1. Introduction

The usefulness of providing interactive topographic and thematic spatial data representations over the Web is undisputed. Web mapping applications using various technological approaches (e.g. JAVA applets, Scalable Vector Graphics/JavaScript, HTML/AJAX, proprietary plug-ins) have been around for years. Major GIS vendors like Intergraph, ESRI and Autodesk created their own Web mapping systems with the goal of migrating elements of desktop GIS to a larger audience over the Web. Such Web mapping systems respond to user actions in real time by using map servers, thus offering basic functionalities like zooming, panning, identifying objects and layer management (showing and hiding map layers).

Unfortunately most of these interactive mapping systems were developed as a set of proprietary implementations with no published interfaces [5]. To address this interoperability problem of existing online mapping services, the Open Geospatial Consortium (OGC) undertook standardisation efforts towards open standards to give transparent access to heterogeneous geo-data and geo-processing resources in a networked environment. The standardisation effort was successful with the industry acceptance of the Web Map Service (WMS) [3] and Styled Layer Descriptor (SLD) [4] Specifications. As a result, currently these standards are broadly supported by commercial as well as open source software.

ORCHESTRA (Open Architecture and Spatial Data Infrastructure for Risk Management) is an Integrated Project partly funded by the European Commission’s 6th Framework Programme, under the priority 2.3.2.9 “Improving Risk Management”. The aim of ORCHESTRA is to improve the efficiency in dealing with risks by developing an open service-oriented architecture for risk management [8]. One main goal of ORCHESTRA is to solve the lack of interoperability for risk and environmental information and to provide an information infrastructure which is capable of managing risk information across Europe. As in risk management special attention is paid to an integrated service and data approach, including their spatial, temporal and thematic characteristics,
thematic maps are of interest as a decision support tool that visually communicate risk-related geographical concepts like data distributions, variations and aggregations (e.g. distribution of risk levels, densities, etc…). The Institute of Cartography at ETH Zurich, one of 14 institutions involved in the development of the ORCHESTRA Architecture, is responsible for designing and implementing a Web map service (the Map and Diagram Service) suitable for on-demand map generation of thematic data on the internet.

2. WMS/SLD usage for interactive mapping

The WMS (Web Map Service) is an OGC, and in the same time an ISO standard for Web map services which is broadly supported by commercial as well as open source software. It defines three operations: GetCapabilities (returns service metadata), GetMap (returns a map whose geographic and dimensional parameters are well-defined) and GetFeatureInfo (optional operation that returns information about particular features shown on the map at a specified location).

These standard WMS operations can be invoked using a WMS client by submitting requests in the form of HTTP Uniform Resource Locators (URLs). Each WMS operation has several mandatory or optional request parameters with standardized names. In the case of a GetMap request, the URL contains information about the specific attributes of the invoked request (e.g. version and name), the output image characteristics (e.g. format, width and height), what information is to be shown on the map (layers), how this information is represented on the map (styles), what geographical region is to be mapped (bounding box) and the desired coordinate reference system (CRS). The main output of the request is a map, which can be either in raster (e.g. jpeg, png) or symbolized-vector format (e.g. SVG). A service does not provide a graphical user interface, however, depending on the implementation, a geographic viewer, other map service instance, or even a web browser could act as a WMS client.

As an extension to allow user-defined symbologies, a WMS may provide the capability of accepting SLD (Styled Layer Descriptor), an XML based description language for extending OGC Web Services such as Web Map Services (WMS) and Web Coverage Services (WCS) with a user-defined symbolization. Another OGC standard - Symbology Encoding (SE) - together with the Styled Layer Descriptor Profile for the Web Map Service Implementation Specification is the direct follow-up of Styled Layer Descriptor Implementation (SLD). 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. For this present paper, in order to prevent confusion, the notion of SLD considers also the latest developments from the Symbology Encoding Implementation Specifications [7].

