CO-329

CARTOGRAPHIC ENHANCED GEOPORTALS

TOOMANIAN A., HARRIE L., OLSSON P.O.
GIS Centre, Dept of Earth and Ecosystem Sciences, Lund University, LUND, SWEDEN

1. INTRODUCTION

1.1 Background

The demand for web based spatial data applications are increasing rapidly. A wide area of web based applications, initiated the requirement to disseminate spatial data to the end-users by the use of geoportals. They support searching, viewing and downloading spatial data. In this study we concentrate on the view services.

Cartography is an important issue in the geoportal view services. The possibility to overlay geospatial data layers from different sources requires symbolization methods that support visual integration. One important issue is the visual hierarchy that ranks various data according to their relative importance; that is, data layers that are more important for the application should be visually emphasized. Another important issue is that information in one layer should not obscure or hide vital information in other layers (Figure 1).

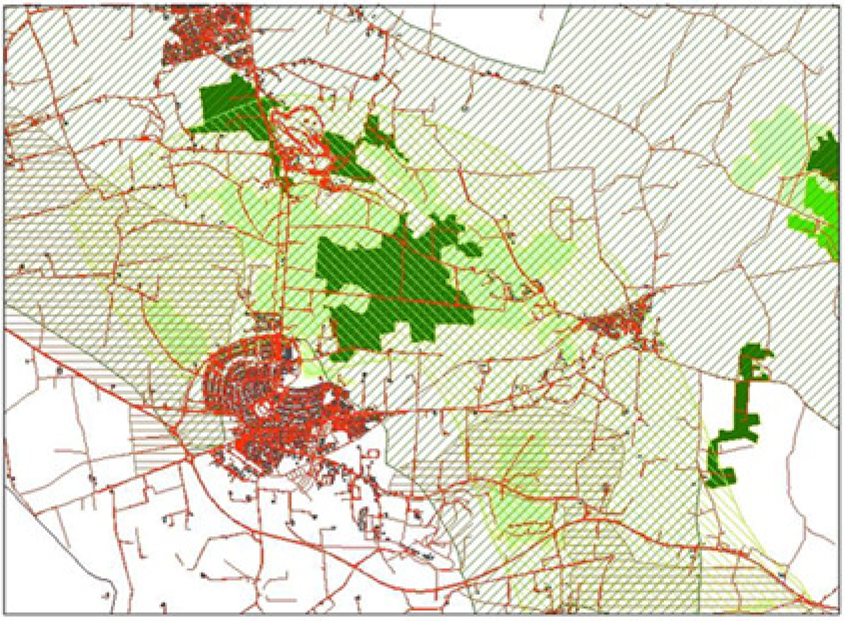

Figure 1. Protected area information placed on top of a base map for planning purposes. Here the protected areas hide important information in the base map.

There are several approaches of handling cartographic presentation in a view service of a geoportal. The most restricted approach, from an end-user perspective, is to only allow selection of layers where the symbologies of all layers are predefined. To provide the end-user with more capabilities he/she is allowed to set the relative importance of each layer (e.g. if a layer is placed in the background, middle ground or foreground in the visual hierarchy). It is also possible to define several symbologies on the geoportal which gives the end-user the possibility to choose between different symbologies. The fourth and final approach is to give the end-user capabilities to create his/her own symbology for each data layer. The methods in this paper are mainly targeting the second approach; the end-user only has the option to set the relative importance of each layer. However, it would also be possible to adjust the presented methods to work with the two latter approaches where the end-user has more freedom.

1.2 Objectives

The aim of this study is twofold. The first aim is to develop and implement a system architecture of a cartographic enhanced geoportal. The second aim is to develop and implement two methods that enable good cartography when combining several data layers in a geoportal. The first method, *polygon overlay method*, aims at establishing a good presentation of a polygon that is overlaid a base map (cf. Figure 1). The second method, *colour saturation method*, aims at creating visual hierarchies by manipulating the colours.

1.3 Related works

In this study we create cartographic support to an end-user by developing a system architecture and two methods; in all parts the study has borrowed ideas from other studies.

Extended geoportals/ clearinghouses

This paper proposes a cartographic enhanced geoportal; however, there are several attempts to enrich the functionality of clearinghouses and geoportals. Mansourian et al. (2010) proposed to complement clearinghouses with expert systems and semantic matching. The expert system facilitates automatic determination of candidate data sets and conversion of available data to the desired data. A schema translator is also used to find similar data that might be used in other disciplines or other datasets by semantic matching. De Longueville (2010) presents how geoportals can benefit from the Web2.0 features. He provides an overview to support the next generation geoportal development by defining connected concepts, emphasising on the pros and cons of this approach, and proposing suitable implementation strategies.

