

# Hidden features and useful tips for UMN MapServer powerusers

09.09.2010, FOSS4G 2010, Barcelona, Spain



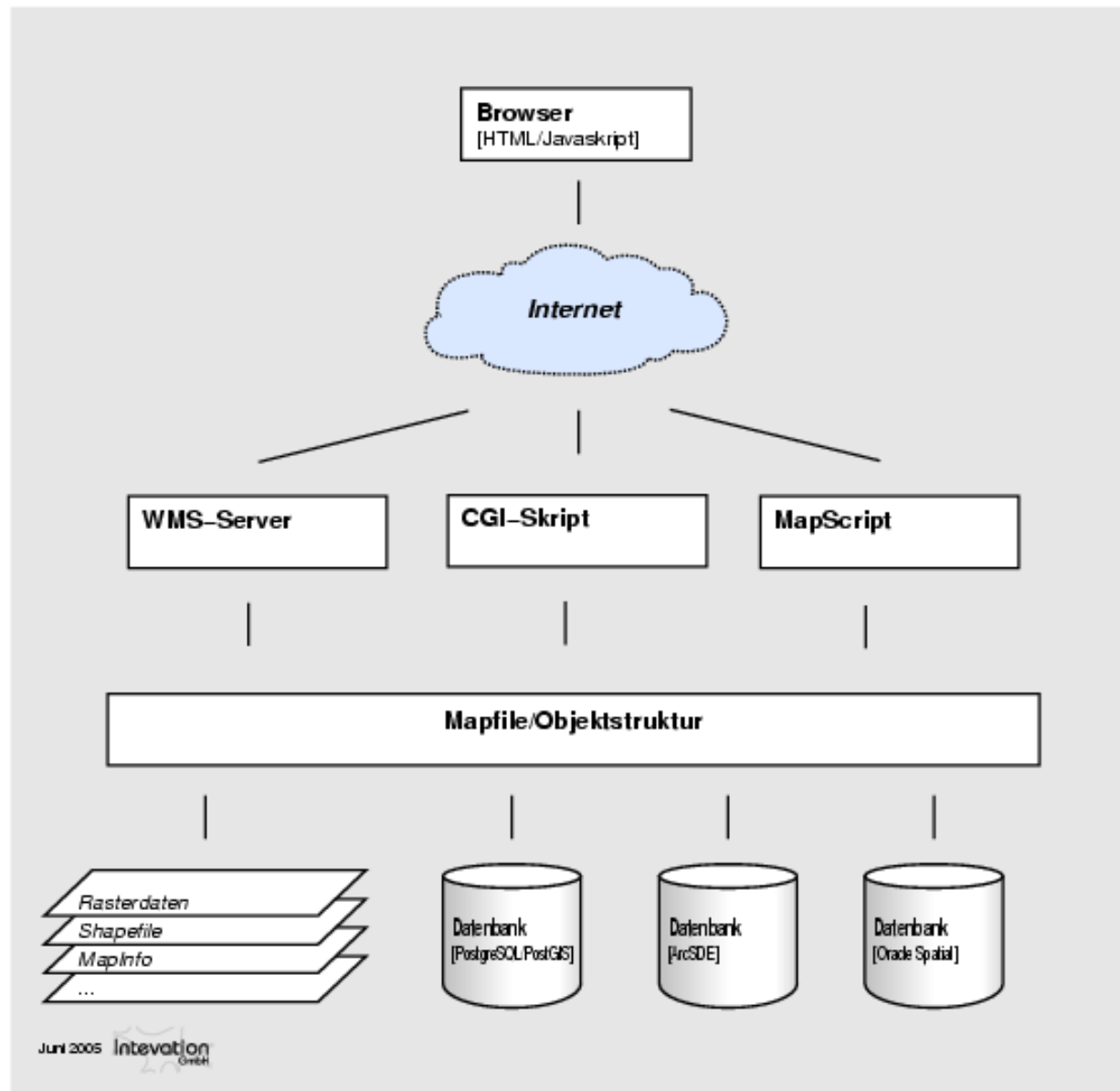
FOSS4G 2010  
Barcelona

- What is UMN MapServer?
- Mapfile-Hacking
- MapServer-based Services with GMAP and VE
- Cartodiagrams
- Performance-Tuning
- Debugging
- What's next?

