Getting involved with GeoSciML

Why do we need GeoSciML?

Globally Geological Surveys are experiencing an ever increasing digital data deluge. Our customers are becoming more digitally sophisticated and are no longer satisfied with images and portrayals of data. Customers want digital data in standardized formats that can be used immediately in applications. Hours, days or weeks spent merging data sets obtained separately from multiple agencies is time wasted.

The eXtensible Markup Language (XML) and web-based data delivery is the technology that enables geoscience agencies to service this need. It allows agencies to implement standardized interfaces to standardized re-usable data products, enabling applications to use global geoscience data in real time.

GeoSciML is a geoscience specific XML-based GML (Geography Markup Language) application that supports interchange of geoscience information. It is being developed through the Interoperability Working Group of the Commission for the Management and Application of Geoscience Information (CGI), a commission of the International Union of Geological Sciences (IUGS). The Working Group is (currently) comprised of geology and information technology specialists from agencies in North America, Europe and Australia.

What exactly is GeoSciML?

The GeoSciML application is a standards-based data format that provides a framework for application-neutral encoding of geoscience thematic data and related spatial data. GeoSciML is based on Geography Markup Language (GML – ISO DIS 19136) for representation of features and geometry, and the Open Geospatial Consortium (OGC) Observations and Measurements standard for observational data. Geoscience-specific aspects of the schema are based on a conceptual model for geoscience concepts and include geologic unit, geologic structure, and Earth material from the North America Data Model (NADMC1, 2004), and borehole information from the eXploration and Mining Markup Language (XMML). Development of controlled vocabulary resources for specifying content to realize semantic data interoperability is underway.

Intended uses are for data portals publishing data for customers in GeoSciML, for interchanging data between organisations that use different database implementations and software/systems environments, and in particular for use in geoscience web services. Thus, GeoSciML allows applications to utilize globally distributed geoscience data and information.

GeoSciML is *not* a database structure. GeoSciML defines a format for data interchange. Agencies can provide a GeoSciML *interface* onto their existing data base systems, with no restructuring of internal databases required (Figure 1).

Where can I learn about GeoSciML?

Six International and 2 state surveys, stretching from Australia to Europe to North America, participated in a proof-of-concept demonstration of GeoSciML at the International Association of Mathematical Geologists (IAMG) meeting in Liege, Belgium in September 2006.

The demonstration showed that it was possible to access information in real time from globally distributed data sources. Geological map polygons and attribute information, and borehole data, were displayed, queried and re-portrayed using web applications hosted by the Geological Survey of Canada and the BRGM. GeoSciML data could also be downloaded. PowerPoint presentations on GeoSciML from the IAMG 06 meeting are available at https://www.seegrid.csiro.au/twiki/bin/view/CGIModel/GeoSciMLPresentations .

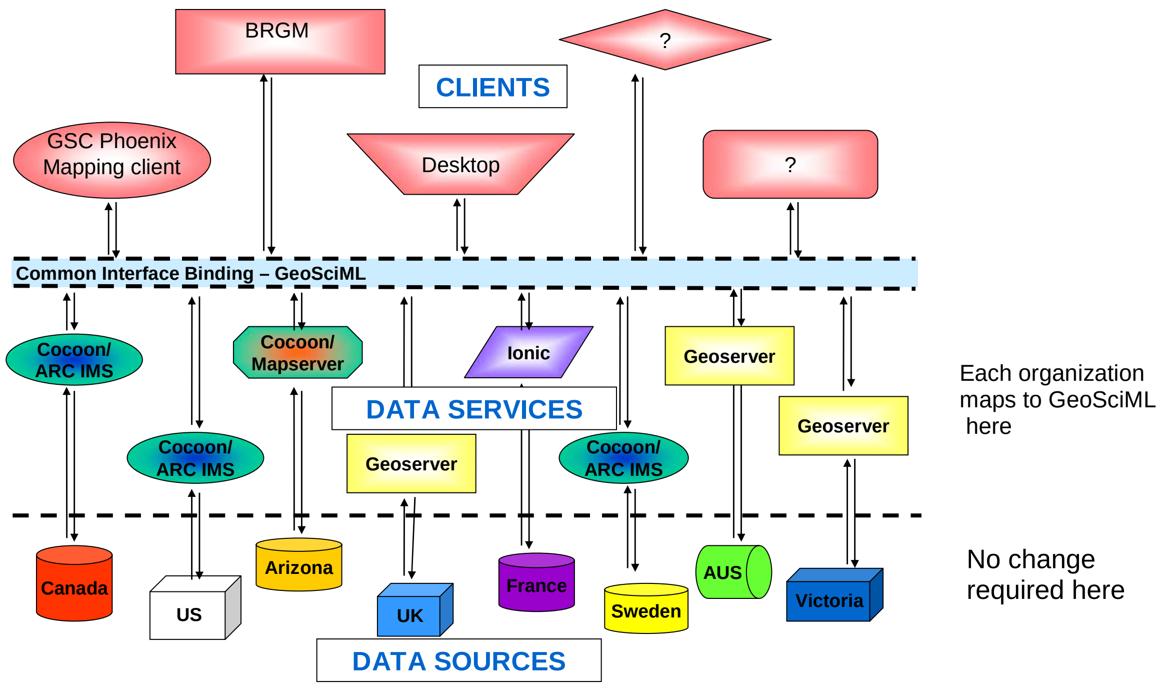

Figure 1: Architecture of the GeoSciML Test Bed 2

How can I join GeoSciML?

