Orchestrating OGC web services to produce thematic maps in a spatial information infrastructure

Victoria Rautenbach[a], Serena Coetzee[a], Adam Iwaniak[b]

[a] Centre for Geoinformation Science, Department of Geography, Geoinformatics and Meteorology, University of Pretoria, Pretoria 0002, South Africa
[b] Wrocław University of Environmental and Life Sciences, ul. Grunwaldzka 53, 50-357 Wrocław, Poland

Abstract

An intelligent geoportal orchestrates (automatically coordinate) web services to prepare, discover and present information to the user. Implementations of web service standards by the Open Geospatial Consortium (OGC) and ISO/TC 211, *Geographic information/Geomatics*, such as, the Web Map Service (WMS), Web Feature Service (WFS) and Styled Layer Descriptor (SLD) enable the display of spatial data in a geoportal. Ultimately, our goal is to intelligently orchestrate web services to produce thematic maps. As a first step towards our ultimate goal, in this article, we present the results of experiments with the orchestration of OGC web services to produce thematic maps. Our goal here was to evaluate whether orchestrating OGC web services can produce thematic maps. The results prove that this is possible, but show that there is customised functionality that has to be wrapped into WPSs. This poses a challenge to on the fly intelligent orchestration, which is required in an intelligent geoportal.

Keywords: SDI, spatial data infrastructure, web service, geoportal, WMS, thematic mapping

1. Introduction

One of the aims of a spatial data infrastructure (SDI) is to share and make data available to a wide audience for their economic and societal benefit. In an SDI, a geoportal is typically used to provide access to the spatial *data* and associated web services for discovery, display, editing and analysis (Maguire & Longley, 2005). For example, the focus of the European SDI, the **IN**frastructure for **SP**atial **I**nfo**R**mation in **E**urope (INSPIRE) is on data interoperability: datasets are harmonized to facilitate the integration of individual datasets is possible.

The terms data and information, as well as SDI and spatial information infrastructure (SII), are sometimes used interchangeably and the distinction between data and information can be subjective. Also, in practice SDIs sometimes provide access to both data and information. For the purposes of our research, we distinguish between data as the raw facts that are unstructured and convey no specific message (Bellinger et al., 2004), and information as data

that is processed and prepared so as to be useful in a particular context. An SII then provides access to *information*, i.e. data that has been processed and prepared so as to be useful in a particular context. An SII geoportal requires intelligence to orchestrate (automatically coordinate) web services that prepare and present information, instead of data, to the user. We call this an *intelligent geoportal*: a geoportal that provides complex functionality through a simple user interface for a user in a specific application domain (Iwaniak et al., 2011). The focus in an SII is on interoperability of *processed data*, making it possible to retrieve and present information rather than just raw data. This information makes specific relationships in the data visible.

A thematic map is an example of the presentation of geographical information: the original raw data is processed, additional information is generated (e.g. standard deviation, mean, etc.), and the resulting thematic map is useful to the user in a particular context. In an SDI interoperability of spatial *data* publishing services is implemented mainly at the level of search, view and exchange of data. In an SII interoperability of *information* services is also required at the level of search, view and exchange, but in addition, intelligence is required at a level of dynamic processing, interpretation and visualization of data. The intelligence requirement is in line with the observed current trend of SDIs evolving towards more distribution and dynamism (Coetzee, 2011). Thematic maps have become increasingly popular in the last few years and interest has been especially in the use of web services to produce thematic maps; this can be contributed to the increased use of web service in general, but in SDIs specifically (Maguire & Longley, 2005).

Interoperability of web services, whether in an SDI or an SII, requires the development and use of uniform standards. There are two levels of web service interoperability: syntactical and semantical (ISO 19119:2005, Yue *et al*. 2007). Syntactical interoperability assures that there is a technical connection enabling the transfer of data between web services. With semantical interoperability the services have a common understanding of the information that is processed and transferred between the services.

As spatial data infrastructures developed, the role and activity of standardisation bodies has increased significantly. Standards published by the International Organization for Standardisation (ISO) and the Open Geospatial Consortium (OGC) are widely accepted and implemented in SDIs, such as INSPIRE. ISO/TC 211, *Geographic information/Geomatics* is the ISO technical committee that deals with geographic information standards. The OGC is an industry consortium that develops open standards for spatial content and associated services and has developed a non-propriety web mapping approach based on open-interfaces, encoding and schemas (OGC, 2004). The main goal of OGC is to come up with a set of specifications to be used as a guide to internet GIS design for different software vendors so that their designed systems can be communicated or interoperated with each other (Peng & Tsou, 2003). ISO/TC 211 and the OGC liaise closely through the Joint Advisory Group (JAG) to ensure that standards from the two bodies are harmonized. For example, the OpenGIS Web Map Server Implementation Specification is also published as ISO 19128:2005, *Geographic Information – Web Map Server Interface*.

Multiple OGC web services are required to prepare and present thematic maps over the web. This can be achieved through web service orchestration, the automated arrangement of web services in a predefined pattern based on local decisions about their interactions with one another at the message and/or execution level (Sun *et al.*, 2010, Suazo & Aguirre, 2005). It uses a single process, referred to as a central controller, to control the underlying processes, executes the processes in the particular order in which they were specified. Ultimately, our goal is to intelligently orchestrate OGC web services to produce thematic maps. As a first step, in this paper we orchestrated OGC web services to produce thematic maps. Our experiments show that not all required functionality was available as OGC web services: some functionality had to be wrapped into WPSs. This poses a challenge to on the fly intelligent orchestration, which is required in an SII's intelligent geoportal.

The remainder of the paper is structured as follows: in section 2 a brief background of cartography and computer science for thematic web services is provided; in section 3 related work is discussed; in section 4 we describe the thematic web service that we orchestrated from existing OGC web services; and in section 5 the results are discussed; section 6 has the conclusion.

2. Background

Our paper touches on two disciplines: cartography and computer science. In this section we provide a brief background on the cartography and computer science behind thematic web services.
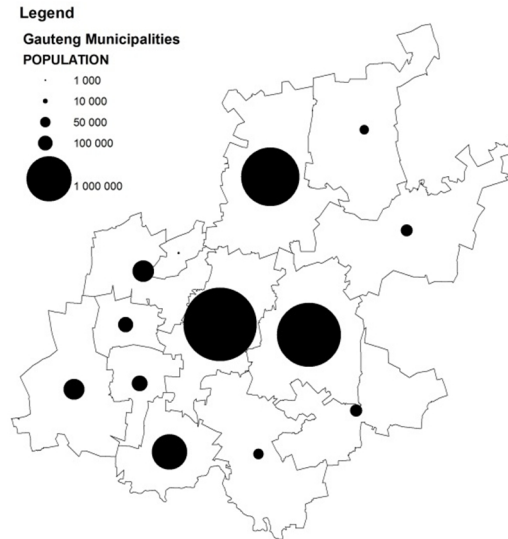
2.1 Cartography

Cartography is the science and art of making a map. The cartographic process involves the transformation of unmapped data into map form involving symbolization and the map-reading activities whereby map users gain information (Dent, 1996). Thematic cartography is a subsection of cartography that deals in detail with the production of thematic maps. A thematic map is a specific type of map that is designed to communicate information about a single topic or theme, such as population density or geology (Wade & Sommer, 2006). A thematic map portrays the distribution of features, relationship amongst features or classifications related to a specific topic (Parr, 2000, Robinson *et al.*, 1995). Cartographic methods are required to prepare information from raw spatial data for presentation in an SII's intelligent geoportal.
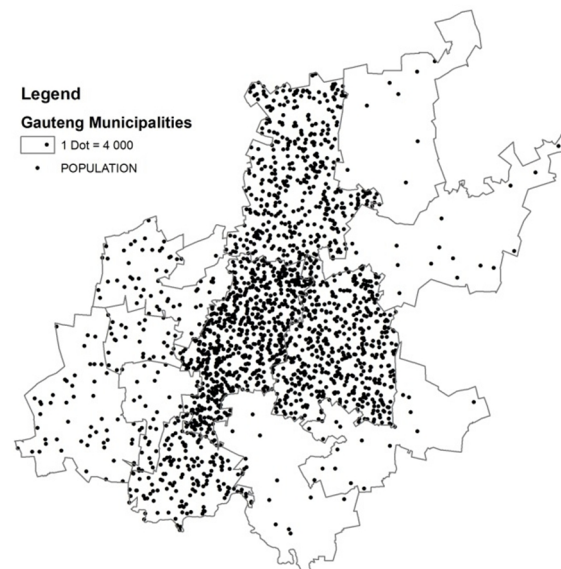
2.2 Thematic cartography