Cartographic support

Iosifescu-Enescu et al. (2009) utilized an enriched cartographical approach for OGC standards to fulfil the complex visualization requirements coming from environmental management. They used cartographic extensions for expressing cartographic rules with spatial operators and advanced feature filtering for layer masking, flexible point symbolization, and patterns and gradients for all spatial features. By using such an idea, the critical point of creating thematic maps is also solved with extensions for intuitive choropleth and various diagram types generation. Bucher et al. (2007) address some cartographic issues for designing efficient on-demand maps in service oriented architecture (SOA). According to their study, current standards do not support some crucial steps of an on-demand map design process and may greatly benefit from knowledge that has been formalised by cartographers such as the definition of styles to draw geographical data on the map. They aim at integrating parts of the cartographic knowledge, mostly semiologic rules, in web map service oriented architectures. They also employ a set of web services dedicated to facilitating the definition of accurate legends with respect to user objectives and data. Brewer & Buttenfield (2007) provides methods to create a map from a multiple representation database. They emphasize map display changes using symbol design or symbol modification. In addition, it comprises a demonstration of establishing specific map display scales at which symbol modification should be imposed.

Polygon overlay method

The polygon overlay method in this study, is based on representing a polygon with icons. This method is partly based on previous studies: Harrie et al. (2004) developed a method to place icons in a least disturbing position and Harrie & Revell (2007) developed a method to place icons in a semirandom pattern. In the latter study the basic idea was to optimize the distance between the neighbouring icons in such a way that the result mimics manual placement of icons. This way of describing pattern using neighbouring objects have also been used for characterizing dot maps (Ahuja, 1982; Sadahiro, 2000). Another study was conducted in the context of mobile cartographic services (Edwards and Burghardt, 2004). They stress the spatial relationships between icons and the base map features and propose solutions through better models of spaces. To perform this, they use a combination of generalization techniques and spatial modeling.

To optimize the placement of icons in this study, we use a combinatorial optimization approach in a similar fashion that is made in label placement (cf. Zoraster, 1986; Christensen et al., 1995; Zoraster, 1997; Ware and Jones, 1998; Zhang and Harrie, 2006).

Colour saturation method

Chesneau et al. (2005) established a method based on Itten colour contrast theory where each graphic sign analyses colour contrasts with its neighbours. Then, this analysis is validated at a more global level. If problems in colour contrast are detected, another graphic solution is proposed. The process is repeated until a more legible map is obtained. Also Buard and Ruas (2009) proposed a number of processes to improve colour contrasts of topographic on demand maps.

2. APPROACH AND METHODS

This section starts with a description of system architecture of a cartographic enhanced geoportal; then follows a description of two cartographic methods to integrate several data layers.

2.1 System architecture

To support geoportals with a good cartography we propose a system architecture based on following of components (Figure 2):

• *Client* is a normal WMS-client with added functionality to create a visual hierarchy (it enables the end-user to define if a layer is in the back-, middle- or foreground, cf. Figure 3).

• *Registry service* manages the registry of spatial services to be used by the geoportal.
• *Cartographic enhanced geoportal* is a geoportal with added functionality to enable good cartography. The geoportal consist of three components:
- *Web map* program is an entry point to geographic data on the web. It is a web site on the Internet where download and view services are registered. It provides users (clients) with the capability of viewing and/or developing spatial data.
- *Cartographic core* determines the symbolization of the user-selected layers. In this component several cartographic methods can be implemented.
- *The symbolization library* contains one (or several) symbolization(s) for each dataset that is registered in the geoportal.
• *Geospatial web services* are standard services for distributing geographic data (WFS).

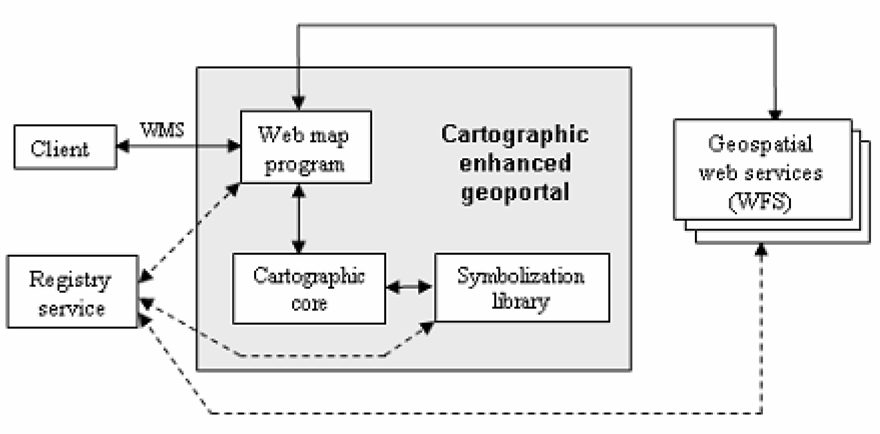

Figure 2. System architecture.

