Prototype for Wrapping and Visualizing Geo-Referenced Data in A Distributed Environment Using XML Technology

Jianting Zhang Muhammad Javed Amir Shaheen Le Gruenwald

The University of Oklahoma, School of Computer Science, Norman, OK, 73019

Contact author email: ggruenwald@ou.edu, phone: 1-405-325-3498

ABSTRACT

This paper proposes a prototype for integration and visualization of geo-referenced information (GRI) in a distributed environment in general and World Wide Web in particular. This prototype adopts a three-tier architecture and includes three main components: GRI wrapper for distributed GRI web sites, GRI integration mediator and client side visualization interface.

In this prototype, XML is used as a communication protocol between distributed web sites that provide GRI and the mediator, and between the mediator and clients. Java Servlets are written to translate data in distributed websites into XML documents. Data in distributed websites can be stored in a flat file, relational database, object-oriented database or object-relational database. Java Servlet in the mediator server retrieves data from related distributed websites in an XML format upon a request from the client side, parses the retrieved XML documents, performs merge or other operations on the retrieved XML documents to build a new XML document and sends it to the client side. When the client side gets the requested data from the mediator server, it will parse the returned XML document and draw it inside the browser window by using a Java applet.

Keywords

Geo-Referenced Information, XML, Integration, Visualization

1 INTRODUCTION

Geo-referenced information (GRI) is essentially a kind of spatial information with earth related projections. It is extensively involved in governmental administration, public affairs and private everyday lives. Although geo-referenced information can be expressed as numeric values like other textual information but it makes little sense to the end user in the numeric form. Instead, a map is the most commonly used form to display geo-referenced information. Unfortunately, GRI is among the unconventional data types which are not specified either in HTML or XML. To enable web-based GRI browsing, generally two strategies are adopted, namely image-based and vector-based ([9]).

Most of current web based mappings adopt the two-tier architecture since it is simple and more efficient if no integration issue is considered. However, in the near future, our vision is that, due to wide applications of GPS, personal-based instead of governmental based geo-referenced information will be overwhelming. Geo-referenced information will not be provided only by few data providers any more. It will be the case that some data vendors provide basic geo-referenced information while end users put thematic geo-referenced information together with text information associated with it for domain specific applications. Instead of buying such data for each application and physically combining it with user's own thematic data, it is better to provide a mechanism to let user logically refer to the GRI data in public websites as the base map and allow him to simply integrate it with his own thematic data. This is essentially a problem of geo-referenced information integration in a web environment.

Compared to intra-enterprise information integration that usually has well-developed protocols, information integration in a web environment is more difficult. This is mainly because each website involved is heterogeneous and autonomous. The visualization nature of HTML, which lacks the capability of conveying semantic information, is another important factor. The newly emerged XML technology is a promising technology for information integration in web-environment.

In this paper we present a prototype for wrapping, integrating and visualizing geo-referenced information in a web environment. This prototype has the three major features:

1. Using a three-tier XML based architecture that allows flexible web databases integration.
2. Integrating geo-referenced information from distributed and heterogeneous web sites that provide geo-referenced information.
3. Visualizing integrated geo-referenced information for better understanding and decision-making.

The organization of the rest of the paper is as follows. Section 2 discusses related work. Section 3 provides an overview of our prototype. The details of the prototype implementation are given in Section 4. Finally the experiment results and future work are described in Section 5.

2 RELATED WORK

The idea of wrapping heterogeneous databases to provide a common interface for information integration is also well discussed ([15]). XML has been recognized as a good method for web-based information integration ([6]). More recently, XML based information mediation has been extensively studied in the MIX project ([2]). These studies suggested that XML has the potential capability to serve as a common protocol for information integration in a web environment.

As we had mentioned before, XML has no specifications on visualization issues. It is often the case that XML is combined with HTML in web-based applications, i.e., XML is responsible for information exchange and HTML is responsible for display. In this case XML is finally mapped to HTML usually through XSL (XML Style Sheet Language) for display. However, there are many XML elements that cannot be mapped to HTML elements, and thus, cannot be visualized directly. [4] uses Java applets to display Z specifications written in XML.

