CARTOGRAPHIC WEB SERVICES AND CARTOGRAPHIC RULES – A NEW APPROACH FOR WEB CARTOGRAPHY

Ionut Iosifescu, Marco Hugentobler, Lorenz Hurni
ETH Zurich, Institute of Cartography
Wolfgang-Pauli-Str. 15
CH-8093 Zurich, Switzerland
E-mail: {iosifescu, hugentobler, hurni}@karto.baug.ethz.ch

1. Introduction

With the advent of open standards and Web services, nowadays many institutions are exchanging and processing spatial information in real-time over the Web. Consequently, a major requirement for Web Cartography is to produce maps directly from geospatial data available in geospatial databases and distributed repositories (Hurni et al., 2006).

The research for improved cartographic visualization of GIS data in a web service context was triggered by two European projects for the environmental management domain, namely ORCHESTRA (Open Architecture and Spatial Data Infrastructure for Risk Management) and SANY (Sensors Anywhere). In ORCHESTRA, managing risks requires that decisions are taken rapidly based on cartographic visualisations of up-to-date data from various systems. Similar requirements for Web Cartography are present in the IST - 6th Framework Integrated Project SANY, which focuses on interoperability of in-situ sensors and sensor networks. Sensor and spatial data as well as the results of scientific analyses and fusion algorithms must be dynamically presented to the decision makers in a legible map and in real-time.

OGC standards and especially the combination of Web Map Service (WMS), Styled Layer Descriptor (SLD) and Symbology Encoding (SE), provide already an open framework for Web mapping services. However, as presented in (Sykora et al. 2007), from the cartographic point of view the OGC standards offer limited graphical representation possibilities in comparison with state of the art cartographic products. As such, the melding of state-of-the-art in distributed cartographic visualisation with open standards was accomplished within Cartographic Web Services.

Cartographic Web Services are standards-based map services which are cartographically enriched. Primarily, cartographic extensions for the Symbology Encoding (SE) standards allow expressing cartographic rules with advanced point symbolisation, patterns for all spatial features, gradients, advanced features filtering and thematic mapping. In addition to fulfilling the most complex visualisation requirements, Cartographic Web Services formalize the map creation process and allows the exchange of cartographic rules.

2. Anatomy of Cartographic Web Services

2.1 Cartographic Interfaces

Cartographic interfaces refer to a software interface defining high quality map symbolization adapted to the area of distributed mapping. Although there many efforts for the creation of cartographic interfaces, unfortunately there are almost no developments based on accepted OGC standards that promote interoperability. Therefore in the context of interoperability and standardization, the only map visualisation interface that fulfils the requirements comes from the Open Geospatial Consortium (OGC).

Styled Layer Descriptor (SLD) is an XML(Extensible Markup Language) based description language for extending OGC Web Services such as Web Map Services (WMS), Web Feature Services (WFS) and Web Coverage Services (WCS) with a user-defined symbolization. Main advantages of SLD are the structuring of the style attributes and understandability for computers as well as for users.

Since OGC Web Services are based on layers, each layer can be symbolized with user-defined styles. Another OGC standard - Symbology Encoding (SE) - together with the Styled Layer Descriptor Profile for the Web Map Service Implementation Specification (SLD WMS profile) is the direct follow-up of the original Styled Layer Descriptor Implementation standard. It is the most recent OGC standard for portrayal of geographic information. The SLD specification document was split up into two documents to allow the parts that are not specific to WMS to be reused by other service specifications. In avoid any confusion, it can be noticed that the notion of Styled Layer Descriptor Implementation includes, among other, Symbology Encoding syntax.

2.2 The Map and Diagram Service Interface

Web Map Services are considered to be those services that visualize, symbolize and enable geographic clients to interactively visualize geographic data provided in vector and raster formats. In this context, the map is defined as a “portrayal of geographic information as a digital image file suitable for display on a computer screen” (ISO, 2005). Cartographic Web Services are more than WMS, in the sense that they offer a cartographic interface that combines advanced cartographic output with the generality, flexibility and standardization of SLD.