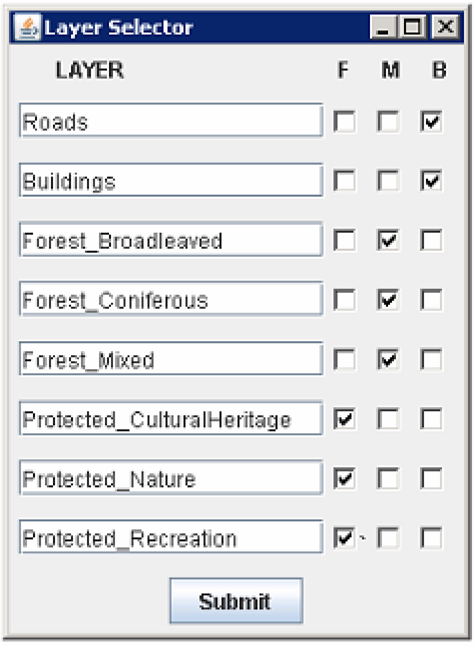

Figure 3. Graphical interface for selecting hierarchical levels for the data layers (B-background, M-middleground and F-foreground). The selection of layers created the map in Figure 1.

It is required that all geospatial web services to be registered in the geoportal. In this registration process metadata about the service (including web addresses) are stored in the web map program project files. It is also required that symbolizations are stored in the symbolization library and information about geometry type in the geospatial web services.

For the end-user the cartographic enhanced geoportal is similar to any other geoportal. The only difference is that the user is requested to state in which level in the visual hierarchy a layer should be. The following levels are defined (cf. example given in Figure 3):

• Foreground – Additional information layers that are of high relevance for the application.

• Middle ground – Data layers in the base map that are vital for the application.

• Background – Less prominent data layers in the base map.

The ordering of the layers in the final map (that is sent to the client) is decided by the cartographic core. The rules are:

• A layer in the foreground is always on top of a layer in the middle ground, and a layer in the middle ground is always on top of a layer in the background.

• Within each level (back-, middle- and foreground) point layers are on top, line layers in between and the polygon layers are at the bottom. The ordering of two layers with the same geometry in the same level is set according to the order in the WMS *getMap* request.

To enhance the feeling of a visual hierarchy, special methods can be implemented in the cartographic core. Below we propose two such methods:

1. *Polygon overlay method*: This method is applicable for polygon layers that are defined to be in the foreground. This could typically be planning regions, restricted areas etc. in a planning application.

2. *Colour saturation method*: The aim of this method is to deemphasize data in the background by decreasing the saturation in their symbology.

2.2 Polygon overlay method

One common problem is when polygons in the foreground hide important information in the middle and/or background (Figure 1). In this section we propose that the polygons can be symbolised by a combination of a boundary line and icons, where the icons are placed so they do not obscure vital information. This method does not make any changes to the base map.

The polygon overlay method is defined as a combinatorial optimisation problem and consists of the following steps:

1) Initialisation step – The polygon is symbolised by the boundary and icons placed in preliminary positions.

2) Cost function step – A cost function is created for the positions of the icons.

3) Optimisation step – In this step the positions of the icons is determined by finding the optimal solution to the cost function.

Two elements are utilized for the whole process. First, we use a *minimum bounding box* (MBB) to cover the whole polygon as a frame and all other steps are operated within this area. Within the MBB we create a fine grid. This grid will act as the resolution of the icon placement. The icon centre can only be placed in a centre of a cell in this grid (cf. Figure 4) and the size of the icon ($is_{x,y}$) is defined as:

$$is_x = (2k+1) \cdot \Delta x, \forall k \in Z \qquad (1)$$

where Z is an integer, Δx is the resolution in x–direction in the fine grid and is_y is defined analogously.

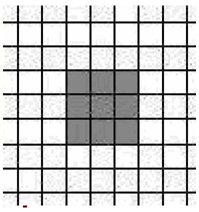

Figure 4. The relation between the icon and the dense grid. Here, k=1 in Equation (1).

2.2.1 Initialisation step

The main task in the initialisation step is to decide the number of icons and to give them preliminary positions. To perform this, the fine grid covered by the MBB is utilised. First, a coarse grid, with size $cg_{x,y}$ is defined as:

$$cg_x = (2k+1) \cdot \Delta x, \forall k \in Z \qquad (2)$$

where Δx is the resolution in x–direction in the fine grid and cg_y is defined analogously.