As far as geo-referenced information is considered, there have been continuous efforts to integrate multiple sources and multiple formats of geo-referenced information. Generally, there are two directions towards this goal. The ideal one is to follow the same standard. In this field the most significant contribution is Open GIS Consortium (OGC). OGC has published its Abstract Specifications Version 4 and implementation specifications for OLE/COM, CORBA and SQL ([HREF 1]). The practical direction is to develop a set of APIs for each data format. Geo-referenced information can be integrated by calling APIs locally or in a distributed environment. The experiment on developing an interoperable test-bed on Open GIS in Japan follows the first direction ([13]). In this system, the authors developed a three-tier model that consists of clients, legacy database wrappers and Geo Spatial Mediator (GSM), and also proposed the CORBA implementation. Alternatively, [11] followed the second direction and proposed a practical solution to integrate geo-reference information which is now commercially available ([HREF 2]).

The XML technology also has a great influence on web-based geo-referenced information applications. Oracle has announced its Oracle8*i* GeoXML Server and has put it in the OGC website for testing ([HREF 3]). It transfers spatial information from ORACLE database to XML but no visualization function has been provided. OGC has recently announced its Geography Markup Language (GML) Recommendations ([HREF 4]). However, no implementation of GML has been reported yet.

With more and more web-based semi-structured information expressed in the SGML or XML format, the demand for integrating XML and geo-referenced information has also emerged. A related study ([1]) shows that although it is possible to integrate SGML with commercial GIS software to provide browser-based visualization and query, the cost is not negligible. It would be very interesting if all the information, both text-based and geo-referenced information, is expressed in the XML format, then the task of integration might be easier. As far as our literature review is concerned, no study on integrating and visualizing geo-referenced information using the XML technology has been reported.

Metadata plays important roles in geo-referenced information applications. Both OGC and national (such as Federal Geographical Data Committee -FGDC) and international organizations (such as ISO /TC211) have contributed a lot to the metadata research ([7]). Unlike current proposals that store metadata and data itself separately, tag based XML has the capability of integrating metadata and data in one document. At the same time, Data Type Definition (DTD) for XML enables carrying semantics for the semi structured GRI and validating its XML document in real time.

3 PROTOTYPE OVERVIEW

In this prototype research, we use XML as the communication protocol between distributed web sites and the mediator, and between the mediator and clients. Java servlet programs are written to translate data in distributed websites into an XML document. Data in distributed websites can be stored in a flat file, relational database, object-oriented database or object-relational database. A Java servlet program in the mediator server will retrieve the data from related distributed websites in the XML format upon a request from the client side, parse the retrieved XML documents, perform merge or other operations on the retrieved XML document to build a new XML document and send it back to the client side. When the client side gets the requested data from the mediator, it will parse the returned XML document and draw it inside the browser using a Java applet. The architecture is shown in Figure 1.

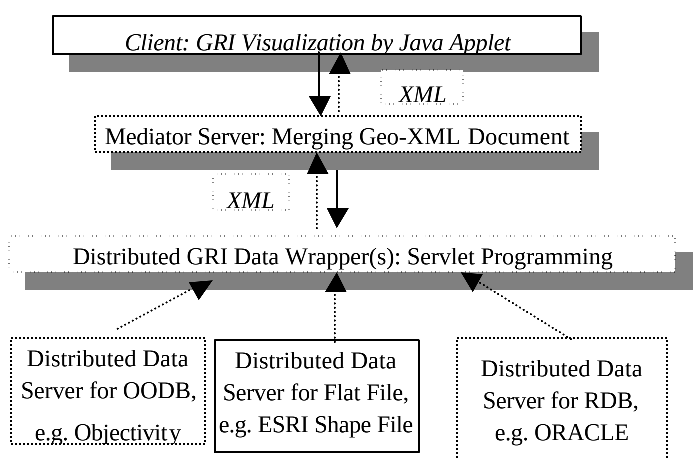

Figure 1. Prototype System Architecture

Although there have been studies in web-based database wrapping, XML based information integration, applet-based XML elements visualization, web-based geo-referenced information mapping, geo-referenced information integration with three-tier architecture, our proposal of integrating all these features to build a prototype for wrapping, integrating and visualizing geo-referenced information in a web environment is novel. Table 1 shows the comparison of our prototype with existing studies.