The Map and Diagram specifications were created specifically from a cartographer’s point of view with the purpose of allowing dynamic composition of web maps for environmental management without sacrificing visual differentiation of features or the legibility of the map. The main design consideration was not to replace existing standards, but to extend them for cartographic usage (Iosifescu, 2007). To support interoperability and use of open standards, the Map and Diagram Service Interface is based on and enhances the WMS (Web Map Service) and SLD (Styled Layer Descriptor) specifications. Therefore, it supports the requests 'GetCapabilities', 'GetMap' and 'GetFeatureInfo' in a compatible manner with the corresponding OGC standards. A short description of the semantics of each operation is shown in Fig. 1.

Operation Name	Description
getMap	Returns a map of spatially referenced geographic and thematic information as an image document.

getDiagram	Returns a diagram representation of tabular data as an image document.
getFeatureInfo	Returns information about the features rendered in a certain point of a map or diagram layer.
getLegendGraphic	Returns a legend symbol (corresponding to a layer) as an image document.
getLayerDescription	Returns a layer description document containing schema information for a layer.
getStyle	Returns a style associated with a layer.

Fig. 1. Operations of the Map and Diagram Service

The current HTTP GET interface defined for WMS has some major limitations preventing to use the full power of web-services in the cartographic domain. Not only that it is not possible to transport large SLD and GML messages but also dynamic discovery and integration of such services is reduced due to the absence of a machine-understandable standard interface description. The Map and Diagram Service provides a SOAP (Simple Object Access Protocol) interface for all its operations with the goal of making Web map services more accessible to a large variety of applications requiring integration of mapping components based on dynamic discovery.

In addition there are many aspects in which Map and Diagram Service Specifications goes further than the current state of the OGC specifications:

- Data in Geographic Mark-up Language (GML) format can be embedded directly into the request
- Supports any OGC compliant web service, including Sensor Observation Services (SOS)
- The resolution and quality of the output image is controlled with a DPI parameter
- Diagram symbolization and thematic mapping is possible
- Cartographic rules are expressed in Symbology Encoding (including spatial filtering, SVG point symbols, patterns)

3. Core concepts in Cartographic Web Services

3.1 Cartographic Rules

The core of a cartographic web service is represented by rules. Cartographic rules can be defined as detailed symbology descriptions for spatial features that specify their precise visual representation on a map. By abstracting from physical data, cartographers define cartographic rules considering only the logical data organisation of geometric features and linked attributes. Cartographic rules are defining the desired symbolization that can be applied by Cartographic Web Services to various distributed spatial data sources, e.g. GML/WFS, PostgreSql, Oracle Spatial, IBM DB2, ArcSDE, or KML.

Cartographic rules not only specify the basic properties and superimposing of cartographic symbols, but they may also define rotations, displacements and even or geometry manipulations of symbols with exactly the same goal as in traditional computer-assisted cartography. Symbology Encoding (SE) is the grammar for styling map data independent of any service interface specification. As symbology encoding is independent of the data itself, it provides a powerful basis to build the cartographic extensions. From the cartographic point of view, the symbology (that is to be applied to a feature type) is defined by a FeatureTypeStyle, which contains several parameters, most import being rule definitions. On a conceptual level, cartographic rules need three parts: generic conditions when cartographic rules are to be applied considering the overall suitability for a specific map (e.g. scale-dependent map symbolization), specific conditions for selecting (and expressing) the geometric features to be symbolized, and finally the definition of symbology. This conceptual classification can be directly mapped to the SE, as Rules are used to group rendering instructions by feature-property conditions and map scales. Therefore SE Rules are an appropriate mechanism to express cartographic intentions.