Secondly, icons are placed in all cell centres, in the coarse grid, which are within the polygon (Figure 5).

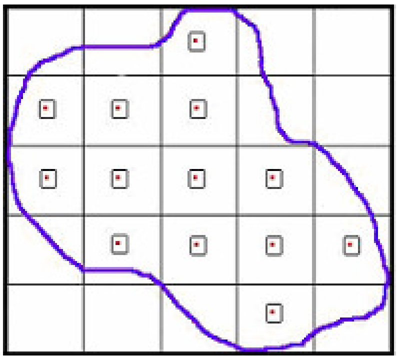

Figure 5. Initial positions of the icons.

2.2.2 Cost function step

To find proper positions for the icons we have to consider several aspects. Our approach is to associate all these aspects with a cost and then find the solution with the lowest total cost. We consider the following costs for the icons: placement cost, spatial distribution cost and removal cost.

Placement cost

Placement cost calculates the cost for hiding and distributing other objects by the icons. The cost computations are based on the fine grid. We also introduce two terms: *object cost* and *cell value*.

Each object creates an object cost around the symbology in such a way that the cost decreases with the increase of distance from the symbology in a linear fashion until it reaches a threshold value (Figures 6). Figure 7 shows an example of the object cost where we focus on a part of the road layer (green) and a building (red) to describe the method in details. In this example, the cells with white colour with minimum cost (e.g. ‘0’) are highly ranked for icon placement and the cells with dark colours are not suitable.

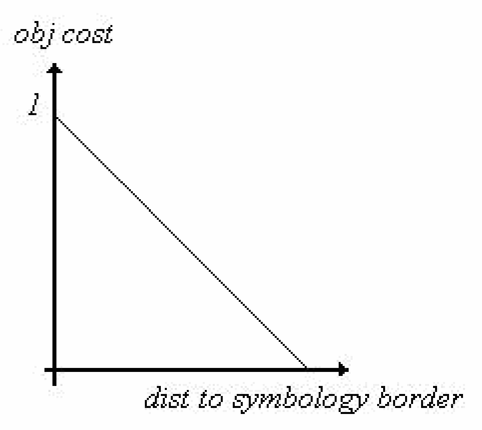

Figure 6. The relation between an object cost and the distance to symbology border

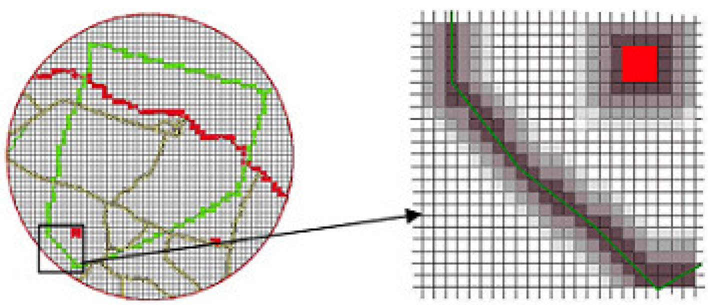

Figure 7.Illustration of the object cost.

The cell value is the value for each cell in the fine grid. We define a cell value for cell k (cv_k) as:

cv_k =999 if the cell overlaps any symbology

else

$$cv_k = \sum_{i=1}^{n} object\ cost \qquad (3)$$

where n is the total number of objects.

According to the definition of a cell value, first, cells that overlap with any symbology (even if a part of the symbology such as the width of a road) are set to a high cost (e.g. ‘999’). That means they are not a candidate for icon placement.

Each icon covers a number of cells, and the cost for each icon depends on the cell values for all cells covered at a certain position. We use the following function to calculate the placement cost (c_p).

$$c_p = \sum_{i=1}^{n} \sum_{j=1}^{m_i} cv_{ij} \qquad (4)$$

where c_p is the placement cost, n is the number of icons, is the number of cells covered by and cv_{ij} is the $cell_j$ covered by $icon_i$.

Spatial distribution cost

Spatial distribution cost is stated to describe the difference between the distances in ideal and other places when moving an icon from the centroid to another location in a cell. Here, the cost is calculated as:

$$c_{sd} = \sum_{i=1}^{n} |d - id| \qquad (5)$$

where c_{sd} is the spatial distribution cost, d is the distance between neighbouring icons, id is the corresponding ideal distance and n is the number of neighbourhood relationships.

In this step we define a neighbouring icon as an icon lying in any of the eight neighbour cells in the coarse grid (Figure 8).