The reasons to use XML as the common protocol for geo-referenced information exchange are as follows:

1. Text-based information exchange protocol ensures platform independence and easy implementation.
2. Data provider can be any system with any data format as long as it can generate XML documents.
3. The generated XML document is application independent, which can be used in any applications as long as the application can parse XML documents.
4. It is more efficient in storing and transferring data in the XML format for GRI because usually there are many attributes in a whole data set but few attributes are associated with each object, i.e., the associated relational table is sparse

Table 1. Features Comparison of Related Existing Studies and the Proposed Prototype

Ref	GRI	XML	Wrapping	Integrating	Visualization
(Amann, 1999)	Yes	Yes (SGML)		Yes	Yes
(Baru, 1999)		Yes	Yes	Yes	
(Ciancarini, 1999)		Yes			Yes
(Shimada, 1999)	Yes		Yes	Yes	
(Sorokine, 1998)	Yes				Yes
(Thiran, 1999)			Yes		
This Study	Yes	Yes	Yes	Yes	Yes

4 SYSTEM IMPLEMENTATION

As we have explained in Section 3, there are three main components in the prototype system: geo-referenced information wrapper, mediator and client side visualization interface. Before discussing each component, we will introduce some technical issues on XML generation and parsing, which are the basis for XML-based web information integration.

4.1 XML: Semi-structured Web Database, Its Generation and Parsing

Basically there are two major issues in using XML, one is how to generate an XML document and the other is how to parse an XML document. Up to now, major software companies have announced their supports for XML ([HREF 5,6,7,8]). In this prototype, we are using SUN's XML API to generate and parse XML document. There are two ways to manipulate an XML document. One is based on SAX (Simple API for XML) and the other is based on DOM (Document Object Model). Both of them are W3C recommendations ([HREF 9]).

Based on our experiences, the tree-based DOM model is more suitable for geo-referenced information integration using the XML technology. The reason is that a geo-referenced object often has multiple levels of attributes that can hardly be arranged in one XML element. In the SAX based method, each element in an XML document is processed sequentially in a first-in-first-out manner, and thus, can hardly maintain the hierarchy of geo-referenced objects. On the contrary, the DOM model allows the user to travel through the whole document tree. The tree hierarchy is a good representation of multiple levels of attributes of a geo-referenced object.

4.2 Wrapper: Generating Geo-XML from Flat Files

Point, polyline and polygon are three major data types in geo-referenced information. We adopted a DTD definition that is a subset of Oracle GeoXML ([HREF 3]). The format of the flat files that we used is the ShapeFile of GIS industry leading company ESRI. To read data from a file into memory, several Java classes, developed by the computational geography center at the University of Leeds, UK ([HREF 10]), are used. Our work is to map those classes into XML documents. The process is shown as Figure 2.