The combination WMS/SLD already provides an open framework for Web mapping services. Since Web Map Services work with layers, each layer can be symbolized with user-defined styles (e.g. the layer "rivers" with a blue stroke or “lakes” with a blue fill). However, it is not possible to visualize multiple data values, e.g. with diagrams such as pie diagrams or bar diagrams. Proportional maps symbols such as proportional circles or squares can be drawn, however they are limited to a few predefined geometric types or to external graphics. Only simple point symbols/markers, lines, polygons, texts and raster images are integrated in the SLD language, which is the cause why WMS instances are often used for topographic map services, but not for thematic map services.

In comparison with Web-based Atlas information Systems, the WMS and SLD standards lack in advanced cartographic functionalities. Extensions to the standards in order to allow thematic mapping are a requirement for many application domains.

3. Cartographic functionalities via Map Services – Specifications of the Map and Diagram Service

The Map and Diagram Service is a service that visualizes, symbolizes and enables the geographic clients to interactively visualize thematic data. Its main task is to transform geographic data (vector or raster) and/or thematic data (e.g. census data, results of risk susceptibility analysis) into a graphical representation using symbolization rules. To support interoperability and use of open standards, the Map and Diagram Service is based on and enhances the WMS and SLD standards. The Map and Diagram Service Specifications, extend the WMS/SLD standards with cartographic features like various diagram types (e.g. pie diagrams, bar diagrams), definition of complex point symbols, data distributions, transparency levels for individual layers, patterns, advanced texture mapping and definitions of label placement rules.

The proposed interface for a Map and Diagram Service is presented in Fig.10. The service covers the various functional requirements for a comprehensive cartographic visualisation of geo-spatial data including both topographic and thematic aspects.
The Map and Diagram Service interface features different operations that can be remotely called in an interoperable way, operations that fulfil many of the advanced requirements for advanced symbolization.

The GetMap operation assists in the creation of spatial navigation controls and layer manager with a minimal effort, as precise navigation (zooming, panning, etc.) and the layers displayed can be precisely controlled by the parameters of the request. The GetFeatureInfo and GetDiagram are useful for attribute data display. The GetDiagram operation allows the display of information as complex diagrams and returns a diagram in form of an image document. The diagrams can be constructed based on the meta-information retrieved in real time by the GetLayerDescription operation, as this operation offers extended information about the data layer (including the attribute names and corresponding value types and ranges).

The GetLayerLegend operation used in combination with the GetFeatureInfo supports the client-side implementation of bidirectional linked legends. GetLayerLegend operation returns a map or diagram legend in form of an image document. GetFeatureInfo operation returns information about the features present at/near a specified point in an image document (map or diagram).

Furthermore, for some of the above operations the Map and Diagram Service supports the clients to send their data to be rendered as part of the request message. Currently only preconfigured data hosted on the server or WFS/WCS data is considered in the WMS standard. This implies that a client should also have a preconfigured and fully
functional WFS/ WCS server on the local device. With the option of sending data directly, the Map and Diagram Service allows creating more complex service clients (overlaying of user data layers, e.g. visualisation of digitized field data). The current Map and Diagram Service Implementations allows the clients to send their data in GML as part of the request message, providing the needed flexibility for various usage scenarios.

Other important features of the Map and Diagram Service specifications are cartographic enhancements of the WMS/SLD standards. As there is little to no support for creating thematic maps, different symbolization possibilities, including various diagram types, are specified to support thematic web mapping.

Current SLD symbolisation is provided by five types of Symbolizers: Line, Polygon, Point, Text and Raster Symbolizers. Although diagram symbolization of geographic features is an effective way of visualizing statistical data in a spatial context [10], it is not currently possible 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.

