

# Publication of Spatial Data via WMS in Marushka Design



**GEOVAP**

## CONTENTS

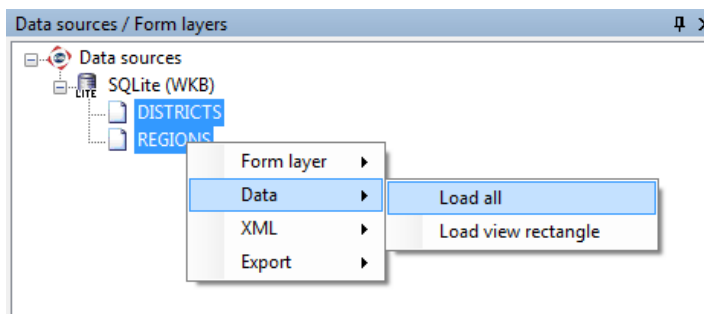
1	AIM OF THE EXAMPLE.....	2
2	WORKING WITH EXAMPLE .....	2
3	DIALOG BOX SAMPLE.....	3
4	A BRIEF DESCRIPTION OF THE EXAMPLE IN MARUSHKADESIGN .....	5

## 1 Aim of the Example

In this example we will demonstrate how to publish spatial data via WMS (web map services) in MarushkaDesign. This example was created in version 4.0.1.0, so it does not have to be compatible with older versions.

## 2 Working with Example

- Unzip the **WMS\_EN.zip** into **c:\MarushkaExamples\** folder. The target folder must be respected due to interconnection of paths with the project. In the case of placing the files in the different folder, it would not be possible to work with an example.
- Open the **WMS\_EN.xml** in MarushkaDesign environment.
- Select form layers **REGIONS** and **DISTRICTS** in SQLite (WKB) data store, in the context menu choose *Data – Load all*:



- In map window choose “Fit all”:



- Launch the local web server:



### 3 Dialog Box Sample

Fig 1: Publish layers structure in MarushkaDesign

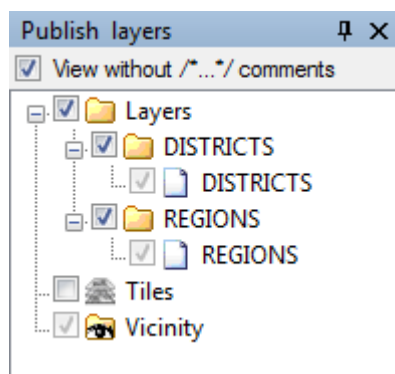


Fig 2: Query *GetCapabilities* result sample

Tento XML soubor nemá připojeny žádné informace o vzhledu prvků. Strom XML dokumentu je zobrazen níže.

```

- <WMS_Capabilities xsi:schemaLocation="http://www.opengis.net/wms http://schemas.opengis.net/wms/1.3.0/capabilities_1_3_0.xsd" version="1.3.0">
- <Service>
  <Name>WMS</Name>
  <Title>
  <Abstract/>
  <KeywordList/>
  <OnlineResource xlink:type="simple" xlink:href=""/>
- <ContactInformation>
  - <ContactPersonPrimary>
    <ContactPerson/>
    <ContactOrganization/>
  </ContactPersonPrimary>
  - <ContactAddress>
    <AddressType>postal</AddressType>
    <Address/>
    <City/>
    <StateOrProvince/>
    <PostCode/>
    <Country/>
  </ContactAddress>
</ContactInformation>
<MaxWidth>2560</MaxWidth>
<MaxHeight>1600</MaxHeight>
</Service>
- <Capability>
- <Request>
  - <GetCapabilities>
    <Format>text/xml</Format>
  - <DCPType>
    - <HTTP>
      - <Get>
        <OnlineResource xlink:type="simple" xlink:href="http://localhost:21574/MarushkaService/handlers/wms.ashx?ThemeId=15&" />
      </Get>
      - <Post>
        <OnlineResource xlink:type="simple" xlink:href="http://localhost:21574/MarushkaService/handlers/wms.ashx?ThemeId=15&" />
      </Post>
    </HTTP>
  </DCPType>
</GetCapabilities>