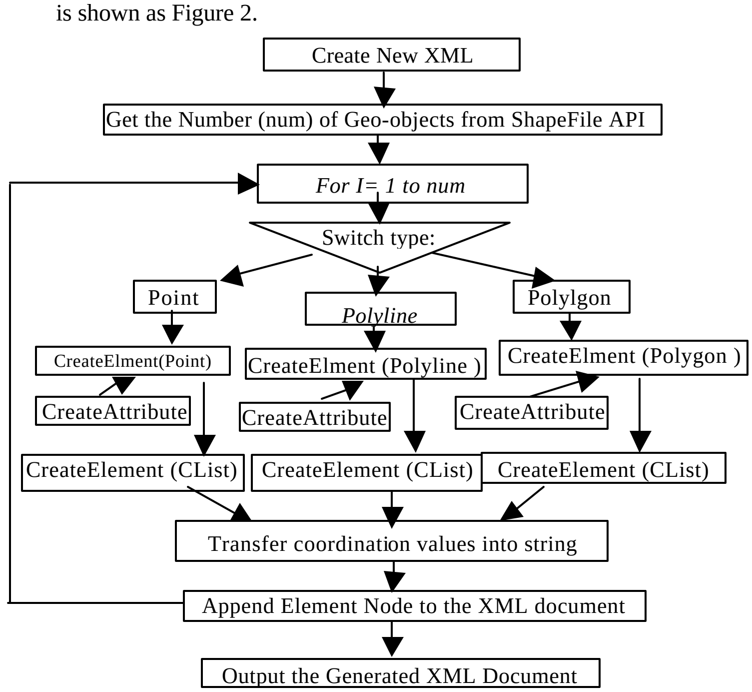

Figure 2 Process of generating an XML document from a ShapeFile by the GeoXML Wrapper

We put the wrapper procedure in a Java servlet called GeoXmlServlet. A wrapper can provide one XML document or multiple XML documents by taking parameters. It serves two purposes: first, the user can get the data in the XML format by visiting the specified URL directly, and second, the mediator can get the needed data by sending the standard URL request to the distributed Geo-XML wrappers.

The syntax to retrieve a GeoXML document from our website in this prototype is as follows:

http://129.15.192.159:8081/servapp/servlet/GeoXmlServlet?layers=num

Where the parameter "*num*" after the servlet name is the layer number.

We have experimented with two sets of data, one containing cities, rivers and countries of the world and the other containing places names, rivers and census tracks in Cleveland County, Oklahoma, USA. Both of the sets have point (cities/place names), polyline (rivers) and polygon (countries/census tracks) data types. Both of the experiments were successful. In our two experiments, the layer numbers and their contents are listed below:

Table 2. Layers Used in the Experiments

Layer Number	Experiment 1 (World)	Experiment 2 (Cleveland County, OK, USA)
0	Cities (point data)	Places Names (point data)
1	Rivers (polyline data)	Rivers (polyline data)
2	Countries (polygonal data)	Census Tracks (polygonal data)

4.3 Mediator: Geo-Referenced Information Integrator

The mediator is also built into a Java servlet, which enables the end user to get the integrated information in the XML format by visiting the mediator URL directly to support multiple purposed applications.

The mediator is designed to visit the distributed data servers automatically and gathers the metadata. These data are stored in the mediator's Meta database. It accepts the user's high-level query and finds relevant web sites from its Meta database. Then the mediator divides the query into sub-queries and sends them to related web sites. Finally, the mediator gathers the returned XML documents from the distributed web sites to build a new XML document with integrated information and sends it to the client side for visualization. However this aim has not been implemented fully. In the first version of our prototype, currently our mediator supports merging of XML documents to build a new XML document. The merging strategy is relatively simple; it extracts elements of each XML document and puts them into a new XML document. Also, the metadata stored in the mediator's Meta database is manually set. The process of GeoXML document integration is shown in Figure 3.

In our prototype, the syntax for retrieving an integrated XML document from the mediator (wrapper server) is as follows:

Retrieving meta data (the word "meta" after "layers" is the reserved key word and is treated as a special layer):

http://129.15.192.159:8081/servapp/servlet/GeoXmlMserv?layers=meta

Currently, a brief XML document description, URL, number of geo-objects and layer sequence number are stored in the mediator's Meta database.

Retrieving integrated XML data:

http://129.15.192.159:8081/servapp/servlet/GeoXmlMserv?layers=110

Where the parameter "layers" after the servlet name is the combination of the layers requested. We use 1 to indicate the layer is requested while 0 is not. Note that the visualization applet on the client side will also use the same syntax to retrieve the information from the mediator, which we will describe in detail below.

