

Aggregating Remote Map Services with Local Cascading WMS Server

TOMAS BRUNCLIK
 Institute of Environment Protection
 University of Pardubice
 Studentská 84, 53210 Pardubice
 CZECH REPUBLIC
<http://mapserver.upce.cz>

Abstract: Map servers are mostly used to publish data owned by an organization to the Internet. In this paper we discuss possibilities of map server use in the opposite direction, where cascading OGC WMS map server on local network is used to aggregate and standardize access to several remote map services. This approach allows not only simplified access to selection of most used map services from one virtual source, but also to change many parameters of the cascaded services on the fly. It is also possible to transform some types of non-WMS map services to virtual WMS services with the help of a simple script.

Key-Words: Geographic information system; Map server; Map services; Cascading; WMS; Open GIS Consortium; Interoperability

1 Introduction

When using Geographic information systems (GIS), and especially when teaching GIS and its special applications, lack of GIS data is common problem. One of the solutions is to use map services of various public map servers on the Internet. During extensive use of such services with various desktop GIS applications and thin clients, other problems may arise, however:

First, while there is great number of available public map services, there are also many kinds of map servers. Although the Open GIS Consortium (OGC) standards as Web Map Service (WMS) [1] were already adopted by most major map server producers, it is still possible to find many map services not complying to these standards.

Second, even if there are two map services perfectly compliant with WMS standard, it is not guaranteed that it would be possible to open them together in one GIS project. It depends on properties of these map services, especially on available cartographic projections and coordinate systems of each, and also on capabilities of the client.

Third, working with several map services at a time may be a time consuming task. Especially searching for the right service and opening the layers from these services is not always as comfortable and takes more time compared to opening local GIS layers nicely organized on disk or in a database.

All these issues could be addressed by customized cascading WMS server located on local area network (LAN) or intranet. Moreover, it is possible to change some parameters of original map services not only to make them compatible enough to open together, but also to tune their properties for

intended way of use or for specific capabilities of used client applications or to combine layers from different services into one virtual layer on cascading WMS server.

2 Cascading Web Map Server

2.1 Common Cascading WMS Model

Cascading WMS server works as WMS client of the primary WMS server (Fig. 1). It sends its WMS requests, based on requests of its own WMS clients, to the primary WMS server. Then it cascades the content acquired from primary server back to the clients [1]. “Cascades” does not necessarily mean just “forwards”, cascading server can modify the content.

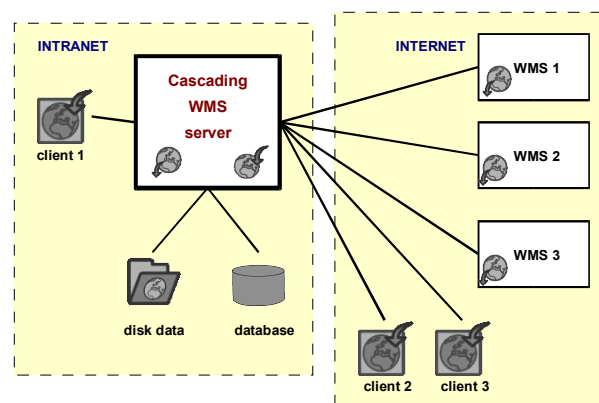


Fig. 1: Cascading WMS server.

2.2 Implementation of Cascading WMS

To address the issues mentioned in section 1, cascading WMS server had been implemented on the network of the University of Pardubice. Initial work of the implementation was part of thesis [2].

University of Minnesota (UMN) Mapserver [3] had been chosen as the WMS server software framework, mainly for its versatility and cofigurability.

Linux server with the Apache [9] web server had been dedicated to the map server. Links to its map services are available from site <http://mapserver.upce.cz>. The purpose of this web site is presentation of our GIS projects and support of GIS teaching on Institute of Environment Protection. Most techniques described in this paper are demonstrated in map service "pardubicko" available at the site.

Client software most used with the cascading WMS server is OpenJump [10], a java GIS desktop application.

All the software used in connection with this project is opensource [11] and is free to use.