3.2 Cartographic Algorithms

The cartographic rules handle advanced filtering and symbolisation of features. For improved cartographic representation, the data should be enriched with attributes to control rotation, scale, size or even transparency. As example, for an optimal display of labels, the dataset should preferably contain additional attributes that can be used for specifying the font family, style, size, rotation and most important placement or displacement information for textual labels. However, such attributes can be defined only for special cases or data, and it may be required that additional attributes are to be computed dynamically each time a new map request is made. In this case algorithms are required to adapt the intelligence of the data according to specific map scale and bounding box.

Unfortunately such algorithmic complexity is beyond the expressive power provided by formalized cartographic rules expressed in SLD (even with cartographic extensions). The OGC Filter defines the notion of function as a named procedure that accepts zero or more arguments, performs a distinct computation and generates a single result. A function is composed of the name of the function, encoded using the attribute name, and zero or more arguments. However the implementation of these functions is hardcoded in the software (in our case a Cartographic Web Service). Therefore the reuse or exchange of such functions and the underlying algorithms are impossible.

Ontologies are introduced as backend for the standardisation of cartographic algorithms and functions that go beyond the expressivity of cartographic rules. With the underlying concepts and syntax set by a cartographic ontology (accepted by all implementations of cartographic web services), then algorithms can be expressed in any interpreted programming language. In this manner advanced cartographic algorithms are becoming scripts that can be interpreted and executed by cartographic web services. Thus they can be exchanged without restrictions as long as they use only the “API” fixed by the ontology. A proof of concept was performed using the Python scripting language. It is left open what scripting language cartographers will prefer - depending on their familiarity with a specific scripting or programming language. However it is an attempt to create the theoretical concepts and the methodology framework for interoperable cartographic algorithms.

3. Implementation of Cartographic Web Services

3.1. Design Considerations

The conceptual specification of Cartographic Web Services is enhanced with a design for efficient and cost-effective implementation based on existing components. This design is illustrated with QGIS mapserver (QGIS mapserver, 2009), an advanced Cartographic Web Service built on open-source GIS and graphic libraries. QGIS mapserver is a server compliant with WMS and SLD standards but the design of the software allows native support for the concepts of cartographic symbolisation as defined by the concept of Cartographic Web Services.

Regarding the software implementation, it is important to understand that desktop mapping software has been designed to be used locally on one computer and to get input from and draw output to screen. However, with the trend towards distributed mapping systems, these desktop mapping systems are more and more faced to concurrency from distributed software systems. Therefore practical considerations advises that there is no point in dropping existing desktop mapping software if the requirements of service based computing and distributed mapping come into play. Rather, the desktop software should be enhanced such that it is accessible as a service over the Internet, thus becoming a map server.

3.2. Implementation of a Cartographic Web Service

The Map and Diagram Service specifications have been implemented in the open source software QGIS mapserver. QGIS mapserver (QGIS mapserver, 2009) is compatible with the OGC specifications and it implements improved cartographic functionalities and output as defined by the Map and Diagram Specifications. QGIS is an open source desktop GIS with user-friendly graphical user interface that provides viewing, editing, printing and also analytical functionality for geodata. QGIS has a clear separation of GUI code and non-gui code. The non-gui code is compiled into the core library and thus can be used from by any application, provided it is licenced under the GNU GPL. This modularity leads to the decision to write a mapserver on top of QGIS rather than to replace it with an existing mapserver.

QGIS mapserver uses SLD (Styled Layer Descriptor) as its language both for predefined layers and styles configured by the server administrator as well as for user defined layers and styles. Consequently, the design of QGIS mapserver reflects the concepts of SLD. It includes the following components:

- Request handler classes to parse incoming requests
- A Server class that controls the execution of the different WMS request types
- A Parser class that provides an entry point to interpret cartographic rules and create layer and style objects from SLD
- A Rule class to represent individual cartographic rules (e.g. classifications items)
- A Filter class to test if a rule applies to a specific spatial feature
- A Renderer class to store the representation of vector layers and to apply it to individual spatial features

3.3. Implementation of Cartographic Rules

As each spatial layer may have one or more cartographic rules. A rule may have a filter that describes for which features it should be applied. If there is no filter, the rule is valid for every feature available in the spatial layer. The rule also consists of the symbology description, i.e. the polygon fill, line stroke, the point symbol, etc.