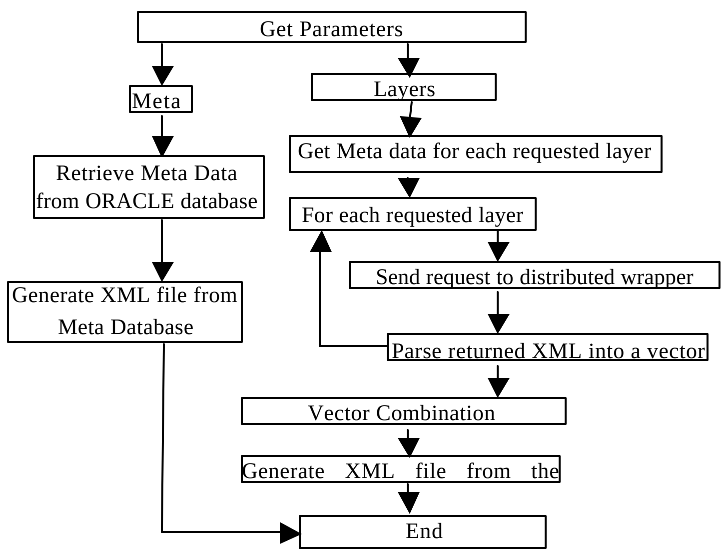

Figure 3. Process of integrating GeoXML Documents by the GoeXML Mediator

4.4 Client Side Visualization

The client side responses to the user's choices and communicates with the mediator to retrieve integrated information and then visualize it. A client can either be any standalone program that can parse XML documents or can be any object embedded into a web browser under the system architecture. In our prototype, we use a Java applet to embed the visualization functions into the web browser. The process is shown as Figure 4.

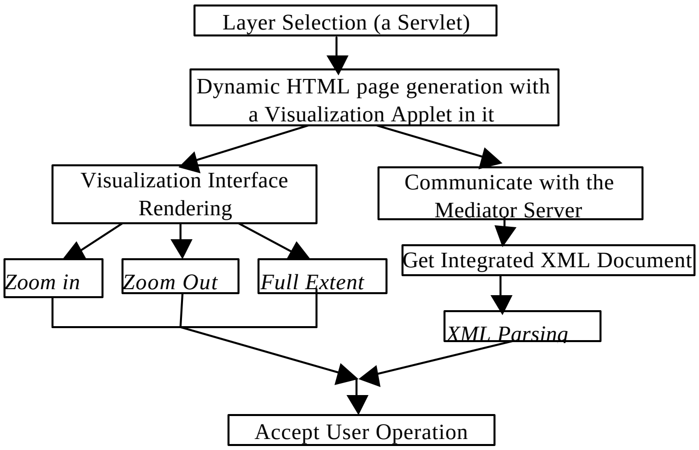

Figure 4. Process of Client Visualization based on GeoXML

First, a form based user-interface using servlet is developed to get user choices. Another servlet is built to dynamically generate an HTML page containing the visualization applet together with the parameters of the user's selection passed by the first servlet. Then the client side visualization applet will contact the mediator and retrieve data from it according to the parameters. After the data is retrieved, the applet will pass the XML document into java classes and render them in a graphic form for visualization.

Suppose there are three data sets registered in the mediator, the user can select any combinations of them. The syntax is:

http://129.15.192.159:8081/servapp/servlet/GotoXML

By interpreting the submission of the user's selection, the servlet will guide the user to the following URL for visualization:

http://129.15.192.159:8081/servapp/servlet/GenApplet

Currently, the frequently used geo-referenced information visualization functions such as "Full Extent", "Zoom In" and "Zoom Out" are implemented in our visualization applet. Compared to existing commercial products, the functions of our prototype are very limited. However, the architecture is developed to allow additional functions to be implemented easily, such as panning a map, identifying the attributes associated with the geo-referenced objects and hyper-linking a location to some other URLs. In fact, the architecture allows the implementation of a full-scale web-based geographical information system to support complex queries and spatial analysis, which is of our long-term goal.

