SCHEMA TRANSLATIONS BY XSLT FOR GML-ENCODED GEOSPATIAL DATA IN HETEROGENEOUS WEB-SERVICE ENVIRONMENT

L. Lehto and T. Sarjakoski

Finnish Geodetic Institute, PO Box 15, 02431 Masala, Finland - (Lassi.Lehto, Tapani.Sarjakoski)@fgi.fi

Commission IV, WG IV/2

KEY WORDS: Web based, Integration, Interoperability, Mobile, Internet

ABSTRACT:

The paper discusses online integration of XML-encoded datasets in the current Web services environment, especially concentrating on the required schema transformations. The approach is based on the use of a generic XML transformation technology, called Extensible Stylesheet Language Transformations (XSLT). The role of the data integration process in a layered service architecture framework is described. The basics of the applied XML technology are introduced. A case implementation, built as the prototype service of the GiMoDig project, demonstrates the benefits yielded by the approach. The required schema translations can be carried out as a single process, together with the needed coordinate transformations. The same base methodology can be applied both to query and data transformations. The declarative nature of the transformation definition, represented as templates stored in an XML-encoded text file, makes the transformation process easy to debug and fine-tune. As the final result, cross-border map visualization is presented.

1. INTRODUCTION

Integration of distributed geospatial databases is a rather well established research area. Many solutions have been proposed over the years, based on various database tools and technologies. The recent development of the Web as a common platform for geospatial data delivery and processing makes the traditional problem of database integration to emerge in a new setting.

Simple Web-based map services have become commonplace. As more sophisticated service types are being introduced, the focus in research and development is shifting away from issues related to pure map visualization to processes involving real spatial datasets. Extensible Markup Language (XML) based data encoding has been widely adopted in various application areas, geographic information not being an exception (W3C, 2004a) (Lehto, 2000). Geography Markup Language (GML) is a well-known XML vocabulary designed for applications dealing with geospatial data (OGC, 2003).

The new challenge encountered in this approach is to integrate XML-encoded heterogeneous geospatial datasets in real-time. The data integration process involves at least two aspects: schema translations and coordinate system transformations. These two integration tasks are dealt with in the following. The issues are discussed in the context of a seamless, cross-border mobile map service, based on open system architecture.

In this paper the need for real-time data transformations is first motivated and the chosen general service architecture introduced. The applied generic XML transformation technology is then discussed. Finally, as a case implementation a prototype cross-border GML data service developed by a four-nation EU-funded project: "Geospatial Info-Mobility Service by Real-Time Data-Integration and Generalisation", (GiMoDig, 2004) is presented.

2. NEED FOR DATA INTEGRATION

The rapidly expanding European integration development increases the demand for consistent, continent-wide geospatial data services. A few European Commission-backed initiatives are already working to facilitate this process. These initiatives include projects like GINIE, GETIS, EULIS, INSPIRE, GiMoDig etc. Various data harmonization processes of the EuroGeographics also aim at same target. The main results of this work can be seen in the development of the European-wide datasets, like EuroGlobalMap, EuroRegioMap and SABE (EuroGeographics, 2004).

In the case of the major national datasets, like the topographic map series, it has become obvious that the Pan-European geospatial data provision must be based on databases maintained in the national data models by the national authorities. Standardized access interfaces and spatial data encoding mechanisms would be applied to achieve the desired cross-border accessibility. The INSPIRE Architecture and Standards Position Paper (INSPIRE, 2004) proposes that the OGC's implementation level specification Web Feature Service (WFS, 2003), or rather the upcoming ISO-released official version of it, be used as a query interface by the data providers. According to the WFS specification, the dataset provided by the service is to be encoded in XML format - in compliance with the GML specification of the OGC.

In the case of the proposed European geospatial data service architecture there are two basic integration tasks. Firstly, the coordinate values presented in the individual national coordinate systems have to be transformed into a common Pan-European reference frame. Secondly, the data models applied in each country for comparable data contents, like the information normally presented on large-scale topographic maps, vary considerably from country to country. This fact necessitates

schema transformations to be carried out before the datasets can be combined.

3. OPEN SERVICE ARCHITECTURE

The task of data integration can be seen as a part of a larger distributed data processing framework. The research described in this paper has produced a design for layered service architecture. This architecture has been developed in the context of a cross-border map service aimed at mobile users, but the same approach can be adapted in many different kinds of online services involving geospatial data (GiMoDig, 2003a).