```

Fig 3: Query *GetMap* result sample, layer REGIONS

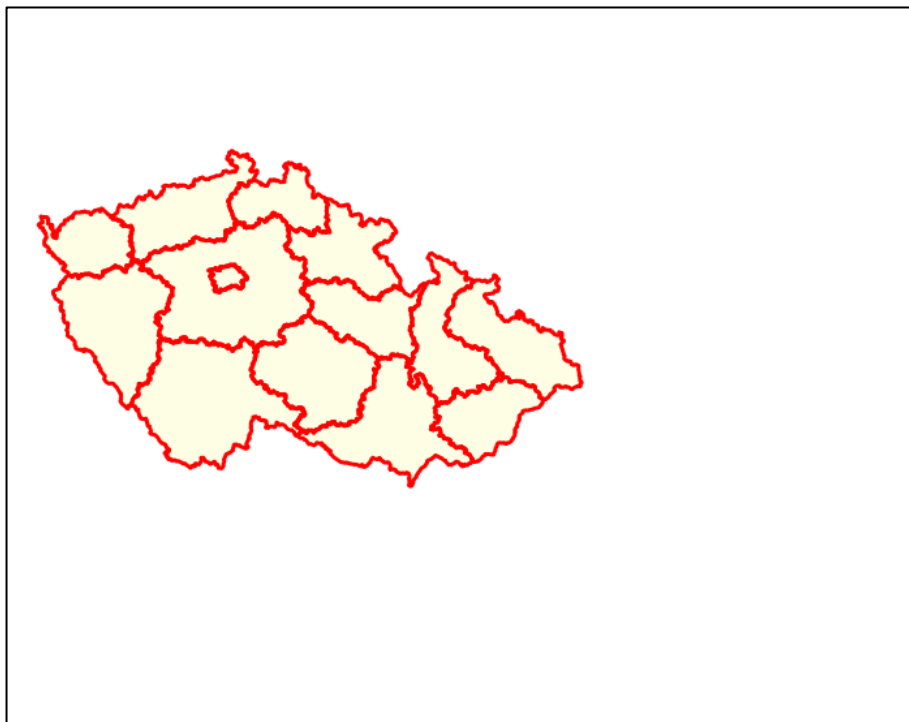


Fig 4: Query *GetFeatureInfo* result sample, layer DISTRICTS

Layer: DISTRICTS	
<b>Result of query: Info 2</b>	
id	110
geom	System.Byte[]
xmin	-775700.374819201
ymin	-1014786.99971609
xmax	-731894.249722404
ymax	-977541.500289403
NAME	Litoměřice
NAME_ENG	Litomerice
NAMEREG	Ústecký
NAMEREG_EN	Ustecky
<b>Result of query: Info 3</b>	
NAME	Litoměřice

## 4 A Brief Description of the Example in MarushkaDesign

### 4.1 Publication of Spatial Data via WMS server:

In MarushkaDesign, the spatial data are published through publish layers, which are named lists of physical layers for publication purposes, for HTML client and other supported services (WMS, WFS, TILE). The end user gets offered just these named lists - publish layers. Each publish layer can contain either one or more form layers displayed at different scales (this is set using *Map composition explorer*). For publish layers can be used wavy notation, for tree view of the individual layers. Publish layers structure in MarushkaDesign is shown in Fig 1.

Publish layers are uniquely identified by *Gid*, which is a unique code within the project. This code is generated automatically, you can change it, but you must remember that it must be genuinely unique. You can find it in the publish layers properties *1. Identifier and titles*, item *Gid*. This code is in the case of WMS layer publication, identifier of the individual publish layers.

For each item in the publication layers list, can be set, if the item will be published through WMS or not. It is set in publish layer properties *4. OGC Web Map Service*, item *AllowWMS*. If you want to publish specific item, set its value to "*True*", if you want to disable its publication, set it to "*False*".

If there are any information queries related to individual publish layers, it is possible to select which queries will be displayed in the publication. This can be set in publish layer properties *4. OGC Web Map Service*, item *GetFeatureQueries*, where can be assigned specific queries to specific publish layers.

WMS service can have authenticated access, i.e. it can require login and password. This authentication is usually handled by *Basic access authentication*, more [here](#). In MarushkaDesign it is possible to insert username and password directly into the URL. It is done in a way, that at the end of the URL is joined string *&UserName=[UserName]&Password=[Password]*, replaced with specific values for user name and password. This method of authentication is not encrypted, so it needs to be handled with carefully.

[Here](#) you can read the essential information and features of WMS service.