This logical structure is also reflected in the implemented class hierarchy. The Renderer contains the vector symbology and describes how a feature is to be rendered. An instance of the Renderer may have one or more associated Rule objects. For every feature that is to be rendered, the method Renderer::renderFeature is called. Inside this method, the Renderer goes through the list of rules until it finds the first one that is valid for the given feature. If one has been found, the representation of this spatial feature is linked to the appropriate Rule object.

4. Cartographic Visualisation with Web Services

4.2. Topographic symbolization

Features such as spatial filtering, masking, definition of complex point symbols, data distributions, transparency levels for individual layers and selected features, patterns, advanced texture mapping and various diagram types (e.g. pie diagrams, bar diagrams), are just a few of the capabilities required for correct cartographic output for topographic maps.
The possibilities for advanced cartographic symbolization can allow the creation of expressive topographic and thematic maps in all situations. Furthermore by accepting multiple cartographic rules for a selection of features in a layer, complex representations for topographic features according to cartographic conventions can be achieved.

A visual illustration of the expressivity of cartographic rules for Web map creation is presented in Figure 2. Patterns for line and polygon features, double line, visual differentiation of contours and custom point symbols are among the advanced cartographic symbolization possibilities that can be observed. Furthermore, complex representations of features according to the most demanding domain requirements can be achieved by accepting multiple cartographic rules (expressed as SLD rule elements) for the selection and symbolisation of features in a layer.

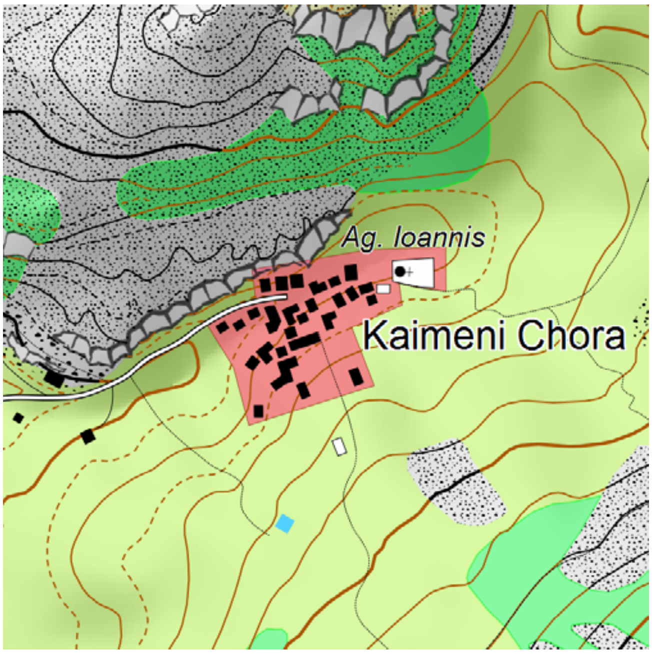

Fig. 2. Detail of a web map symbolized with cartographic rules

4.3. Thematic symbolization

Slocum et al. (2005) confirms that diagram symbolization of geographic features is an effective way of visualizing statistical data in a spatial context. Unfortunately it is not currently possible in SLD/SE to visualize multiple data values using diagrams. More over in the SLD, the classification of the features has to be fully specified. There are no options like 'graduated symbol' or 'unique value' which make the creation of thematic maps straightforward in current desktop GIS software.

For this reason cartographic extensions for thematic mapping were defined in order to produce thematic representations. There are different diagram types currently specified, e.g. pie, bar, line, area, ring and polar - some of them having additional options like normal, stacked or percent. Beside diagrams, proportional symbols and dot density are specified. The creation of choropleth maps is also enhanced by defining different classification methods (beside the manual definition of classes): Equal Interval, Quantile, Natural Breaks and Standard Deviation. An example of a thematic layer overlay with pie diagrams is presented in Figure 3.