When presenting information, the choice of thematic map type is very important: depending on how the information is presented, different patterns are visible and either aid or hinder the understanding of the map. One of the more popular thematic maps is the choropleth map; however, it is just one of several types available. Several types of thematic maps are available: choropleth map (refer to Figure 1a), proportional symbols maps (refer to Figure 1b), dot density maps (refer to Figure 1c), cartograms (refer to Figure 1d), isarithmic map, dasymetric map, animated maps and flow generalization Map.
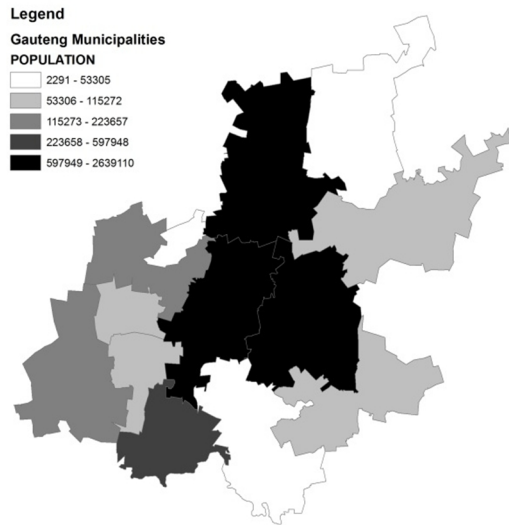
a) Choropleth map with five classes

b) Proportional symbol map

c) Dot density map

d) Cartogram

Figure 1 Thematic maps of population per municipality in Gauteng, South Africa (StatsSA, 2001)

Figure 2 depicts the thematic cartography process for a choropleth and proportional symbol map. From the diagram it can be seen that the processes are similar, with some minor differences. For example, data for the choropleth map requires standardisation, but this is optional for the proportional symbol map. The main difference between the two processes is in step 7 and 8: during the symbolization process, colour is used for a choropleth map and symbols for a proportional symbol map. The thematic cartography process can be summarised into the following steps (see figure 2):

Step 1: Determine the goal of the thematic map.
Step 2: Select the dataset.
Step 3: Obtain dataset from source.

Step 4: Specify the reference area.
Step 5: Choose the projection of the map, e.g. EPSG service.
Step 6: Decide on the thematic cartographic method.
Step 7: Standardise the dataset, if required.
Step 8: Classify the dataset, if the user selected this option.
Step 9 and 10: Symbolization process.
Step 11 and 12: Create map and additional information, e.g. legend and layout.

It is important to understand the steps involved in creating the thematic maps, before attempting to orchestrate web services to produce thematic maps. We used the process depicted in figure 2 to produce a workflow and to determine the required web services, before orchestrating these by a central controller. Web services can be orchestrated to perform the entire process depicted in figure 2, however our ThematicWS will only perform steps 3 and 7 through 12; in our implementation the cartographer still has to perform steps 1, 2, 4 to 6, and after the map has been produced the cartographer needs to ensure that the map is correct and conveys the intended message.
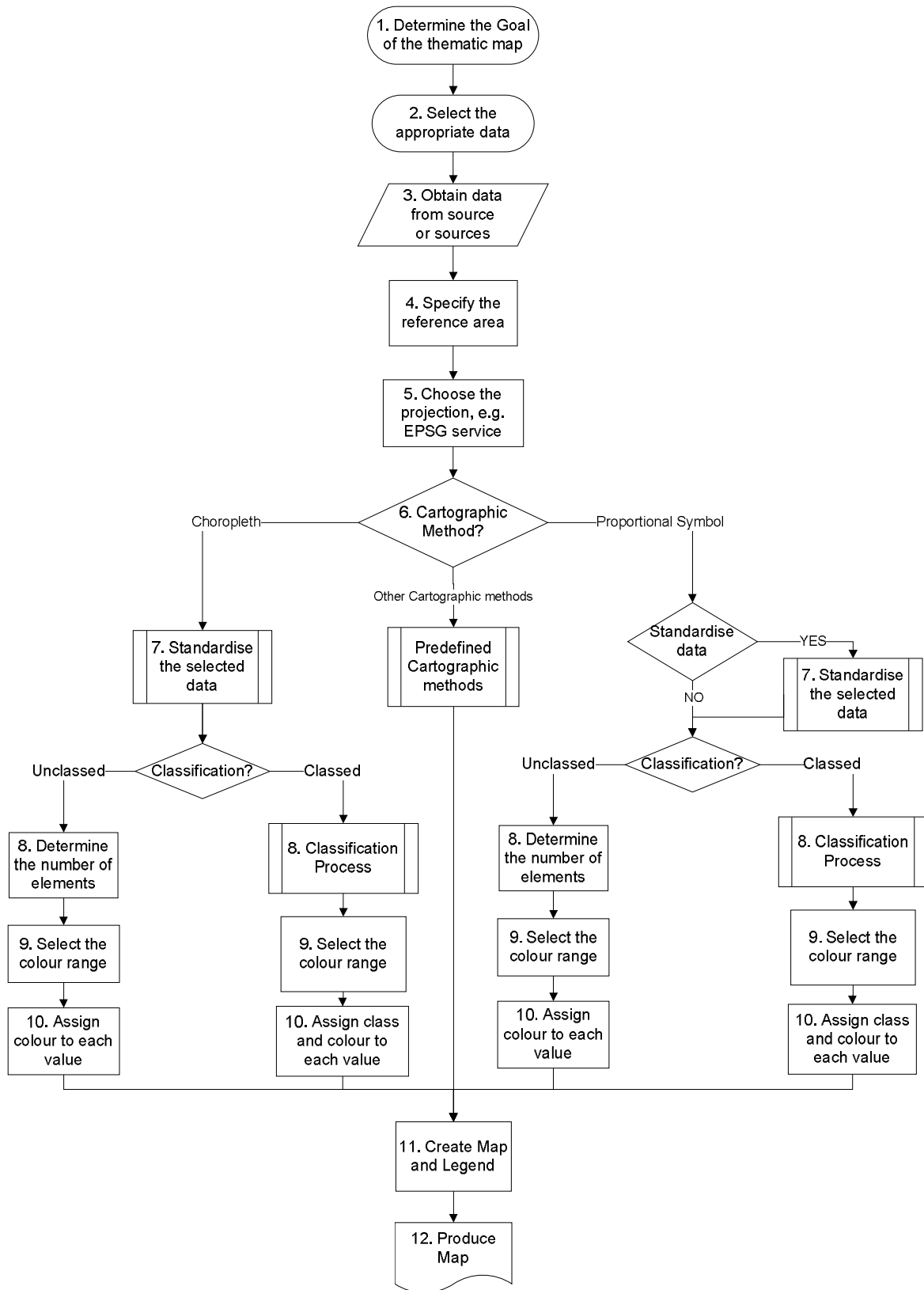
Figure 2 Flow diagrams depicting the processes involved in thematic mapping

2.3 Web mapping

Cartography has evolved with the advent of the Internet and other related technologies, which brought about new methods of geographic visualisation and data exploration (Slocum *et al.*, 2009). Web mapping is the process of designing, implementing, generating and delivering maps on the World Wide Web (WWW) (Li *et al.*, 2009). The Internet has made mapping available to a wider audience, initiating neogeography and web mash-ups (Haklay *et al.*, 2008), which have become increasingly popular. Examples are OpenStreetMap (http://www.openstreetmap.org) and Google maps (http://maps.google.com) with Panoramio photos (refer to Figure 3).

