GeoServer - WMS - Simple Example

Added by Karolina Zdzitowiecka, last edited by Aleda Freeman on Sep 11, 2012

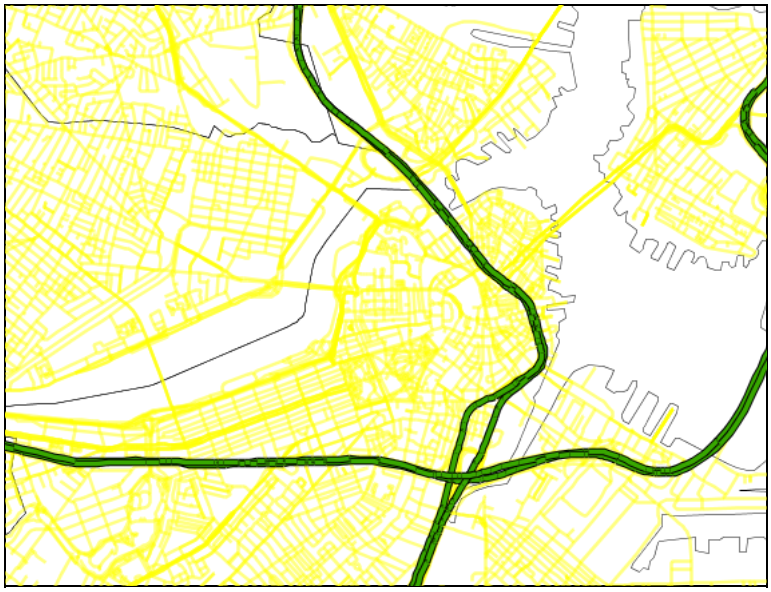
A Beginner's Tutorial to Web Map Service 'GetMap' Request

The request

The following URL is a WMS 'GetMap' request:

http://giswebservices.massgis.state.ma.us/geoserver/wms?VERSION=1.1.1&REQUEST=GetMap&SERVICE=WMS&LAYERS=massgis:GISDATA.TOWNS_POLYM,massgis:GISDATA.NAVTEQRDS_ARC,massgis:GISDATA.NAVTEQRDS_ARC_IN

Here is the map which it generates:



Viewing the map

- right-click the WMS URL above, and choose "copy link location" (in Firefox) or "copy shortcut" (in internet explorer)
- Open a new web browser window
- paste the URL you copied in step 1 into the URL bar and press enter

You should see a map with *__yellow__* streets, *__green__* interstates and *__black lines__* at town boundaries

__It's that easy.__ You just create a URL with the right parameters and load that URL in a web browser. The WMS will make your map for you and return it so your web browser can display it.

Breaking down the request

That big, long URL is actually made up of many small bits, separated by '&' characters.

Here is the request, broken up so that each bit is on its own line, and with bits re-arranged so they flow better:

```
http://giswebservices.massgis.state.ma.us/geoserver/wms?

VERSION=1.1.1&

REQUEST=GetMap&

SERVICE=WMS&

LAYERS=massgis:GISDATA.TOWNS_POLYM,massgis:GISDATA.NAVTEQRDS_ARC,massgis:GISDATA.NAVTEQRDS_ARC_INT&

STYLES=Black_Lines,GISDATA.NAVTEQRDS_ARC::ForOrthos,GISDATA.NAVTEQRDS_ARC_INT::Default&

BBOX=232325.38526025353,898705.3447384972,238934.49648710093,903749.1401484597&

SRS=EPSG:26986&

WIDTH=570&HEIGHT=435&

FORMAT=image/png&

TRANSPARENT=TRUE
```

The Parts of the Request

The protocol, host and path

```
http://giswebservices.massgis.state.ma.us/geoserver/wms?
```

The first line of the request specifies that we're connecting via *__HTTP__* to the server at the address *__giswebservices.massgis.state.ma.us, and we're accessing the path **/geoserver/wms__* on that server.

When making requests to the *__MassGIS WMS__* you will always use

- http://giswebservices.massgis.state.ma.us/geoserver/wms (http://giswebservices.massgis.state.ma.us/geoserver/wms?**)

as the protocol/host/path for your request.

Other WMS servers may have different host addresses and different paths, but most will use HTTP as their protocol.

The 'VERSION' parameter

```
VERSION=1.1.1&
```

The next bit of the request is the *__VERSION__* parameter. The WMS introduction page mentions that different versions of the WMS specification have been published. Each version has slightly different rules about how to interpret a request and how to return a response. We tell the WMS with the *__VERSION__* parameter that we want to use the rules defined in the WMS 1.1.1 specification.

In general, the rules are similar enough from specification to specification that it doesn't really matter which version number you use here. Feel free to stick with 1.1.1.

The 'REQUEST' parameter

```
REQUEST=GetMap&
```

The *__REQUEST__* parameter tells the server which operation you'd like to perform. The 'GetMap' request tells the server you want to fetch a map image. Other request types include 'GetLegendImage', 'GetFeatureInfo', and 'GetCapabilities'

To find out every request type for the WMS, read the WMS specification. It's really quite readable!

The 'SERVICE' parameter

```
SERVICE=WMS&
```

The *__SERVICE__* parameter tells the server which exact service you're sending your message to. In some cases, the service endpoint might work for multiple services, and this parameter could be used to specify whether you're sending your 'GetMap' request to the WMS or the WFS.