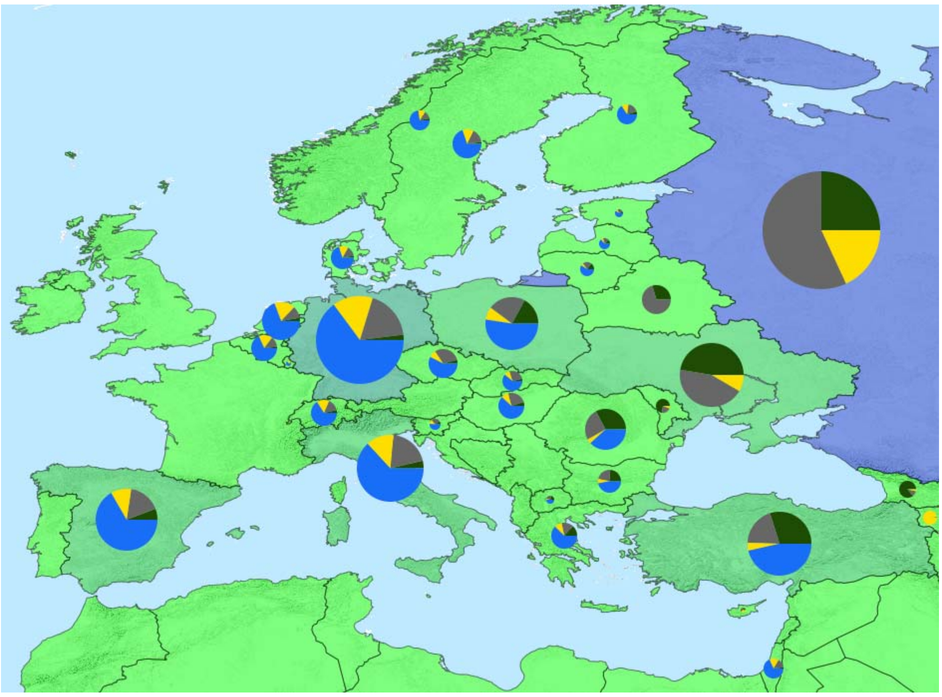

Fig. 3. Thematic layer overlay dynamically generated with a GetMap request

4.3. Specific cartographic reprezentations for risk and crysis management

Sensors are an important source of spatial information in risk management. This kind of data usually is very dynamic and therefore it is appropriate to distribute the data via a web service. In this manner redundancy is avoided and all the clients are able to use data that is up-to-date.

To facilitate interpretation of sensor data, QGIS mapserver offers the possibility to display sensor data as coloured raster (thematic symbolization) and with contours (topographic symbolization). In the case of point data with associated measurement values, the data model needs to be transformed in real-time. For example a cartographic rule can define that the discrete point data has to be interpolated to a raster and then contour lines are derived from the raster.

Figure 4 shows a temperature map generated with this approach, which in addition can also include the location and status of the sensors that generated the data. The colour of the points representing the sensors can be linked with the data uncertainty (e.g. green is to be used for sensors delivering certain data and red for sensors delivering data which is below a predefined threshold).