2.4 Thematic maps over the web

Thematic mapping is not a default functionality of web mapping, but provides additional data exploration and spatial analysis functionalities to the user; refer to figure 4 and 5 for examples. While advances in technology have made it possible to create animated maps, choropleth and proportional symbol maps are still the most frequently implemented thematic maps on these sites.
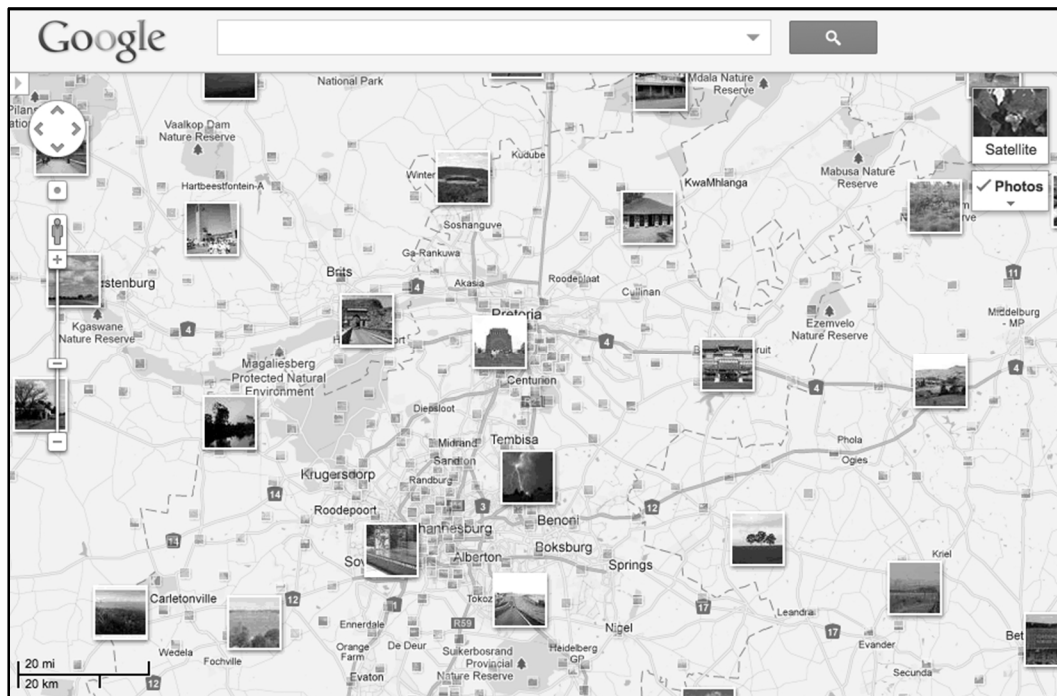

Figure 3 Google Maps with geotagged Panoramio photos

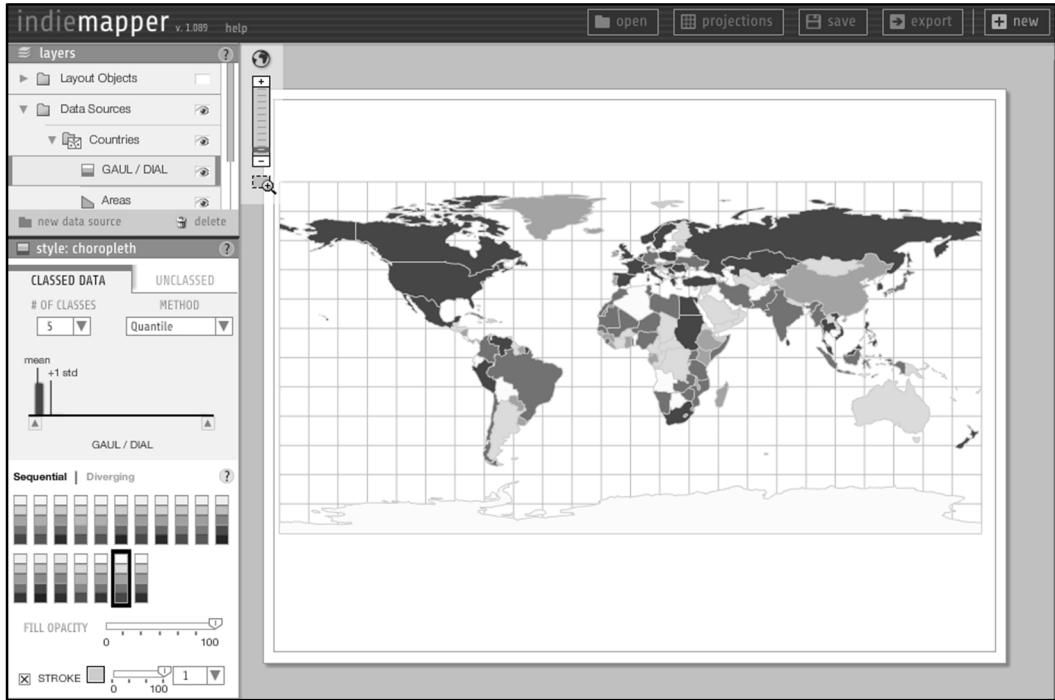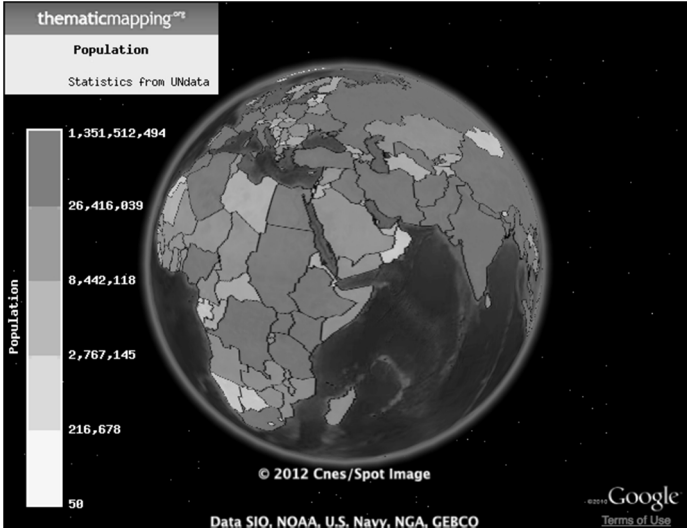**Figure 4 Indie mapper being used to produce a thematic map, on the left panel the available options are shown, http://indiemapper.com**

a) The interface selecting the parameters to produce a thematic map

b) Produced thematic map displayed on Google Earth

Figure 5 Thematic API interface for producing thematic maps, http://thematicmapping.org/engine

2.5 OGC web services

The OGC web services are modular components based on eXtensible Markup Language (XML) technology. These services are self-describing and can be published, located and invoked from anywhere on the Web or within any local network. Refer to Figure 6 for the general architecture of the OCG web service standards. The web service standards specify the input and output parameters of a web service. The graphical user interface (GUI) of a web mapping site is built by getting the input parameters from the user, sending it to a web service and presenting the web service output to the user. Our work is not concerned with the GUI, but rather with the work that the web services do. Our experiments use three OGC web services, described in the following paragraph.

The Web Map Service (WMS) provides a simple HTTP interface for requesting georeferenced map images from one or more distributed geospatial databases (OGC, 2006). Web Feature Service (WFS) specifies an interface for retrieving and manipulating geographical features (OGC, 2010b). Web Processing Service (WPS) describes a standard interface for describing a service that provides processing functionalities to be executed in a web environment (OGC 2007).
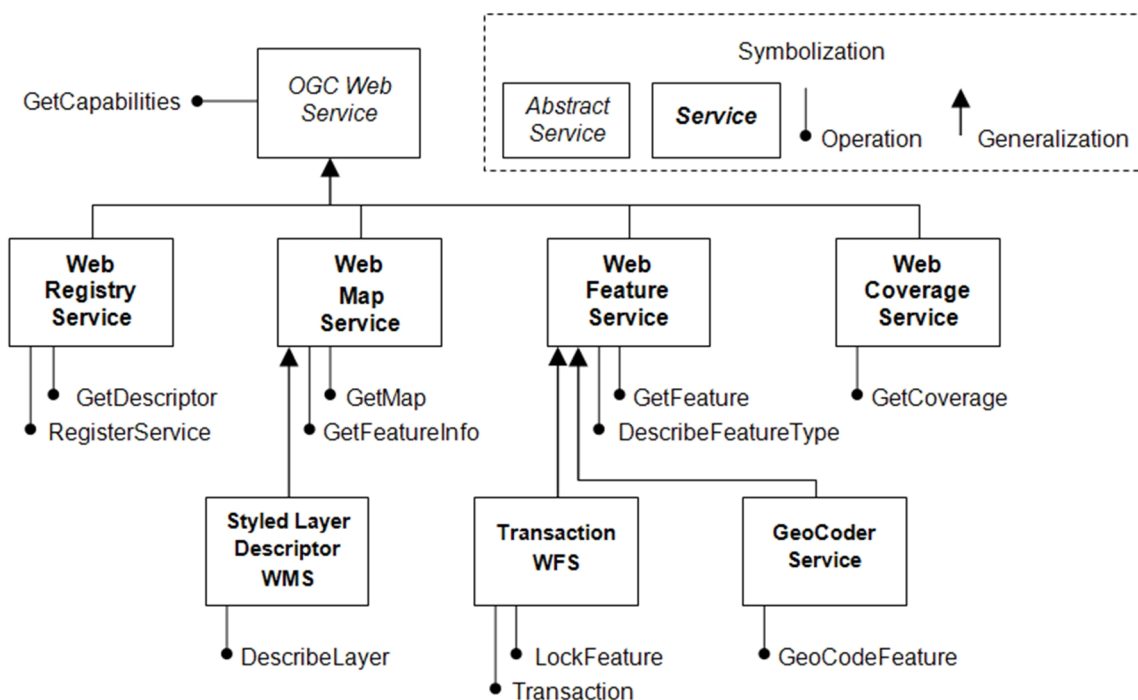
Figure 6 The OGC web services architecture (OGC, 2002)