The open system architecture could be based on a layered service stack, in which a service would make queries to the service below it, do some processing on the data received as a response, and provide the results of this process as a service to the service layer above it. The level of detail in specifying the layers is a matter of discussion, but if the services were to be run on separate computers communicating through network, too fine-grained service definition would create a significant disadvantage in terms of overall system performance.

For the above-mentioned reason, five-level system architecture is proposed (Figure 1) (Lehto, 2003). On the first level the data providers (e.g. NMAs) would run a Data Service providing raw spatial data in an XML-encoded form. Above the data services is the Data Integration Service layer. The responsibilities of this layer include for instance coordinate transformations to a common reference frame and other data integration procedures, like schema transformations.
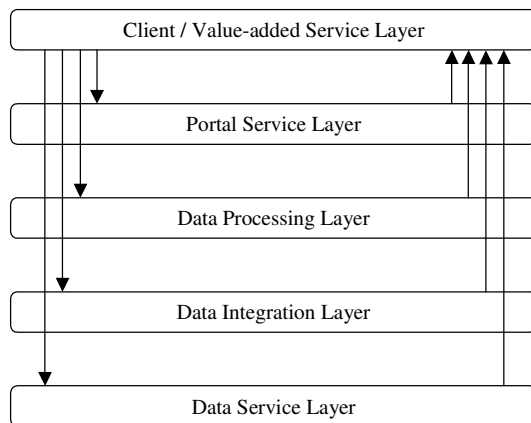

Figure 1. The Open Layered Service Architecture

On the third level in the architecture is the Data Processing layer. This layer is responsible for various data processing and analysis tasks, like map generalization or dynamic labeling.

The fourth layer in the system architecture is called Portal Service. The main responsibilities of this layer can be listed as: provide basic metadata service to the client, process the service requests coming from the client subsequently forwarding the request in an appropriate form to the Data Processing layer below, and transform the resulting piece of geospatial data into an visual representation, according to the capabilities of the client platform in question. It should be noted that in the service

architecture the query results are represented in the form of XML-encoded spatial data (e.g. GML) up to Portal Layer. Only there is the query dataset transformed into a visual map image, styled appropriately for the client environment in use.

On the fifth layer are finally the client applications. An advantage of the layered architecture approach is that the results can be adapted to a wide set of different client environments. For example the following three client platforms could be considered: the traditional Web browsing on a PC platform, the more restricted Web access on PDA devices and the various different client applications on mobile phones.

4. XSLT TECHNOLOGY

As the number of XML-based spatial data services increases on the Web, the need to employ XML-technologies in the processes involving geospatial data becomes obvious. One of the most significant technologies developed for processing XML-encoded data is called Extensible Stylesheet Language Transformation (W3C, 1999b), a mechanism for transforming an XML document into another XML document.

The Extensible Stylesheet Language (XSL) specification has been developed by the World Wide Web Consortium as a tool for defining presentation characteristics of an XML dataset (W3C, 2004b). In connection to this work the W3C has created a specification for transforming XML documents, XSL Transformations (XSLT). XSLT is primarily designed for transforming XML documents for presentation purposes. Typical examples include dynamic creation of the table of contents, and creation of a tabular presentation of some data values in the source document.

As an analogy in the geospatial data domain, XSLT could be used to transform a dataset from an application-specific spatial data structure into a map image, for instance in the form of the new Web vector graphics standard, Scalable Vector Graphics (W3C, 2003) (Lehto et al., 2001). The other transformations being considered in geospatial applications include data model transformations, coordinate transformations, and generalization of spatial data (Lehto and Kilpeläinen, 2001).

The XSLT specification is a promising tool to carry out the tasks encountered when integrating spatial datasets in real-time. Most simple integration operations are readily available. These include tasks like changing the naming system applied, grouping data from several feature classes into one class or dividing data from one feature type into several types, changing code tables etc.

More sophisticated integration operations can be added via the XSLT extension mechanism. The extensions can be programmed e.g. in Java, offering an environment for procedural programming. Typical examples include different coordinate manipulations, like coordinate reference system transformations, changes in geometric primitive types (e.g. area collapsed to a point) etc. The integrated datasets are written out as XML data, presented in a common GML application schema. The extension mechanism available in the XSLT process enables arbitrary, application-specific functions to be introduced into the transformation process. Several XSLT processes can also be chained together, if the task is too complicated to be expressed as one individual transformation.