As such, the Diagram Symbolizer is the most important extension to SLD. An example of a Diagram Symbolizer SLD definition is visible in Fig. 2 and the corresponding result in Fig. 3.

```xml
<slid:DiagramSymbolizer>
  <slid:Name>Energy Pie Diagrams</slid:Name>
  <slid:Diagram>
    <slid:Rotation>0</slid:Rotation>
    <slid:Scale>
      <slid:Value>458143</slid:Value>
      <slid:Size>50</slid:Size>
      <ogc:PropertyName>TOTAL_ENER</ogc:PropertyName>
    </slid:Scale>
    <slid:WellKnownName>Pie</slid:WellKnownName>
    <slid:Category>
      <slid:Title>Renewable Energy</slid:Title>
      <ogc:PropertyName>TOTAL_ALTE</ogc:PropertyName>
      <slid:InteriorGap>0</slid:InteriorGap>
      <slid:PolygonSymbolizer>
        <slid:Fill> <slid:CssParameter sld:name="fill">#00ff00</slid:CssParameter> </slid:Fill>
      </slid:PolygonSymbolizer>
    </slid:Category>
    <slid:Category>
      <slid:Title>Non Renewable Energy</slid:Title>
      <ogc:PropertyName>TOTAL_NON</ogc:PropertyName>
      <slid:InteriorGap>0</slid:InteriorGap>
      <slid:PolygonSymbolizer>
        <slid:Fill> <slid:CssParameter sld:name="fill">#ff0000</slid:CssParameter> </slid:Fill>
      </slid:PolygonSymbolizer>
    </slid:Category>
  </slid:Diagram>
</slid:DiagramSymbolizer>
```

Fig. 2. Diagram Symbolizer Example
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.

Other extensions to the SLD standard allow flexible point symbols definitions. SLD defines only basic symbol which has coloring applied to it. The WellKnownName element gives the well-known name of the shape of the mark. Allowed values include at least “square”, “circle”, “triangle”, “star”, “cross”, and “x”. The alternative to a WellKnownName is an external graphic format which has visible limitations. The proposed solution is to enhance the “Mark” SLD element with SVG (Scalable Vector Graphics) definitions.

```xml
<svg:path d="M 0 0 L 7 0 L 3.5 7 z" fill="blue" stroke="red"/>
```

*Fig. 4. Mark definition using the svg:path in extended SLD*

Advanced graphical output as supported by SVG (pattern and gradients) is also included among the extensions. SLD standard currently allows the use of the following SVG/CSS styling parameters for a stroke: “stroke” (color), “stroke-opacity”, “stroke-width”, “stroke-linejoin”, “stroke-linecap”, “stroke-dasharray”, and “stroke-dashoffset”.
The “stroke-linejoin” and “stroke-linecap” SvgParameter elements encode enumerated values telling how line strings should be joined (between line segments) and capped (at the two ends of the line string). The “stroke-dasharray” SvgParameter element encodes a dash pattern as a series of dashes separated by spaces. The “stroke-dashoffset” SvgParameter element specifies the distance into the “stroke-dasharray” pattern at which to start drawing.

Beside line type definitions currently supported by the SVG parameters, advanced definition of patterns for polygon geometries has also been defined (Fig. 5).

Fig. 5 Different stroke and fill patterns possible in extended SLD

The definition of the polygon patterns and gradients as SLD extensions are based also on SVG. An SVG pattern and the corresponding result is presented in Fig 6. Gradients are defined in a similar manner by integrating the SVG standard.

Fig. 6 Definition of polygon fill patterns in extended SLD

As presented above the extensions are minimal but in combination with the Raster Symbolizer and the SLD support for the OGC Filter standard allows for creation of thematic maps in all situations. Furthermore by accepting multiple SLD “Rule” elements for a selection of features in a layer, complex representations for topographic features according to cartographic conventions can be achieved (e.g. for road maps).
Based on the SVG symbol definition, also customized diagrams can be in theory constructed by allowing descriptions of construction principles. However, this will be possible only in combination with additional tools supporting the creation of SLD descriptions in order to mask the complexity of such a process.