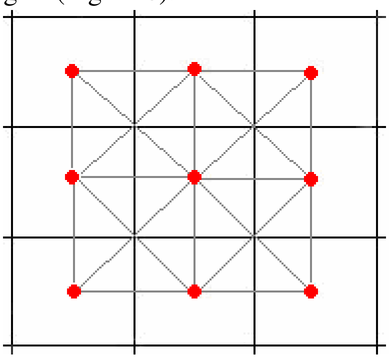

Figure 8. Definition of a neighbourhood relationship

Removal cost

Removal cost analyses the situations where a number of icons are too close due to the movement in previous steps (spatial distribution cost). Here, if the distance reaches a minimum threshold then either one or some icons have to be removed. Also, all neighbourhood relations are removed as well. Removal cost is calculated as:

$$c_r = Constant\ value\ for\ each\ icon * numbers\ of\ removed\ icons \qquad (7)$$

where c_r is the removal cost.

2.2.3 Optimisation step

Total cost via the following cost function:

$$tc = w_p c_p + w_{sd} c_{sd} + w_r c_r \qquad (8)$$

where tc is the total cost and is the weight for each cost achieved from the optimization step.

The cost function can act as an objective function in a combinatorial optimization. To optimize the cost function, we apply the combinatorial optimization method simulated annealing (see e.g. Russell and Norvig, 1995). This method is a stochastic hill-climbing algorithm that provides the possibility to escape from local minima.

The expected output of the polygon overlay method is illustrated in Figure 9. In this figure, the background layers are roads and buildings and there are two foreground layers: protected nature and protected recreation (cf. Figure 3).

2.3 Colour saturation method

A major problem in maps from a visualization perspective is using high saturated colours in base maps and background layers. To solve this problem, colour saturation method is used to deemphasize background layers in this study.

The colour saturation method is simple but it needs supplementary computation and conversion. Here, the initiative is to decrease the colour saturation with a specific range (e.g. 20%) or any constant values for each symbol in the background layer. The procedure for the conversion is as follows (assuming colour stored as RGB values):

1. Convert the colour format from RGB to HSV (cf. Foley et al., 1996).
2. Decrease the saturation according to the predefined ranges.
3. Reconvert the new colour to the RGB format.

By using these steps the background colours are less distinct so that one can focus on the foreground layers.

3. CASE STUDY

To evaluate the cartographic enhanced geoportal a prototype system was implemented according to the architecture in Figure 2, and a case study was performed in the municipality of Lund, southern Sweden where an urban planning map was created (cf. Figure 1).

3.1 Data

• Road- and building layers are provided by the municipality of Lund.

• Polygon layers from the County administrative board of Skåne (http://www.gis.lst.se/lstgis/): Protected areas of national interest for (1) cultural heritage, (2) recreation and (3) nature.

• Corine land cover vector data (http://www.eea.europa.eu/data-and-maps/data/): Broad-leaved forest, coniferous forest and mixed forest.

3.2 Implementation

The implementation consists of a client, a cartographic enhanced geoportal and external web services (Figure 9). The communication between these components follow OGC standard, but since a user needs to define the visual hierarchy an additional parameter is added to the *GetMap* request. This parameter (LayerPriority – *fore*-, *middle*-, or *background*)) acts as a WMS *Vendor Specific Parameter* (VSP) (OGC, 2010b). When including a VSP the service may choose not to include it in a capabilities response, which is the case in this implementation.

• Client is a WMS client written in Java with added functionality for a user to select if a layer should belong to fore-, middle-, or background (Figure 3).

• Cartographic enhanced geoportal consists of three components:

• Web map program is *MapServer* (MapServer, 2010) that is run, both via Common Gateway Interface (CGI) and Java MapScript API.

• Cartographic core has two methods implemented: (1) the polygon overlay method that utilises *OpenJUMP* (OpenJUMP, 2010) to convert GML-files to Well Known Text (WKT) format and *Java Topology Suite* (JTS) (JTS, 2010) for the geometry computations, and (2) the colour saturation method that is a Java program.

• Symbolization library consists of *Styled Layer Descriptor* (SLD) documents (OGC, 2010a).

• External web services are normal OGC WFS services. In this study two WFS-services are used, running on two different platforms: *Geoserver* (GeoServer, 2010) and Mapserver. Both store their data in *PostgreSQL* databases (PostgreSQL, 2010), with *PostGIS* extensions to allow storage of spatial data (PostGIS, 2010).