2.2.1 Extended Cascading WMS model

In the Czech Republic, most public map services are based on ArcIMS by ESRI with a third party superstructure extension (in particular T-MapServer [4] and Mapmaker [5]) or on a stand-alone map server provided by a local commercial provider (e.g. WebMap [6]). Most of such servers include WMS-compatible interface in current versions, but not all currently available map services are always covered.

This situation, even worse back in 2004, brought need to find a way to provide connection to these non-WMS services from our WMS clients. The solution found is a slightly modified cascading WMS model with added capability to connect to specific non-WMS servers (Fig. 2).

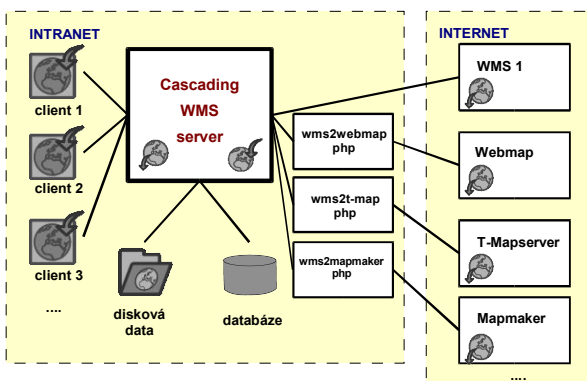


Fig. 2: WMS server extended to cascade non-WMS services.

As the main purpose of the cascading map service is to aggregate remote map services for better access from within the University, clients are shown only within the University network. It is currently available also from internet however, for the purpose of presentation of techniques described in this paper.

The solution is based on the fact, that all mentioned non-WMS map services use similar type of map requests as UMN Mapserver [7]. It is *http get* data request by universal resource locator (URL) with parameters specifying map extent, requested layers, file format and so on, just like in a WMS request. The parameters have only different names and sometimes different logic. Thus, it is possible to place relatively simple script between the cascading WMS server and the non-WMS server. This script translates WMS request parameters to the specific non-WMS map server request parameters and sends the modified request. Non-WMS server sends the requested map image back to the script and the script forwards it to the cascading WMS server.

What is different from normal WMS request of cascading server is the target of the request (local script instead of the target server) and one additional parameter specified in the UMN Mapserver .map file [8] connection URL (Fig. 3). This special parameter contains real target URL, that is, URL of the non-WMS server.

```
LAYER
NAME      zm10
...
CONNECTION "http://localhost/wms/import/wms2mapmaker
2.php?adresa=http://map.env.cz/mapmaker/cenia/map/mapa
_run.php&server=luthien.ceu.cz&service=ceu_b_rzm10_sde&"
...
END
```

Fig. 3: Part of .map file with Connection URL to wms2mapmaker2.php script.

From the point of view of WMS client applications, there is no difference when requesting map originating from WMS or non-WMS server. They send standard WMS request, based on standard WMS capabilities information, to the cascading WMS.

Main advantage of this approach is relative simplicity of deploying connectivity to new types of map servers. Unlike a full featured WMS connector, which would provide similar functionality, the script used with cascading WMS server does not have to implement transformation of GetCapabilities requests, because cascading WMS server is always configured with knowledge of target server capabilities. Thus, the script would be quite simple (Fig. 4).

```

<?php
$adresa=$HTTP_GET_VARS["adresa"];
$server=$HTTP_GET_VARS["server"];
$service=$HTTP_GET_VARS["service"];
$height=$HTTP_GET_VARS["HEIGHT"];
$width=$HTTP_GET_VARS["WIDTH"];
$bbox=$HTTP_GET_VARS["BBOX"];

list($x1, $y1, $x2, $y2)= split('[,]', $bbox);

$url = $adresa."?service=".$service."&server=\
".$server."&map_width=".$width."&map_height=\
".$height."&rect=".$x1.".".$y1.".".$x2.".".$y2;

header('HTTP/1.0 200 Ok');
header("Content-type: image/png");
$im = imagecreatefrompng ($url);
imagetruecolortopalette ($im, FALSE, 256);
imagepng ($im);
?>

```

Fig. 4: Script WMS2Mapmaker2.php (without comments)

Three php scripts were developed so far within the framework of the mapserver.upce.cz project “pardubicko” for non-WMS servers connection.

