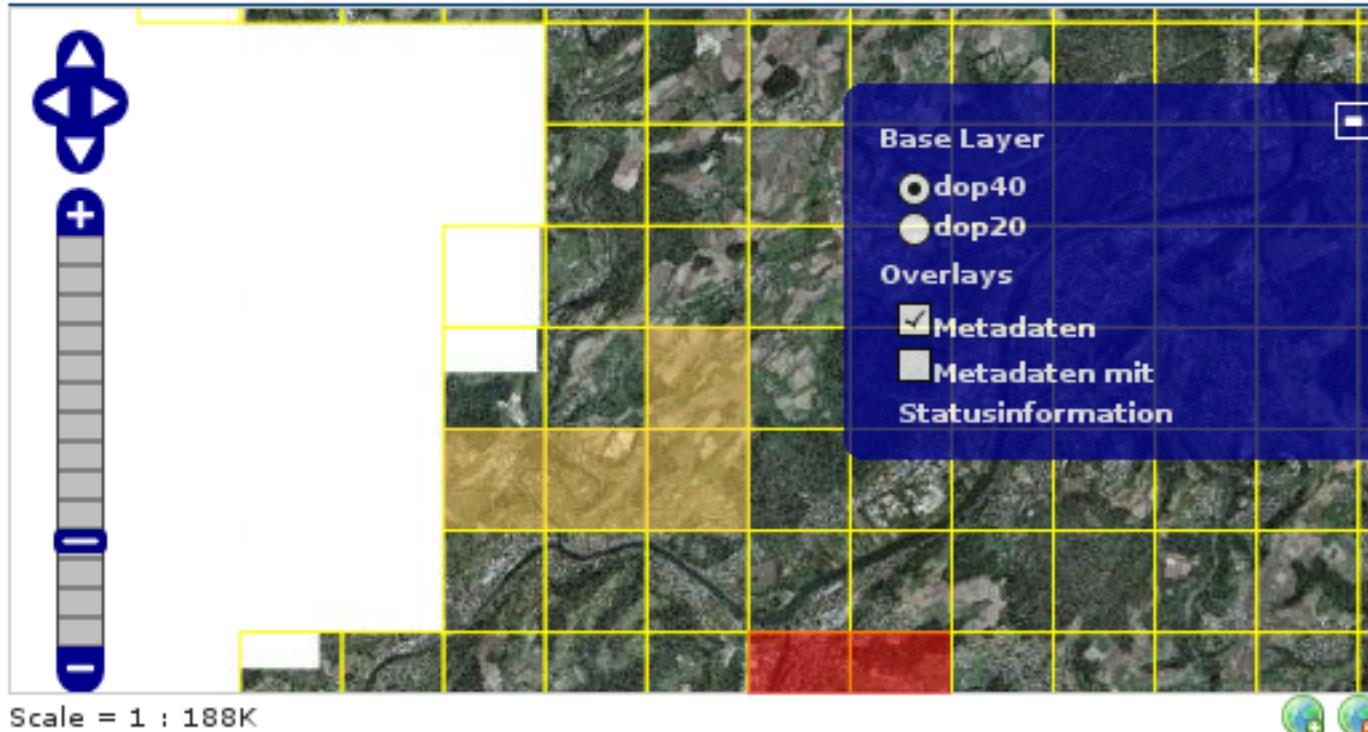
Beispiel Geo4

- Orthofotoserver der Landesvermessung Rheinland-Pfalz (LvermGeo)
- implementiert eine Prozesskette zur performanten Auslieferung von Luftbildern per WMS mit GDAL/MapServer/Python
- einzelne Kacheln sind über HTTP-Schnittstelle aktualisierbar
- VRT als Basisformat zur Aggregierung und Generalisierung der Kacheln

- Eingesetzte Prozesskette
 - Komprimierung der TIF-Dateien mit 80% JPEG-Komprimierung und internem Tiling (gdal_translate)
 - Zusammenfassen von Kacheln über VRTs (gdalbuildvrt)
 - Erzeugen von Overviews (gdaladdo)
 - Reduktion der räumlichen Informationen (gdal_translate)
 - Erzeugen von BBOX-Shapes für MapServer

Beispiel Geo4

Statuskartenansicht



Metadaten

Klicken Sie auf eine Kachel um Metadaten zu dieser Kachel zu erhalten.

Beispiel Geo4

Lieferscheine aktivieren

Lieferscheine aktivieren

Einfuegen Aktualisieren

Systemzeit definieren

Systemzeit definieren

Startdatum:	2011-10-11
Startzeit:	04:00:00
Enddatum:	2011-10-11
Endzeit:	05:00:00

Systemzeit aufheben Systemzeit setzen

Loadbalancer konfigurieren

[Load Balancer Manager auf georgia.atlas](#)

- GDAL-Werkzeugkasten beinhaltet eine Vielzahl von hilfreichen Werkzeugen
- Einfach und effizient nutzbar für Massendatenverarbeitung
- Verfügbar für Windows, Linux, Mac OSX

Vielen Dank für Ihre Aufmerksamkeit!

Kontakt:

Intevation GmbH

Stephan Holl <stephan.holl@intevation.de>

Neuer Graben 17
49074 Osnabrück
0541 – 335083 663