OGDI: Toward Interoperability among Geospatial Databases

Gilles Clement,[1] Christian Larouche,[2]
Denis Gouin,[3] Paul Morin,[4] and Henry Kucera[5]

Abstract

The growth of the geomatics industry is stunted by the difficulty of obtaining and transforming suitable spatial data. This paper describes a remedy: the Open Geospatial Datastore Interface (OGDI), which permits application software to access a variety of spatial data products. The discussion compares the OGDI approach to other standards efforts and describes the characteristics and use of OGDI, which is in the public domain.

1. Introduction

The geomatics industry has grown continuously over the last two decades, and the number of application domains that use geospatial information increases each year. But the industry's growth has not met the predictions of 10 years ago, and lags behind such information technologies as Internet, Office Automation, and Sales Automation. Some experts estimate that the geomatics industry would generate 10 to 12 times more revenue if the geospatial data barrier could be eliminated.

This paper describes one way to alleviate the problem: the Open Geospatial Datastore Interface (OGDI). OGDI is a public-domain application programming interface (API) that permits applications software to read "foreign" data formats directly, without translation.

The discussion that follows describes the geospatial data barrier and outlines remedies (Section 2). Section 3 outlines the characteristics and architecture of OGDI. Future plans and instructions for obtaining the source code are in Section 4.

2. Breaking the geospatial data barrier

Billions of dollars have been invested worldwide in the production of geospatial datasets [Buehler and McKee 1996]. Datasets now exist for municipal planning, forestry, mining, environmental monitoring, and military command-and-control. Each geospatial application tends to have its own particular data requirements (e.g., raster or vector representation, level of detail, projection, datum), so data products targeted at different applications also tend to use different formats.

Producing geospatial data is expensive. For example, the cost of producing a single 1:250,000 VMap vector map by converting existing digital files is about $3,000. Purchasing geospatial data also is expensive. A 1:250,000 digital map sheet sold by National Resources Canada (NRCan) costs about $600. Data sharing is the best hope for minimizing costs. Three solutions to break the data barrier have evolved: data translation, data standards, and open frameworks.

2.1 Ad hoc translation

Most geospatial data today are used in geographic information systems (GISs), and most GIS vendors use their own proprietary data format, designed to maximize their own systems' performance. To use "foreign" data in a commercial GIS (e.g., data not in the internal or "native" format), the common approach is to translate foreign data to the native format. This works, but there are drawbacks.

- Transformations between formats can result in loss of information because different formats support different types of data..
- Translation ties up large amounts of secondary storage because it is performed off-line, in bulk.
- Translators can be costly to develop and generally are sold at a loss. A separate translator is required for every system for every format.

[1] Logiciels et Applications Scientifiques (LAS) Inc., 1570 Chabanel St. W., Montreal, Quebec H4N 1H4, gc@las.com

[2] LAS Inc. (see above), cl@las.com

[3] Defense research Establishment Valcartier (DREV), 2459 Pie-XI Blvd. N., Val-Belair, Quebec, G3J 1X5, denis.gouin@drev.dnd.ca

[4] J2 Geomatics, 615 Booth St., Ottawa, Ontario, K1A 0K2, pmorin@ncs.dnd.ca

[5] Mercator Systems Ltd., 84 Moss St., Victoria, BC V8V 4L8, hkucera@ibm.net

- Some data providers repackage products for particular GISs. This favors some vendors over others, and hampers healthy competition for market share. It also causes proprietary formats to become *de facto* standards, regardless of their true worth.

2.2 Format standards

Standardization of data formats is the approach favored by many data producers. A number of geospatial data exchange standards now exist. Their use does not eliminate the need for translation, but it reduces the *number* of translators needed and improves interoperability by specifying the information to be exchanged. One standard can be used for many different data products, and can be read into different application software. An example of such a format is DIGEST (DIgital Geographic information Exchange STandard).

DIGEST was issued and approved first as a NATO standard, then as a Canadian standard [DGIWG 1994]. DIGEST was designed to exchange map and chart data, which are of two dimensions with an optional z value. DIGEST can support vector, raster, and matrix data type; it permits different levels of topological description and feature codings, and uses a hierarchical structure. As part of the Global Geographic Information & Services Initiative [McKellar 1995], eleven NATO countries have undertaken a $100 million geospatial data production project to generate worldwide coverage in the DIGEST vector format.

Data standardization is a positive step toward solving the geospatial data barrier, but by itself it isn't enough. The industry is unlikely to adopt a single standard. More likely, many standards will coexist.

