
AJAX TECHNOLOGIES APPLICATION FOR GEODATA PUBLISHING ON THE INTERNET

Vojtěch Honzík¹, Tomáš Dolanský²

¹Department of geoinformatics, The Faculty of Environment, University of Jan Evangelista Purkyně in Ústí nad Labem,
Králova Výšina 3132/7,
400 96, Ústí nad Labem Czech Republic
vojtech.honzik@ujep.cz

²Department of geoinformatics, The Faculty of Environment, University of Jan Evangelista Purkyně in Ústí nad
Labem, Králova Výšina 3132/7,
400 96, Ústí nad Labem Czech Republic
tomas.dolansky@ujep.cz

Abstract.

The AJAX technologies (Asynchronous Javascript and XML) widely spreading during last two years took effect in expressive enhancement of user-friendly interface of web mapping applications. Google Maps or Mapy.cz can serve as the examples. AJAX technologies do have a number of benefits; the essential one is removing the need of reloading and refreshing of whole page after every event as necessary in standard html page. Properly design of the application and successful component selection results in measurable lower load of web server in geodata rendering and publishing as there is no need of linking and sending of whole html document but just the relevant part that needs to be changed. There is proven fact by general practice that using of ready-to-go web application (either free or commercial software) takes advantage only when no changes or minimum of them is required. In the case when the project requires large application logic changes seems to be wise to design application from scratch and minimize the use of particular software components.

In this article will be presented web application for publishing archival maps of Ústí nad Labem and its surrounding micro regions utilizing tiny AJAX, UMN Mapserver, Ka-Map, OpenLayers and PostGIS technologies and software components. Presented application takes advantage of tinyAJAX for fast HTML coding of AJAX enabled application. OpenLayers is object oriented library in Javascript library that allows without any dependencies on the server side integration of geodata from such different sources like Mapserver, Ka-Map, Google Maps, GML files or services like WMS or WFS. Only one source file from Ka-Map project was used to perform geodata tiling and server side caching. This combination of available free software components allows web application effective development using most of the technologies and methods usually used in commercial web mapping applications.

Keywords: web mapping, AJAX, tiles, modular application design

There are typical trends of short development cycles, implementation and wide spreading of new technologies in recent years. There is no other trend on the field of geoinformatics and Internet technologies globally. The development moves from the simple static images inserted into the source code of the simple markup languages towards usage of the scripting languages on the server side and the extensive usage of the client side proprietary web browser plugins. This approach brings several constraints to a user in a browser selection and the need of installing the third party plugins. Without these additional technologies there was almost impossible to support a sufficient interactivity and the server is being loaded with the repeated requests for the same spatial data.

The new approach to the web geodata access technologies has been enabled with the AJAX technologies (Asynchronous Javascript And XML) in recent two years. The common name is an envelope putting together web technologies that allow updating the web page parts without the need of performing full-page reload. The AJAX technology merges together the technologies for the information publication (HTML + CSS), the displaying and dynamic changes of document (DOM + Javascript) and the technology for asynchronous server-client data transfer (usually XMLHttpRequest).

This approach sets usual standard for web-based spatial applications, the usual functions and tools. These standards are de facto defined by the applications like Google Maps. If there has been made a good application design and component selection, there is a rapid decrease of a web server load in the geodata rendering and the HTTP protocol transfer as there is no need of sending a whole document but just the requested changes that could be reasonably small tiles in the properly designed structure for example.

The working experience shows that the implementation of an application that is ready-to-use (either commercial or free software) rarely fits the needs for particular area of interest and is applicable only in such a cases where there is no need of additional custom changes or if they are just the minor ones. In case the project requires the deep changes of the application logics there is wise to plot the application design from scratch and to minimize the amount of ready-to-use components into a maximally effective unit.

The AJAX applications development is possible without any additional tools but regarding to a high source code redundancy and combination of the different structural and markup programming languages is wise to ease the coding with some AJAX framework. For PHP as selected primary project language exist hundreds of frameworks and at least dozens of them contains AJAX oriented features. For the process of getting familiar with AJAX technology was selected TinyAjax framework [1] for its minimalist approach and the compact documentation.

Despite the fact that UMN Mapserver was and is the best free web mapping software and most probably will remain in the close future, its role in the complex applications has been changed thanks to well documented API from the role of main component into the separated tool for tiles rendering that are utilized by the next components. UMN Mapserver itself does not have any features for raster tiling support and even the tools for their administration. The lack of this feature is fully compensated using the software components like Tilecache or script tile.php – originally the part of web application Ka-Map. The above-mentioned components are usable not only for simple on-request tiling but for different zoom levels and different layers pre-caching.