An example of a simple XSLT declaration is shown in the Figure 2. XSLT declarations are expressed in the form of templates. The template in the example selects all elements representing buildings ('Rakennus' in Finnish) from the source tree that match to the selection phrase (expressed in a language called XPath) (W3C, 1999a), then filters out all elements for which the given test phrase inside the 'xsl:if' element does not hold. All elements inside the template not belonging to the xsl-namespace are written to the result tree. For instance, in the example the Building-element forms part of the target common vocabulary, so the effect of the transformation in this case would be a change in the naming system (from the Finnish to English terms) and a change in the collection criteria (only buildings with area ('pinta-ala' in Finnish) larger than the threshold value will be included). After the instruction: 'xsl:apply-templates', the process continues down the XML tree. The transformation goes on until no more matching elements are found.

```
<xsl:template match="Rakennus">
  <xsl:if test="pinta-ala &gt; 200">
    <Building>
      <xsl:apply-templates/>
    </Building>
  </xsl:if>
</xsl:template>
```

Figure 2. A sample XSLT template

5. CASE: GIMODIG PROTOTYPE

5.1 General

The XSLT-based data integration transformation described is being tested in a European Union funded project GiMoDig. The Finnish Geodetic Institute acts as a coordinator for the project. The other participants are the University of Hanover and the NMAs of Finland, Sweden, Denmark and Germany (Sarjakoski et al., 2002).

The objective of the GiMoDig project is to develop and test methods for delivering geospatial data to a mobile user by means of real-time data-integration and generalization. The project aims at creation of a seamless data service providing access, through a common interface, to the primary topographic geo-databases maintained by the NMAs in various countries. A special emphasis is put on providing appropriately generalized map data to the user depending on a mobile terminal with limited display capabilities.

In the GiMoDig system architecture each participating NMA provides geospatial data through the WFS interface, encoded in a country-specific XML-format (GML Application Schema). These datasets are processed by a middleware service on the Data Integration layer to integrate the pieces of data coming from individual countries into a common application schema and coordinate system. The middleware service employs the XSLT technology extensively in the process. The service is implemented as a Java servlet environment and the XSLT Processor used is a product called Xalan from the Apache community (Xalan, 2004).

5.2 Global Schema and Service Architecture

In the GiMoDig project there are four different national data models involved. Each of the participating countries has organized the topographic map data in an individual way. For the purposes of the data integration a common data model, named GiMoDig Global Schema has been developed (GiMoDig, 2003b), (Afflerbach et. Al, 2004). This data model consists of 17 different Feature types. These types are selected based on the data availability in the national databases on one hand, and on the requirements of the selected mobile use cases on the other. The list of the Global Schema Feature types is shown in Table 1.

Feature type	Geometry type
Administrative Boundary	Line
Water (except inland)	Area
Watercourse	Area or Line
Lake / Pond	Area
Marsh / Swamp	Area
Park	Area
Building	Area
Contour Line (Land)	Line
Cropland	Area
Named Location	Point
Built-Up Area	Area
Railway	Line
Road	Line
Trail / Footpath	Line
Airport / Airfield	Area
Forest	Area
Grassland	Area

Table 1. The GiMoDig Global Schema Feature types

The GiMoDig prototype service is built according to the system architecture illustrated in the Figure 1. As the aim in the project is to develop as open service architecture as possible, the access interfaces to each of the service layers are based on internationally accepted standards. Consequently, the four national data services implement the OGC's WFS interface (OGC, 2002).

The Data Integration service also implements the WFS interface providing a single access point to all of the participating national data services. As such, the Integration Service plays the role of a Cascading WFS. Query interface for the Data Processing layer (Generalization Service) has been developed in the project, as one does not exists as a result of the international standardization efforts. On the Portal layer the widely recognized Web Map Service (OGC, 2001) interface is used, together with the newer OpenLS Presentation Service interface (OGC, 2004).