5 RESULTS AND FUTURE WORK

In this prototype system, we have implemented an XML wrapper for geo-referenced information from a distributed node in a flat file format, developed a Geo-XML mediator which merges XML documents from distributed websites and provides integrated information in the XML format, and finally developed a visualization Java applet to retrieve geo-referenced information from the mediator and visualize it.

Due to space limitation, we only show a part of our experiment results in this section. Figure 5 shows integrated Geo-XML documents in the text format from two different GRI wrappers displayed hierarchically in Internet Explore 5.0. Figure 6 shows the client side GRI visualization interface.

At the same time, we are fully aware of the limitations of the prototype system. Further directions include better visualization functions at the client side, full implementation of spatial query and analysis at the client and/or mediator side, building GRI search engine inside mediator to automatically retrieve metadata and implementation of GRI wrappers for relational database and object-oriented database.

6 REFERENCE

[1] B.Amann, D.Vodislav, J.Fernandes, G.Coste, "Browsing SGML Documents with MAPS: The French "Inventaire" Experience", LNCS 1460,1999, 250-259

[2] C. Baru, A. Gupta, B. Ludaescher, R. Marciano, Y. Papakonstantinou, P. Velikhov, "XML-Based Information Mediation with MIX", In Exhibitions Program of ACM SIGMOD 99

[3] S.S.Bhowmick, W-K.Ng, E-P.Lim, S.K.Madria, "Join Processing in web Databases", LNCS 1695, 1999, 647-657

[4] P.Ciancarini, F.Vitali, C.Mascolo, " Managing Complex Documents Over WWW: A case study for XML", IEEE trans. Knowledge and Data Engineering, 11(4), 1999, 629-638

[5] T.Connoly, "Chapter 24: web Technology and DBMS", Database System, 2nd edition, Addison-Wesley, 1999

[6] D.Florescu, A.Levy , A.Mendelzon, "Database Techniques for the world wide web: A survey", ACM SIGMOD Record, 27(3), 1998,

[7] S.Gobel, K.Lutze, " Development of meta database for geospatial data in the WWW", ACM GIS'98, 1998, 94-99

[8] R. Goldman, J. Widom, "Interactive Query and Search in Semistructured Databases", WebDB'98, 1998

[9] A.Hardie, 1998, "The Development and Present State of web-GIS, Cartography", 27(2), 1998, 11-26

[10] D. Konopnicki , O.Shmueli, "Information Gathering in the world wide web: The W3QL query language and the W3QS System", ACM trans. Database Systems, 23(4), 1998, 369-410

[11] C.Larouche, G.Clement, " A practical solution to the geospatial data barrier problem", ACM GIS'97, 1997, 62-66

[12] S.Laurent, E. Cerami, "Building XML applications", McGraw-Hill, 1999

[13] S.Shimada, H.Fukui, "Geospatial mediator functions and container-based fast transfer interface in Si3CO Test-Bed", LNCS 1580, Springer, 1999, 265-276

[14] A.Sorokine, I.Merzliakova, "Interactive map applet for illustrative purpose", ACM GIS'98, 1998, 46-51

[15] Ph. Thiran, A.Chougrani, J-M Hick, J-L Hainaut, "Generation of Conceptual Wrappers for Legacy Database", LNCS 1677, Springer, 1999, 678-687

[HREF 1] OpenGIS Specifications, http://www.opengis.org/techno/spe7cs.htm [Apr 20, 2000]

[HREF 2] GlobalGeo Products, http://www.globalgeo.com/products/ Apr 20, 2000]

[HREF 3] Oracle8*i* GeoXML Server, http://feature.opengis.org:8080/oracle/ [Apr 20, 2000]

[HREF 4] OpenGIS Geography Markup Language, http://www.opengis.org/techno/specs [May 20, 2000]

[HREF 5] ORACLE XML: http://technet.oracle.com/tech/xml/ [Apr 20, 2000]

[HREF 6] MICROSOFT XML: http://msdn.microsoft.com/xml [Apr 20, 2000]

[HREF 7] IBM XML: http://www.ibm.com/developer/xml/ [Apr 20, 2000]

