Status Report on my Research Work Done So Far

59 Questions and Answers
Grouped based on Research Topics

By Ahmet Sayar

Research Committee:

- *Prof. Geoffrey C. Fox (Principal Advisor)*
- *Prof. Randall Bramley*
- *Prof. Kay Connelly*
- *Prof. Melanie Wu*

Indiana University
Computer Science - Community Grids Lab (CGL)
April, 2007

Contents

Introduction **2**

Summary of Research Topics: *2*

1. Background **3**

Key Terms and Definitions **3**

2. OGC Compatible Web Map Services with Web Service Principles **7**

3. Smart Map Tools **16**

3.1. Integrating AJAX Approach to SMT (Q36 to Q45): *18*

4. Architectural Framework for Streaming Map Movies **22**

5. **Sci-Plotting Services *(mostly completed)*** **26**

Introduction

In the following chapters, concepts introduced here under the "summary of research topic" are explained in the **QUESTIONS AND ANSWERS** manner. By this way, we think research issues and challenges are highlighted and, concepts are made more concrete and understandable. A lot of questions listed here came from real application users by e-mail or as comments on my blogs. You can also see another version of this document uploaded to my blog in topic base. Please see them here http://ahmetsayar.blogspot.com .

Summary of Research Topics:

i. Background
Key Term and Definitions

ii. Building OGC Compatible Web Map Services with Web Service Principles
- Integration with WFSs and other WMSs
- Implementing as Web Services
- Capability-Based WMS Federations
- Performance issues with archived data access and rendering

iii. Smart Map Tools (SMT) Tools
- WMS Client
- Interactive data querying
- Streaming map movies capability for the time-dependent feature data
- Integrating AJAX Approach
    - Layer based integration of Google maps and OGC's WMS maps
    - Overlaying features from WFS over Google Maps

iv. Building Architectural Framework for Creating Streaming Map Movies:
- Publishing streams through JMF libraries.
- Enhanced SMT with movie capabilities
- Easy to display movie streams
    - from Collaboration sessions
    - from standalone JMF client program

v. SCI-Plotting Services (%60 completed)
- Implementing in Web Service principles
- Data Conversion (from … to VOTable)
- Provides layers compatible to be overlaid on WMS maps a and/or Google Maps

1. Background

Key Terms and Definitions

1. What is GIS?

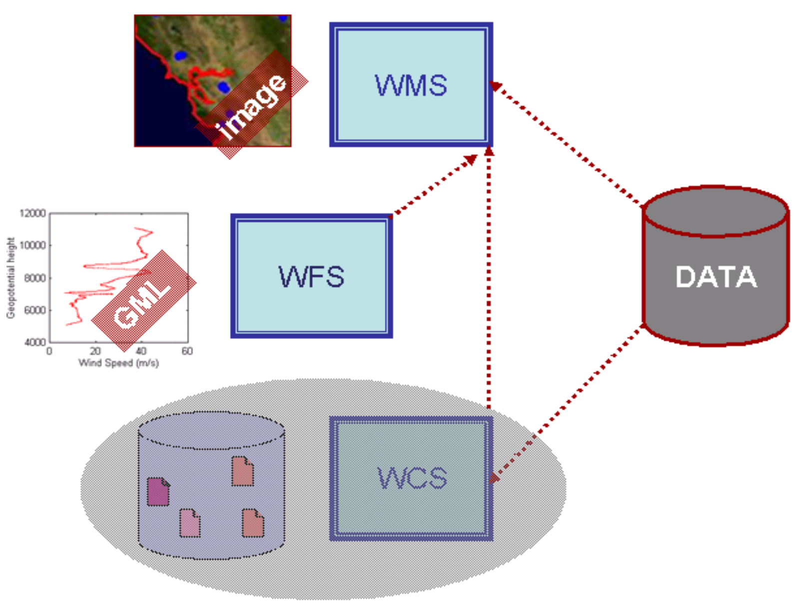
Geographical Information Systems (GIS) introduce methods and environments to visualize, manipulate, and analyze geospatial data. The nature of the geographical applications requires seamless integration and sharing of spatial data from a variety of providers. Interoperability of the services across organizations and providers is the main goal for GIS.

2. What is OGC?

The Open Geospatial Consortium (OGC) defines a number of standards (both for data models and for online services) that have been widely adopted in the Geographical Information System (GIS) community. OGC is a non-profit, international standards organization that is leading the development of standards for geographic data related operations and services. OGC has variety of contributors from different areas such as private industry and academia to create open and extensible software application programming interfaces for GIS.
The most important OGC specifications are WMS, WFS, WCS and GML.

3. What are the main OGC services in order to create the simplest GIS?

In simplest case, you should have a database (MySQL, Oracle or any other), Web Feature Server providing data from this database and Web Map Server to create map images by rendering data coming from WFS or/and any other WMS. See the below figure for the simplest representation of an OGC compatible GIS systems.



4. Is there any relation between ISO (International Standards Organization) and OGC?

OGC has a close relationship with ISO/TC 211 (Geographic Information/Geomatics). The OGC abstract specification is being progressively replaced by volumes from the ISO 19100 series under development by this committee. Further, the OGC standards Web Map Service and Simple Features are ISO standards. GML will soon be approved as an ISO standard.

5. What is WMS?

Web Map Service produces maps from geographic data. A map is not the data itself. Maps create information from raw geographic data. Maps are generally rendered in pictorial formats such as JPEG, GIF, PNG. WMS also produce maps from vector-based graphical elements in Scalable Vector Graphics (SVG).

WMS provides three operations protocols (GetCapabilities, GetMap, and GetFeatureInfo) in support of the creation and display of registered and superimposed map like views of information that come simultaneously from multiple sources that are both remote and heterogeneous.

Basically there are two types of WMS defined in the specifications. These are basic WMS and SLD-enabled WMS. For the basic WMS, there are three operations defined. These are getCapabilities, getMap, GetFeatureInfo.

WMS publishes its ability and data holdings in its capabilities document. This document is encoded in XML. WMS classifies its geographic data holdings in the “Layers” and gives information about the styles available for these Layers. Each layer can have sub layers and the sub layers can have different styling defined for them.

A Web Map Service is usually not invoked directly. More often, it is invoked by a client application that provides the user with interactive controls. This client application may or may not be web-based

6. What is WFS?

Web Feature Service (WFS) is an intermediary server sitting in front of the archived data kept in the Databases or files systems and, provides data manipulation operations.

Clients interact with a Web Feature Service by submitting database queries encoded in Open Geospatial Consortium Filter Encoding Implementation and in compliance with the Open Geospatial Consortium’s Common Query Language.

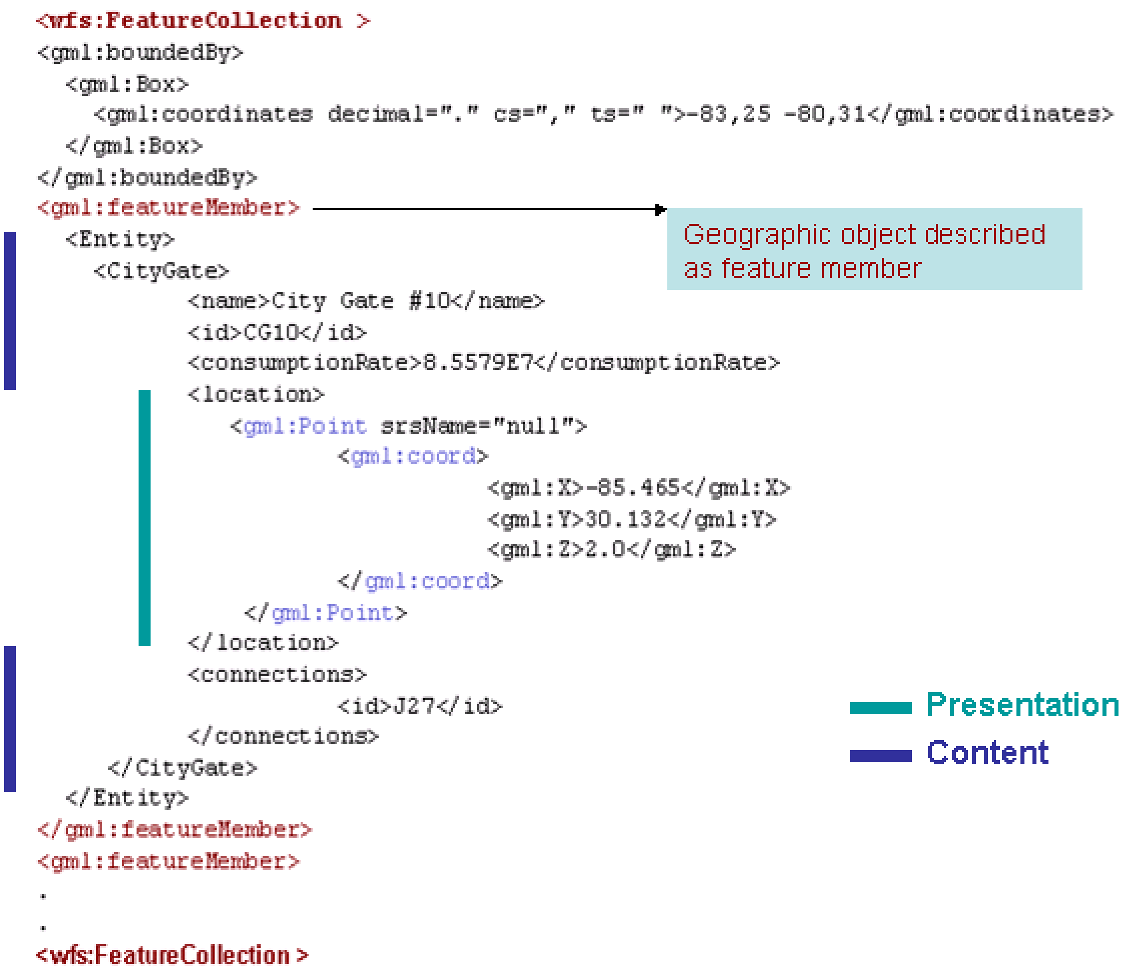
As a minimal requirement a basic Web Feature Service should be able to provide requested geographical information as Geographic Markup Language feature collections. However, more advanced versions also support “create, update, delete and lock operations”. Three operations must be supported by a basic Web Feature Service: GetCapabilities, DescribeFeatureType and GetFeature.