When the component is correctly configured and there is enough storage space available, the component for tiles rendering and pre-caching is ready to use for rendering particular scale range for selected layers. Therefore there is no need to render requested tiles on demand, but the server can use the tiles from its hard drive cache – this approach gives much better response times and reduces server load. The data temporal validity can be set if the frequent updating of geodata is expected. When the data are requested later, the particular tile from cache is automatically replaced with the new one.

The hierarchical structure application is not limited only for particular web application. Despite of (or maybe due to) being a simple scripting tool there is possible to utilize this tool for the preliminary adjustment of large raster datasets for next processing or publication. Old maps can serve as an example as they usually have oblique oriented coordinate system axes when reprojected into any of recently used coordinate systems. Filling null value data area with transparent color is usually simple but a large area of particular non-collinear tiles is empty and makes the tiles large in dimensions and size as well. There is possible to process the raster data using this tool for publication over free or proprietary standards like Web Mapping Service (WMS) or ArcGIS.

In order to exploit the potential of asynchronous data transfer between server and client, the raster data has to be divided into a grid of adequate dimensions and an optimal scale sequence. If the tile would be extremely large, the effect of faster response time from the web service would not be much distinctive; in the case of the small tiles there would be an increase of the overhead costs on a server due to the multiple storage access, loading and parsing a tiles georeference and loading the particular tiles including the headers.

Regarding to the patterns of storing the data into computing memory and due to Geotiff format data structure there is most efficient to have the number of rows and columns equal to power of 2. Beside this there is no reason to have different number of rows and cells. Upper limit on tile size is set to 2048 pixels [2, p. 48], lower limit should be set upon the result of the selected testing project performed with the selected software serving

data over WMS protocol. Considering OSGeo (Open source Geospatial Foundation) suggestion [3], the tile size was set to 256 pixels and several preliminary tests has been performed. Regarding to API functions of UMN Mapserver featuring formal WMS requests and results was possible to perform the basic tests on WMS response times in dependence on generated image size. The test was very simple, consisted of the repeated requests of the area with the same bounding box and the scale numbers were of geometric progression with the scale factor 1.05 (5% increment in each step) so that the width and height grows in the same progression. The process was scripted and the batch had been performed ten times; the acquired times had were in agreement with hypothesis and the growth was quadratic. The test was automatically canceled when the response time was longer then 60 seconds – the default limit for PHP script execution on webserver. A logical step was to try the same testing procedure on pyramid structure tiles with the scale factor 0.5 - next zoom level had half resolution than previous one but physical size of tile was still the same. As for resampling method, there is obviously no preferred method and the selection is bound to the specific data character, purpose and target group users customs. In the case of an application for a general public there ought be need for a web oriented graphic designer to be a member of team.

Results of the test performed with the tiles in a scale range structure proved noticeable improvement in response time. There is still visible quadratic trend but before the response time can grow into unacceptable values for workaday traffic, next zoom level has turn. There is presumption that the results of second test was affected by the testing environment of web server – the server is VMware virtualized and has limited operational memory for testing reasons and because of the specific conditions; the factor of limited operational memory is crucial for the web services.

The option of pyramid structure not in separate files but in the tiles of highest resolution has been taken into consideration as well. The problem is technically simple as the GeoTiff raster format allows storing of encapsulated thumbnails and there are available utilities like gdaladdo from supportive tools of GDAL library [7]. There are two practical issues that obstructs such a procedure – the updating of raster datasets would be too complicated and there would be violated the limit on tile size [2]. There is necessary to figure out a multilevel directory structure in storing tiles for temporary and production purposes. The directory structure has to be designed in such a way to allow a simple maintenance and the theoretical maximum of tiles would not negatively affect the file system performance. The directory structure design has to be done conformable to the features of a specific selected file system with regards to the case studies of the available file systems – there are dozens of such a case studies.

Directory structure that has levels for the particular data sets, the zoom levels and the suitable selected coordinate intervals for both axes is usually sufficient. The coordinate interval is wise to set for example to round kilometers or tens kilometers regarding to the size of area of interest and original resolution. Tile storing system should be consistent for all layers of the map server for the purpose of easily repeated deployment of relevant parts of a source code. There had been used a set of ortogonalized aerial photographs of Ústí nad Labem and its suburban in the pilot project. The original resolution was 0.5 m/px. The selected structure for storing the tiles had following common format, coordinates are in rounded kilometers:

```
..\<dataset_name>\<lower left corner X coordinate>\< lower left corner Y coordinate >\
..\ortofoto_UL2\757\972\
```

Tiles files are named with following template:

```
<dataset name> <tile resolution> < x_LL>_< y_LL>
ortofoto_UL_0#50_757576_972112.tif
```

Tiles generated of any method can be therefore easily transferred into target location.

