

17.3 INTEGRATING SCIENCE DATA INTO GEOGRAPHIC INFORMATION SYSTEMS

Ted Habermann*
NOAA/NESDIS, Boulder, CO
John Cartwright and Roland Schweitzer
CIRES, University of Colorado, Boulder, CO
Ian Barrodale and Eric Davies
Barrodale Computing Services Ltd., Victoria, BC

1. INTRODUCTION

Geographic Information Systems (GIS) provide well-developed tools for query, analysis and display of point, line, and polygon data, and there are well-developed communities that use these data types and rely heavily on GIS tools. Many meteorological datasets consist of combinations of time series and grids. Tools for query, analysis and display of these data types are less well developed in the geospatial community. Developing approaches to integrating these datasets into GIS is a critical step toward bridging the gap between science data and GIS. We describe our experience with three approaches that cover the spectrum of geospatial systems from desktops to distributed geodatabases.

2. OPENDAP GEOTIFF SERVER

The goal of the OPeNDAP GeoTIFF server is to make a large number of gridded science datasets available to the GIS community. OPeNDAP is a well known data access protocol formerly known as DODS. There are many gridded datasets being served by OPeNDAP servers all over the world (<http://www.unidata.ucar.edu/packages/dods/>). These servers make these datasets available using URLs and http.

The GeoTIFF format is a well known format for geolocated images. It extends the Tagged Image File Format by adding tags that include geolocation and projection information.

The OPeNDAP GeoTIFF server (<http://panther.ngdc.noaa.gov/SDG>) takes an OPeNDAP URL, retrieves the data from a remote server and provides a GeoTIFF file (or a set of files) to the user. This file contains the actual data values suitable for further analysis in the GIS.

This approach is useful for a desktop GIS user (or internet map server) that wants to view and/or analyze a science dataset in a GIS. They can create the GeoTIFFs they need once and download them to local disk to use as often as they need them. This approach also allows a science data provider with an OPeNDAP Server to

provide images for GIS users without translating all existing datasets or time slices into GeoTIFFs.

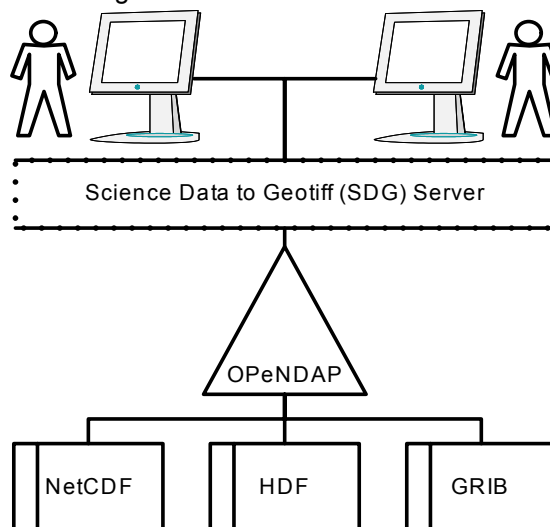


Figure 1. The Science Data to GeoTIFF Server.

3. ARCIMS DATA SOURCE

In a partnership with OPeNDAP and the Environmental Systems Research Institute (ESRI) we have developed a custom data source for ArcIMS that allows map service providers to include gridded science data sources from OPeNDAP servers in ArcIMS sites directly. The data appears as a layer in the internet map integrated with traditional geospatial data from local or distributed geospatial databases. In this case what is returned to the user is an image of the data rather than the data themselves. This approach is useful for groups serving maps using ArcIMS that want to integrate data from local or distributed OPeNDAP servers into their Internet maps.

4. GEOSPATIAL DATABASES

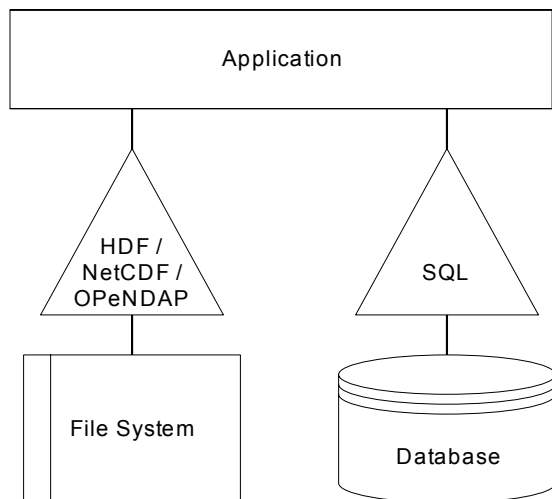
Most traditional scientific and geographic information systems store data in files and then access and manipulate data from those files. This approach has led to a proliferation of standard and non-standard file formats that are used by some

* Corresponding author address: Ted Habermann, NOAA / NGDC E/GC1, 325 Broadway, Boulder, CO, 80305, Ted.Habermann@noaa.gov

group of applications, and there are many difficulties in sharing data between these groups. In the GIS world, this problem was exacerbated by the fact that the formats were generally complex and proprietary (i.e., ArcInfo Coverages and Grids). The GIS applications included data access and management capabilities. (i.e., the info in ArcInfo).

Several years ago the developers of ArcInfo (ESRI) began to explore the potential of using relational databases to store spatial data rather than proprietary file formats. This simplified their programming requirements by allowing them to offload many of the data management capabilities onto the database systems. In addition, it increased their users' overall capabilities by allowing them to take advantage of features of the database systems.