2.3 Open frameworks

Over the last three years, interoperability has become a major goal among vendors and users of geospatial data. Interoperable approaches can use object-oriented technology, open architectures, and Intranet/Internet communications. To be interoperable, a system must be able exchange both information and applications with other systems; thus it must offer standard interfaces for geospatial data and geoprocessing operations in both stand-alone configurations and across networks. The increasing popularity of the electronic highway and the growing use of commercial satellites to distribute information has accentuated the need for interoperability. Some of the efforts that relate particularly to this work are reviewed here.

ISO/TC 211. Formed in 1994, ISO/TC 211 is an international effort to establish standards in geomatics. The standard specifies methods, and services to manage geospatial data; and their acquisition, treatment, analysis, presentation, and transfer in digital format between different users, systems, and sites. TC 211 is potentially the most important component in the movement toward compatible geographic standards because it provides a politically neutral forum for all proponents to align their respective standards [Lam 1996].

ISO SQL3 and SQL/MM. ISO/IEC JTC1 SC21 WG3 has the mandate to develop standards for database operation, language, models, and intercommunication. An extension to the SQL3 standard [ISO 1996], called SQL/MM [Ashworth 1996], covers multi-media and spatial data. *SQL/MM Part 3: Spatial* is aimed at providing database capabilities to increase interoperability and management of spatial data. SQL/MM Spatial supports the use of spatial objects through abstract data types; the objects can have up to three time-stamped dimensions, with some limitations on the use of the third dimension. Requirements for spatial metadata are drawn from TC 211. The core of SQL/MM Spatial will be finalized in July 1997. Other parts are still being studied.

OGIS. The Open GIS Consortium provides a forum for software vendors to seek interoperability. The Open Geodata Interoperability Specification (OGIS) [Buehler and McKee 1996; http://www.opengis.org] is a specification of a software framework for distributed access to geodata and geoprocessing resources. The specification is developed under a non-profit, open consortium with participants from major sectors of the industry. OGIS plans to develop a detailed common interface template for writing software that will interoperate with other OGIS-compliant software written by different software developers. There is regular interaction between TC211, SQL/MM, and OGIS through formal liaison and informal discussion.

DGIWG. The Digital Geographic Information Working Group is a NATO standards committee. DGIWG's objectives focus on the DIGEST map data transfer format. They include:

- promote the use of DIGEST through freeware tools;
- provide interoperability of geospatial data;
- provide direct ("on-the-fly") and transparent access (platform and projection independent) to DIGEST-compliant data;
- conduct R&D in open frameworks; and
- provide worldwide data access through communications networks (Internet/Intranet, and both local- and wide-area networks).

OGDI was developed to meet the goals of DGIWG. The rest of this discussion focuses on the characteristics, capabilities, and availability of OGDI.

3. Open Geospatial Datastore Interface

What would remain of the geospatial data barrier if all systems could simply read all formats? Just as multilingual people communicate easily with one another, a "multilingual" system would have few problems with the multitude of data formats now in existence. Multilingual people can share information with each other without having to translate into their mother tongue. For example, if English-speaking people were to learn to *read* Chinese and vice versa, then English people wouldn't have to learn to *write* Chinese, and vice versa. OGDI is based on this "non-ethnocentric" approach. It reads different geospatial data formats directly, without translation or conversion.

3.1 Definition

OGDI sits between the application software (e.g., a GIS) and various geospatial data products. It follows the style of Open Database Connectivity (ODBC) [Microsoft 1994], an API for accessing databased data using Structured Query Language (SQL). Like ODBC, OGDI uses drivers to access each geospatial data format.

3.2 Main characteristics

OGDI is a collaborative and coordinated effort within NATO. It is based on the TCP/IP protocol and can use the Internet to distribute geospatial data products. OGDI facilities can be accessed either through a C or Tcl/Tk [Ousterhout 1994] API, called from Win32 or UNIX applications. OGDI provides tools for many geospatial data integration tasks, including:

- conversion of various formats into a uniform transient data structure,
- transparent adjustment of coordinate systems and cartographic projections,
- transparent transformation of platform-dependent representations,
- retrieval of geometric and attribute data.

3.3 Transient data model

The transient data model supports both geometric and attribute data. Geometric data can be vector (area, line, point, text) or raster (line or tile access). Metadata include geographic coverage, cartographic projection, and sources. Table 1 shows the C data structures used to describe vector elements in the transient data model. Table 1's "c_len" variable indicates the number of coordinates that describe a linear feature.