The key factor in web application development is decision whether the user interface would be developed „from scratch“ or would be wiser to use a third party component. Due to the fact that major part of web applications use virtually identical set of functions for moving around a map or functions for turning the layers on and off as an example, development of new user interface (with the exception of really specialized applications) seems not to be very effective.

The framework Openlayers seems to be the best one for a basic web application [4] – it has all of the basic tools that web-mapping application should contain. It could be easily modified - in the case of the pilot project aimed to the old maps covering Usti nad Labem and its suburban publication was the need of transparency scroll bar and layer order changes, these features can be considered to be crucial in old maps research. The biggest benefit of OnenLayers is a simple integration of the different geodata formats and the different service types. The list of

integrated formats is extensive – one can easily integrate the data from Mapserver, Google Maps, WMS and WFS services and from several others.

There is possible to utilize middle tier between Openlayers and the raster data itself to exploit the full potential of tile cache structure technology and its encapsulation. There are two equal tools available for this approach – the part of source code extracted from Ka-Map [5] (interesting application itself) and Tilecache [6], the specialized software on geodata caching conformable with WMS-C standard. Despite the fact that the software TileCache is a compact and advanced tool, it is not the preferred component in our pilot project despite of its benefits. The reasons are mostly irrational – not being familiar with the Python programming language in which is Tilecache written and that there is much bigger potential security risk in CGI application when not properly configured. The last decision to be made was the choice between direct geodata access and an access over another service. If the service were WMS, there would be another way for geodata publication and there would be easy to split the project into data storage and application parts not necessarily on the same server. This feature has not been implemented regarding to the testing status of the pilot project. All three application architecture scenarios are displayed on the figure 1. Thanks the facts that OpenLayers is fully AJAX enabled and the map window is inserted into a HTML file as an independent object, the final improvement in the response times and accelerated operations is obvious. All layers tiles are preloaded into the currently visible area. The order of tiles download depends on a layer order and a percentage of the tile area that is visible in a current view. After that the surrounding tiles are downloaded regarding to proper configuration. A small shift in a visible area does not affect the overall performance.

The progression in recent years shows that one approach in the future of the web based geodata publication is in the sophisticated combination of independent components that are capable to grant the similar functionality and robustness of commercial monolithic applications. The big advantage is that the developer can easily adapt the package to the specific needs and the available hardware without the need of the additional costs in a project budget.

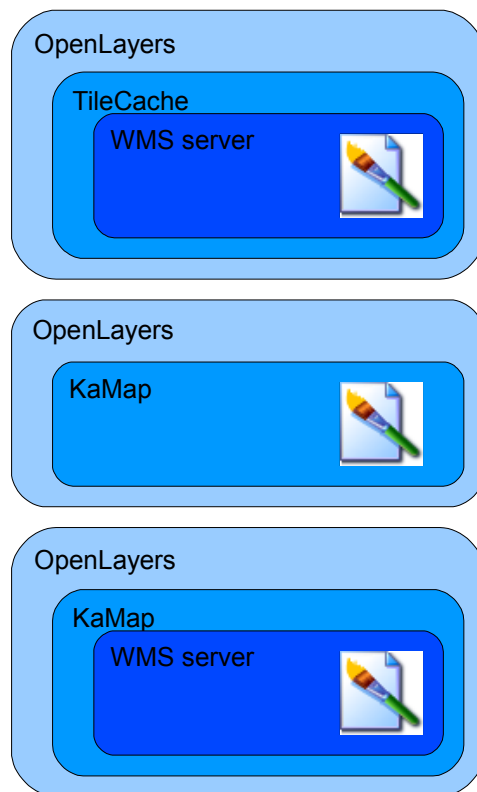


Fig. 1. Different approaches to the modular system of web based raster data publication