7. What is GML?

GML is an XML encoding for the transport and storage of geographic information, including both the spatial (attributes) and non-spatial (geometric) properties of geographic features. GML defines a data encoding in XML that allows geographic data and its attributes to be moved between disparate systems with ease XML encoding of geospatial information. It is human-readable and separates contents from presentation. Presentation is basically represented under the “geometry” tag elements.

If some one wants to define a GML schema for a feature, he should import the Geometry schema (geometry.xsd) the Feature Schema (feature.xsd) and the XLinks schema (xlinks.xsd) as base schemas.

WMS requests feature data from the WFS in the form of GML. WFS store and serve the geospatial data encoded in GML.



8. What is feature?

A feature is an abstraction of a real world phenomenon; it is a geographic feature if it is associated with location relatives to the earth such as faults, rivers, roads, lakes. All the feature instances should be created according to the schema files and encoded in GML.

9. What are the vector and raster data?

Vector data deals with discrete phenomena, each of which is conceived of as a feature. The spatial characteristics of a discrete real world phenomenon are represented by a set of one or more geometric primitives (points, curves, surfaces, or solids). Other characteristics of the phenomenon are recorded as feature attributes. Usually, a single feature is associated with a single set of attribute values.

Raster data deals with real world phenomena that vary continuously over space. It contains a set of values, each associated with one of the elements in a regular array of points or cells. It is usually associated with a method for interpolating values at spatial positions between the points or within the cells.

10. What is coverage data?

OGC uses the term "coverage" to refer to any data representation that assigns values directly to spatial position. Coverage is a feature that associates positions within a bounded space (its spatiotemporal domain) to feature attribute values (its range). Examples include a raster image, a polygon overlay, or a digital elevation matrix. The spatio-temporal domain of coverage is a set of geometric objects described in terms of direct positions. Commonly used spatio-temporal domains include point sets, grids, collections of closed rectangles, and other collections of geometric objects.

11. What is spatial and geo-spatial data

Spatial data are a kind of data that pertains to the space occupied by objects. Example spatial data from the real world are cities, rivers, roads, states, crop coverage, mountain ranges etc. In the implementation these are represented by points, lines, rectangles, surfaces, volumes and etc. Spatial data have some common characteristics. These type of data are geometric data and in high dimensions. These data can be either discrete (vector) or continuous (raster). GIS applications are applied on these types of data.

Geospatial data are spatial data associated with a location relative to the Earth.

12. What is the term "capability" in OGC used for?

Each GIS service defined in OGC has a capability file encoded in XML. There is no common format for the capability valid across all the OGC defined services. All the OGC services has getCapability service interface in order to return capabilities metadata. Capabilities function as "service" metadata, providing information about what the service offers. Clients to the services determine whether they can work with that server based on its capabilities.

```xml
<?xml version='1.0' encoding="UTF-8" standalone="no" ?>
<!DOCTYPE WMT_MS_Capabilities SYSTEM "http://toro.ucs.indiana.edu:8086/xml/capa.dtd">
<Capabilities version="1.1.1" updateSequence="0">
   <Service>
      <Name>CGL_Mapping</Name>
      <Title>CGL_Mapping WMS</Title>
      <OnlineResource xmlns:xlink="http://www.w3.org/1999/xlink" xlink:type="simple“
                              xlink:href="http://toro.ucs.indiana.edu:8086/WMSServices.wsdl" />
      <ContactInformation>
         .....
       </ContactInformation>
  </Service>
  <Capability>
      <Request>
        <GetCapabilities>
           <Format>WMS_XML</Format>
             <DCPType><HTTP><Get>
              <OnlineResource xmlns:xlink="http://w3.org/1999/xlink" xlink:type="simple“
                                 xlink:href="http://toro.ucs.indiana.edu:8086/WMSServices.wsdl" />
             </Get></HTTP></DCPType>
         </GetCapabilities>
         <GetMap>
            <Format>image/GIF</Format>
            <Format>image/PNG</Format>
             <DCPType><HTTP><Get>
              <OnlineResource xmlns:xlink="http://w3.org/1999/xlink" xlink:type="simple“
                                 xlink:href="http://toro.ucs.indiana.edu:8086/WMSServices.wsdl" />
             </Get></HTTP></DCPType>
        </GetMap>
      </Request>
      <Layer>
         <Name>California:Faults</Name>
         <Title>California:Faults</Title>
         <SRS>EPSG:4326</SRS>
         <LatLonBoundingBox minx="-180" miny="-82" maxx="180" maxy="82"  / >
       </Layer>
   </Capability>
</Capabilities>
```

2. OGC Compatible Web Map Services with Web Service Principles

13. Why did you implemented GIS services in Web Services principles? What is the gain of extending OGC GIS as Web Services?

Web Services give us a means of interoperability between different software applications, running on a variety of platforms. A Web Services support interoperable machine-to-machine interaction over a network.

Distribution: It will be easier to distribute geospatial data and applications across platforms, operating systems, computer languages, etc.

Integration: It will be easier for application developers to integrate geospatial functionality and data into their custom applications.

Infrastructure: The GIS industry could take advantage of the huge amount of infrastructure that is being built to enable the Web Services architecture – including development tools, application servers, messaging protocols, security infrastructure, workflow definitions, etc.

Easy to extend: Web Services have lots of specifications. By using any configuration of these specifications we will make our implementation more secure, more robust and fault tolerant as Web Service specs improve themselves.

14. OGC introduces capability concept to describe services. What are the differences between WSDL and capability?

WSDL is **W**eb **S**ervice **D**escription **L**anguage. Web Service is software that makes services available on a network using technologies such as HTTP, XML and SOAP. Web Service enables loosely coupling of services and interoperability.

WSDL describes Web Services' service functionality in operations and messages.

WSDL enables access to functions through describing data encodings, location and communication protocols

WSDL enables client to invoke the service's operations through generating of stubs and skeletons

SO, Web Services are service based, in contrast capability is service+data related. Below figure shows the WSDL structure roughly with fundamental tag elements.

```
<description xmlns=http://www.w3.org/2004/08/wsdl targetnamespace=".. ">
        <types>
                <!– XML Schema description of types being used in messages-->
                .....
        </types>
        <interface name="...">
                <!-- list of operations and their input and output -->
                ......
        </interface>
        <binding name="..." interface="..." type="...">
                <!– message encodings and communication protocols -->
                .....
        </binding>
        <service name="..." interface="...">
                <!--combination of an interface, a binding, and a service location-->
                .......
        </service>
</description>
```

See the capability structure represented with its major tag elements in Section 1 Q12.

15. OGC introduces capability concept to describe services. What is the difference between metadata and capability?

Metadata is data describing data. Data might be paper, article, image, sound, spatial, temporal etc. To give an example, suppose data is article, so metadata might be consists of “author”, “title”, “subject”, “source type”, “format” and possibly more. Metadata enables describing and sharing information between different data sources and applications.

SO, metadata is data based, in contrast capability is service+data related.

Here is a sample metadata describing a resource created by merging Dublin Core in RDF.

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF PUBLIC "-//DUBLIN CORE//DCMES DTD
   2002/07/31//EN"
    "http://dublincore.org/documents/2002/07/31/dcmes-
   xml/dcmes-xml-dtd.dtd">
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-
   ns#"
         xmlns:dc="http://purl.org/dc/elements/1.1/">
  <rdf:Description
   rdf:about="http://www.nap.edu/books/0309064996/html/">
    <dc:title>The Digital Dilemma</dc:title>
    <dc:creator>National Research Council</dc:creator>
    <dc:date>2000-06-22</dc:date>
  </rdf:Description>
</rdf:RDF>
```

16. How is the data stored, will it be in GML, what database is mostly used.

The Web Feature Service starts processing user requests by extracting the database query from the request (encoded as a GML Filter – see below figure) and uses this to construct a corresponding MySQL query. The results of the database querying process are used to create a GML FeatureCollection which usually contains multiple features.

Sample query “getFeature” encoded in XML (WMS sends to WFS to get feature data):

```xml
<?xml version="1.0" encoding="iso-8859-1"?>
<wfs:GetFeature outputFormat="GML2"
xmlns:gml="http://www.opengis.net/gml"
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:ogc="http://www.opengis.net/ogc">
  <wfs:Query typeName="boundary_lines">
 <wfs:PropertyName>FNODE</wfs:PropertyName>
 <wfs:PropertyName>TNODE</wfs:PropertyName>
 <wfs:PropertyName>WORLD</wfs:PropertyName>
 <wfs:PropertyName>coordinates</wfs:PropertyName>
 <wfs:PropertyName>minx</wfs:PropertyName>
 <wfs:PropertyName>miny</wfs:PropertyName>
 <wfs:PropertyName>maxx</wfs:PropertyName>
 <wfs:PropertyName>maxy</wfs:PropertyName>
   <ogc:Filter>
     <ogc:BBOX>
       <ogc:PropertyName>coordinates</ogc:PropertyName>
       <gml:Box>
         <gml:coordinates>-155,28 -120,50</gml:coordinates>
       </gml:Box>
     </ogc:BBOX>
   </ogc:Filter>
  </wfs:Query>
</wfs:GetFeature>
```

17. How does WMS get the data from WFS and in what format?

WMS uses basic WFS services. These are getCapabilities, describeFeatureType, and getFeature. Invocation is done through client stubs created at the WMS side according to WFS's WSDL file before run-time. Before invocation WMS also sends a getCapabilities requests to WFS to learn which feature types WFS provides and what operations are supported on each feature type.

When WMS receives any getMap or getFeatureInfo request, it creates a getFeature (see the figure in previous question -18-) request and sends it to WFS. As a response WFS sends back the set of result features meeting the search criteria to WMS as a GML encoded in XML. In case of using streaming WMS, the result feature set is returned in streaming through NaradaBrokering messaging middleware in the form of byte arrays. In case of using pure Web Service version the result set is returned as XML document in the built-in format <xsd:string>. We prefer to use streaming data approach by using NaradaBrokering.

18. Can you give a sample "getCapabilities" request for Web Map Server?

WMS receives getCapabilities request in below format and returns its capabilities listed in Q12.