The 'LAYERS' parameter

```
LAYERS=massgis:GISDATA.TOWNS_POLYM,massgis:GISDATA.NAVTEQRDS_ARC,massgis:GISDATA.NAVTEQRDS_ARC_INT&
```

The *__LAYERS__* parameter is the meat of the request. It lists for the WMS the exact layers you wish to have drawn, as a comma-separated list. The order in which you list the layers is the //order in which they're drawn//. So layers *__listed first__* are *__drawn first__*.

In this case we're asking for the following layers:

- massgis:GISDATA.TOWNS_POLYM
- massgis:GISDATA.NAVTEQRDS_ARC
- massgis:GISDATA.NAVTEQRDS_ARC_INT

The 'STYLES' parameter

```
STYLES=Black_Lines,GISDATA.NAVTEQRDS_ARC::ForOrthos,GISDATA.NAVTEQRDS_ARC_INT::Default&
```

The *__STYLES__* parameter tells the server //how// to draw the layers you've specified in the *__LAYERS__* parameter. There must be exactly as many comma-delimited styles in the *__STYLES__* parameter as there are layers in the *__LAYERS__* parameter. The MassGIS WMS is smart enough to use a default style if you omit a style for a given layer, but the default style might not be suitable for your map. In addition, you must at least provide an "empty" style entry (a comma with nothing after it) for layers using the default style.

For example, the following would be an acceptable *__STYLES__* parameter for this request:

```
STYLE=Black_Lines,,
```

This would draw the first layer (massgis:GISDATA.TOWNS_POLYM) using the named style 'Black_Lines', and would use the default style for the other two layers.

The 'BBOX' parameter

```
BBOX=232325.38526025353,898705.3447384972,238934.49648710093,903749.1401484597&
```

The *__BBOX__* parameter tells the WMS what spatial extent to use for this map. It is specified as four ordered spatial points: //minx,miny,maxx,maxy//

So the map in this request is defined by a box with:

- the lower-left corner (minx, miny) at 232325.38526025353, 898705.3447384972
- the upper-right corner (maxx, maxy) at 238934.49648710093, 903749.1401484597

//*__NOTE:// These coordinates are *not__* lattitude/longitude. Nor are they UTM coordinates (if you know what those are). They are coordinates in the coordinate system defined in the next parameter.

For a good explanation of coordinate systems, see this page.

The 'SRS' parameter

```
SRS=EPSG:26986&
```

The *__SRS__* parameter tells the WMS which coordinate system the *__BBOX__* parameter is expressed in. The SRS parameter is written as an EPSG code.

There is a slightly tricky point to be made here:

- Most of MassGIS's data is stored natively in EPSG:26986 (Mass State Plane Meters, NAD-83)
- The *__SRS__* parameter does **not** specify what the native data projection is...it simply tells the WMS how to understand the coordinates in the *__BBOX__* parameter
- This allows you to request the map with a BBOX expressed in any supported coordinate system (and you'd be hard-pressed to find a coordinate system we don't support)

So to make a request using lat/long, you'd find out the EPSG code for lat/long (hint, it's EPSG:4326), and then make a request with the following parameters:

```
BBOX=your-min-long,your-min-lat,your-max-long,your-max-lat&SRS=EPSG:4326
```

The 'WIDTH' and 'HEIGHT' parameters

```
WIDTH=570&HEIGHT=435&
```

The *__WIDTH__* and *__HEIGHT__* parameters specify the width and height of the created image.

__//NOTE://__ The WMS //will// stretch images to comply with your request. If you want un-distorted images, be sure that the ratio of your width-to-height is the same as your ratio of geographic-x-extent to geographic-y-extent. Otherwise the WMS will try to squish a square map extent into a rectangular image, causing it to look "warped".

The 'FORMAT' parameter

```
FORMAT=image/png&
```

The *__FORMAT__* parameter specifies the format of the returned image. This parameter takes the value of the MIME type you wish for your image.

Supported output formats are:

```
application/pdf
image/png
image/jpeg
application/vnd.google-earth.kmz
image/svg+xml
image/geotiff
image/tiff
image/gif
application/vnd.google-earth.kml+xml
```

The 'TRANSPARENT' parameter

```
TRANSPARENT=TRUE
```

The *__TRANSPARENT__* parameter specifies whether areas of the map which are not otherwise drawn should be marked as transparent in the response image. Note that the JPEG standard does not support transparent pixels, so this parameter will be ignored for requests with the *__FORMAT__* parameter set to 'image/jpeg'.

Other parameters

There //are// other parameters which the WMS will handle. To see the full list, read the WMS specification.

Changing the Request

When making maps, you'll clearly want to change what the map displays from time to time.

The examples below show exactly how to change each parameter. We suggest you go through these examples in-depth, to see how the WMS request works.

- zoom in or out (change the BBOX parameter)
- make the map larger or smaller (change the WIDTH and HEIGHT parameters)
- change which layers are shown (change the LAYERS parameter)
- change how certain layers are displayed (change the STYLES parameter)

= Next steps=

After going through this tutorial you should feel comfortable

- Knowing what the WMS does
- Knowing how to change parameters in a WMS request to Zoom in, Zoom out or Pan the map
- Knowing how to change parameters in a WMS request to make layers "look" different
- Knowing how to change parameters in a WMS request to add new layers to your map
- Knowing how to take a WMS request and "see" the map it produces (by pasting the URL into a browser bar)

The next steps are

- Learning about the WMS capabilities document

None