Point Feature	**Line Feature**	**Area Feature**

Point features are composed of one instance of ecs_Coordinate.
 	
struct ecs_Point {
ecs_Coordinate c;
};
 	
struct ecs_Coordinate {
double x;
double y;
};
 	
typedef struct ecs_Coordinate ecs_Coordinate;
 	
Note: the current model supports only 2D vector representations.
	Line features are composed of two or more coordinates. Line features must be homogeneous in direction.
	
	struct ecs_Line {
	struct {
	u_int c_len;
	ecs_Coordinate *c_val;
	} c;
	};
		Area features have one or more rings. Rings are similar to line features except that the last coordinate is always equal to the first.
		
		struct ecs_FeatureRing {
		ecs_Coordinate centroid;
		struct {
		u_int c_len;
		ecs_Coordinate *c_val;
		} c;
		};
		
		typedef struct ecs_FeatureRing ecs_FeatureRing;
		
		struct ecs_Area {
		struct {
		u_int ring_len;
		ecs_FeatureRing *ring_val;
		} ring;
		};

Table 1: OGDI Transient Vector Structure

3.4 Architecture

A dynamically loadable C language library is normally used to access OGDI facilities. First, an application that needs to perform geospatial processing calls functions through the C API and retrieves the results. Alternatively, a Tcl/Tk API can also be called from an application. Tcl/Tk (Tool Command Language/ToolKit) is a simple framework for building and accessing GUI applications via a scripting language in a shell-like environment [Ousterhout 1994]. Drivers are used to access various geospatial data formats, one for each format. A driver is also a dynamically loadable library that processes C language API requests for a specific datastore. Once a driver is loaded, it receives requests, fetches data from the datastore, translates the data into the transient data structure, and returns the results to the application. All the APIs are available for UNIX (Solaris and Linux) and for Windows NT and Windows 95. Figure 1 illustrates the components of the OGDI architecture.

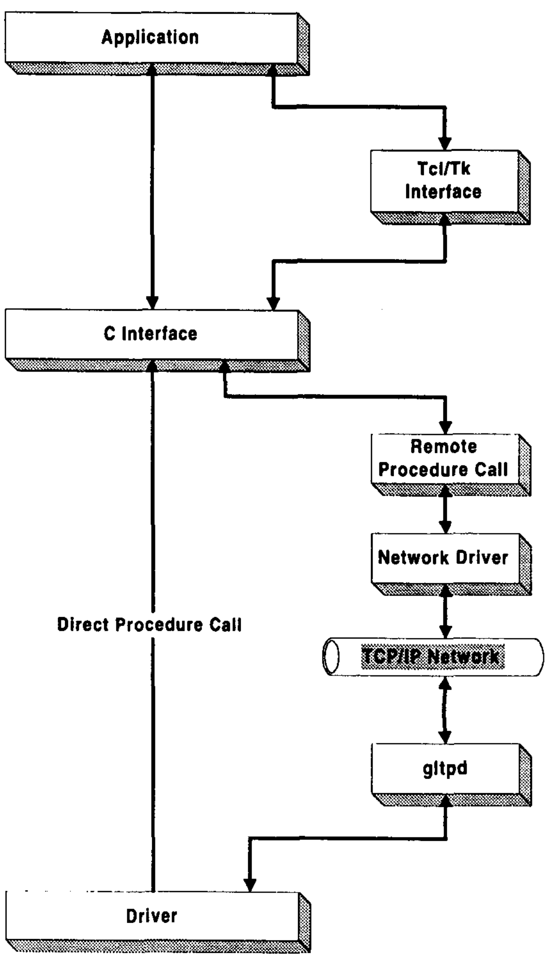

Figure 1: OGDI Architecture

Drivers can be accessed either directly, for local datastores, or remotely, for external datastores. For remote procedure calls, other modules are needed. The *gltpd* module (geographic locator transfer protocol deamon) is a small utility program that mimics the behavior of the C API on a remote computer. The network driver is a dynamically loadable library that relays calls from the C API to a *gltpd* process running on a remote computer. The *gltpd* process and the network driver are used together to link the application to a remote driver through a TCP/IP network, allowing the application programmer to access remote drivers as if they were local, using a client/server paradigm.

When the *gltpd* process receives its first request from an application, it creates a new thread (a fork) of itself, which loads the requested driver type, takes control of the communication process with the network driver, and serves all subsequent OGDI calls coming from the application. The combination of the *gltpd* process and a specific driver becomes a server to the client (i.e., the application's connection).