2.6 Web service orchestration

Web service orchestration is the automated arrangement of web services in a predefined pattern based on local decisions about their interactions with one another at the message and/or execution level (Sun et al., 2010, Suazo & Aguirre, 2005). Orchestration enables sophisticated forms of interaction with inherited intelligence or implicit automatic control. It uses

a single process, referred to as a central controller, to control the underlying processes, executes the processes in the particular order in which they were specified and assigns the necessary input parameters for the WPS processes

Web service 'chaining' and web service 'composition' are two examples of orchestration. These terms are sometimes used interchangeably - but are they really the same? According to ISO 19119 (2005), Yue et al. (2007) and Alameh (2003) service *chaining* is the process of combining monolithic services into a dependent series or pipeline to create a new web service. Web service *composition* on the other hand, according to Danapaquiame and Bhavani (2010) and Zeng et al. (2003), refers to the combination of interoperable web services into a tree-like structure to create a new web service. Service chaining is thus a serial process (one web service is executed after the other), whereas service composition has a hierarchical nature (one service invokes one or more other services as part of its execution). Web service chaining and composition both take advantage of existing web services to create new, more advanced web services. Web services, conforming to ISO and OGC standards, can be used to develop such advanced geospatial web services, either through chaining or through composition. Both chaining and composition of geospatial web services can provide the intelligence required in an SII's intelligent geoportal.

3. Related work
In this section we provide a brief overview of previous research related to our work.

There have been a few proposals in recent years for improving cartographic representations of spatial data over the web. Lu et al. (2007) described research issues related to the use of Geography Markup Language (GML) for visualization, such as extending style sheets for mapping purposes. Styled Layer Descriptor (SLD) is an OGC standard XML schema that can be used together with the WMS to define styles for the publication of raster and vector data on the web. Sae-Tang and Ertz (2007) proposed an extension to SLD and Symbology Encoding (SE) for thematic mapping, called SLD-T. Similar to SLD-T, Rita et al. (2010) proposed an extension to SLD and SE for cartograms. The OGC standards working group on SLD and SE 1.2 is currently working on integrating some of the recommendations proposed in these projects into the latest versions of SLD and SE. Sanvik (2008) proposed using geobrowsers together with a modified version of the Keyhole Markup Language (KML), which has been implemented successfully and made available in Google Earth. However, this implementation is limited to Google Earth and cannot be used with web services (refer to figure 5). Our work is different in that we do not attempt to improve the thematic maps, but rather attempt to automate the process of producing the thematic maps.

Service chaining is possible with WPS services, but its functionality is limited. For example, processes cannot be run asynchronously and cannot be managed while executing (Kiehle 2010). However, wrapping existing geoprocessing functionality in a WPS makes it possible to deploy these services in the cloud, which is especially useful for resource intensive computations (Ludwig & Coetzee, 2010).

Business process execution language (BPEL) is an executable language standard that composes services by choreographing service interactions (Weerawarana *et al*., 2005). BPEL was developed by the Organisation for the Advancement of Structured Information Standards (OASIS). It is an XML-based language used to describe the control logic for web service orchestration in a business process (OASIS, 2003). BPEL makes use of web service standards as a basis: the final process created with BPEL is exposed as a web service through the Web Services Description Language (WSDL). BPEL allows the creation of a centralized process, which invokes all the containing processes in sequential order, thus controlling the entire workflow. Complex processes based on a syntactical connection between web services can be created, but web services connected through semantics are not yet supported.

Various researchers have shown that BPEL cannot be used for orchestrating OGC web services and that other solutions are required (Weiser & Zipf, 2007, Stollberg & Zipf, 2007, Schaeffer, 2008, Fleuren & Muller, 2008). The main obstacles identified at the time were the lack of SOAP support, raw binary data served by OGC services, and the manual creation of the WSDL document for each process. These obstacles created barriers so that BPEL orchestration engines cannot be used for the orchestration of OGC web services. As a solution to the transfer of raw data, Fleuren and Muller (2008) proposed using a PostGIS database to store intermediate data for orchestration with an ActiveBPEL engine. Stollberg and Zipf (2007) investigated the use of WPS for the orchestration of OGC web services, as an alternative to BPEL. WPS services were successfully used in three different ways to orchestrate OGC web services in these experiments; the WPS service was however used in manners for which it was not intended. In 2010, Foerster *et al*. stated that composition of OGC web services is still difficult, due to interoperability problems. They present a common service classification to improve semantic interoperability of web services. Experiments to perform map generalization and transformation were done using the 52° North WPS framework. Amirian *et al*. (2010) identified problems for orchestrating OGC services and other web services. They implemented a common backend, façade service and metadata-exchange approaches as solutions for data type handling, functionality mapping and standard metadata delivery, respectively. In line with these findings, we used a workflow script, and not BPEL, for the orchestration of OGC web services.

4. Orchestrating the ThematicWS

In this section we describe ThematicWS, the web service that produces choropleth and proportional symbol maps by orchestrating implementations of WMS, WFS and WPS. The purpose of our experiments was to investigate the production of thematic maps with an orchestrated web service consisting of standard OGC web services. The expected result of the experiments is an orchestrated web service, which can be called using HTTP GET/POST.

We used the Common Query Language (CQL), SLD and SLD extensions in GeoServer 2.1 in our experiments. GeoServer is an open source Java-based server that allows geospatial data to be displayed and edited (OpenGeo, 2009). GeoServer is built on the GeoTools API and

implements OGC standards. CQL is a formal language, a combination of the powerful and expressive languages such as SQL and the simple and natural query language such as Google's Query Language (The Library of Congress, 2008). We used the 52° North WPS platform for orchestrating the OGC web services. The platform provides a pluggable architecture for process and data encoding and adheres to the OpenGIS specification (http://52north.org/wps).

Figure 7 (without the ThematicWS) shows how most SDI geoportals use OGC web services in their implementations. It also shows how the ThematicWS in our experiments makes use of existing standard OGC components, WFS, WMS and WPS, to implement a new orchestrated web service that is available through the SII intelligent geoportal. Figure 7 shows that the service is composed from existing standard components: WFS, WMS and WPS. The required customised programming to prepare the SLD style sheet and statistical processing are each wrapped into a WPS. The WPS provides a standard interface to invoke a processing service that e.g., does standardisation and classification. A standard OGC web service for the standardisation and classification of data does not exist.
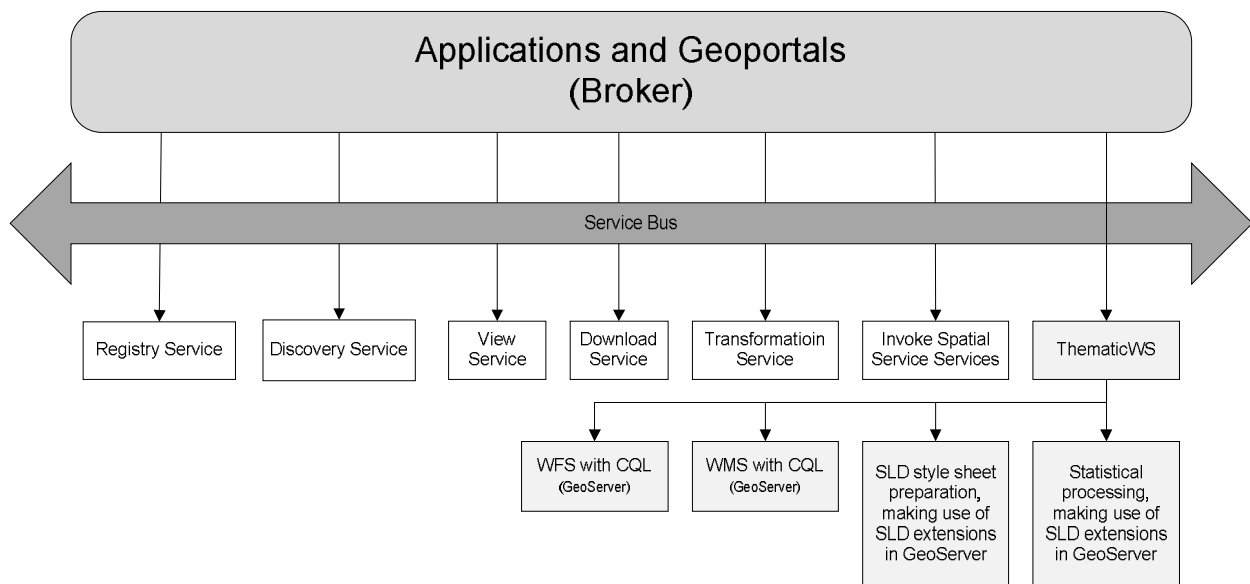
Figure 7 Components of the ThematicWS.