```xml
<GetCapabilities xmlns="http://www.opengis.net/ows">
    <version>1.1.1</version>
    <service>wms</service>
    <exceptions>application_vnd_ogc_se_xml</exceptions>
    <style>full</style>
</ GetCapabilities>
```

19. Can you give a sample "getMap" request for Web Map Servers?

WMS receives getMap request and returns a map in a form defined in the tag "format" which is under the tag "image".

```xml
<GetMap xmlns="http://www.opengis.net/ows">
        <version>1.1.1</version>
        <service>wms</service>
        <exceptions>application_vnd_ogc_se_xml</exceptions>
        <Map>
            <BoundingBox decimal="." cs="," ts=" ">
                -124.85,32.26,-113.56,42.75
            </BoundingBox>
            <Elevation>5.0</Elevation>
            <Time>01-01-1987/12-31-1992/P1Y</Time>
        </Map>
        <Image>
            <Height>400</Height>
            <Width>400</Width>
            <Format>video/mpeg</Format>
            <Transparent>true</Transparent>
            <BGColor>0xFFFFFF</BGColor>
        </Image>
        <ns1:StyledLayerDescriptor version="1.0.20" xmlns:
                                        ns1="http://www.opengis.net/sld">
            <ns1:NamedLayer>
                <ns1:Name>Nasa:Satellite</ns1:Name>
                <ns1:Description>
                    <ns1:Title>Nasa:Satellite</ns1:Title>
                    <ns1:Abstract>Nasa:Satellite</ns1:Abstract>
                </ns1:Description>
            </ns1:NamedLayer>
            <ns1:NamedLayer>
                <ns1:Name>California:States</ns1:Name>
                <ns1:Description>
                    <ns1:Title>California:States</ns1:Title>
                    <ns1:Abstract>California:States</ns1:Abstract>
                </ns1:Description>
            </ns1:NamedLayer>
        </ns1:StyledLayerDescriptor>
</GetMap>
```

20. Can you give a sample “getFeatureInfo” request for Web Map Servers?

WMS receives getFeatureInfo request and returns content of the feature data falling in close neighborhood (small bbox) of the clicked point on the map. Return type is mostly html or text.