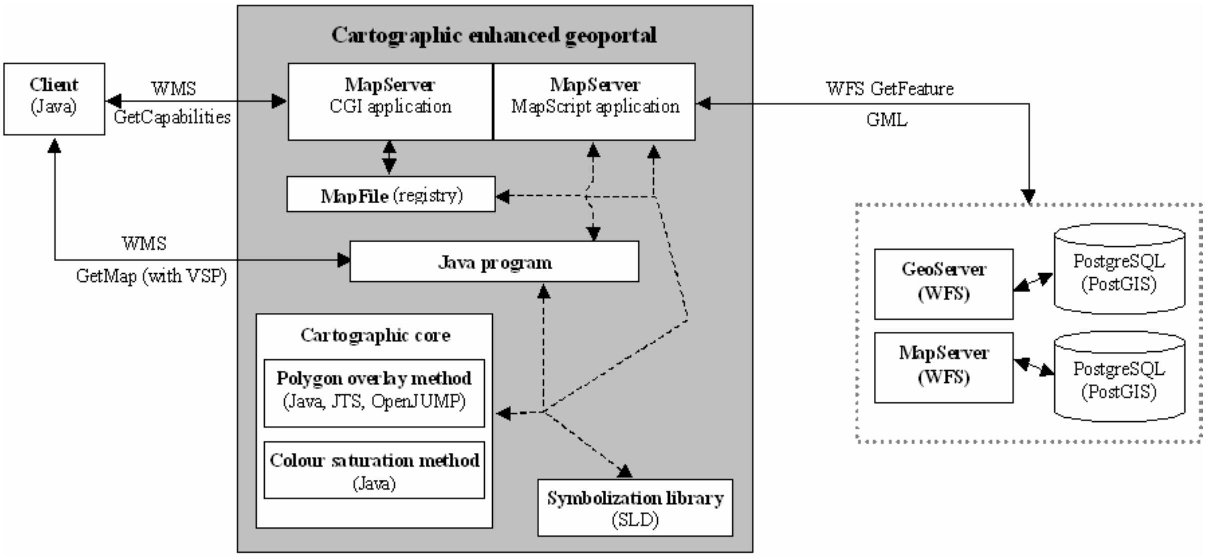

Figure 9. Implementation of a Cartographic enhanced geoportal.

The workflow for the implementation is as follows: The client sends a WMS *GetCapabilities* request to the cartographic enhanced geoportal following the OGC standard. The request is sent as a CGI-command to the CGI-application of MapServer. The geoportal responds to the request by returning an XML-document describing the capabilities; that is a list of available layers, styles from the symbolization library etc.

For this, MapServer does not send a *GetCapabilities* request to external services; instead MapServer gets the capabililties from a Mapfile. That is a MapServer specific configuration file, containing information about available data: where the different layers are stored, available styles etc. When an external service registers in the registry, the capabilities of that service, and the connection to the service, are added to the Mapfile. In this implementation the registry is not included, instead the Mapfile is updated manually.

From the capabilities the user formulates a *WMS GetMap* request. That is a request defining which layer(s) to include, which style(s) to use for symbology, format of the output map image etc. The next step is what differs the cartographic enhanced geoportal from an ordinary WMS-service; namely to select if a layer should belong to fore-, middle- or background. That is the VSP *LayerPriority* introduced above. When a VSP is introduced, the WMS standard requires that a GetMap request does return a map image also if the VSP is missing (OGC, 2010b). The geographic enhanced geoportal solves this by setting LayerPriority to middleground if no value is given.

The request, with *LayerPriority* included, is sent to the Java program, which acts as a spider, in the geoportal. For the client the Java program acts as a server and the communication is handled by a TCP/IP connection.

Since a WMS GetMap request only returns a map image, and the polygon overlay method works with the geometry of the different layers a WFS *GetFeature* request must be sent to the external services; that is a service that returns the actual data as a GML-file. The Java program formulates a WFS *GetFeature* request and sends it via the MapScript application of MapServer to the external services.

The Java program then interacts with the cartographic core, symbolization library and MapFile (registry). In the cartographic core, the polygon overlay method utilises OpenJUMP, to convert GML-files to WKT format, and JTS for the geometry. The colour hierarchy method is a Java program, which computes the new values for RGB (see 2.3), and applies them to the SLD to use.

The output from the polygon overlay method is one new point layer for each polygon layer sent to the method. These point layers give the positions for the icons that will be used to present a polygon layer together with the boundary from the polygon layer itself. The result of the colour saturation method is temporary SLDs that are updated according to *LayerPriority* in the *GetMap* request.

The new and modified data are then made available for MapServer by the Java program, which also sends an OGC standard (without VSP) GetMap request to MapServer. MapServer then creates the map image according to the original *GetMap* request from the client. The image is finally sent via the Javaprogram back to the client.