[HREF 8] SUN XML: http://java.sun.com/xml [Apr 20, 2000]

[HREF 9] W3C XML Specification and recommendation http://www.w3.org/XML/ [Apr 20, 2000]

[HREF 10] ShapeFile Reader http://www.geog.leeds.ac.uk/pgrads/j.macgill/java/JavaShapefile.html, [Apr 20, 2000]

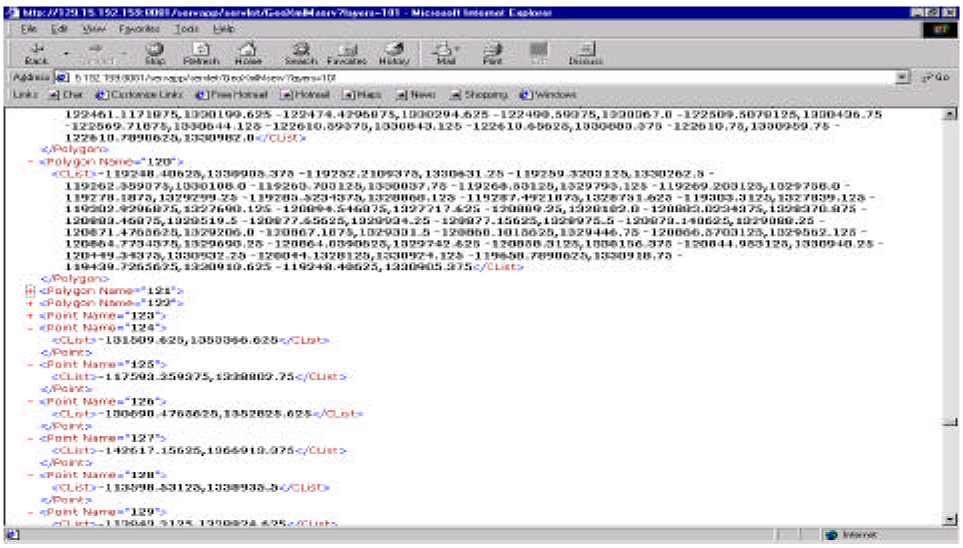

Figure 5. Integrated GeoXML document retrieved from distributed GRI wrappers generated by mediator

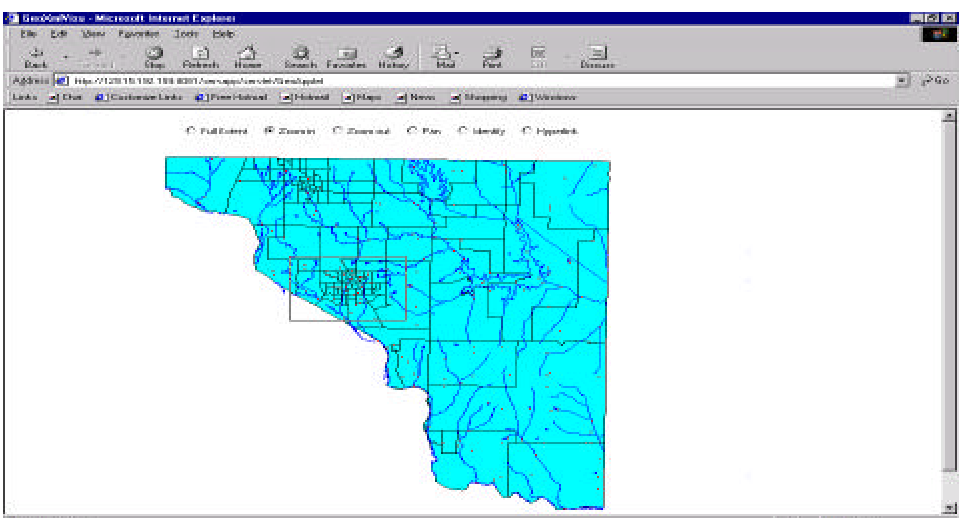

Figure 6. Client side GRI visualization based on GeoXML document retrieved from the mediator