The following steps are executed to generate information from our ThematicWS. Figure 8 illustrate the steps.
1. The client (SII intelligent geoportal) sends a request for a thematic map to the ThematicWS.
2. ThematicWS requests data from a WFS.
3. ThematicWS requests statistical processing on the data source. We wrapped this into a WPS.
4. ThematicWS receives the requested results.
5. The SLD is prepared according to the statistical results. We also wrapped this functionality into a WPS.

6. ThematicWS requests an image from the WMS. A reference to the SLD is passed to the WMS.
7. WMS returns the image to the ThematicWS.
8. ThematicWS returns the image to the client (SII intelligent portal).
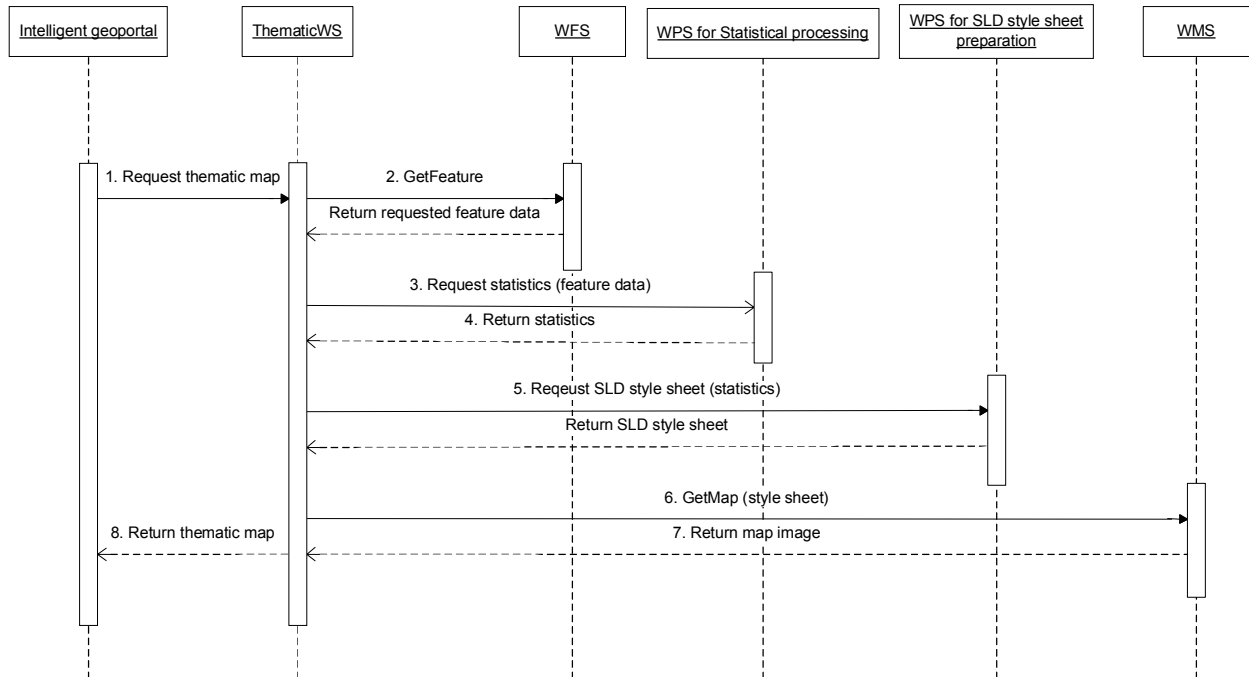
Figure 8 Sequence diagram illustrating the orchestration of web services in ThematicWS

The custom web services required for the ThematicWS were developed as WPSs using the 52° North WPS platform. Initially, the orchestration involved no complex logic and was executed sequentially by the central controller (see Figure 8). Later we added control logic to the process by specifying constraints for the classification results (see Figure 9). Different classification methods were executed until a classification method's results satisfied the constraints set.

Figure 9 Flow diagram of the implementation of the ThematicWS with constraints for controlling the classification of data

Thematic styling is possible with an implementation of SLD, but these maps are static (i.e. their styles and classes cannot vary at run-time, based on actual data values) and the generation of dynamic maps can be very demanding and time-consuming. Before the user can start thematic styling, some statistical processing is necessary, such as, the classification and standardisation of data. With the standard 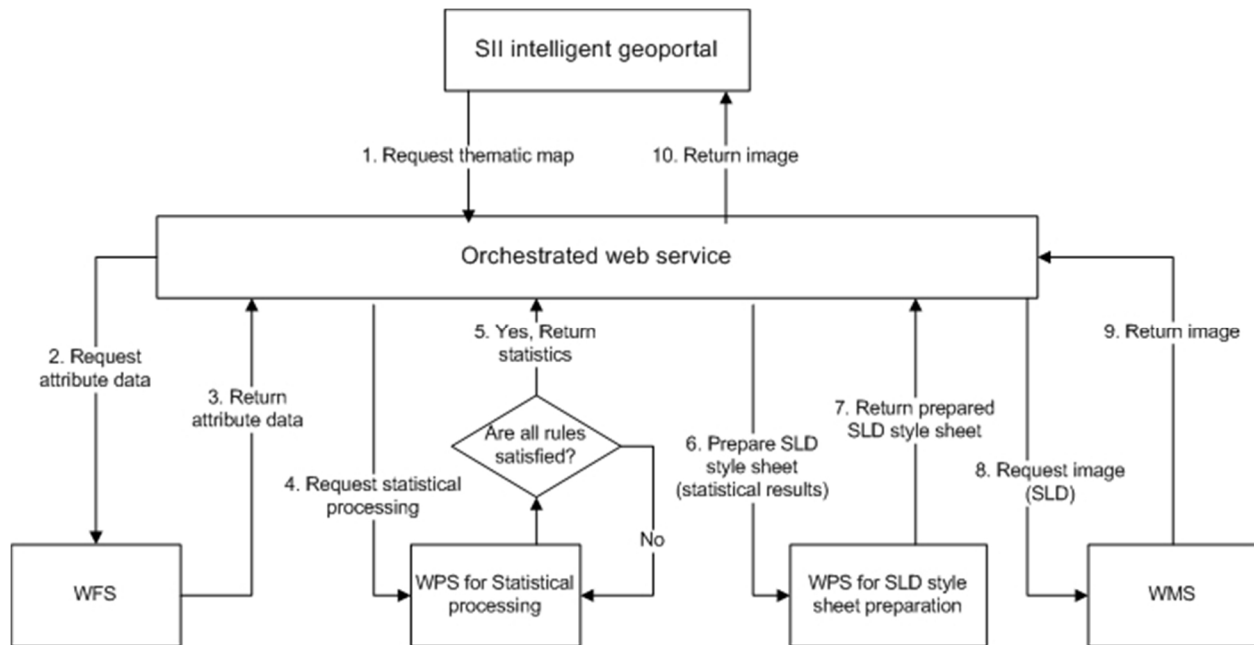SLD this processing has to be performed in a separate tool and the results have to be copied manually to the SLD. Steps 3 to 6 in Figure 8 represent this processing in our ThematicWS. Each class or value of the thematic map is represented by a rule element inside the SLD: thus, the creation of an unclassed map from a dataset with thousand different values requires thousand rules! Each of these rules has to be either prepared in a separate tool and exported to an SLD, or written manually by a user. Steps 5 and 6 refer in Figure 8 represent the preparation of the SLD rules in our ThematicWS. Figures 10 and11 are examples of rules to describe one class for a choropleth and proportional symbol map respectively.

```
 1  <Rule>
 2          <Name>SmallPop</Name>
 3          <Title>Less Than 200,000</Title>
 4          <ogc:Filter>
 5                  <ogc:PropertyIsLessThan>
 6                          <ogc:PropertyName>population</ogc:PropertyName>
 7                          <ogc:Literal>200000</ogc:Literal>
 8                  </ogc:PropertyIsLessThan>
 9          </ogc:Filter>
10          <PolygonSymbolizer>
11                  <Fill>
12                          <CssParameter name="fill">#00FF00</CssParameter>
13                  </Fill>
14                  <Stroke>
15                          <CssParameter name="stroke">#000000</CssParameter>
16                          <CssParameter name="stroke-width">1</CssParameter>
17                  </Stroke>
18          </PolygonSymbolizer>
19  </Rule>
```