5.3 Use of XSLT in the Integration Service

In the GiMoDig Integration Service the XSLT technology is used extensively. As the incoming queries are encoded in XML, according to the WFS specification, and expressed in terms of the Global Schema and the common coordinate system, the XSLT mechanism can be equally well applied also for the query transformation.
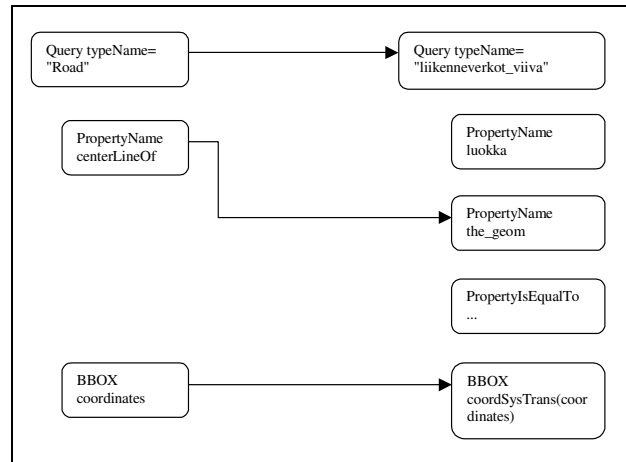
The query typically includes a spatial query window. The extent of the window has to be investigated to determine which national data services need to be included into the subsequent local WFS query processing. The initial query window, given in the common coordinate system, needs to be transformed into each of the involved local coordinate systems.

The Feature types and their requested properties need to be mapped to the corresponding local types and properties. In many cases simple one-to-one relationship between the global and local types does not exist, but additional conditions must be introduced to the local queries. In some of the cases one global query actually generates two or more local queries. Thus, a schema transformation is also applied on the incoming queries, not only on the outbound datasets.

In the GiMoDig prototype service the WFS query transformations are carried out with the help of the XSLT mechanisms. The code sample in Figure 3 illustrates a query transformation declaration from the GiMoDig global query to the corresponding Finnish local query (Feature type 'Road'). The following schema illustrates the process also in graphic form.
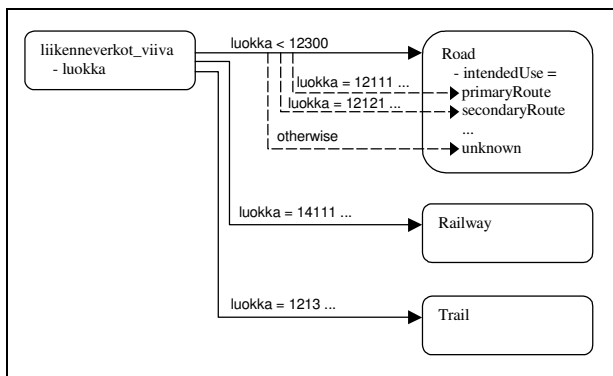
The first template demonstrates how additional conditions are introduced to the local query (PropertyIsEqualTo). The last template shows how an XSLT extension function is called to carry out the needed query window coordinate system transformation (ETRS2NLS:coordSysTrans).

The resulting GML dataset is transformed from the local national data model to the defined Global Schema applying the XSLT process declared in the sample code shown in Figure 4. The code illustrates that the local Feature type (nls:liikenneverkot_viiva) actually contains three Global Schema Feature types (gmd:Road, gmd:Railway, gmd:Trail). These are differentiated in the local data model by a property (nls:luokka). The same property also identifies the road classification and can thus be used to determine value for the corresponding Global Schema property 'intendedUse'. The figure also contains a graphic illustration of the transformation procedure to help the code interpretation.


```
<xsl:template match="wfs:Query[@typeName = 'Road']">
  <Query typeName="liikenneverkot_viiva" handle="q14">
    <ogc:PropertyName>luokka</ogc:PropertyName>
    <xsl:if test="wfs:PropertyName = 'centerLineOf'">
      <ogc:PropertyName>the_geom</ogc:PropertyName>
    </xsl:if>
    <ogc:Filter>
      <ogc:And>
        <ogc:Or>
          <ogc:PropertyIsEqualTo>
            <ogc:PropertyName>luokka</ogc:PropertyName>
            <ogc:Literal>12111</ogc:Literal>
          </ogc:PropertyIsEqualTo>
          …
        </ogc:Or>
        <xsl:apply-templates select="ogc:Filter"/>
      </ogc:And>
    </ogc:Filter>
  </Query>
</xsl:template>
<xsl:template match="ogc:Filter">
  <xsl:apply-templates/>
</xsl:template>
<xsl:template match="ogc:BBOX">
  <ogc:BBOX>
    <ogc:PropertyName>the_geom</ogc:PropertyName>
    <gml:Box srsName="EPSG:2393">
      <gml:coordinates>
        <xsl:value-of
select="ETRS2NLS:coordSysTrans(gml:Box/gml:coordinates)"/>
      </gml:coordinates>
    </gml:Box>
  </ogc:BBOX>
</xsl:template>
```