3.3 Results

Figure 10 shows the difference between a typical visualization and the cartographic enhanced method in planning application. The figures include the layers:

1. National outdoor interest (foreground) in blue
2. Highway (middle ground) in brown
3. Coniferous layers (middle ground) in green
4. Municipality (background) in orange

In Figure 10 (b), boundary and icons (walking man) are used to visualize the foreground layer using the polygon overlay method and the middle ground and background layers are faded by color saturation method.

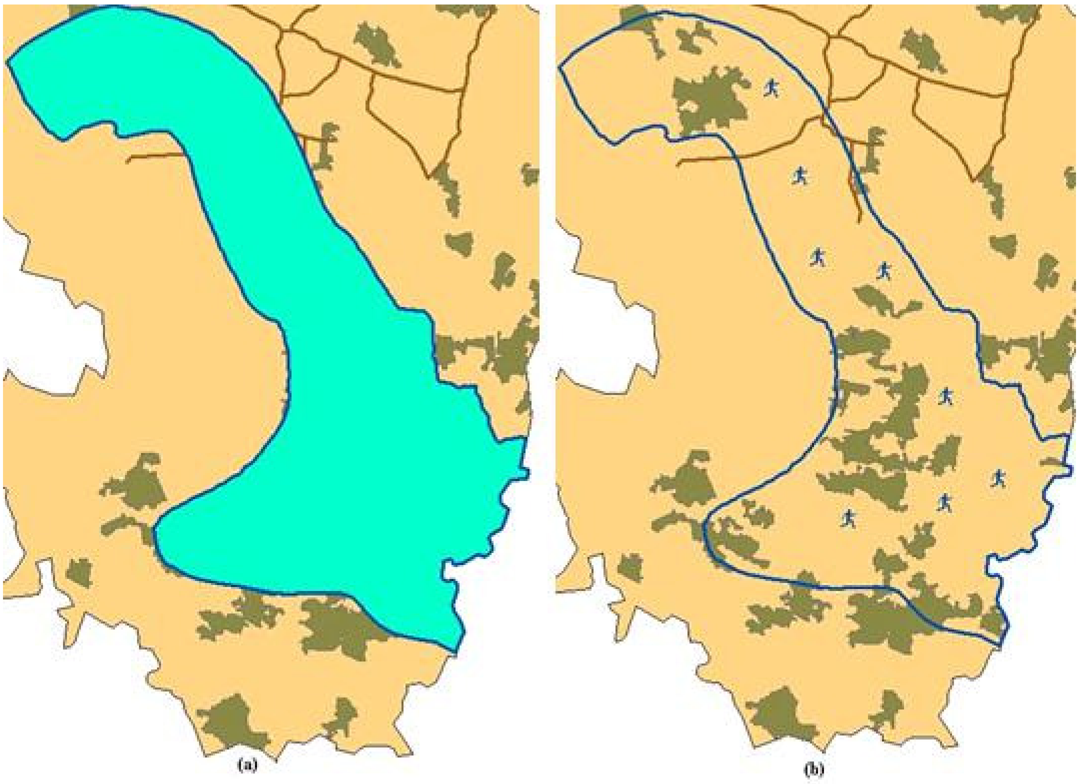

Figure 10. (a) Typical visualization with the case study dataset, (b) The method result for the case study:

The details of the results are described as follows. As mentioned in the methodology, simulated annealing is used for optimization step. The weights used in the cost function, are set as:

w_P=1, w_{SD}=0.3 and the w_R=0.1.

Also, the threshold for removal cost is set to 1000 to remove the icons.

The simulated annealing algorithm calculated the cost with 400 iterations with the following parameters:

alpha=0.999, *temperature*=400, *epsilon*=0.001

The result for total cost for the optimization step is calculated as:

tc=4722

where c_{SD}= 9803, c_p= 1780.0 and c_r= 0 (no removal).

The colour saturation method keeps the foreground layer border and decreases the saturation of the background to 80 percent and the middle ground layer to 90 percent of the original colour.

4. Conclusion

In this paper we described the methodology and the implementation of a cartographic enhanced geoportal. Our prototype system contains two methods: *polygon overlay* and *colour saturation*. In the polygon overlay method, the optimization of the cost function is based on simulated annealing approach. The results show that these methods are appropriate techniques to visualize the overlaying layers without any data lost.

ACKNOWLEDGEMENT

The European Union funding program, Erasmus Mundus "External Cooperation Window" (EMECW lot8), financed the first author of this study. The financing is acknowledged. The authors are thankful to *municipality of Lund, County administrative board of Skåne and European environment agency* for providing data. Also thanks to Mikael Johansson, Erik Gustafson and Vianney Dugrain for providing some implementation ideas, and Anders Söderman and Ali Mansourian for interesting discussions.