# What is UMN MapServer?

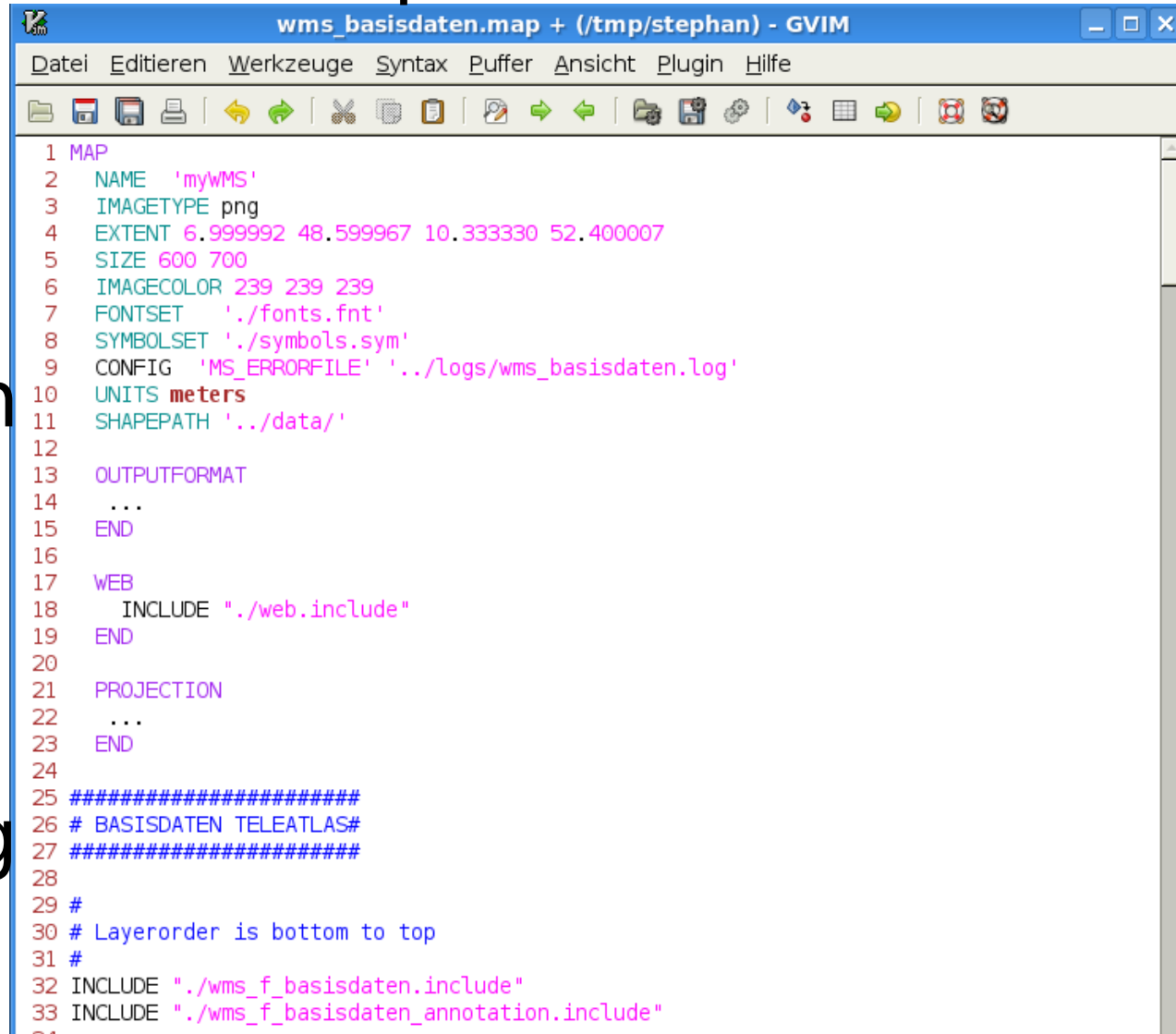
- C-based Mapping-plattform for creating spatial data and interactive mapping-components for the internet
- MapServer is **NO** complete GIS and it will not be in the future.
- Basically it's a rendering-software for creating maps
- MIT-style license, OSGeo-project
- plattform independend (Windows, Linux, MacOSX)
- Supports vector- und rasterformats as well
  - SHP, TIFF, GDAL/OGR, Oracle Spatial, Arc SDE, PostGIS and many more