Figure 3. A sample WFS Query Transformation

```
<xsl:template match="nls:liikenneverkot_viiva">
  <xsl:choose>
    <xsl:when test="nls:luokka &lt; 12300">
      <gmd:Road fid="Road.NLS.{substring-after(@fid, '.')}">
        <xsl:apply-templates select="nls:the_geom"/>
          <xsl:choose>
            <xsl:when test="nls:luokka = 12111 or nls:luokka = 12112">
              <gmd:intendedUse>primaryRoute</gmd:intendedUse>
            </xsl:when>
              …
            <xsl:otherwise>
              <gmd:intendedUse>unknown</gmd:intendedUse>
            </xsl:otherwise>
          </xsl:choose>
      </gmd:Road>
    </xsl:when>
    <xsl:when test="nls:luokka = 14111 or nls:luokka = 14112">
      <gmd:Railway fid="Railway.NLS.{substring-after(@fid, '.')}">
        …
      </gmd:Railway>
    </xsl:when>
    <xsl:when test="nls:luokka = 12313 or nls:luokka = 12314">
      <gmd:Trail fid="Trail.NLS.{substring-after(@fid, '.')}">
        …
      </gmd:Trail>
    </xsl:when>
  </xsl:choose>
</xsl:template>
```

Figure 4. A Sample GML data transformation declaration

As can be seen from the above examples, the same base technology (XSLT) can be used to transform both the initial data query and the resulting dataset. The approach makes it straightforward and logical to define the transformations. As the transformation declarations, represented as XML text-files, are read into the integration process during the query processing, the procedure is easy to debug and fine-tune.

The Figure 5 shows a sample map view, received as a WMS response from the GiMoDig Portal Service. The map includes data coming from the Swedish (the left-hand side of the image) and from the Finnish GiMoDig WFS node (the right-hand side of the image). The datasets were transformed to the common coordinate system (ERTS89) and to the common data model (GiMoDig Global Schema) by the GiMoDig Integration Service. Subsequently, the integrated dataset has been transformed into an SVG image by the Portal Service, and finally displayed in the client device.

Figure 5. A Sample map display, provided by the GiMoDig Portal Service

6. CONCLUSION

The provision of adequate information services is becoming a challenge in the increasingly heterogeneous European Union. Accession countries are about to add a new dimension also to the puzzle of spatial data service provision in the Union. Local data management procedures and data models differ significantly from each other. It is essential that the spatial data may be managed according to the primary, local requirements, but at the same time offered for international use in compliance with a harmonized data model.

Advanced, spatially sophisticated services cannot be created on basis of – perhaps visually harmonized – map series; they require availability of rich source datasets in a common data model. The huge amount of data involved, together with the active maintenance routines carried out by local organizations, effectively prohibits approaches based on centralized data base management. Online data integration, based on real-time transformations, seems to be the only solution for advanced information services with Pan-European coverage.

Syntactic harmonization is rapidly being achieved with the general acceptance of the XML as the common data encoding mechanism. In the geospatial domain the GML vocabulary is equally widely recognized as the language for spatial data encoding on the Web. However, due to the differences in the national data models, the local GML Application Schemas are different and thus require schema transformations to be carried out before datasets can be integrated.

XSLT, the generic transformation tool for XML-encoded data, can be applied to perform schema transformations also on geospatial data. The approach presented in the paper enables real-time schema and coordinate transformations to be carried out both on spatial data queries, and on the resulting GML datasets alike. Development of integration services becomes straightforward, as the same declarative mechanism can be used on both ways.