```
<?xml version="1.0" encoding="UTF-8"?>
<GetFeatureInfo xmlns="http://www.opengis.net/ows">
        <version>1.1.1</version>
        <service>wms</service>
        <exceptions>application_vnd_ogc_se_xml</exceptions>
        <Map>
                <BoundingBox decimal="." cs="," ts=" ">
                        -124.85,32.26,-113.56,42.75
                </BoundingBox>
        </Map>
        <Image>
                <Height>300</Height>
                <Width>400</Width>
                <Format>image/jpg</Format>
                <Transparent>true</Transparent>
                <BGColor>0xFFFFFF</BGColor>
        </Image>
        <QueryLayer>
                Nasa:Satellite, California:Faults, California:States
        </QueryLayer>
        <InfoFormat>text/html</InfoFormat>
        <FeatureCount>999</FeatureCount>
        <x>117</x>
        <y>218</y>
</GetFeatureInfo>
```

21. Why do you prefer using streaming data approach through NaradaBrokering?

In case of sending the result GML set in the form of string causes some problems when the GML is larger than some amount of size. Since the WFS returns the resulting XML document as an <xsd:string>, this has to be constructed in memory and the size will depend on several parameters such as the system configuration and memory allocated to the Java Virtual Machine etc. Consequently there will be a limit on the size of the returned XML documents.

For these reasons we have investigated alternative ways for data transport and researched use of topic based publish-subscribe messaging systems for streaming the data. Our research on NaradaBrokering shows that it can be used to stream large amount of data between nodes without significant overhead. Additional capabilities such as reliable messaging and support for different transport protocols already inherent in NaradaBrokering show that it is a powerful yet easy to integrate messaging infrastructure. For these reasons we have developed a novel Web Map Service and Web Feature Service that integrate Open Geospatial Consortium specifications with Web Service-SOAP calls and NaradaBrokering messaging system.

22. How does data streaming from WFS to WMS work to display a scientific data?

The clients make the requests with standard SOAP messages but for retrieving the results a NaradaBrokering subscriber class is used. Through first request to web service getFeature, WMS gets the topic (publish-subscribe for a specific data), IP and port on which WFS is streaming requested data. Second request is done by NaradaBrokering Subscriber. Even whole data is not received by WMS; WMS can draw the map image with the returned science data. This depends on the WMS's implementation.

23. In order to get further information about the feature data displayed on the map, I selected "*i*" map tool and clicked on the feature data on the map and saw the content data displayed in a pop-up window. How does it work?

This is done through invoking "GetFeatureInfo" interface of WMS. GML is consists of content and presentation parts. WMS's "GetFeatureInfo" interface enables displaying of content part of the data falling in a small area. This small area is around the point that user defines by clicking on the map and formulated by WMS Client as bbox and embedded in getFetatureInfo query. Area calculation in bbox changes depending on the zoom level. If you can't get any result you should change your resolution level by zoom-in or out.

When WMS get GetFeatureInfo, it creates getFeature request and sends it to WFS. Upon receiving returned GML, it extracts the content part (all but geometry elements) and converts it to HTML by using deployed XSLT and XSLT machine. It then returns HTML file to the client. The returned HTML document's format is defined in XSLT.

24. How do you convert feature data into HTML?

When WMS get GetFeatureInfo, it creates getFeature request and sends it to WFS. Upon receiving returned GML, WMS extracts the content part (all but geometry elements) and converts it into HTML by using deployed XSLT and XSLT machine. It then returns HTML file to the client. The returned HTML document's format is defined in XSLT.

25. How do you transfer and display large amount of scientific data (such as 60MB) in XML format?

In order to increase the performance we use innovative techniques. Regarding the transfer of large data sets see my explanation in Q21 and Q22. Regarding the parsing and rendering the returned GML, we use Pull Parsing technique (see Q26).

26. Why do you use pull parsing? What are the advantages over DOM and SAX parsing techniques?

In case of using pull parsing, the parser only parses what is asked for by the application rather than passing all events up to the client application. The pull approach of this parsing model results in a very small memory footprint (no document state maintenance required – compared to DOM), and very fast processing (fewer unnecessary event callbacks - compared to SAX).

Pull parsing does not provide any support for validation. This is the main reason that it is much faster than its competitors. If you are sure that data is completely valid, or if validation errors are not catastrophic to your system, or you can trust validation on the server, then using XML Pull Parsing gives the highest performance results.

Since we know that returned data is GML and know all possible tag names of the possible geometry elements, it is very advantageous using pull parsing.

XML Pull Parsing Web Site: http://www.xmlpull.org.

Please see the below article where XPP2 document model is compared with other leading Java based XML parsing implementations.

http://www-128.ibm.com/developerworks/xml/library/x-injava/index.html

27. How do you convert GML to SVG?

We convert GML to SVG by using XSLT.

28. How does WMS send the map images to the requester (WMS Client)?

There two ways one is a common pure OGC approach through HTTP and other is through SOAP messages using client stubs for the getMap interface.

In case of using HTTP, result is sent to client through java's servlet classes. For example result is written to client port by using HttpServletResponse's java.io.PrintWriter object. Client does not need to write any client stub or anything. He just put the URL on the web browser in a correct format defined in OGC specifications, and gets the map on the browser.

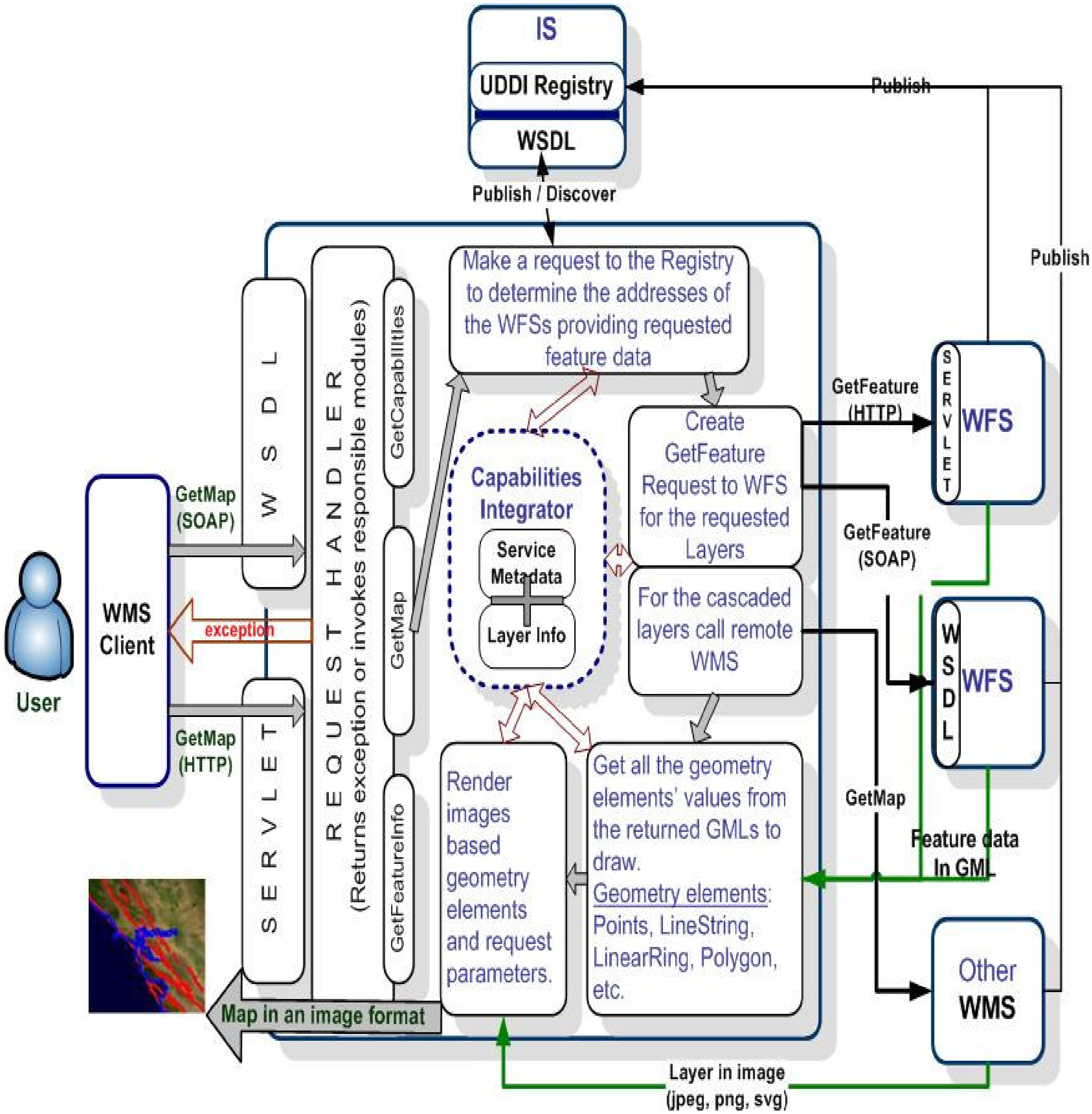
In case of using SOAP, WMS returns maps in image MIME types such as image/jpeg, before returning WMS converts maps into javax.activation.DataHandler object and attaches it to SOAP message. Client needs to create stubs before runtime in order to invoke Web Service interface of "getMap" by using online WSDL file of the WMS.

29. How do you create map images from the GML data after getting it from WFS? And, how do you overlay vector data from WFS over another layers coming from other WMSs?

There are twp possible options to create a map from the WMS point of view. One is creating SVG first from the GML through XSLT and converting SVG to any image type by using java libraries. Another is extracting geometry elements by using Pull parsing technique and drawing and displaying the map by java Image object. From our experience and performance tests we figured out that second approach has higher performance results then the first approach.

Web Map and Feature Services cooperate to create map images. XML-represented geographical features obtained from Web Feature Services are converted by Web Map servers into images using matrix-pixel (raster data), or points, lines and polygons (vector data). The basic operations related to mappings include discovering the physical address of the data providers, transferring the data from data servers and rendering it to create a map.

The figure below helps to explain a few more questions here (Ex. Q17, Q28 and Q30). Information Service (IS) in the figure is not completed exactly. Currently, we read the addresses of WFS and other Map services providing requested data from the WMS capability file. In the future, we plan to use any Information Service completing the architecture illustrated below.



30. How does WMS provide services through two different protocol in your implementation, one is over HTTP and other as Web Services?

The difference of these two is at the interface level. I have written a request handler for both protocols. Request handlers get the request (in case of Web Services, they are structured XML format – getMap, getFeature or getCapabilities) and parse and extract all the parameters. After having request name and related parameters, remaining jobs are all handled through the same classes and handlers.

3. Smart Map Tools

- **WMS Client**

31. What is SMT?

SMT is acronym of Smart Map Tools. We call it smart because tools allow users to manipulate data, create interactive queries and analyze the spatial information. SMT also enables Geo-Science application results interrelated with maps at the "layer" level. SMT also allows creating movie streams or animations from time dependent data.

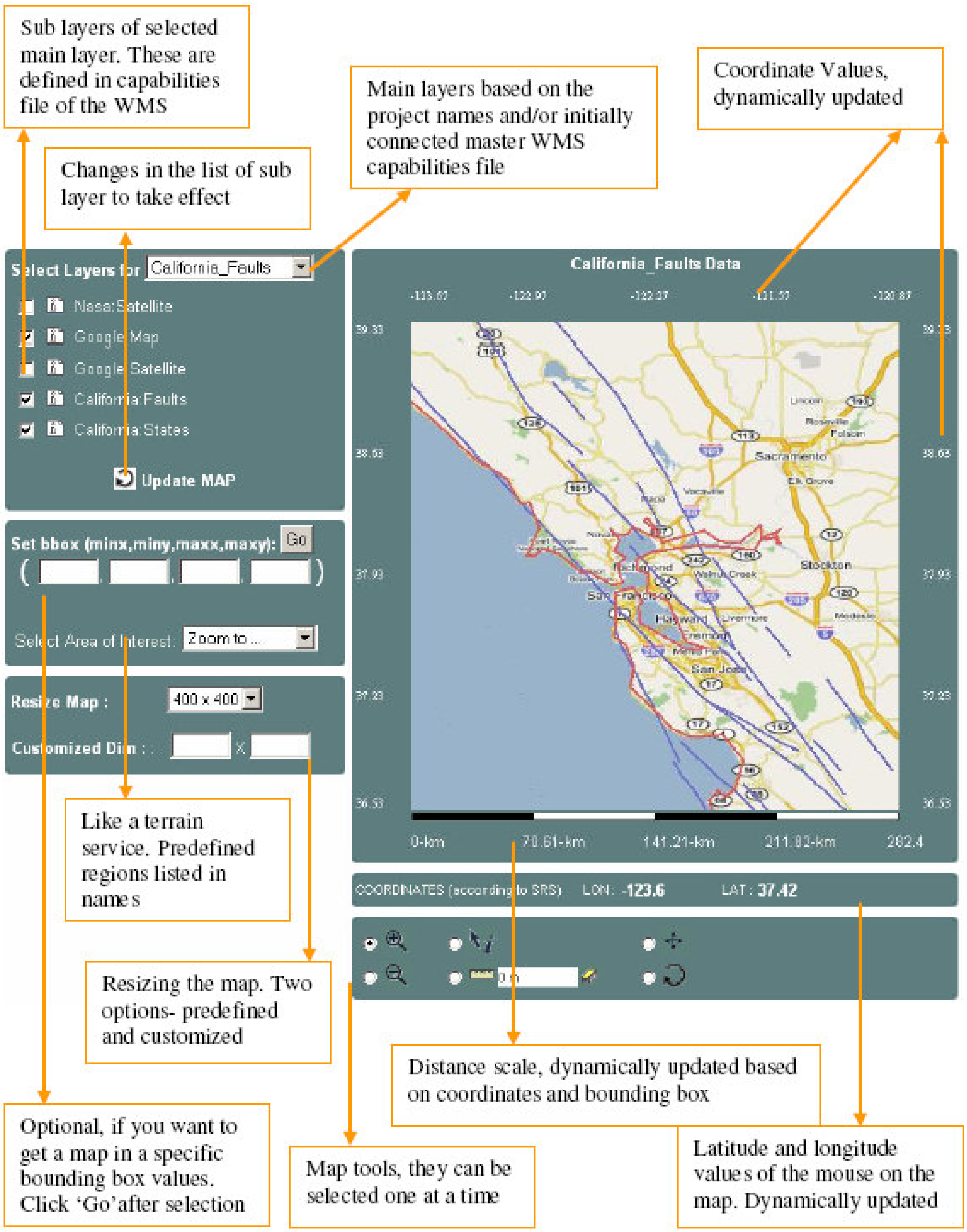
32. What functionalities does it provide?

In order to give brief idea about what SMT provides, I give development stages from the initial to the current.

1. I first implemented it as "WMS Client". At that time, SMT was just getting map images and displaying through standard map tools enabling zooming, moving and calculation distance between two selected points on the map etc.
2. I have added interactive query capability. Users are able to query data just by clicking on a specific data on the map. They do not write any complicated SQL queries or any other type of scripts.
3. I have added project based functionality to give additional interpretation to the data.
4. Embedding additional interfaces for the layers based on time dependent data in order to create streaming map movies. User is given option of defining time intervals such as per one year or per one month etc.
5. I created AJAX based approach. It enabled layer based integration of Google maps and OGC's WMS maps at the "layer" level and, overlaying features from WFS over Google Maps.
6. NOW – I am trying to create a flexible framework enabling integration of all these Map services with Geo-Science Grids. it includes interacting with Sci-Plotting services to plot Geo-Science Grids simulation outputs over Google maps or WMS maps, interacting with HPSearch Job Manager in accordance with Sci-Plotting services. Sci-Plotting services (by Sayar) and HPSearch Job manager (by Gadgil) are two of CGL projects.

33. Can you give a snapshot from SMT?

This is a snapshot from the very early stage of our SMT implementation. For more advanced one please see the snapshot at Q59. Explanations in orange framed boxes give the basic functionality of the SMT. It is implemented by using Java Server Pages (JSP), JavaScript and DHTML technologies.



34. Do you have any GIS demo running?

We have a demo running at http://156.56.104.187:8083/aaa/maptools/newmap.jsp. This is my implementation of user interface for the WMS Client and our proposed Map Services and Geo-Science grids integration framework.

Here is the *user manual* for instructions about how to use smart map tools to get and manipulate maps and run geo-science applications through smart map tools interactively.
http://complexity.ucs.indiana.edu/~asayar/gisgrids/docs/usermanual.pdf

35. What is the streaming map video?

Another capability SMT provides is the set of tools enabling creation of the streaming map movies. Visualizing changes over time is achieved by integrating temporal information on a map. Usually the result is a series of static maps showing certain themes at different moments. In addition to creating static maps, WMS also has the ability to combine the static maps correspond to a specific time interval data and combine them in an animated movie. Movies created by WMS are composed of a certain number of frames. Each frame represents a static map that corresponds to a time frame defined in request. Please see the Section 4 for more detailed information about the streaming map movies.

3.1. Integrating AJAX Approach to SMT (Q36 to Q45):

36. What is AJAX?

AJAX (Asynchronous JavaScript and XML) is a web development model for the browser based web applications. AJAX is a style of web application development that uses a mix of modern web technologies to provide a more interactive user experience. AJAX is not a technology. It is an approach to web applications that includes a couple of technologies. These are JavaScript, HTML, Cascading Style Sheets (CSS), Document Object Model (DOM), XML and XSLT, and XMLHttpRequest as messaging protocol.
AJAX drew some attention in the public, after Google started to develop some new applications with it. Some of the major products Google has introduced over the last year by using AJAX model are Google Groups, Google Suggests, and Google Maps. Besides the Google products Amazon also have used AJAX approach in its search engine application.

37. What inspired you to use AJAX and Web Services together for the GIS Vis Systems?

AJAX provides high interactivity and performance results for the browser based web applications. AJAX and Web Services are XML Based structures, so that they are able to leverage each others strength. Both AJAX and Web Services uses widely-used and well-known technologies such as XML and HTTP. Last but not least is that they are not competing but complementary technologies.

38. What are the advantages of using AJAX in your GIS Visualization systems?

AJAX has a couple of advantages for the browser based web applications developers. It eliminates the stop-start nature of interactions, user interactions with the server happen asynchronously, data can be manipulated without having to render the entire page again and again in the web browser, and requests and responses over the XMLHttpRequest protocol are structured XML documents.

Using AJAX approach in our system enables users to be able to integrate Google Maps with the Geo-science data in the form of layers (from WMS) or plotting (feature data from WFS). Regarding WFS, it provides vector data (points or line strings) in the GML format encoded in XML. GML contains content and presentation information for the feature data. Google maps are able to be plotted by using its own libraries after extracting the presentation data (OGC call them as Geometry elements such as points and line strings) through parsing of the GML.
By layering or/and plotting geo-science data on the Google maps, we obtain more detailed and informatory maps. As an example, I can see if my house is constructed on an earthquake fault through layering and in right zoom level.

39. What are the disadvantages of using AJAX and Web Services together (if there is…)?

This framework introduces some extra work. Extra work mostly comes from the conversion of parameters to be able to make compatible requests to remote Web Services. In order to make valid requests, the proxy server should be deployed locally and client stubs for Web Service invocations should be created before running the application.

40. What are the challenges in your architecture?

The biggest challenge is the handling the messages going through different protocols before accessing the receiver. Web Services and AJAX use different protocols for the data transfer. AJAX uses HTTP based XMLHttpRequest protocol and Web Services use SOAP over HTTP protocol (they also use SMP but in our implementations we use SOAP). They have no common message protocols to transfer messages and data.

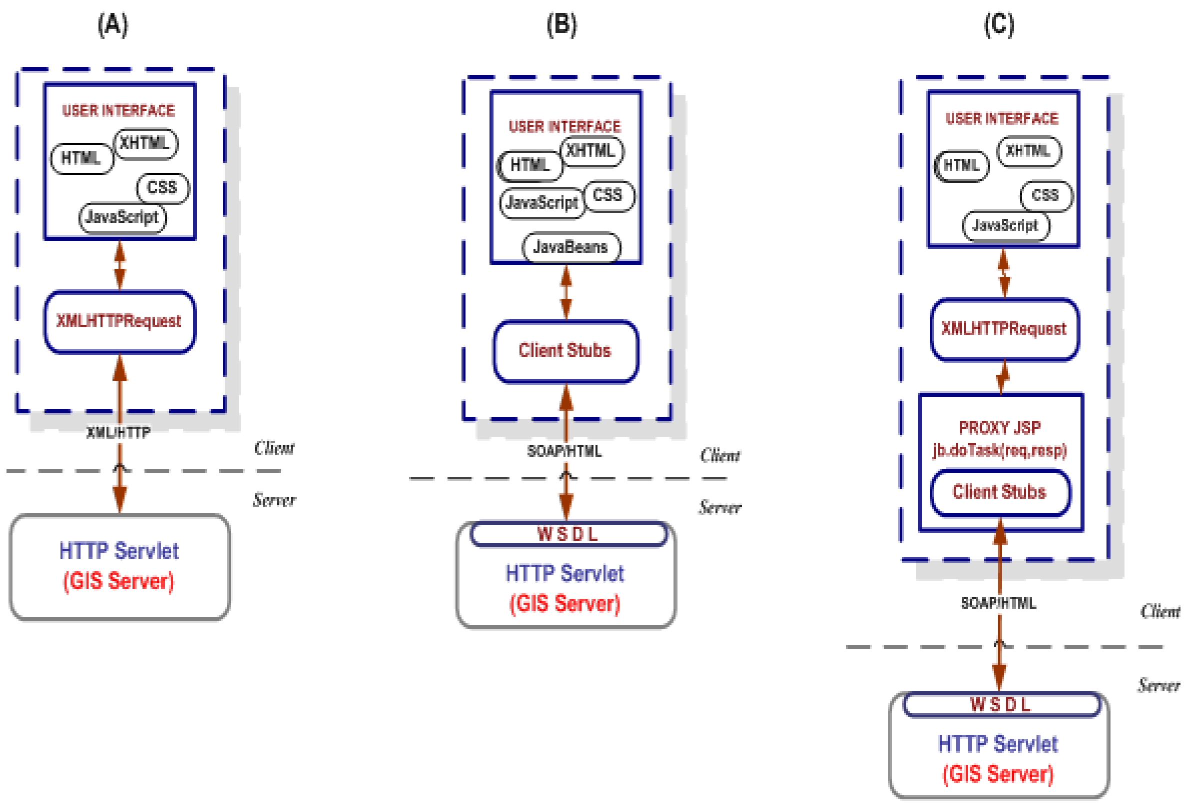
41. Can you use Google Maps to display geo-science data directly instead of using your integration approach?

First of their concern is commercial not scientific. It is not efficient for integrating and displaying scientific data over the Google Maps by using their libraries. There are a couple of reasons for that but the most important one is that they use DOM parsing internally and it is not memory and performance efficient when you need to display data larger then about 4MB. With our approach we can display even 30MB of XML based geo-science data in a reasonable time period.

42. What is the general architecture (approach for the involving AJAX)?

The major contribution of the proposed architecture is that users interact with the Web Service based GIS services through the AJAX enabled GIS client user interface.
AJAX uses HTTP GET/POST requests (through JavaScript's XMLHttpRequest) for the message transfers (see (A) in Figure). Web Services use Simple Object Access Protocol (SOAP) as a communications protocol (see (B) in Figure). In order to be able to integrate these two different message protocols, we convert the message format from one protocol to another. Since there is no ready to use common protocol to handle messages communications between AJAX and Web Services, we implemented a simple message conversion technique (see (C) in Figure). This essentially works by having the XMLHttpRequest communicate with a servlet, which in turn acts

as a client to a remote Web service. This allows us to easily convert between SOAP invocations and HTTP POSTS. It also has the benefit of avoiding JavaScript sandbox limitations: normally the XmlHttpRequest object in the browser can only interact with its originating Web server.

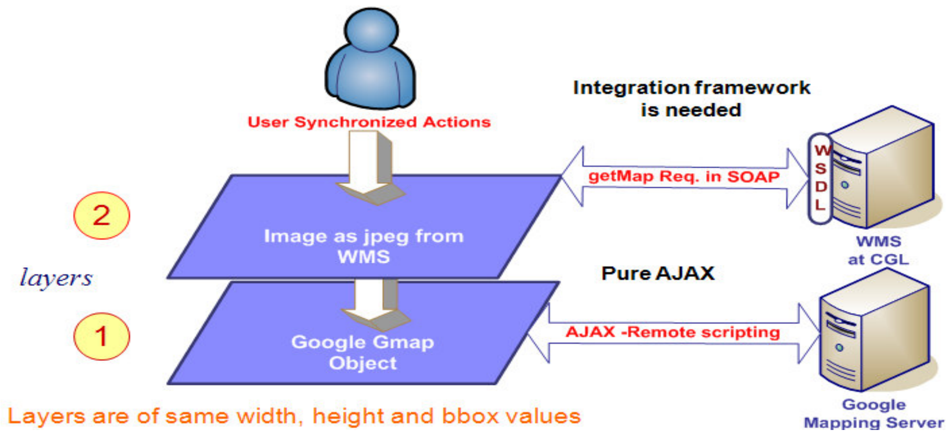


43. How do you integrate Google Maps with WMS?

This is actually a usage scenario of the general architecture explained in Q42 with some extensions. In order to fulfill this task we need to use base architecture.

Please see the figure below in order to understand the architecture in general before reading below explanations. There are two different path working in parallel by the given user parameters created by the client actions. Actions are interpreted by the browser through the Google Mapping tools on Layer 1. JavaScript captures these actions by ActionListeners and Google Binding APIs and gives to Layer-2 object.

Each layer is represented as JavaScript object. In our architecture we can utilize all the WMS Web Service interfaces such as “getMap”, “getCapabilities” and “getFeatureInfo”. The client does not need to make rendering or mapping jobs to create the map image. The map is consists of two layers one is coming from Google and other is coming from WMS. The layer returned by WMS is in a ready to use format such as JPEG or PNG or TIFF. The Layer returned by Google is a JavaScript object. In order to overlay these, we convert the layer coming from WMS into a JavaScript object before overlaying.

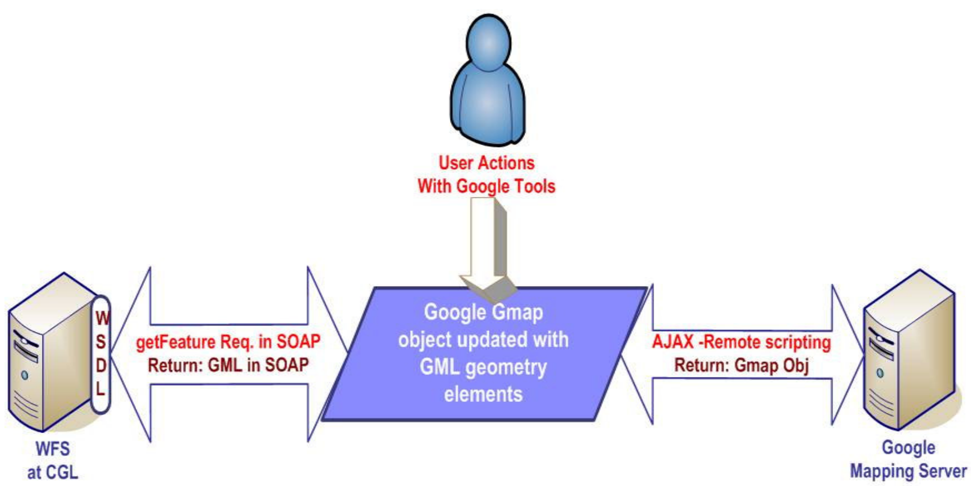


44. How do you integrate Google Maps with WFS?

This is actually a usage scenario of the general architecture explained in Q42 with some extensions. In order to fulfill this task we need to use base architecture.

Before reading the answer please see the figure below in order to better understand the architecture. WFS provide feature data in vector format and vector data are encoded in GML according to OGC WFS specifications. GML is an XML encoding for the transport and storage of geographic information, including both the geometry and properties of geographic features.

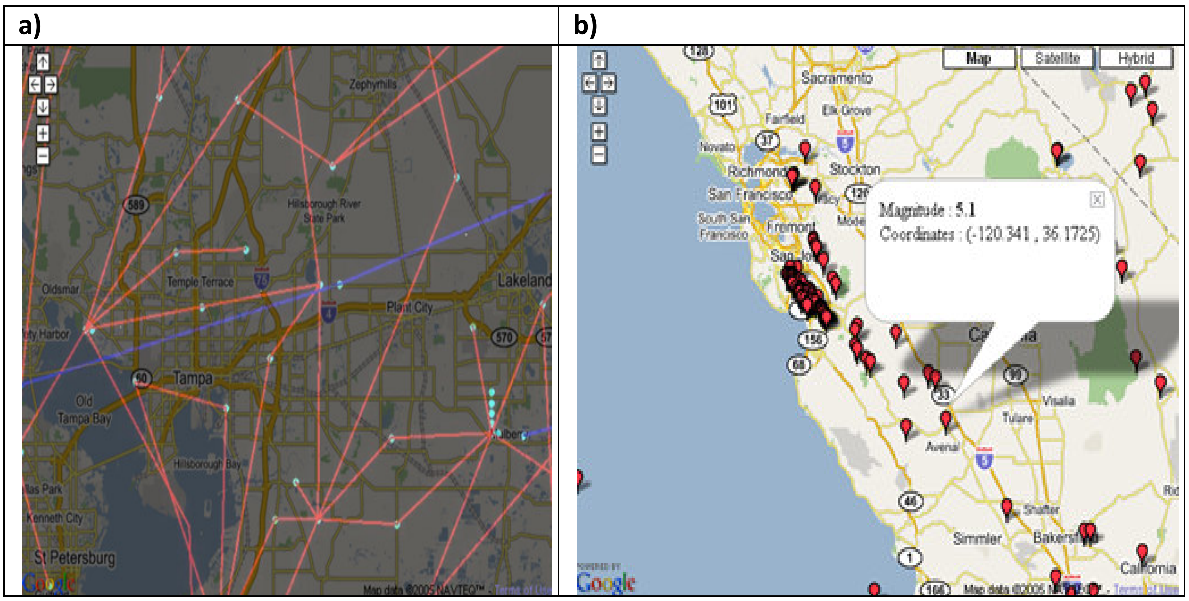
Client makes “getFeature” request and, the GML file encoded in XML is returned in a SOAP envelope. After getting a response, the client extracts geometry elements. The most important and commonly used geometry elements are Points, LineStrings, LinearRings, and Polygons. Even though Google Mapping API supports just two of them, Points and LineStrings, the other geometry elements can also be converted to these two types with minor updates. Having extracted and obtained geometry elements, these elements are plotted over the Google Map by using “GPoints” and “GPolylines” objects and the “mapOverlay” function of the Google Map API. By setting returned GML’s non-geometry elements and using ‘GMarker” object of the Google API, this architecture also provides the “getFeatureInfo” functionalities of the OGC WMS services. All these tasks are achieved by using XMLHttpRequest API and JavaScript functionalities.



45. Did you apply your architecture to any project? If so, can you give us sample screen shots of your outputs?

Figure-a is about Google and WMS integration. The layer is coming from WMS is overlaid on Google map. This is a real output from the IESS application done in los Alamos National Lab. Layers coming from WMS are consists of two feature data. These are Natural gas and electric power pipelines.

Figure-b is about Google and WFS integration. Application is done at Community Grids Lab (CGL) for Pattern Informatics, earthquake forecast application. In this sample application output we show California Earthquake seismic data plotted on Google maps as GIcons. You can even query data interactively by clicking on it as shown on the figure.



4. Architectural Framework for Streaming Map Movies

46. What is streaming map movie?

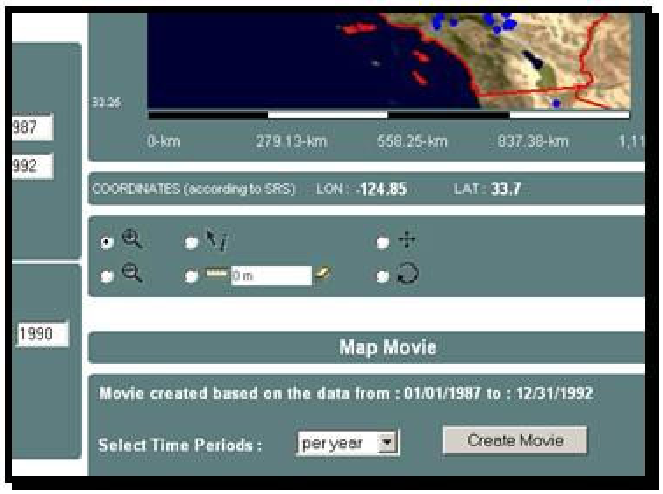
Visualizing changes over time is achieved by integrating temporal information on a map. Usually the result is a series of static maps showing certain themes at different moments. In addition to creating static maps, WMS also has the ability to combine the static maps correspond to a specific time interval data and combine them in an animated movie. Movies created by WMS are composed of a certain number of frames. Each frame represents a static map that corresponds to a time frame defined in request.

47. Why do you need to create map movie?

Standard map servers produce static images, but many types of geographic data are time dependent. In order to understand geographic phenomena and characteristics of temporal data it is necessary to examine how these patterns change over time for these types of data.

48. How does WMS create a streaming map movie?

Movies in general are composed on static frames. Frames are actually images. Movies are obtained by playing these static frames sequentially. In our case of map movies, frames are static map images created for a specific bounding box and a specific time interval. Sequential frames are created based on successive time intervals. Range of interval is given by the client from the user interface. See below figure, time interval is set to “P1Y”.



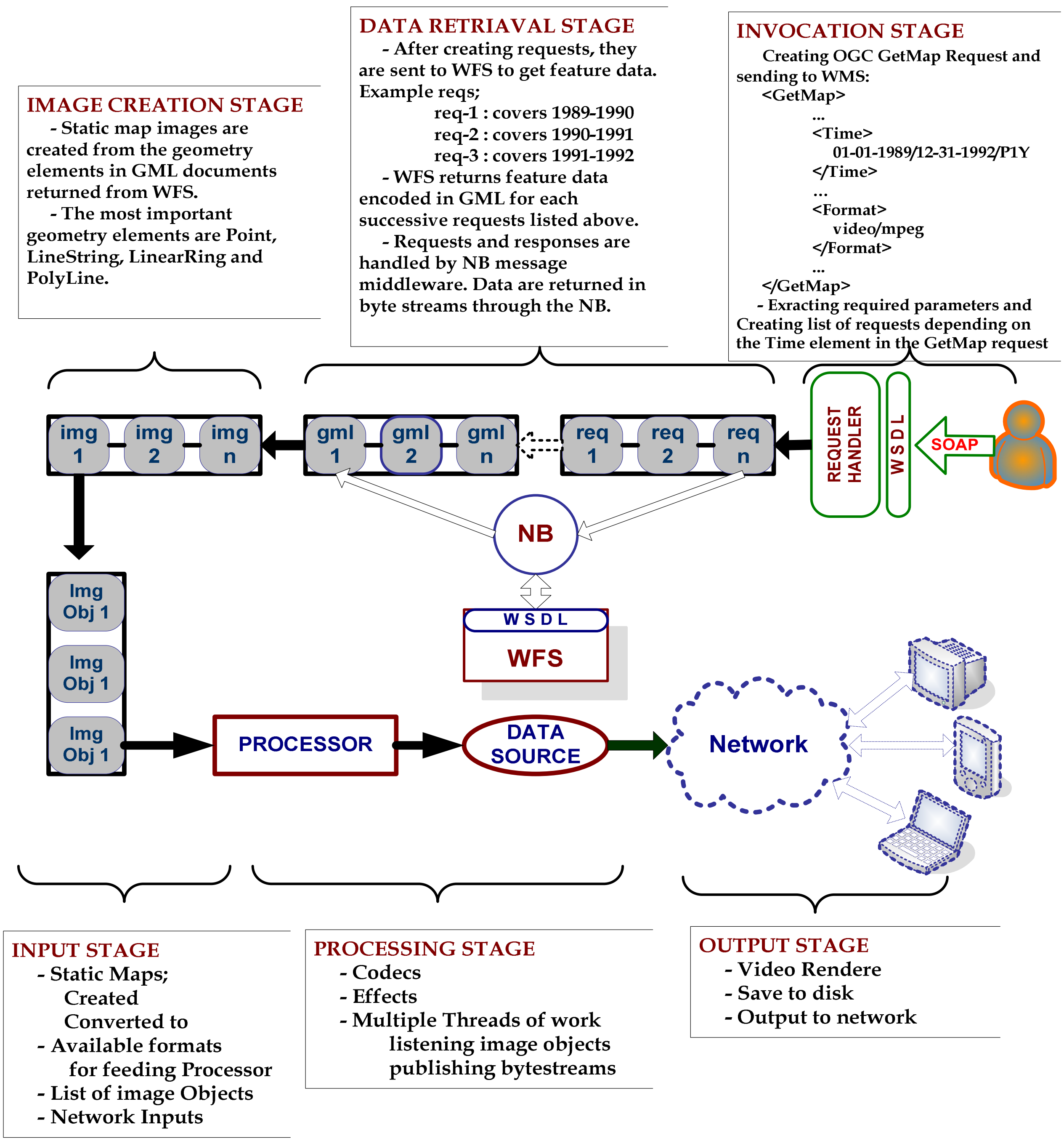
Movies are created just for time-dependent data. WMS notify the clients for what layers it can create movie streams through its capabilities file. If a client makes a request to get a movie for a specific layer, to succeed, this layer should have a time dimension defined under this layer element in the capabilities file. Here is an example dimension tag under the layer tag in capability. Layer is created based on a data which is provided yearly.