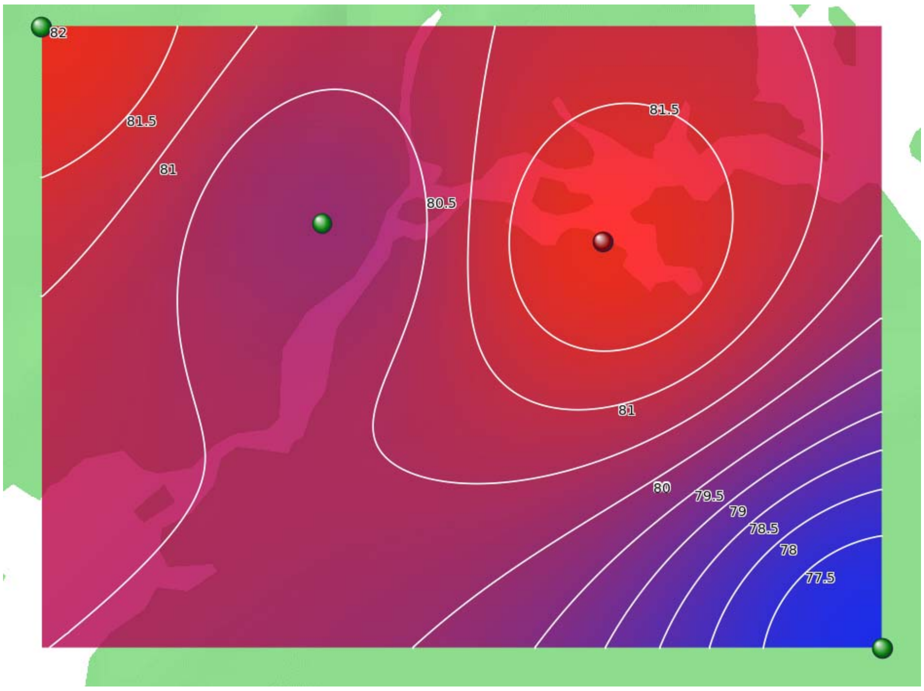

Fig. 4. Real-time visualisation of Sensor data with contours

5. Conclusions and outlook

Cartography can offer an important contribution regarding the visualisation of geospatial and thematic data over the Web. The goal of the type of cartographic research presented in this paper, was to achieve cartographic quality for maps created in real-time. The extreme requirement for real-time mapping of continuously changeable datasources increases the difficulty of the problem, possibly starting the field of 'extreme cartography'.

In this context, cartographic rules represent a formalisation for cartographic knowledge as well as promoting interoperability and exchange. With cartographic rules the cartographer has the means to formalize the cartographic knowledge used in the process of map making. Therefore cartographers now have expressive power and flexibility required for the creation of cartographic products in real-time. Moreover the creation and exchange of more complex cartographic rules may also play an important role in the detection and elimination of cartographic conflicts.

An implementation of the discussed cartographic enhancements is provided with the open source project QGIS mapserver. With this implementation it is proven that the concept of Cartographic Web Services is not only valid but also cost-effective while enabling cartographic functionality to be transferred on the server side.

Acknowledgments

The development of cartographic web services is supported by the ORCHESTRA and SANY projects. ORCHESTRA and SANY are 6th Framework European Integrated Projects co-funded by the Information Society and Media DG of the European Commission within the RTD activities of the Thematic Priority Information Society Technologies.

References

Hurni, L., Gogu, R., Iosifescu, I. (2006) Cartography for Environmental Systems. Proceedings of the Workshop Intelligent Embedded Systems, July 20, 2006, Romanian Academy Library, Bucharest, pp. 9-17.

Iosifescu, I. (2007) Specification of the Map and Diagram Abstract Interface, Technical Report, ORCHESTRA Consortium.

ISO (2005) ISO/DIS 19128, International Organization for Standardization, ISO Technical Committee 211

OGC (2002) Styled Layer Descriptor Implementation Specification, Open Geospatial Consortium

OGC (2005a) Styled Layer Descriptor Profile of the Web Map Service Implementation Specification, Open Geospatial Consortium

OGC (2005b) Symbology Encoding Descriptor Implementation Specification, Open Geospatial Consortium

Slocum, A., R. McMaster, F. Kessler, Howard H., (2005) Thematic Cartography and Geographic Visualization, second Ed., Pearson Education Inc, 518 pp.

Sykora, P., Schnabel, O., Iosifescu, I. (2007) Extended Cartographic Interfaces for Open Distributed Processing. Cartographica, Volume 42, Number 3, 2007, pp. 209 – 218

QGIS (2009) http://qgis.org, accessed 2009-05-29

QGIS mapserver (2009) http://karlinapp.ethz.ch/qgis_wms, accessed 2009-05-29

W3C (2003) Scalable Vector Graphics (1.1) Specifications, World Wide Web Consortium.