A new component-- *gltp*--is used by OGDI. It is similar to the *http* or *ftp* prefixes of the Web, and it gives direct access to a local database or remote access to an external database. The *gltpd* protocol was preferred to *http* because of the need for a statefull protocol. Stateless protocols (like *http)* process each query independently. OGDI required a protocol with "memory" to handle successive queries.

A programmer using OGDI would see no difference between a local driver and a remote one. The *gltpd* process and the network driver manage communication protocol transparently, and automatically transform data between incompatible processor architectures.

3.5 Available Drivers

OGDI offers drivers for the major NATO map data formats, including Vector Relational Format (VRF), Digital Terrain Elevation Data (DTED), and ARC Digitized Raster Graphics (ADRG). US National Imagery and Mapping Agency (formerly Defense Mapping Agency) has completed the Compressed ADRG (CADRG) and the Controlled Image Base (CIB) drivers.

Drivers for GRASS, Arc/Info, and PAMAP also are available. Drivers will soon be developed for other commercial formats, including GeoTIFF, Autocad DWG and DXF, Intergraph DGN, USGS DLG-3, and Mapinfo MID/MIF.

3.6 Using the API in an application

The C-language API is composed of functions that:

- manage and load the geospatial data driver;
- provide an entry point to OGDI functions for each driver;
- allocate storage for geometric and attribute data;
- perform "garbage collection" of previously allocated storage;
- validate parameters and sequences for OGDI calls;
- transform coordinates and projections.

A complete set of instructions for programmers can be found in [LAS 1996]. The underlying philosophy of OGDI is to encapsulate the tasks related to file access in one API. With OGDI, an application is isolated from the details of data access. The task of navigating through a datastore falls to the drivers; the C API retrieves information, regardless of data format.

To interact with a datastore involves the steps shown in Figure 2. The application establishes a connection, selects a geographic region and map layer, then extracts objects. It processes the results and finally terminates the connection.