Figure 10 Example of an SLD rule for a class in the choropleth map of Figure 1a)

```
 1  <Rule>
 2          <Name>SmallPop</Name>
 3          <Title>1 to 50000</Title>
 4          <ogc:Filter>
 5                  <ogc:PropertyIsLessThan>
 6                          <ogc:PropertyName>population</ogc:PropertyName>
 7                          <ogc:Literal>200000</ogc:Literal>
 8                  </ogc:PropertyIsLessThan>
 9          </ogc:Filter>
10          <PointSymbolizer>
11                  <Graphic>
12                          <Mark>
13                                  <WellKnownName>circle</WellKnownName>
14                                  <Fill>
15                                          <CssParameter name="fill">#0033CC</CssParameter>
16                                  </Fill>
17                          </Mark>
18                          <Size>8</Size>
19                  </Graphic>
20          </PointSymbolizer>
21          <PolygonSymbolizer>
22                  <Stroke>
23                          <CssParameter name="stroke">#000000</CssParameter>
24                          <CssParameter name="stroke-width">0.5</CssParameter>
25                  </Stroke>
26          </PolygonSymbolizer>
27  </Rule>
```

Figure 11 Example of an SLD rule for a class in the proportional symbol map of Figure 1b)

The first extension to GeoServer implements a dynamic symbolizer to overcome some of the limitations of SLD. This extension was implemented in GeoServer 1.7. The dynamic symbolizer is an extension of the PointSymbolizer described in OGCs SLD standard and provides the user with the following three additional functionalities: 1) create external references that contain

feature attributes as variables; 2) use decorative true type fonts as markers in the map; and 3) program one's own dynamic symbolizers that extend existing ones and have full access to an individual feature's attributes at run-time (Jordan, 2010). We used this dynamic symbolizer in our implementation of ThematicWS.

In the SLD, the Mark element of the PointSymbolizer in the standard SLD specifies the graphic element to be used on the thematic map. The value of the Mark element can be a well-known name such as 'square' or 'circle'. The ExternalGraphic element of the PointSymbolizer in the standard SLD specifies a URL to a GIF or PNG image to be used on the thematic map (Woźniak et al., 2011). In standard SLD both Mark and ExternalGraphic are static elements, implying that they cannot vary according to individual features (i.e. all features on the thematic map are represented with the same Mark or ExternalGraphic). With the GeoServer dynamic symbolizer, a user can embed attribute names into these elements for value expansion at run-time, i.e. the value of the attribute at run-time determines the value of the Mark or ExternalGraphic element (thus the thematic style).

Another feature of the dynamic symbolizer that we used in our implementation is a valid CQL expression for the WellKnownName sub-element of the Mark element and the OnlineResource/@xlink:href sub-element of the ExternalGraphic element. This feature allows for more compact styling descriptions in the SLD in cases where the symbol name can be derived from a feature attribute value (Geoserver, 2011). The dynamic symbolizer in GeoServer is implemented in the PointSymbolizer only and can thus be successfully used to create proportional symbol maps. A choropleth map, however, requires the use of the PolygonSymbolizer, for which the dynamic symbolizer is not implemented.

Figure 12 is an example of the dynamic symbolizer in the OnlineResource element that we used in our ThematicWS. The OnlineResource element contains a CQL expression that is evaluated at run-time. The SLD rule in Figure 12 was used to create an unclassed map, displaying a user defined image for each unique value in the dataset. This method is optimal for unclassed symbol maps.

```
 1   <FeatureTypeStyle>
 2       <Rule>
 3           <Title>Dynamic Symbolizer Example</Title>
 4           <PointSymbolizer>
 5               <Graphic>
 6                   <ExternalGraphic>
 7                       <OnlineResource xlink:type="simple"
 8                           xlink:href="http://mysite/geoserver_images/${municipal_type}.jpg"/>
 9                       <Format>image/gif</Format>
10                   </ExternalGraphic>
11               </Graphic>
12           </PointSymbolizer>
13           <PolygonSymbolizer>
14               <Stroke>
15                   <CssParameter name="stroke">#000000</CssParameter>
16                   <CssParameter name="stroke-width">0.5</CssParameter>
17               </Stroke>
```

```
18              </PolygonSymbolizer>
19          </Rule>
20  </FeatureTypeStyle>
```

Figure 12 Example of a style sheet implementation of the dynamic symbolizer

The second GeoServer extension to standard SLD that we used in our implementation is the ability to pass parameter values from the WMS request into the SLD style sheet. This provides the user with the ability to dynamically change colours, fonts, sizes and filter thresholds in the SLD style sheet (GeoServer, 2011). In the GetMap request a list of env parameters are added as follows:

```
    ...&env=paramName:value;otherParam=otherValue&...
```

The env function can be specified in any element of an SLD rule where an OGC expression is used, such as in the size, offsets, filter and CssParameter elements. Figure 13 is an example of an SLD rule for a choropleth map using the env function for parameter substitution in the filter and CssParameter (contains an SVG/CSS graphical-formatting parameter, e.g. stroke and fill): lowerlimit and lowercolour are specified to be env parameters in the filter and CssParameter elements respectively. Default values for each parameter are specified (200,000 and 00FF00). The default value is used if the parameter is not passed with the GetMap request. Based on the results of the data classification and standardisation, we specified the lower limit and colour for each class in the parameters of the GetMap request. This example shows how we were able to dynamically change the parameters for the resulting choropleth map.

```
1   <Rule>
2         <Name>SmallPop</Name>
3         <Title>Less Than 200,000</Title>
4         <ogc:Filter>
5               <ogc:PropertyIsLessThan>
6                     <ogc:PropertyName>population</ogc:PropertyName>
7                     <ogc:Function name="env">
8                           <ogc:Literal>lowerlimit</ogc:Literal>
9                           <ogc:Literal>200000</ogc:Literal>
10                    </ogc:Function>
11              </ogc:PropertyIsLessThan>
12        </ogc:Filter>
13        <PolygonSymbolizer>
14              <Fill>
15                    <CssParameter name="fill">
16                          <ogc:Function name="env">
17                                <ogc:Literal>lowercolour</ogc:Literal>
18                                <ogc:Literal>00FF00</ogc:Literal>
19                          </ogc:Function>
20                    </CssParameter>
21              </Fill>
22              <Stroke>
23                    <CssParameter name="stroke">#000000</CssParameter>
24                    <CssParameter name="stroke-width">1</CssParameter>
25              </Stroke>
26        </PolygonSymbolizer>
27  </Rule>
```

Figure 13 Example of a style sheet implementing parameter substitution

Finally, we used the chart extension of GeoServer in the implementation of our ThematicWS.
This extension combines the functionalities of dynamic symbolizers and Google's Charts API
(GeoServer, 2011) and makes the development of diagram maps possible. A number of styling
options are provided. Refer to the example in Figure 14.

```
1   <Rule>
2       <PointSymbolizer>
3           <Graphic>
4               <ExternalGraphic>
5                   <OnlineResource
6                       xlink:href="http://chart?cht=p&amp;chd=t:${100*employed/population},
7                           ${100 * unemployed / population}&amp;chf=bg,s,FFFFFF00" />
8                   <Format>application/chart</Format>
9               </ExternalGraphic>
10              <Size>
11                  <ogc:Add>
12                          <ogc:Literal>20</ogc:Literal>
13                          <ogc:Mul>
14                                <ogc:Div>
15                                      <ogc:PropertyName>population</ogc:PropertyName>
16                                      <ogc:Literal>20000000.0</ogc:Literal>
17                                </ogc:Div>
18                                <ogc:Literal>50</ogc:Literal>
19                          </ogc:Mul>
20                  </ogc:Add>
21              </Size>
22          </Graphic>
23      </PointSymbolizer>
```
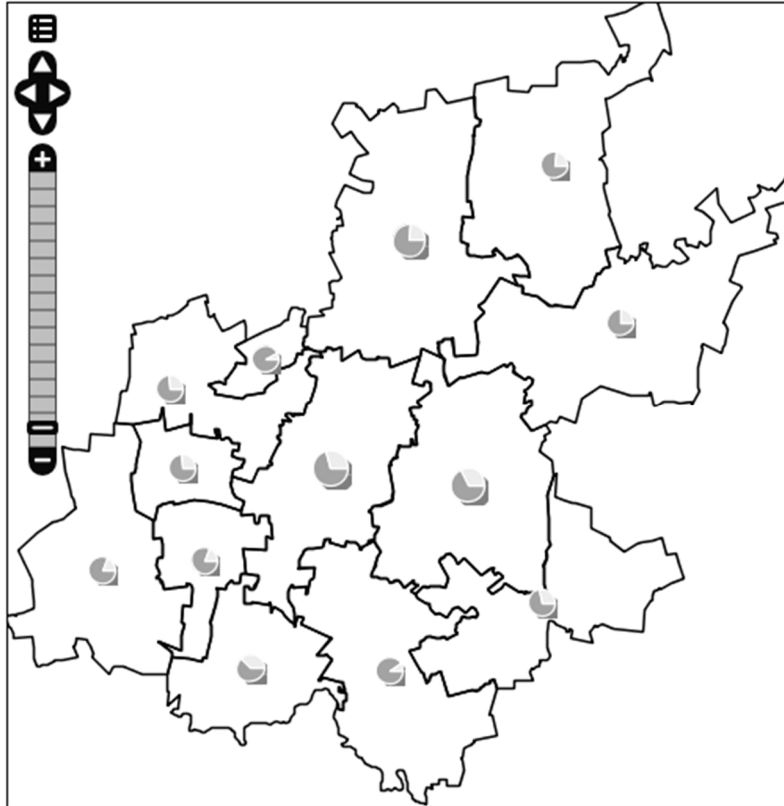
```
24    </Rule>
```