A similar evolution has taken place in the development of Science Information Systems (SIS) with the emergence of standard formats (i.e. HDF, NetCDF, GRIB) and related application-programmer interfaces (APIs). These APIs hide the details of the formats and allow the community to share tools based on the format they are using (i.e., use Ferret and LAS to display gridded data from netCDF files). In this case the netCDF libraries provide the API to the data. The OPeDAP Libraries are an extension of the same concept to include independence from the underlying file format. In the database world Structured Query Language (SQL) provides the API.



The experiment with migrating spatial features into a database environment has gone very well for the GIS community. Many major database vendors are now incorporating spatial data types and operators into their core products. These products are generally referred to as geospatial

databases. There are a number of advantages to this approach. Some of these include:

Uniform treatment of data items. By storing all data types in the same framework, applications can use the same SQL interface to perform complex queries that are based on any of these data items, including dynamically-derived properties of the data.

Client-server issues. In a traditional database or file-based environment, custom functionality is implemented purely on the client side, resulting in "fat clients". Modern DBMSs, in addition to allowing user defined types (UDT), also allow user defined routines (UDR) to be created in languages such as C or Java, which provide much greater control over where functionality resides. Data can be assembled, processed and/or disassembled on either the client or the server, or both, with attendant tradeoffs.

Synchronized concurrent access to data. Multiple users of the database can safely query the same data simultaneously. With a file-based approach, there is a danger that the update activity of one application might result in another application seeing inconsistent data.

There are still a number of challenges to be faced in the geospatial database world. Some of these are related to polar data or spatial features that cross 180 degrees. There are also challenges related to the semantics of the data in the database. ESRI has developed a number of domain specific data models that include data objects and specify relationships between those objects to address these challenges. These models fill a similar role as COARDS and other semantic conventions in SIS. Finally, there are challenges related to managing gridded datasets in geospatial databases.

5. GRIDS IN GEOSPATIAL DATABASES

Integrating gridded data into geospatial databases has gone more slowly than has been the case with other data types. ESRI includes grid storage and access mechanisms in their Spatial Database Engine (SDE). In that case, the grids are implemented as ESRI data types in the database and SDE handles all grid operations. This allows SDE users to store grids in any of the underlying databases supported by ESRI.

Certain DBMSs include the capability to integrate gridded data more tightly into the database core. In particular, object-relational DBMSs provide the support for UDTs and UDRs required to realize these advantages. Barrodale Computing Services (<http://www.barrodale.com>) used such support to develop a Grid DataBlade for use with

the IBM Informix object-relational DBMS. Using this DataBlade involves ingesting gridded data into the database and accessing it using spatial extensions to Structured Query Language (SQL).

An example of this would be the following pseudo-SQL query, which returns all pressures at a depth of 100 m, collected by a particular device:

```
select GRDExtract(gridColumn, "extraction_specification")
from tablename
where inside(100, depth_range)
and collection_device = "pressure_transducer";
```

This basic query could be extended by modifying the text of the extraction specification to support more refined requests, such as selecting pressures collected during the last five hours, converting to a Lambert conformal projection, etc. In contrast, gathering this information when the gridded data is on a file system would involve a complex application of not only queries but also code to locate, transform and subset the relevant data.

Other features of the Grid DataBlade are that it:

- is designed to handle 4D (and "5D") grids as a standard. Grids of lesser dimensionality are handled by taking one or more of the grid dimensions to be 1.
- stores grids using a tiling scheme in conjunction with random-access BLOBs, with user control over the tile size. This allows very efficient generation of data products that involve only a small portion of the data in the overall grid (e.g., extracting planar oblique slices from volumetric grids, or retrieving 1D probes or sticks from 3D or 4D grids).
- can store the data in, and convert it between, more than 40 different planar mapping projections supported by the IBM/Informix Spatial DataBlade (the second demo provided at http://www.barrodale.com/grid_Demo uses the Grid DataBlade and the Spatial DataBlade to resample and re-project a 386 MB raster image of the World).
- can handle irregularly spaced grids in any or all of the grid dimensions.
- can handle the presence of multiple vector and/or scalar values at each grid point.
- provides interpolation options using N-linear or nearest-neighbor schemes and allows users to insert custom interpolation schemes.
- provides for convenient loading and extraction of grid files to/from the database via a specific form of the commonly used NetCDF format, termed Grid Import-Export Format (GIEF).

- provides application programming interfaces for C, Java and SQL.
- often provides more than 50-fold increases in speed of data product generation compared to the conventional approach that does not involve tiling or SmartBLOBs.
- is supplied with full user/programmer documentation.

6. NOAA AND THE GRID DATABLADE

Many offices in NOAA create and manage gridded data products and model results using traditional file based frameworks with formats developed for particular applications. Others use standard formats and access tools (i.e. NetCDF, HDF, GRIB, OPeNDAP). Many offices in NOAA are also using geospatial databases for managing point, line, and polygon datasets.

The Geospatial Data Services Group at NOAA's National Geophysical Data Center is working with Barrodale Computing Services and NOAA's Comprehensive Large Array-data Stewardship System (CLASS) is to test the Grid DataBlade for NOAA applications. This project will initially involve gridded satellite products created by NESDIS.

As part of this effort, Barrodale Computing Services has developed the capability to load COARDS compliant NetCDF files into the Grid DataBlade. This capability will allow CLASS to explore and apply Grid DataBlade capabilities along side of existing gridded product archives. Future developments may include the capability to insert data extracted from remote OPeNDAP Servers into the Grid DataBlade and to extract grids from the DataBlade using the Open GIS Consortium Web Coverage Server specification.