```
<Dimension name="time" units="ISO8601" default="2000-08-22">
        1999-01-01/2000-08-22/P1Y
</Dimension>
```

The last value “P1Y” (in the Dimension tag element above) defines the periodicity of the data collection. According to this value, WMS cuts the whole time (“from/to/periodicity”) into multiple values and for each time slice, it makes a "getFeature” request to WFS to get feature data in GML. After receiving GML, WMS creates static map images for each time slice. These images will be frames of the result map movie.

Clients should make the “getMap” standard request to WMS to get the movie for a specific layer but the “format” variable should be set to “movie/ <movietype>” and “time” variable should be set to a value in an appropriate format. The last value in the Dimension tag element (P1Y) is defined based on the user’s selection from the drop-down list shown in above figure. (See also a sample request for the movie in Q49 below).

Below figure is the BIG PICTURE for the creation of the map movies. In order to get further information for the map movies please see the *draft* technical report http://complexity.ucs.indiana.edu/~asayar/gisgrids/docs/movie_jmf.doc.



49. Can you give a sample movie request sent to WMS?

The request is actually "getMap" request with specific parameters set to specific values. We might need to request a map movie based on the layer created from a data. WMS defines this layer in its capabilities file with the "dimension" tag as displayed in Q48. In order for WMS to create map movie we need to send a “getMap” request as below. Here we request an mpeg movie created from frames created with a yearly data. First frame starts with the year 1987 and last frame ends with the year 1992. So, movie will be created from five frames.