Figure 14 An example of a diagram map created with the GeoServer charts extension

5. Discussion of Results

In our experiments we orchestrated OGC web services to produce a thematic map. The orchestration was performed with a workflow script, which relied on syntactical interoperability of the services. Aside from the standard components available by OGC, we used three extensions to SLD in GeoServer to produce the thematic maps. Our experiments show that a thematic web service can be orchestrated from standard components, such as WMS, WFS and WPS, that make use of other standards, such as CQL and SLD. However, there are two limitations. Firstly, the customized programming for the statistical processing and classification needs to be wrapped into a WPS to make orchestration of standard web service components possible (refer to Figure 8). This poses a challenge to on the fly *intelligent* orchestration of web services in an intelligent geoportal.

Figure 2 provides an outline of the thematic cartography process. We showed in our experiments that there are standards that can be used in some of the steps of this process, as shown in Table 1. There is not a standard for the standardisation and classification of data and the symbolization functionality of the standards is lacking. Therefore implementations of these standards do not yet allow on the fly *intelligent* orchestration of web services. Improvements on these standards and/or semantic descriptions of these web services are required for intelligent orchestration.

19

There are several advantages to standardising the input and output of a web service for the statistical processing of data: this will make it possible to produce off-the-shelf components that can be re-used by many web thematic mapping applications (eliminating the cost of developing components); modularity and code re-use are important elements of an SOA; and thematic maps produced by different web services can more easily be compared.

Table 1. Existing standards for thematic map production

Thematic cartography process	Standard
Step 1: Determine the goal of the thematic map.	Not applicable
Step 2: Select the dataset.	Not applicable
Step 3: Obtain data from source.	WFS gets the data
Step 4: Specify the reference area.	Not applicable
Step 5: Specify the projection of the map.	Not applicable
Step 6: Decide on the thematic cartographic method.	Not applicable
Step 7: Standardise the dataset, if required.	**A standard is required**
Step 8: Classify the dataset, if the user selected this option.	**A standard is required**
Step 9 and 10: Symbolization process.	SLD allows symbolization but has limitations for dynamic thematic mapping.
Step 11 and 12: Create map and additional information, e.g. legend and layout.	WMS renders a thematic map

Secondly, the GeoServer extensions to SLD automate some of this programming, but relying on them limits interoperability, because the dynamic symbolizer, for example, is not a standard. Thus, a solution implementing the SLD extensions cannot readily be ported to another OGC compliant WMS implementation, i.e. independent intelligent orchestration of web services is not possible. Hopefully, the ongoing work in the OGC standards working group on 'SLD and SE 1.2' will solve some of these problems. At present, the standards working group 'SLD and SE 1.2' are busy with integrating a DiagramSymbolizer (similar to the PointSymbolizer) into SLD. The DiagramSymbolizer allows the user to visualize multiple data values using diagrams.

The work on the new version of SLD and SE was prompted by numerous change requests to extend SLD and SE for improved thematic mapping, some as early as 2007. A recent change request posted in the 2011 re-voting process proposes the addition of classification methods to the SLD standard. However, implementing these methods requires the DescribeLayer operation in WMS to contain Count, Maximum, Minimum, Sum, Mean and StandardDeviation properties for every ogc:PropertyName element. Therefore, the revised WMS (currently being prepared) has to precede that of the revised SLD. The GeoServer development team submitted a change request in 2011 to include all the GeoServer extensions and improvements to SLD. Work on these requests is still in progress. In addition, the extensions to SLD in GeoServer are not comprehensive, for example, CQL expressions in the LineSymbolizer and PolygonSymbolizer are not supported, but OGC is not yet looking at these.

6. Conclusion

In this paper we presented our experiments with the orchestration of a thematic web service, ThematicWS, from existing OGC web service implementations in GeoServer. Our goal was to evaluate the role of orchestration of OGC web services for thematic maps. In our experiments we showed that thematic maps can be produced by orchestrating OGC web services, but that there are limitations. Firstly, quite a lot of customised programming is required that needs to be wrapped in WPS services to standardise and classify data and also to prepare the rules for the SLD. Secondly, the extensions to SLD implemented in GeoServer, as well as those proposed by others, allow some flexibility for varying the styles of thematic maps at run-time. However, because we used these GeoServer specific extensions in ThematicWS, this part of the implementation cannot be exchanged (i.e. it is not interoperable). If only standardised OGC web services were used, interchanging different implementations of the OGC web services would be possible. The current work in OGC on improving the SLD standard may in future eliminate the need to use customised extensions to SLD.

Orchestrating OGC web services to produce thematic maps has advantages: different OGC web service implementations can be exchanged (interoperability), existing OGC web service implementations can be re-used, saving on costs (code re-use), and well-known standardised OGC protocols are used. A disadvantage of using standardised OGC web services is the lack of flexibility: with a standard one only gets the standard functionality, nothing more and nothing less. However, there will always be a trade-off between flexibility and code re-use.

The lack of standardized web services for some parts of the thematic web map production creates obstacles in the efforts to develop a spatial information infrastructure. At present, an implementation of thematic web mapping using vendor independent web services is complex. The results of our experiments show that the implementation of SII with vendor independent services can be simplified through the standardization of web services. Producing OGC web service standards, for example, for the statistical processing and classification of data, will make it possible to create vendor independent components that can be used to build an SII.

Our results prove that orchestration of web services based on syntactical interoperability is successfully applied to predefined problems, such as the automated production of thematic maps. However, semantically interoperable web services are required to solve more complex problems where, for example, a relevant service needs to be discovered at run-time.

Our experiments show that a computer can automate the process of thematic map production, but there are risks that a computer could prepare thematic maps that are misinterpreted by humans. In our research, we did not attempt to mitigate this risk, but it definitely requires further research.

In further work we aim to propose a standard web service interface for the standardisation and classification of data. We plan to extend the implementation of our ThematicWS from choropleth and proportional symbol maps to other kinds of thematic maps. In this paper we only experimented with the orchestration of web services; however, the ultimate goal is the *intelligent* orchestration of web services to produce thematic maps. For this we plan to employ semantic

web service descriptions. In addition, we aim to develop a metadata schema that describes thematic maps, which will provide cartographic information interoperability. Such a schema will be useful, for example, when evaluating climate change: the schema will enable the overlay of different thematic maps (e.g. using different classification methods and symbolization) showing sea water temperatures in different years.

Acknowledgements

References

52° North. (2010). 52° North WPS. Retrieved August 16, 2011, from 52° North: http://52north.org/wps

Alameh, N. (2003). Service chaining of interoperable geographic information web services. *Internet Computing*, 7(1), 22 - 29.

Amirian, P., Alesheikh, A., & Bassiri, A. (2010). Standards-based, interoperable services for accessing urban services data for the city of Tehran, *Computers, Environment and Urban Systems*, 34, 309–321.