Another aspect considered in the design of the Map and Diagram Service is a basic user authorization method. A possible solution would be to enrich each request with two optional parameters (user and password) over an encrypted connection (HTTPS). Based on a functional user authorization, any web map server could provide an interface for the administrators to load/remove layers and styles. Using these features, the deployment time and effort as well as the software administration costs will be significantly reduced.

4. Web Map Services – a SOAP Approach

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 W3C recommendation for web services is WSDL/SOAP/UDDI (Web Services Description Language/Simple Object Access Protocol/Universal Description, Discovery and Integration) [1, 2]. Among the standards for Web services, the combination WSDL/SOAP/UDDI have gained strong industry support, and are likely to become the standard way for doing web services.

Alongside the standard HTTP GET request method for the Web Map Service, this paper introduces a SOAP binding using the HTTP POST request method (since the POST binding is declared optional in the standard WMS specifications).

The SOAP binding presented in this paper together with the service description in WSDL tries to follow and improve the latest change proposals of the OGC to offer support for WSDL and SOAP [6]. This approach allows building clients directly from service interfaces expressed in WSDL.

The proposed WSDL for a basic Web Map Service must include at least the mandatory operations GetMap and GetCapabilities.

In the SOAP GetCapabilities request, all the parameters found in the HTTP GET request method are embedded in the SOAP message. This paper also introduces a generic form for GetCapabilities SOAP request applicable to any kind of geospatial web services:
In response to this simplified request the server should reply only with its service metadata. This generic form is useful for the discovery of web mapping services when they are part of a larger network of geospatial web services (e.g. web feature services, web coverage services, etc.), having various functionalities and capabilities.

Similarly, the SOAP GetMap request contains the request attributes, the characteristics of the output image as well as of the resulted map. In response to such a request, the service will reply with a SOAP response message containing the map as a SOAP attachment:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <wms:getMapResponse xmlns:wms="http://www.opengis.net/wms">
      <return href="image@mapservice" xsi:type="image/png"/>
    </wms:getMapResponse>
  </soap:Body>
</soap:Envelope>
```

```--MIME_boundary
Content-Type: image/png
Content-Transfer-Encoding: binary
Content-ID: <image@mapservice> […]image_bytes…]
```

The Map and Diagram Service provides a SOAP 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.

The presented Map and Diagram Service specifications are currently being implemented under the GPL license in the QGIS mapserver project [9]. QGIS mapserver is a CGI/FastCGI application written in C++, ensuring portability of the code for most operating systems and computer architectures. A first prototype of the QGIS mapserver is already available and represents a contribution of the ORCHESTRA Project to the open source community.
5. Conclusions

Cartography can offer an important contribution regarding the visualisation of
geospatial and thematic data over the Web. This paper presents some of the cartographic
functionalities provided by the Map and Diagram Service Specifications and the extensions
of the Styled Layer Descriptor (SLD).

The Map and Diagram Service was designed to support a wide-range of distributed
cartographic applications. In this respect, much cartographic functionality can be
transferred on the server side allowing an efficient and cost effective creation of light-
weight clients with full-fledged interactive mapping functionality. The existing
specifications of the Map and Diagram Service covers most aspects neglected in the
existing OGC standards for interactive Web mapping. Such aspects refer mainly to the
generation of complex thematic maps for a combined visualization of spatial and thematic
data (e.g. diagram layers) as well definition of flexible mechanisms for creating user-
declared symbols, patterns and gradients based on other open standards (SVG).

References
Architectures and Applications. Springer, Berlin
SOAP, UDDI & WSDL; O’Reilly & Associates
Committee 211
Consortium
Consortium
Consortium
[9] ETH Zurich, 2007, QGIS mapserver (karlinapp.ethz.ch/qgis_wms/qgis_wms.html), last
accessed on 2006-05-25
and Geographic Visualization, second Ed., Pearson Education Inc.