```xml
<?xml version="1.0" encoding="UTF-8"?>
<GetMap xmlns="http://www.opengis.net/ows">
	<version>1.1.1</version>
	<service>wms</service>
	<exceptions>application_vnd_ogc_se_xml</exceptions>
	<Map>
		<BoundingBox decimal="." cs="," ts="">-124.85,32.26,-113.56,42.75</BoundingBox>
		<Elevation>5.0</Elevation>
		<Time>01-01-1987/12-31-1992/P1Y</Time>
	</Map>
	<Image>
		<Height>400</Height>
		<Width>400</Width>
		<Format>video/mpeg</Format>
		<Transparent>true</Transparent>
		<BGColor>0xFFFFFF</BGColor>
	</Image>
	<ns1:StyledLayerDescriptor version="1.0.20" xmlns:ns1="http://www.opengis.net/sld">
		<ns1:NamedLayer>
			<ns1:Name>Nasa:Satellite</ns1:Name>
			<ns1:Description>
				<ns1:Title>Nasa:Satellite</ns1:Title>
				<ns1:Abstract>Nasa:Satellite</ns1:Abstract>
			</ns1:Description>
		</ns1:NamedLayer>
		<ns1:NamedLayer>
			<ns1:Name>California:States</ns1:Name>
			<ns1:Description>
				<ns1:Title>California:States</ns1:Title>
				<ns1:Abstract>California:States</ns1:Abstract>
			</ns1:Description>
		</ns1:NamedLayer>
		<ns1:NamedLayer>
			<ns1:Name>World:Seismic</ns1:Name>
			<ns1:Description>
				<ns1:Title>World:Seismic</ns1:Title>
				<ns1:Abstract>World:Seismic</ns1:Abstract>
			</ns1:Description>
		</ns1:NamedLayer>