The positive results achieved in the EU-funded development and research project GiMoDig confirm the feasibility of the approach. A prototype four-nation cross-border GML data service has been developed providing data in the common data model (GiMoDig Global Schema) and in the common coordinate system (ETRS89). Various different mobile map services have been built on top of this base service to demonstrate the potential end-user applications enabled by the real-time data integration.

7. ACKNOWLEDGEMENTS

REFERENCES

Afflerbach, S., Illert, A., Sarjakoski, T., 2004. The Harmonisation Challenge of Core National Topographic Databases in the EU-project GiMoDig. International Archives of Photogrammetry and Remote Sensing, Commission IV, Istanbul.

EuroGeographics, 2004. EuroGeographics, Products and Services, http://www.eurogeographics.org/eng/04_products.asp (accessed 26 Apr. 2004)

GiMoDig, 2003a. GiMoDig System Architecture Specification, http://gimodig.fgi.fi/pub_deliverables/Gimodig_D4_1_1-Arch_Spec.pdf (accessed 26 Apr. 2004)

GiMoDig, 2003b. GiMoDig Global Schema, http://gimodig.fgi.fi/pub_deliverables/GiMoDigD521_global.pdf (accessed 26 Apr. 2004)

GiMoDig, 2004. GiMoDig Project Home Page, http://gimodig.fgi.fi (accessed 26 Apr. 2004)

Lehto, L. and Kilpeläinen, T., 2001. Real-Time Generalization of XML-Encoded Spatial Data on the Web, *Proceedings of the GIS Research in the UK, 9th Annual Conference, GISRUK 2001*, University of Glamorgan, Wales, April 18–20, 2001, pp.182–184

Lehto, L., Kähkönen, J. and Sarjakoski, T., 2001. Multi-purpose Publishing of Geo-data in the Web, *Proceedings of the 4th AGILE Conference on Geographic Information Science*, Brno, Czech Republic, April 19–21, 2001, pp. 209–214

Lehto, L., 2000. XML in Web-based Geospatial Applications, *Proceedings of the 3rd AGILE Conference on Geographic Information Science*, Helsinki/ Espoo, May 25–27, 2000, pp. 162–167

Lehto, L., 2003. A Standards-Based Architecture for Multi-purpose Publishing of Geodata on the Web. In: Peterson, M.P. (ed.), *Maps and the Internet*, Elsevier Science, pp. 221-230

OGC, 2001. Web Map Service Implementation Specification, http://www.opengis.org/docs/01-068r2.pdf (accessed 26 Apr. 2004)

OGC, 2002. Web Feature Service Implementation Specification, http://www.opengis.org/docs/02-058.pdf

OGC, 2003. Geography Markup Language (GML) Version 3.0, http://www.opengis.org/docs/02-023r4.pdf (accessed 26 Apr. 2004)

OGC, 2004. OpenGIS Location Services (OpenLS) Core Services, http://portal.opengis.org/files/?artifact_id=3418 (accessed 26 Apr. 2004)

Sarjakoski, T., Sarjakoski, L. T., Lehto, L., Sester, M., Illert, A., Nissen, F., Rystedt, R. and R. Ruotsalainen, 2002. Geospatial Info-mobility Services - A Challenge for National Mapping Agencies. *Proceedings of the Joint International Symposium on "GeoSpatial Theory, Processing and Applications" (ISPRS/Commission IV, SDH2002)*. Ottawa, Canada, July 8-12, 2002, 5 p, CD-rom.

W3C, 1999a. XML Path Language (XPath) Version 1.0, http://www.w3.org/TR/xpath (accessed 26 Apr. 2004)

W3C, 1999b. XSL Transformations (XSLT) Version 1.0, http://www.w3.org/TR/xslt (accessed 26 Apr. 2004)

W3C, 2003. Scalable Vector Graphics (SVG) 1.1 Specification, http://www.w3.org/TR/SVG11/ (accessed 26 Apr. 2004)

W3C, 2004a. Extensible Markup Language (XML) 1.0 (Third Edition), http://www.w3.org/TR/2004/REC-xml-20040204/ (accessed 26 Apr. 2004)

W3C, 2004b. World Wide Web Consortium's Extensible Stylesheet Language Family (XSL), http://www.w3.org/Style/XSL/ (accessed 26 Apr. 2004)

Xalan, 2004. Xalan-Java version 2.6.0, http://xml.apache.org/xalan-j/ (accessed 26 Apr. 2004)