#### The general procedure for publication via WMS in MarushkaDesign:

- 1) Get any project that include at least one publish layer with at least one form layer.
- 2) For publication layers that you want to make accessible via WMS, set in publish layer properties *4. OGC Web map service* item *AllowWMS* to "*True*".
- 3) Launch the local web server.
- 4) Copy the link below and replace the local server address and the current theme id.
- 5) By this URL can be thus created WMS connected to any client supporting WMS services (e.g. GeoStoreV6), or to any other external WMS applications.

```
http://[current local web server
address]/MarushkaService/handlers/wms.ashx?ThemeId=[current theme ID]
```

The specific link may then look like this:

<http://localhost:21574/MarushkaService/handlers/wms.ashx?ThemeId=19>

### The procedure when using project WMS\_EN.xml

- 1) Open the project WMS\_EN.xml in MarushkaDesign.
- 2) Launch the local web server.
- 3) Copy the link below; replace the local server address and the current theme ID.
- 4) By this URL can be thus created WMS connected to any client supporting WMS services (e.g. GeoStoreV6), or to any other external WMS applications.

```
http://[current local web server
address]/MarushkaService/handlers/wms.ashx?ThemeId=[current theme ID]
```

## 4.2 Basic Request Types According to OGC

### A) GetCapabilities

If the client does not know the options and features of managed data, which happens in almost all cases, it must find out these options. Therefore, the client during the first communication with server compiles *GetCapabilities* request. The query must include parameter **REQUEST=GetCapabilities**, it requires also one more mandatory parameter **SERVICE=WMS**.

#### General procedure:

- 1) Open any project.
- 2) Launch the local web server.
- 3) Copy the link below and replace the local web server address and the current theme id:

```
http://[current local web server
address]/MarushkaService/handlers/wms.ashx?ThemeId=[current theme
ID]&REQUEST=GetCapabilities&SERVICE=WMS
```

#### The procedure when using project WMS\_EN.xml:

- 1) Open the project WMS\_EN.xml in MarushkaDesign.
- 2) Select form layers REGIONS and DISTRICTS.
- 3) Form layers -> Data -> Load all.
- 4) In map window choose display „All“.
- 5) Launch the local web server.
- 6) Copy the link below, replace the local server address and the current theme id and paste it into Web browser:

```
http://[current local web server
address]/MarushkaService/handlers/wms.ashx?ThemeId=[current theme
ID]&REQUEST=GetCapabilities&SERVICE=WMS
```

After a valid query compilation a XML file describing the service becomes accessible. These data are map server metadata i.e. they contain all the information needed to communicate with the service. Sample of the *GetCapabilities* request result is shown in Fig 2.

This file can be edited using theme editor, in category *Theme settings*. Here you can choose for example the default image type provided by WMS service (*DefaultWMSpngFormat*), restrictions of services (*AccessConstraints*), contact information (*ContactInformation*) and others. This file can be also saved and edited in any XML editor. Consequently, it is necessary to set an absolute path to this file in

the theme editor, *Theme settings* category, item *ExternalWMS130CapabilitiesPath*, respectively *ExternalWMSCapabilitiesPath*, according to the version of the WMS.

The specific link calling *GetCapabilities* service may then look like this:

<http://10.102.0.19:21574/MarushkaService/handlers/wms.ashx?ThemeId=19&REQUEST=GetCapabilities>

## B) GetMap

This type of request can be regarded as the main (primary), for the reason that the client makes a map in the form of image data in a specific format. Query URL must contain a parameter **REQUEST=GetMap**.

Specification (depending on the version of WMS) requires further mandatory parameters, which provide more detailed information about the displayed map.

### The procedure when using project WMS\_EN.xml:

- 1) Open the project WMS\_EN.xml in MarushkaDesign.
- 2) Select form layers REGIONS and DISTRICTS.
- 3) Form layers -> Data -> Load all.
- 4) In map window choose display „All“.
- 5) Launch the local web server.
- 6) Copy the link below, replace the local web server address and the current theme id and paste it into Web browser:

```
http://[current local web server
address]/MarushkaService/handlers/wms.ashx?&ThemeId=[current theme
ID] &%20&REQUEST=GetMap&SERVICE=WMS&VERSION=1.0.0&LAYERS=41210240&STYLES=&FORM
AT=image/png&BGCOLOR=0xFFFFFFFF&TRANSPARENT=TRUE&SRS=EPSG:102067&BBOX=-968691,-
1444332.50273973,-140888,-811573.497260274&WIDTH=730&HEIGHT=558
```

This link will after completing the current local server address and theme ID return the map image, adjusted by the parameters specified in the query. In the query above are defined parameters according to which will be the map image rendered, including limiting rectangle, height and width of the map window, transparency of layers, coordinate system, layers and others. It is also possible to define the order of displayed layers, but it is unnecessary because in MarushkaDesign the layer order is defined in *Map composition explorer* in the project configuration.