	</ns1:StyledLayerDescriptor>
</GetMap>
```

50. Can you define any attributes such as frame rate for the movie streams published by WMS?

The map video stream has several parameters that can be adjusted. These parameters affect the quality of the produced map video stream. Among these configurable parameters are frame rate and video format of the stream, update rate of the map images in the video stream.
In our current implementation, we define these attributes in hard coding but in the future we will take it from the smart map tools interface. This is necessary because; some clients might not be capable of visualizing video streams with low frame rate or can visualize them with very low quality because of their connection or some other issues. They might need to adjust the attributes and get the best result.

51. How do you display map videos?

When the frames (static maps) corresponding to separate time slices are created, WMS publishes them to a Real Time Protocol (RTP) session as streams. RTP sessions are defined with IP Address and Port Number couples. Movie streams published to an RTP session can be visualized by any client connecting to the same RTP session (same IP and port number defined by the publisher).
If you want to display the movie streams with a standalone program (not in collaboration environment such as AccessGrid and Global MMCS), then download and install JMStudio software (http://java.sun.com/products/java-media/jmf/). From the user interface select "File" menu and click on "open RTP session" and give IP, port and time to live (ttl) values that you already know there is one WMS publishing map movie streams to this session.

52. Do you have any sample map movie to see online?

Yes, see the below link. Movie at the below link is obtained by screen recording of the real PI Geo-Science application's output. Movie is composed of yearly seismic earthquake records. The minimum magnitude is 4.0 (PI spec parameter), and frames are composed of static maps created yearly based.
http://complexity.ucs.indiana.edu/~asayar/gisgrids/html/work/AutoScreenRecorder_01.mpg

5. Sci-Plotting Services *(mostly completed)*

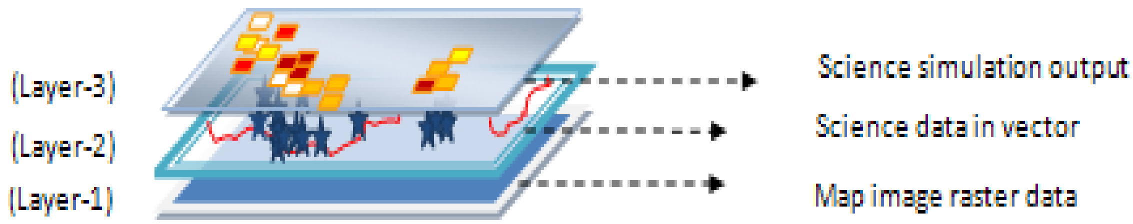
53. What is Sci-Plotting service?

It is a rendering service like WMS providing layers integratable to WMS maps but not map images. They provide layers created from geo-science simulation outputs in tabular format. For the core functionality we use Dislin scientific-data plotting libraries. Dislin is a plotting library containing functions for displaying data graphically as curves, graphs, pie-charts, 3-D color plots, surfaces and contours. All these services are wrapped as Web Services and integrated into the general visualization system. This is a service consists of the functionalities provided as Web

Services through which any geo-science application output data can be interpreted, rendered and displayed as images and graphs.

54. Can you give a sample output layer Sci-Plotting service provides?

In the three layer set structured architecture, Layer-1 and Layer-2 come from WMS and, Layer-3 comes from Sci-Plotting services.



Three-Layer Structure, maps are composed of three classes of layers in above orderings.

55. Did you use Sci-Plotting services in any real geo-science application?

Yes we did. GeoFEST is a two- and three-dimensional finite element software package for the modeling of solid stress and strain in geophysical and other continuum domain applications. GeoFEST uses stress-displacement finite elements to model stress and strain due to elastic static response to an earthquake event in the region of the slipping fault, the time-dependent viscoelastic relaxation, and the net effects from a series of earthquakes. Sample output snapshot at Q59.

56. What are the inputs and outputs of Sci-Plotting services?

Currently raw data (input data) is in the column format and located on an HTTP address which can be accessed by using HTTP protocol. It is actually a plain text file. URL to this file is given as parameter to the request for plotting. For each different application there is a data conversion filter to make the data understandable to the service. For example for PI applications Sci-Plotting server is given a file URL to plot in which data is in tabular format. One column shows X coordinate other column shows Y coordinate and other columns shows the values to be plotted. In the future we plan to use VOTables to represent tabular data.

The VOTable format is an XML representation of the tabular data. The VOTable format's aim is making the online Sci-Data both interoperable and scalable. The interoperability is encouraged through the use of XML standards. When the data is encoded in XML, data interpretation and validation will be easier. During the machine to machine communication, data encapsulation in any kind of messages such as SOAP will also be easier. Application specific metadata can also be embedded in to the VOTable. Metadata can be about the owner of the data, application properties of the data, authentication etc. There is also a price for encoding data in XML. When the data encoded in XML, each data element is tagged. The tagging causes overheads in terms

of the volume. Increasing in the data volume basically causes increasing in the transfer time and the processing time.
Return type is image map layers to be overlaid on other maps or graphs showing statistic information to be displayed separately. Most of the output types are application dependent so far but since our implementation packages are open source you can download and modify it in accordance with your application and needs.

57. What is the most challenging part in creating Sci-Plotting services?

Input data comes from variety of different Geo-Science applications to be plotted. Each of these applications output different number of records, columns, variables, and different data structures. Most of the applications outputs thousands or even millions of lines of records. Storing, updating and reusing of the data are cumbersome. Some times, application requires some restrictions and properties on the data. In that case, there should be some properties and attributes about the archived data to check against these requirements. For example, some data are valid just for some specific time intervals or they might have expiration dates for their validations. Furthermore, data on the net may require authentication for access. The remote data needs to be accessed by using various arbitrarily complex protocols by using the URL syntax, “protocol://location”.
The VOTable format seeks answers and solutions to the problems and challenges explained in the previous paragraph. The main goal is making the online geo-data both interoperable and scalable. The interoperability is encouraged through the use of XML standards. When the data encoded in XML, data interpretation and validation will be easier. During the machine to machine communication, data encapsulation in any kind of messages such as SOAP will also be easier. Metadata about the data can also be embedded in to the data. Metadata can be about the owner of the data, application properties of the data, authentication etc. Everything is not perfect when we used the XML encoded data. There is also a price for it. When the data encoded in XML, each data element is tagged. The tagging causes overheads in terms of volume. Increasing in the data volume basically causes increasing in the transfer time and the processing time.

58. How do you use core Dislin libraries through Web Service Interfaces?

Dislin is originally written in Fortran and C but they have also their corresponding native libraries. Native libraries allow us call codes written in any programming language from a program written in the Java language by declaring a native Java method, loading the library that contains the native code, and then calling the native method. We embedded Dislin libraries into our plotting Web Services which is written in Java by using Java Native Interfaces and native methods. In order to install it into our whole system we needed to publish the plotting services as Web Services. Therefore, we first created service description files (WSDL) and publish them by using java axis web services container.
Service stack is composed of native method declarations. All the declared native methods have their own corresponding native implementation routines in the loaded native libraries. Routines

can be accessed from any java class through the ServiceStack class. You can see the shrunk version of the ServiceStack class as below. Since all the methods are declared as static, class is assumed to be static as well. So, sample contour calls from any java class will be as below;

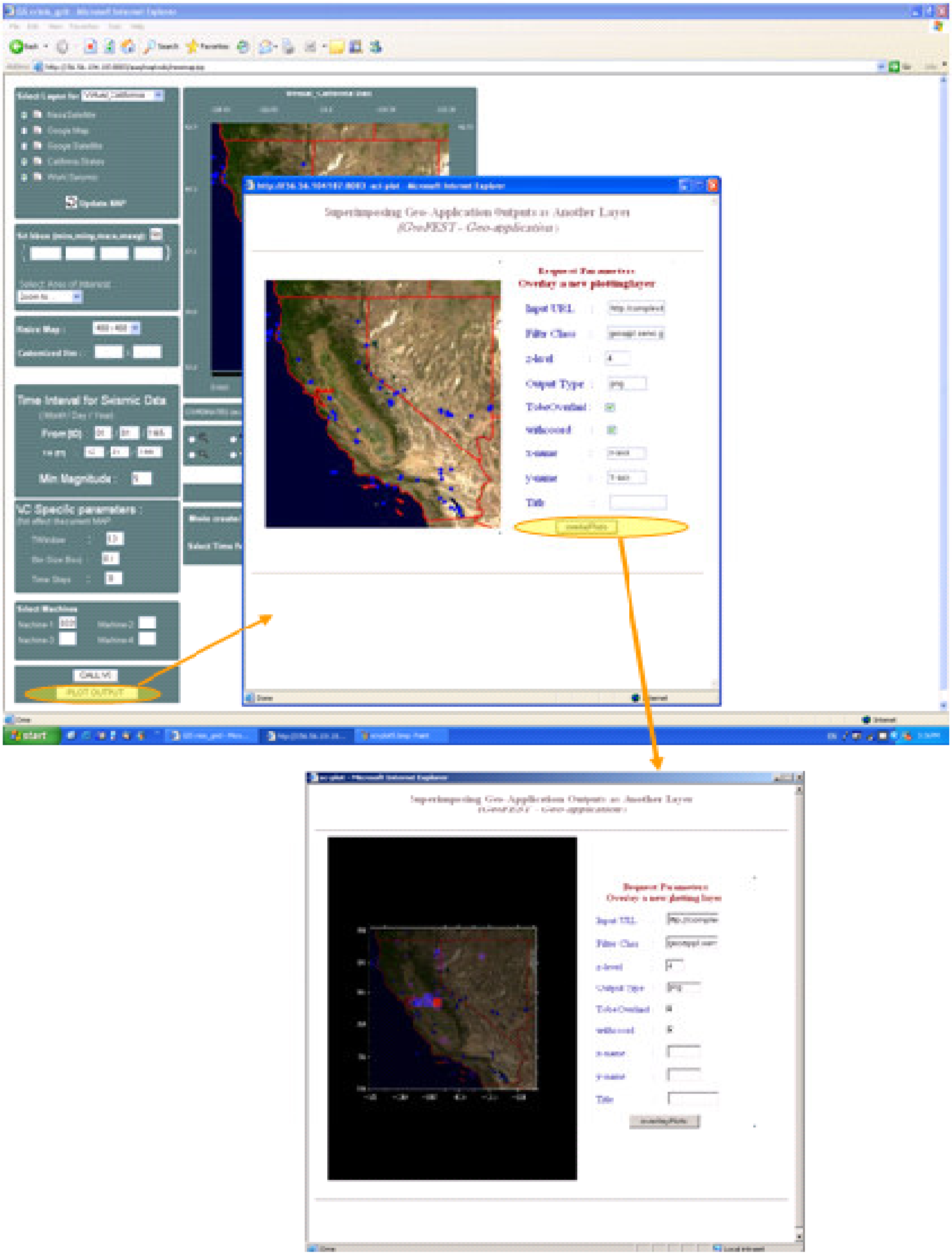
```
// before calling ServiceStack routines first import ServiceStack class
ServiceStack.contour(x-array, x-length, y-array, y-length, data-array, data-length, z-level);