These scripts provide connection to the above mentioned map server types Mapmaker, WebMap and T-Mapserver.

2.2.2 Defining Projection Parameters and Reprojection on the Fly

It is sometimes not possible to display map layers from two different WMS map services together in a GIS application. Similarly, sometimes it is not possible to display layers from a WMS map service together with local map layers. The source of these problems is almost certainly a problem with projection parameters of map services used.

If the client used does not allow reprojection of map layers on the fly, it is not possible to display together two map services not offering common map projection. It is also possible, that a map service does not have defined the projection parameters correctly. If this is the case, it could be almost impossible to view layers of such map service together with local map data or layers of other map services.

These problems could be solved using cascading WMS map server configuration quite easily. The configuration file of UMN Mapserver [8] allows to define which output projections should the server offer, and what is the input projection of each cascaded service. If these projections do not match, the server will reproject the cascaded services on the fly in the same manner as with local map layers. This works also for services cascaded from non-WMS map servers as described in previous section.

The result is, that we get virtual map services with projections used in the organization, regardless of the source servers projection parameters.

2.2.3 Setting Transparency

To effectively combine map layers in a client application, the layers need to have their colors and transparency parameters set properly. Unfortunately, it is not always the case with layers from public map servers, especially with layers originating from raster data.

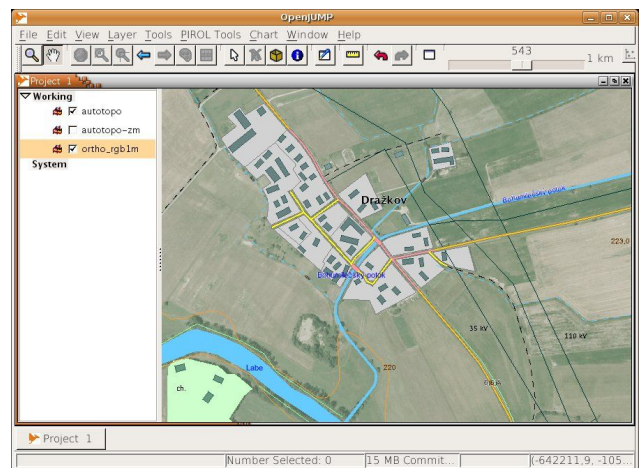


Fig. 5: Overlay of topographic and orthophoto map.

For example raster topographic maps contain usually considerable portion of white space, which do not provide any information. If white color would be set to be rendered as transparent, then it would be possible to overlay such layer for example with an orthophoto layer to get a background map providing much more information (Fig. 5). Topographic maps usually do not have set white color transparency on the public map servers, however.

```

LAYER
  NAME      zm10
  ...
  OFFSITE   252 254 252
  CONNECTION "http://localhost/wms/import/wms2mapmaker2.php?adresa=http://map.env.cz/mapmaker/cenia/map/mapa_run.php&server=luthien.ceu.cz&service=ceu_b_rzm10_sde&"
  ...
END

```

Fig. 6: Definition of color to be rendered as transparent.

It could be again solved by proper configuration of cascading mapserver. It is possible to set white (or other) color as transparent both globally and on per layer basis using the OFFSITE keyword [8]. In the example (Fig. 6), the RGB values of white areas of

cascaded topographic layer are not 255 255 255, which is the global transparent color for the whole map service. Thus, the per layer definition have to be used.

It is also possible to set overall transparency of any layer using keyword TRANSPARENCY in the .map file.

2.2.4 Utilizing Groups of Layers

When using background layers from a public map server, it can happen that the selected layer would not display throughout the whole range of map scale used in a GIS task, or that the layer is not suitable for the whole scale range by its detail level.

This problem could be solved by creating groups of layers with automatic scale-dependent exchange of source layers displayed in .map file of the cascading WMS server [8] (keywords GROUP and MINSCALE, MAXSCALE). Such group of layers can then be opened as one virtual layer by a WMS client.