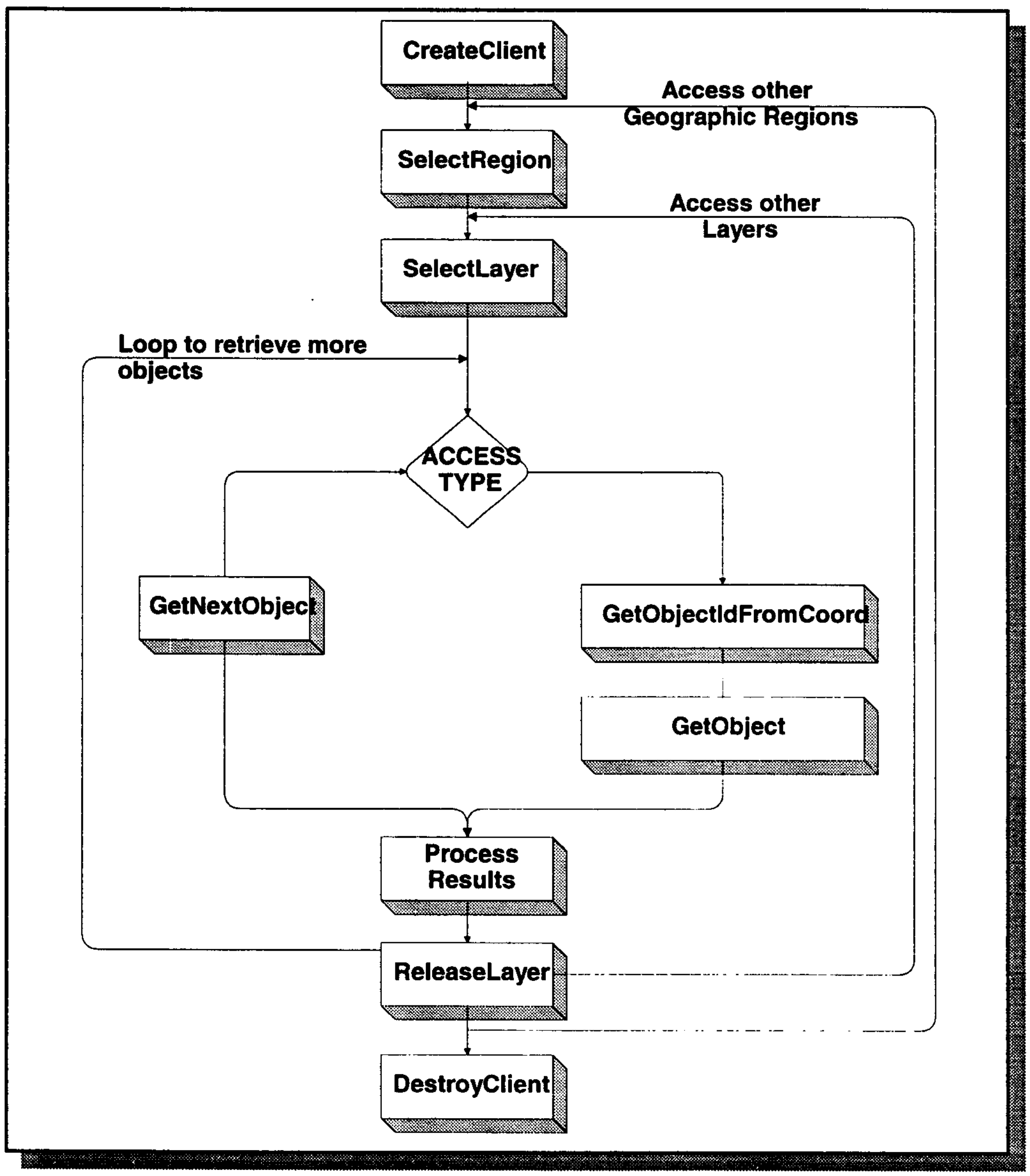

Figure 2: Using the OGDI API in an Application

4. Conclusions and future plans

The previous sections have described how OGDI provides direct and transparent access to a large number of geospatial data formats. Currently, OGDI provides

- retrieval functions for map data comprised of geometry and attributes;
- transparent adjustment of coordinate systems and cartographic projections;
- transparent transformation of platform-dependent representations;
- drivers for GRASS, VRF, ADRG, CADRG, CIB, DTED, Arc/Info, PAMAP;
- the use of the Internet as a medium to distribute geospatial data products.

In the near future, OGDI will be improved in the following ways:

- better remote communications;
- compression/decompression facilities to improve communication between client and server;
- encryption facilities;
- authentication services;
- spatial analysis services;
- new drivers for: MUSE, DXF/DWG, DGN, S-57, TIFF, GIF, PCX, BMP, DLG-3, SAIF, GeoTIFF, ERDAS, MID/MIF;
- a VPF/VRF product generator;
- a Web map-browser plug-in;
- linkages to spatial databases using complex data.

Documentation and source code for OGDI can be downloaded from:

- Defence Geomatics server: http://132.156.33.161/Engineer/Software_Tools/DIGEST_Software_Tools
- LAS server: http://www.las.com/ogdi

Acknowledgments

The authors wish to thank Mr. Dave McKellar, who was instrumental in the adoption of the DIGEST standard and in the elaboration of the GGI&S initiative and the DIGEST Software Tools project, and whose vision has led to many of the concepts in OGDI. The authors also thank the LAS developers for their contribution in the design and development of OGDI, and Gail Langran Kucera of Mercator Systems for her helpful review of this paper.

References

Ashworth, M. (Editor), 1996. ISO Working Draft SQL Multimedia and Application Packages (SQL/MM) Part 3: Spatial, ISO/IEC JTC1/SC21 N10441.

Buehler, K. and McKee L., 1996. The OpenGIS Guide, Open GIS Consortium Inc., OGIS TC Document 96-01,1996, http://www.opengis.org

DGIWG, 1994. CGIS-DIGEST Edition 1.2, Canadian General Standards Board , Ottawa.

ISO, 1996. Working Draft SQL3 Parts 1-8, Part 2: Foundation, ISO/IEC JTC1/SC21 N10489, Part 5: Bindings, ISO/IEC JTC1/SC21 N10490, Part 8: Object, ISO/IEC JTC1/SC21 N10491, documents DBL MAD-6 to MAD-13 or X3H2-96-295 to 302, June.

Lam, S., Achieving Geospatial Data Interoperability Through Data Warehousing, DREV Report in preparation.

LAS, 1996. Open Geographic Datastore Interface (OGDI), Version 1.0: Programmer's Reference, 145p.

McKellar, D., 1995. The Global Geospatial Infoma-tion & Services Initiative, Canadian Geomatics CD-ROM, June.

Microsoft Corporation, 1994. Microsoft ODBC 2.0 Programmer's Reference and SDK Guide: for Microsoft Windows and Windows NT, Microsoft Press, Redmond, Washington.

Ousterhout, J.K., 1994. Tcl and the Tk Toolkit, Addison-Wesley Publishing Company Inc., 458 p.