# How does it work



# The Mapfile

- Configurationfile for MapServer
- Editable in your beloved text editor
- Must end with .map
- Usage of INCLUDEs
- Preprocessing
- Versioning



```
wms_basisdaten.map + (/tmp/stephan) - GVIM
Datei Editieren Werkzeuge Syntax Puffer Ansicht Plugin Hilfe
1 MAP
2 NAME 'myWMS'
3 IMAGETYPE png
4 EXTENT 6.999992 48.599967 10.333330 52.400007
5 SIZE 600 700
6 IMAGECOLOR 239 239 239
7 FONTSET './fonts.fnt'
8 SYMBOLSET './symbols.sym'
9 CONFIG 'MS_ERRORFILE' '../logs/wms_basisdaten.log'
10 UNITS meters
11 SHAPEPATH './data/'
12
13 OUTPUTFORMAT
14 ...
15 END
16
17 WEB
18 INCLUDE "./web.include"
19 END
20
21 PROJECTION
22 ...
23 END
24
25 #####
26 # BASISDATEN TELEATLAS#
27 #####
28
29 #
30 # Layerorder is bottom to top
31 #
32 INCLUDE "./wms_f_basisdaten.include"
33 INCLUDE "./wms_f_basisdaten_annotation.include"
34
```

- Preprocessing
  - Using any sort of preprocessors like cpp
  - Define styling using 'themes'
  - Usage of include-templates -> result is one mapfile (good for debugging)
  - split your mapfile in logical groups and INCLUDE
- Versioning
  - In combination with preprocessing easily maintainable
  - Complex projects make this a must!
  - Need I say more on versioning?

- Projection-block without EPSG-reference; add PROJ.4-string directly in your mapfile (it's fixed, not really needed any more)
- Use [MIN|MAX]SCALEDENOM-parameters
- Sort your CLASSes: frequently used classes on top!
- Simplify expressions
- Symbolset/Fontset: add only those fonts/symbols which are really used!

- Use optimized output-FORMATOPTIONS (depending on your data):
  - QUANTIZE\_FORCE=on
  - QUANTIZE\_COLORS=256
  - QUANTIZE\_NEW=on
- Hide layers from GetCaps which are useless there (e.g. Attributions, tileindex-layers, group-members) [currently requires patching, see #1952 for details]:  
`'OWS_HIDDEN_LAYER' 'TRUE'`



- Requirements:
  - MapServer compiled with PROJ.4-support
  - Every layer needs a PROJECTION-block
  - Spherical-Mercator-Projection:

```
+proj=merc +a=6378137 +b=6378137 +lat_ts=0.0 +lon_0=0.0 +x_0=0.0  
+y_0=0 +k=1.0 +units=m +nadgrids=@null +no_defs
```

- New parameters:
  - mode=tile
  - tilemode=[gmap|ve]
  - tile={x+y+z}|{10231}
- example: <file:///./beispiele/gmaps.html>