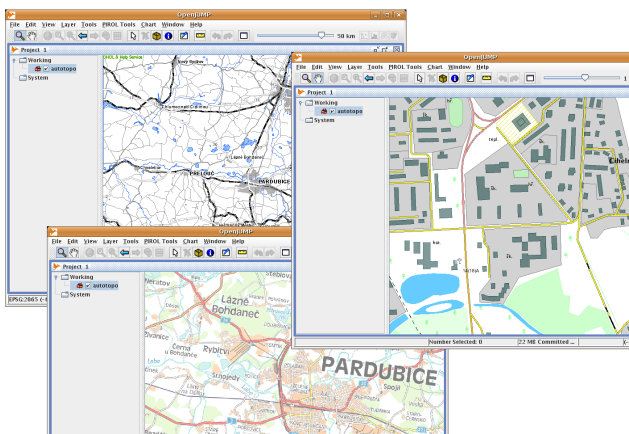


Fig. 7: Scale dependent rendering of virtual layer

In the example (Fig. 7), the layer “autotopo” displays different source layers depending on actual map scale.

Other way of utilization of group of layers is a composite layer, where overlay of two or more source layers is served as a virtual layer. Typical example would be a composition of orthophoto and topographic map, similar to that accomplished within a client in section 2.2.3 (Fig. 5).

In both cases, the source layers of the groups can be combined from local data and cascaded WMS and non-WMS services.

3 Conclusion

When using cascading WMS server to aggregate most used map layers from public map services within an organization, classroom etc., it is possible

to make work with these layers more effective and smooth with available client software.

Moreover, it is possible to extend range of services available to WMS clients by deploying connection to non-WMS services or changing the parameters of map services to be compatible with each other.

This all could be accomplished using completely free open-source software, as have been demonstrated in this paper.

Acknowledgements: This work could have come into being thanks to support from University of Pardubice.

References:

- [1] Beaujardiere, J. (ed.), *OpenGIS Web Map Service (WMS) Implementation Specification*, [online] version 1.3.0 [cit. 2006-12-11], <<http://www.opengeospatial.org/standards/wms>>
- [2] Mercl, T., *Creating of GIS Application about Living Environment*, Pardubice, 2004, University of Pardubice, Faculty of Chemical Technology, Institute of Environment Protection (thesis, supervisor: Brunclik, T.).
- [3] Lime, S., *Welcome to Mapserver* [online], last revision 2006-09-30 [cit. 2006-12-13], <<http://mapserver.gis.umn.edu/>>
- [4] Bradac, J., Novotny, M., *Municipal (G)IS in Czech Republic*, *European ESRI User conference*, 2003-10-10 [cit. 2006-12-13], Innsbruck, <http://www.tmapy.cz/docs/aktualne/clanky/esri_eu_2003.pdf>
- [5] Hradec, J., *MAPMAKER – Environmental Maps’ Server*, *ICC 2005*, La Coruña, Spain. 9-16 Jul. 2005 [cit. 2006-12-14], <<http://www.cartesia.org/geodoc/icc2005/pdf/oral/TEMA29/Session%20JIRI%20HRADEC.pdf>>
- [6] *Webmap* [online], last revision 2005-11-01 [cit. 2006-12-14], <http://www.wmap.cz/hydrosoft/eng/e_webmap.htm>
- [7] Koorman, F., Lime, S., McKenna, J., Morissette, D., *CGI Mapserver* [online], last revision 2005-12-12 [cit. 2006-12-21], <<http://mapserver.gis.umn.edu/docs/reference/cgi>>
- [8] McKenna, J., *Mapfile Reference* [online], last revision 2006-08-16 [cit. 2006-12-21], <<http://mapserver.gis.umn.edu/docs/reference/mapfile>>

- [9] *Apache Http Server Project* [online], last revision 2006-11-13 [cit. 2006-12-21], Apache Software Foundation, 2006 <<http://httpd.apache.org/>>
- [10] *What is OpenJump?* [online], last revision 2006-10-14 [cit. 2006-12-21], <<http://openjump.org/wiki/show/OpenJUMP>>
- [11] *Open Source Initiative OSI* [online], last revision 2006-12-22 [cit. 2006-12-22], <<http://www.opensource.org/>>