Development of GeoSciML is an open process with the intent to involve as many participants as possible. This will ensure development of a schema and services that will meet the needs of a wide variety of geoscience data producers and users. Three types of participation are available: 1) Direct participation in GeoSciML development, 2) Monitoring GeoSciML development via the web-collaboration tools and 3) deploying an internet server to provide data in GeoSciML format.

1) Direct participation in development of GeoSciML. The CGI Interoperability Working Group has established the GeoSciML Interest Community. A mailing list and website will provide information on activities and report on progress. Task groups operate with specific objectives to (i) design GeoSciML v2.0, (ii) collect use cases and validate them in testbeds, (iii) design the service architecture, and (iv) plan service deployment,. Members of the Interest Community are invited to participate.

2) Follow developments on the collaboration portal. The portal is at https://www.seegrid.csiro.au/twiki/bin/view/CGIModel/WebHome . Discussions of future developments, proposed changes, documentation of current efforts and presentations are freely available. Users can subscribe to be informed of changes daily

.

3) **Setting up an internet server to provide data in GeoSciML format.**
See Appendix 1.

Appendix 1. How to set up a GeoSciML server: Technical Overview

Summary of requirements:

1. Data in vector digital form
2. Web server connected to the internet
3. Internet map server that can access data.
4. Software to process OGC Web Map Service (WMS) and Web Feature Service (WFS) requests
5. Software to convert hosted data into GeoSciML based on service requests.

The basic resource to be served is geologic map data in vector digital form. The geometry of points, lines, and polygons comprising a geologic map are represented in the database by ordered collections of points with coordinates related to geographic location through some spatial reference system (e.g. Universal Transverse Mercator projection, or latitude-longitude). Database tables contain attribute data associated with the mapped features. Typically this includes information defining the kinds of lines and polygons represented, geologic units used to classify outcrops represented by polygons, and data collected at point locations (e.g. strike and dip of bedding). This information is typically managed using Geographic Information System (GIS) software, for example PostGIS (open source) or ESRI ArcGIS or MapInfo (commercial).

To get an internet server online, basic hardware requirements are a broadband internet connection (T1, 10-100 megabits per second (Mbs)); a router that interfaces a local area network (LAN) with the internet; and a computer with sufficient storage and calculation capacity (exact requirements depend on volume of data and traffic expected). Software requirements include web serving software (Open source solutions-Apache; Commercial solutions -- IIS); and internet map serving software (Open source solutions GeoServer, MapServer; Commercial—ArcIMS). Various other software will probably be required for security, backup, and web application support. Again exact requirements will vary depending on implementation details.

Currently available internet map serving software can be expected to implement some sort of Web Map Service (WMS) and Web Feature Service (WFS) interfaces. These interfaces are defined by the Open Geospatial Consortium (OGC) and International Organization for Standarization (ISO). These specify the syntax for requests sent to map servers as part of web URL's.

A WMS is a service that returns an image or picture (e.g. tiff, jpeg, pdf) to the requesting client. WMS is designed to send map graphics to lightweight viewer clients. In contrast, a WFS service returns feature data in XML, in this case consisting of some description of geometry that specifies location, with attribute data specifying properties associated with the geometry. An example would be a the shape and description of geologic units cropping out. WFS is designed to provide reusable spatial data to support further processing which may include but is not restricted to portrayal. The WFS response uses GML (Geography Markup Language) to serialize the location and attribute data in a text file.

The role of GeoSciML is to define the structure and syntax of WFS responses to requests for *geoscience* information. In order for this to happen, the physical data store (e.g. in-house GIS) must be mapped into the GeoSciML text file.

In order to deliver GeoSciML, most commercial WFS implementations require a data source with the GeoSciML structure. For more organizations this requires either a redesign of their in-house database, or replication of the database in a transformed version. In general this is seen as undesirable, so various mechanisms have been tested to perform the conversion from an unmodified datastore in real time. These always involve customization based on in-depth understanding of the local data store and the GeoSciML model.

Geoserver does the conversion within the internet map server environment (GeoSciML-enabled server), such that WFS requests are sent directly to the map server, and the map server responds directly with GeoSciML output. Customization requires construction of configuration files that defines the mapping from the back end data store to GeoSciML.

The other major approaches utilize a 'pipeline', in which web requests are routed through a preprocessor that communicates with the internet map server software and in some cases directly with the back end database, and marshals the responses into GeoSciML for return to the client (wrapper implementation). Customization involves setting up the pipeline and generating software to do the mapping from back end database or internet map server responses into GeoSciML. Various test bed participants have implemented wrappers using XSLT, Java, and PHP. The use of a GeoSciML-enabled server requires less software installation and no software program writing, but is less customizable and flexible than the wrapper implementation. Wrapper implementation presents the possibility of constructing a wrapper that may be used with multiple, distributed WFS sources, but requires more software installation, configuration, and coding.

Roll out of a GeoSciML enabled map service requires some mechanism to allow discovery of the service. Possibilities include links from a lightweight web page client, or registration with a data clearing house or catalog (e.g. the GSC Phoenix client, http://ctp.geosemantica.net/collections/layers_folderview.aspx).