*GetMap* request result is shown in Fig 3.

The general procedure would be quite difficult to define, because each version of WMS requires defining also other parameters in the URL. It is also possible to either generate the link or to copy the values from a file generated using query *GetCapabilities*. The browser itself also reports when missing any of the required parameters, or if its value is not valid. It is also possible to manually add the following parameters, but it is rather time consuming.

Example of specific link calling request *GetMap* looks as follows:

<http://localhost:21574/MarushkaService/handlers/wms.ashx?&ThemeId=19&%20&REQUEST=GetMap&SERVICE=WMS&VERSION=1.0.0&LAYERS=KRAJE&STYLES=&FORMAT=image/png&BGCOLOR=0xFFFFFFFF&TRANSPARENT=TRUE&SRS=EPSG:102067&BBOX=-968691,-1444332.50273973,-140888,-811573.497260274&WIDTH=730&HEIGHT=558>



### C) GetFeatureInfo

This type of query returns the client XML file with attributes of the given element on the map on the specific coordinates. Query URL must contain the parameter **REQUEST=GetFeatureInfo**. Specification (according to the WMS version) requires further mandatory parameters, which will provide information to the map server about a specific request.

#### General procedure:

Configuration of the query is performed in two steps:

- 1) Create an information query on physical layer (DB table), which is defined by item *LayerName* of the information query).
- 2) In the publish layer properties for the layer which will publish this query via WMS, in section 4. *OGC Web Map Service*, set item *GetFeatureQueries*, (a visual component will ask to check the particular publish layer) and set item *AllowWMS* to "True".

Publish layer should contain a form layer (query), which generates element from the physical layer (DB table) as defined in the information query as *LayerName*.

Query and form layer must be defined within the same database data store.


#### The procedure when using project WMS\_EN.xml:

- 1) Open the project WMS\_EN.xml in MarushkaDesign.
- 2) Select form layers REGIONS and DISTRICTS.
- 3) Form layers -> Data -> Load all.
- 4) In map window choose display „All“.
- 5) Launch the local web server.
- 6) Copy the link below, replace the local server address and the current theme id and paste it into Web browser:

```
http://[current local web server
address]/MarushkaService/handlers/wms.ashx?ThemeId=[current theme ID]
&REQUEST=GetFeatureInfo&SERVICE=WMS&VERSION=1.0.0&LAYERS=4570FF5A&STYLES=&FOR
MAT=image/png&BGCOLOR=0xFFFFFFFF&TRANSPARENT=TRUE&SRS=EPSG:102067&BBOX=-
968691,-1444332.50273973,-140888,-
811573.497260274&WIDTH=730&HEIGHT=558&QUERY_LAYERS=4570FF5A&X=200&Y=150&
```

In Fig 4 is displayed the result of query *GetFeatureInfo* on layer DISTRICT.

*GetFeatureInfo* result can be also obtained otherwise than by typing fairly complex query in the URL.

In MarushkaDesign it is possible to use a button *Selection of elements from the map*  , substituting for the *GetFeatureInfo* query and returns the result in a HTML table in Information panel. In the case that there is more queries assigned to the same layer, it is possible to choose which query to display.

Example of specific link calling request *GetFeatureInfo* looks as follows:

```
http://localhost:21574/MarushkaService/handlers/wms.ashx?ThemeId=19&REQUEST=GetFeatureInfo&SERVICE=WMS&VERSION=1.0.0&LAYERS=KRAJE&STYLES=&FORMAT=image/png&BGCOLOR=0xFFFFFFFF&TRANSPARENT=TRUE&SRS=EPSG:102067&BBOX=-968691,-1444332.50273973,-140888,-811573.497260274&WIDTH=730&HEIGHT=558&QUERY\_LAYERS=KRAJE&X=200&Y=150&
```