# GMAP-configuration

```
google_maps.html (/home/projects/Ge...ssgis2010/vortraege/beispiele) - GVIM1
Datei Editieren Werkzeuge Syntax Puffer Ansicht Plugin Hilfe
1 <!DOCTYPE html
2 PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
3 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
4 <html xmlns="http://www.w3.org/1999/xhtml">
5 <head>
6 <meta http-equiv="content-type" content="text/html; charset=utf-8"/>
7 <title>Google/MapServer Tile Example</title>
8 <script src="http://maps.google.com/maps?file=api&v=2&key=[YOUR KEY HERE]"
9     type="text/javascript"></script>
10 <script type="text/javascript">
11
12 function load() {
13     if (GBrowserIsCompatible()) {
14         var urlTemplate = 'http://demo.intevation.de/cgi-bin/frida-wms2?';
15         urlTemplate += 'layers=gewaesser_sehenswuerdigkeiten_strassenall&';
16         urlTemplate += 'mode=tile&';
17         urlTemplate += 'tilemode=gmap&';
18         urlTemplate += 'tile={X}+{Y}+{Z}';
19         var myLayer = new GTileLayer(null,0,18,{
20             tileUrlTemplate:urlTemplate,
21             isPng:true,
22             opacity:0.5 });
23         var map = new GMap2(document.getElementById("map"));
24         map.addControl(new GLargeMapControl());
25         map.addControl(new GMapTypeControl());
26         map.setCenter(new GLatLng(52.27, 8.04), 10);
27         map.addOverlay(new GTileLayerOverlay(myLayer));
28     }
29 }
30
31 </script>
32 </head>
33 <body onload="load()" onunload="GUnload()">
34     <div id="map" style="width: 500px; height: 500px"></div>
35 </body>
36 </html>
37
```

- Bar- and piechart-diagrams are supported for AGG and GD-Rendering
- New Layer-Type: TYPE 'CHART'
- Configuration is done with PROCESSING-directives inside the LAYER-block

```
LAYER
  NAME "Ages"
  TYPE CHART
  CONNECTIONTYPE postgis
  CONNECTION "blabla"
  DATA "the_geom from demo"
  PROCESSING "CHART_TYPE=pie"
  PROCESSING "CHART_SIZE=30"
  STATUS ON
  CLASS
    NAME "Population Age 0-19"
    STYLE
      SIZE [v1006]
      COLOR 255 244 237
    END
  END CLASS
  NAME "Population Age 20-39"
  STYLE
    SIZE [v1007]
    COLOR 255 217 191
  END
  END CLASS
  NAME "Population Age 40-59"
  STYLE
    SIZE [v1008]
    COLOR 255 186 140
  END
  END CLASS
END
```

- Optimize/minimize the EPSG-table to the really needed entries your app supports
- Set WMS\_EXTENT manually for your PostGIS-layers
- Use CLOSE\_CONNECTION=defer
- Use Fast-CGI
- Preprocess and prepare your rasterdata as
  - Uncompressed, Tiled, With overviews
- Use good/clever scalehints for your layers, add eventually overviews

- Set Mapfile-parameters:
  - DEBUG [level]
  - CONFIG „MS\_ERRORFILE“ „/tmp/ms\_errors.log“
- Use shp2img:
  - shp2img -m <mapfile> -l <layer> -o <output>  
-all\_debug 10
- Set Environment variables:
  - CPL\_DEBUG=ON: GDAL/OGR-debugging-infos
  - PROJ\_DEBUG=ON: PROJ4-debugging-infos
  - MS\_ERRORFILE=/tmp/file.txt: save debug-messages here

- Commandline-debugging of CGI-mapserver:

```
# export needed environment variables
export MS_MAPFILE=/path/to/mapfile.map
export QUERY_STRING="request=GetCapabilities\
    &Service=WMS&Version=1.1.1"
export REQUEST_METHOD=GET

# run the CGI-process (-nh means no header)
/path/to/cgi-bin/mapserv -nh
```

- Alternatively use cURL for sending requests, especially for POST-WFS-calls

# Where do I get help?

- OSGeo Infrastructure
  - Mailinglists (user/dev)
  - Trac (browse the tickets, there a some treasures included [as patches]).
  - SVN
  - IRC (#mapserver on freenode)
  - Wiki
- Community-conferences (e.g. FOSS4G, German FOSSGIS)
- Commercial partners

- Release 6.0 will (most probably) have
  - Pluggable rendering engines
  - Tile cache support
  - Output formats: GeoJSON, KML, GML3
  - Lots of minor stuff fixed
- More from the 'real' devs in the room!



# Thank you...



Intevation GmbH  
Stephan Holl  
<stephan.holl@intevation.de>  
Neuer Graben 17  
49074 Osnabrück  
+49 (0)541 - 335083 663