public class ServiceStack
{
   public ServiceStack ()   {    }
   public static native void contri(float af[], float af1[], float af2[], int i, int ai[], int ai1[], int ai2[]);
   public static native void abs3pt(float f, float f1, float f2, float af[]);
   public static native void contour(float af[], int i, float af1[], int j, float af2[], float f);
   public static native void angle(int i);
   /*
       Native routine correspondences of all the scientific data plotting libraries
   */
   static
   {
      System.loadLibrary("javaNativeCodeLibraries");
   }
}
```

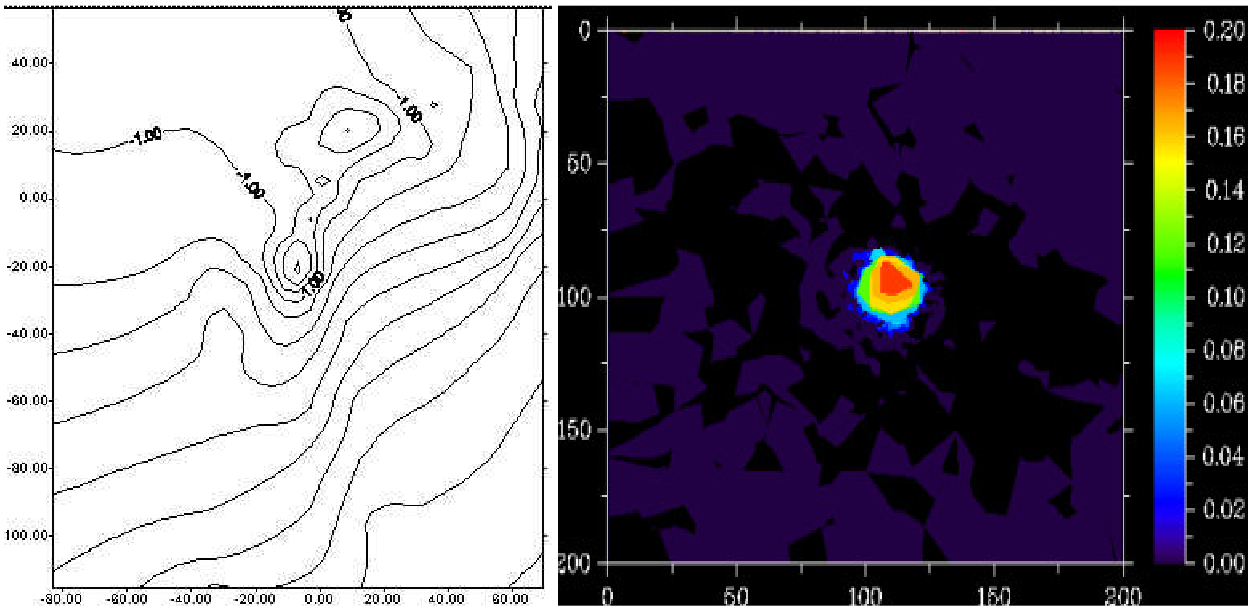
59. How do you invoke the Sci-Plotting services? Can you give sample snapshot from your outputs?

We initially created below user interface as WMS Client. It was enabling just getting maps from WMSs. Later, we enhanced it with smart map tools and movie streaming interfaces. Finally, we have added and integrated new interfaces for invoking scientific plotting Interfaces in accordance with other interacted GIS services.

GUI and map tools enable users to dynamically interact with the Dislin libraries and GIS visualization services. Each action by the user causes the changes in the parameter values and new request to the corresponding services. Depending on the application and user needs, plotting layers can be shown on another pop-up browser or just overlaid on other layers. All these kinds of service parameters are defined in the requests.



Pattern Informatics Geo-Science Application snapshot



GeoFEST Geo-Science Application snapshot