REFERENCES

Ahuja, N., 1982. 'Dot pattern processing using Voronoi neighbourhoods', IEEE Transactions on Pattern Analysis and Machine Intelligence, 4, pp. 336-343.

Brewer, C. A. & Buttenfield, B. P., 2007. Framing guidelines for multi-scale map design using databases at multiple resolutions. Cartographic and Geographic Information Science, 34, 3-15.

Buard, E. & Ruas, A., 2009. Processes for improving the colours of topographic maps in the the context of maps on demand, submitted to 24th ICA Conference, Santiago de Chile, Chile.

Bucher B, Buard E, Jolivet L, Ruas A, 2007. The need for web legend services, Proceedings of the 7th International Symposium on Web and Wireless Geographical Information Systems (W2GIS07), Cardiff, UK.

Chesneau, E, Ruas, A and Bonin, O, 2005. Color contrasts Analysis for a better legibility of graphic signs for risk maps, International Cartography Conference (ICC), Spain.

Christensen, J., J. Marks and S. Shieber, 1995. An empirical study of algorithms for Point-label placement, ACM Transactions on Graphics, 14(3), pp. 203-232.

De Longueville, B., 2010. Community-based geoportals: The next generation? Concepts and methods for the geospatial Web 2.0, Computers, Environment and Urban Systems, 34 (4), Geospatial Cyberinfrastructure, pp 299-308, DOI: 10.1016/j.compenvurbsys.2010.04.004.

Edwards, A. AND Burghardt, D., 2004. Portrayal and Generalisation of Point Maps for Mobile Information Services, In Map-based mobile services – Theories, Methods and Implementations, Reichenbacher, T. and L. Meng (Eds), (Berlin: Springer-Verlag)

Foley, J. D, van Dam, A., Feiner, S. K, and Hughes, J. F, 1996.

Computer Graphics: Principles and Practice, 2nd ed. in C, Addison-Wesley.

Harrie, L, and Revell, P., 2007. Automation of Vegetation Symbol Placement on Ordnance Survey H 1:50000ScaleMaps, The Cartographic journal Vol.44, No.3, Ordnance Survey Special Issue, pp 257 – 267(10).

Harrie, L., H. Stigmar, Koivula, T. and Lehto, L., 2004. An Algorithm for Icon Placement on a Real-Time Map. In Fisher, P. (ed), Development in Spatial Data Handling, Springer, pp. 493-507.

Iosifescu-Enescu, I, Hugentobler, M and Hurni, L, 2009. Web cartography with open standards–A solution to cartographic challenges of environmental management, Environ.Model.Softw, doi:10.1016/j.envsoft.2009.10.017.

Mansourian, A., Omidi, E., Toomanian, A. and Harrie, L., 2010. Expert system to enhance the functionality of clearinghouse services. Computers, Environment and Urban Systems, doi:10.1016/j.compenvurbsys.2010.06.003.

Russell, S. and Norvig, P., 1995. Artificial Intelligence – A Modern Approach, Prentice-Hall.

Sadahiro, Y., 2000. 'Perception of Spatial Dispersion in Point Distributions', Cartography and Geographic Information Science, 27(1), pp. 51-64.

Ware, J. M., and Jones, C. B., 1998. Conflict Reduction in Map Generalization Using Iterative Improvement, GeoInformatica, 2(4), pp. 383-407.

Zhang, Q. and Harrie, L., 2006. A Real-time Method of Placing Text and Icon Labels Simultaneously, Cartography and Geographic Information Science, 33(1), pp. 53-64.

Zoraster, S., 1986. Integer Programming Applied to the Map Label Placement Problem, Cartographica, 23(3), pp. 16-27.

Zoraster, S., 1997. Practical Results Using Simulated Annealing for Point Feature Label Placement, Cartography and Geographic Information Systems, 24(4), pp. 228-238.

WEB REFERENCES

GeoServer, 2010. http://geoserver.org/display/GEOS/Welcome (accessed 2010-10-06).

JTS, 2010. Java Topology Suite, Vivid Solutions. http://www.vividsolutions.com/jts/jtshome.htm (accessed 2010-10-06).

MapServer, 2010. http://mapserver.org/ (accessed 2010-10-06).

OGC, 2010a. Styled Layer Descriptor. http://www.opengeospatial.org/standards/sld (accessed 2010-10-01).

OGC, 2010b. Web Map Service. http://www.opengeospatial.org/standards/wms (accessed 2010-10-01).

OpenJUMP, 2010. http://openjump.org/ (accessed 2010-10-06).

PostGIS, 2010. http://www.postgis.org/ (accessed 2010-10-06).
PostgreSQL, 2010. http://www.postgresql.org/ (accessed 2010-10-06).