Bellinger, G., Castro, D., & Mills, A. (2004). Data, Information, Knowledge, and Wisdom. *Gene*. Retrieved from http://www.systems-thinking.org/dikw/dikw.htm

Coetzee, S. (2011). Reference model for a data grid approach to address data in a dynamic SDI. *Geoinformatica*. DOI: 10.1007/s10707-011-0129-4. Retrieved August 10, 2011, from SpringerLink: http://0-www.springerlink.com.innopac.up.ac.za/content/px350m5045084814/

Danapaquiame, N. & Bhavani, P. (2010). A Novel approach for parallel web service composition. *International Conference on Communication and Computational Intelligence* (pp. 613-616). India: Kongu Engineering College.

Dent, B.D. (1996). *Cartography: Thematic map design* (4th ed.). Wm. C. Brown Publishers.

Fleuren, T., & Muller, P. (2008). BPEL Workflows Combining Standard OGC Web Services and Grid-enabled OGC Web Services. *Software engineering and advanced applications, SEAA Conference*, Parma, Italy, September 3-5, 2008.

Foerster, T., Lehto, L., Sarjakoski, T., Sarjakoski, L., & Stoter, J. (2010). Map generalization and schema transformation of geospatial data combined in a Web Service context. *Computers, Environment and Urban Systems*, 34, 79–88.

Fu, P., & Sun, J. (2010). *Web GIS principles and applications*. ESRI Press.

GeoServer. (2011, June 23). Styling. Retrieved June 27, 2011, from GeoServer: http://docs.geoserver.org/stable/en/user/

Haklay, M., Singleton, A., & Parker, C. (2008). Web Mapping 2.0: The Neogeography of the GeoWeb. *Geography Compass.* Retrieved January 31, 12, from http://onlinelibrary.wiley.com/doi/10.1111/j.1749-8198.2008.00167.x/full

INSPIRE. (2007). *INSPIRE Network Service Architecture*. Retrieved July 5, 2011, from INSPIRE: http://inspire.jrc.ec.europa.eu/reports/ImplementingRules/network/D3_5_INSPIRE_NS_ Architecture_v3-0.pdf

International Organization for Standardisation, (2005). ISO 19119:2005. Geographic information – Services.

International Organization for Standardisation, (2005). ISO 19128:2005. Geographic information – Web Map Server Interface.

Iwaniak, A., Kaczmarek, I., Kubik, T., Lukowicz, J., Paluszyński, W., Kourie, D., Cooper, A., & Coetzee, S. (2011). An Intelligent Geoportal for Spatial Planning. *25th International Cartographic Conference*, Paris, 4-8 July 2011.

Jordan, M. (2010). Dynamic Symbolizers. Retrieved June 25, 2011, from GeoServer: http://blog.geoserver.org/2008/12/08/dynamic-symbolizers-part-1/

Kiehle, C. (2001). OGC Web Processing Service (WPS) – Towards version 2.0. 28th *OpenGridForum*. Munich, Germany. Retrieved August 19, 2011, from OpenGridForum: http://www.ogf.org/OGF28/materials/1971/OGC-OGF-Session1-4-kiehle.pdf

Li, Y., Dong, X., & Chi, G. (2009). Towards spatial information services with applied extension to SVG. International conference on new trends in information and service science. Beijing, China, June 30 – July 2, 2009.

Lu, C., Dos Santos, R.F., Sripada, L.N., & Kou, Y. (2007). Advances in GML for geospatial applications. *Geoinformatica*, 11, pp131–157.

Ludwig, B., & Coetzee, S. (2010). A comparison of platform as a service (PaaS) clouds with a detailed reference to security and geoprocessing services, *WebMGS 2010, 1st International workshop on Pervasive Mapping, Geoprocessing and Services*, Lake Como, Italy, 26-27 August 2010.

Maguire, D., & Longley, P. (2005). The emergence of geoportals and their role in spatial data infrastructures. *Computers, Environment and Urban Systems*, 29, 3–14.

Merriam-Webster. (2012). Merriam-Webster online dictionary, Retrieved February 16, 2012, from http://mw1.merriam-webster.com/

OASIS. (2003). Business Process Execution Language for Web Services 1.1.

OGC. (2002). OGC Web Map Server Implementation Specification 1.1.1.

OGC. (2004). OpenGIS Web Map Server Cookbook 1.0.2. (Kolodziej, K., Ed.)

OGC. (2006). OGC Web Map Server Implementation Specification 1.3.0.

OGC. (2007). OGC Web Processing Service 1.0.0.

OGC. (2007b). Discussions, findings, and use of WPS in OWS-4. (Keens, S., Ed.)

OGC. (2010). OGC Web Services Common Standard.

OGC. (2010b). OpenGIS Web Feature Service 2.0 Interface Standard.

OpenGeo. (2009). What is GeoServer. Retrieved June 21, 2011, from GeoServer: http://geoserver.org/display/GEOS/What+is+GeoServer

OpenLayers. (2011). *OpenLayers*. Retrieved August 8, 2011, from OpenLayers: http://openlayers.org

Parr, D. (2000). *GIS Glossary of Terms*. Park Ridge: URISA.

Peng, Z., & Tsou, M. (2003). *Internet GIS*. New Jersey: John Wiley and Sons, Inc.

Rita, E., Borbinha, J., & Martins, B. (2010). Extending SLD and SE for cartograms. *FOSS4G*.

Robinson, A.H., Morrison, J.L., Muehrcke, P.C., Kimerling, A.J., & Guptill, S.C. (1995). *Elements of cartography* (6th ed.). John Wiley and Sons, Inc.

Sae-Tang, A. & Ertz, O. (2007). Towards web service dedicated to thematic mapping. *OSGeo Journal*, 3, 31-34.

Sandvik, B. (2008). *Using KML for Thematic Mapping*. MSc Thesis, University of Edinburgh, Edinburgh.

Santos, R. D., Lu, C., Lakshmi, N., & Kou, Y. (2007). Advances in GML for Geospatial Applications. *Geomatica*, 11, 131-157.

Schaeffer, B. (2008). Towards a Transactional Web Processing Service. *GI Days 2008*. Münster, Germany, June 16 - 17, 2008.

Slocum, T., McMaster, R., Kessler, F., & Howard, H. (2009). *Thematic cartography and geovisualization* (3rd ed.). Upper Saddle River: Pearson Prentice Hall.

SourceForge. (2011). *GeoServer: Download statistics*. Retrieved August 9, 2011, from SourceForge: http://sourceforge.net/project/stats/detail.php?group_id=25086&ugn=geoserver&mode=year&&type=prdownload

StatsSA. (2001). *South African Census data*. South Africa.

Stollberg, B., & Zipf, A. (2007). *OGC Web Processing Service Interface for Web Service Orchestration*, In: Lecture Notes in Computer Science, Vol. 4857, pp. 239- 251, Springer Berlin and Heidelberg.

Suazo, N.C., & Aguirre, J.O.O. (2005). Aspect-oriented web services orchestration. *2nd International Conference on Electrical and Electronics Engineering and XI Conference on Electrical Engineering*. Mexico City, Mexico. September 7-9, 2005.

Sun, J., Liu, Y., Dong, J.S., Pu, G., & Tan, T.H. (2010). Model-based methods for linking web service choreography and orchestration. *Asia Pacific Software Engineering Conference*. Sydney, Australia, November 30 - December 3, 2010.

The Library of Congress. (2008). Search/Retrieval via URL. Retrieved June 27, 2011, from The Library of Congress: http://www.loc.gov/standards/sru/specs/cql.html

Wade, T., & Sommer, S. (2006). *A to Z GIS*. Redlands, California: ESRI Press.

Weerawarana, S., Curbera, F., Leymann, F., Storey, T., & Ferguson, D.F. (2005). *Web Service Platform Architecture*, Pearson Education, Indiana.

Weiser, A., & Zipf, A., (2007). Web service orchestration of OGC web services. In J. Li, S. Zlatanova, & A.G. Fabbri (Eds.), *Lecture notes in Geoinformation and Cartography* (pp. 239 – 254). Berlin Heidelberg, Springer.

Woźniak, E., Iwaniak, A., & Netzeł, P. (2011). Wykorzystanie standardów OGC do wizualizacji danych w infrastrukturze informacji przestrzennej [Use of OGC standards for data visualization the spatial information infrastructure], *Przegląd Geodezyjny*, Wrocław, Poland, (In Polish).

Yue, P., Di, L., Yang, W., Yu, G., & Zhao, P. (2007). Semantics-based automatic composition of geospatial Web service chains. *Computers & Geosciences*, 33, pp. 649-665.

Zeng, L., Benatallah, B., Dumas, M., Kalagnanam, J., & Sheng, Q. (2003). Quality driven web services composition. *12th International Conference on World Wide Web*, pp. 411 - 421, Budapest, May 20-24, 2003.