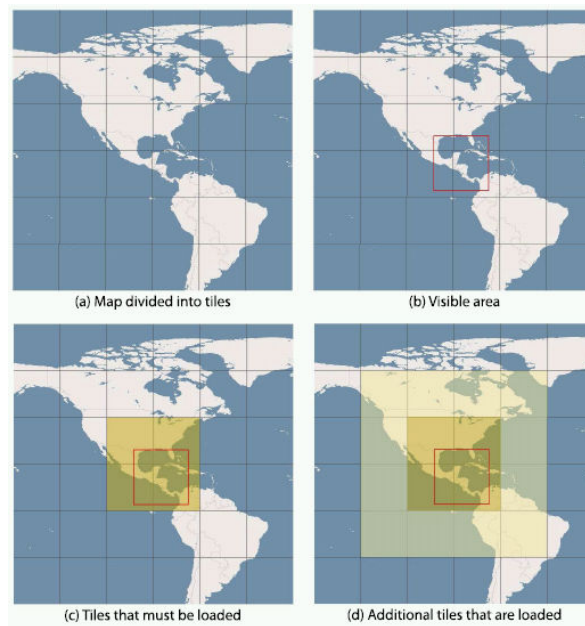
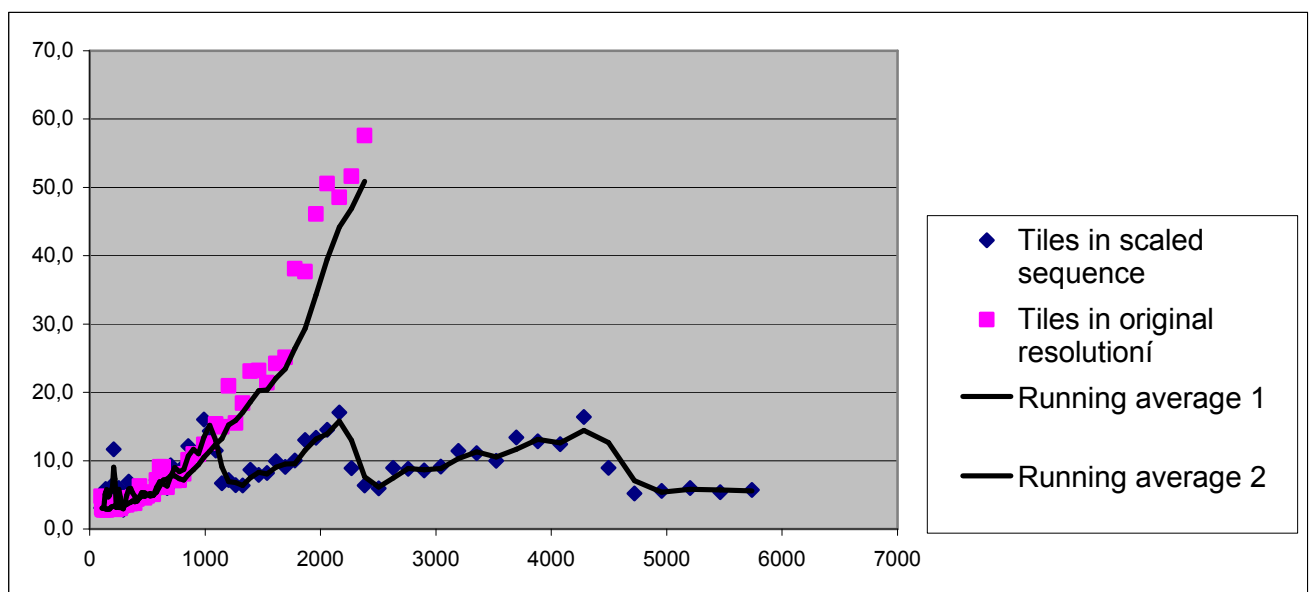


Fig. 2. Preemptive caching of the additional surrounding tiles illustration system of web based raster data publication



Graph. 1. WMS server response times comparison – original and optimized tiling approach

References

1. *TinyAjax* [online]. Mats Karlsson, 14.6.2006. [cit. 2007-10-26] Available from www: <<http://www.metz.se/tinyajax/>>.
2. EMANUEL, S. Current state of technology and potential of Smart Map Browsing in web browsers. [online]. 2007. , 1.6.2007 [cit. 2007-10-26] Available from www: <http://swm.wald.intevation.org/thesis_EmanuelSchuetze.pdf>.
3. OSGeo WMS Tiling Client Recommendation Smart Map Browsing in web browsers. *OSGeo wiki* [online]. 2007. , 3.11.2006 [cit. 2007-10-26]. Available from www: <http://wiki.osgeo.org/index.php/WMS_Tiling_Client_Recommendation>.
4. *OpenLayers* [online]. 2.5. OSGeo, 2007. , 1.6.2007 [cit. 2007-10-26]. Available from www: <<http://www.openlayers.org/>>.
5. *Ka-Map* [online]. 1.0. DM Solutions Group, 2007. [cit. 2007-10-26]. Available from www: <<http://ka-map.maptools.org/>>.
6. *TileCache* [online]. MetaCarta Labs, 2006. [cit. 2007-10-26]. Available from www: <<http://www.tilecache.org/>>.
7. FRANK, W. *GDAL - gdaladdo utilita* [online]. 2007. , 1.6.2007 [cit. 2007-10-26]. Available from www: <<http://www.gdal.org/gdaladdo.html>>.