[DM-01-062] Database Administration

Abstract

Organizations with a responsibility for maintaining large-scale, multi-user spatial databases often turn to server-based relational database management systems to achieve their goals. The administration of such databases has many dimensions. Industry standards in the areas of data storage and services should be researched and applied to ensure a sound, comprehensive database design as well as to promote interoperability with external entities. Data validation tools should be implemented to improve the accuracy and efficiency of data maintenance activities. Metadata should be maintained according to industry standards to protect the organization's investment in data and to increase the likelihood of the data being located by clearinghouse and portal search tools. Database security strategies can prevent unauthorized access to data and lessen the chances of data loss due to accidental data corruption. Database performance should be monitored and strategies implemented to ensure that data can be retrieved from the system with acceptable response times. Finally, trends in the field such as the increasing need to manage large volumes of data call for spatial database managers to be knowledgeable of non-relational data models as well, such as NoSQL data models.

Keywords: database management, domains, indexes, metadata, performance, relational database, roles, security, standards, versioning

Author & citation

Detwiler, J. (2022). Database Administration. The Geographic Information Science & Technology Body of Knowledge (4th Quarter 2022 Edition). John P. Wilson (Ed.).
DOI: 10.22224/gistbok/2022.4.1

This Topic is also available in the following editions:

DiBiase, D., DeMers, M., Johnson, A., Kemp, K., Luck, A. T., Plewe, B., and Wentz, E. (2006). Database administration. The Geographic Information Science & Technology Body of Knowledge. Washington, DC: Association of American Geographers.

Explanation

1. Definitions

Database administration – duties performed by an organization member with greater privileges than other members with the goal of optimizing the database's accuracy, security, performance, and ability to share with external entities

Domain – collection of values that a database element may contain

Metadata – information on the who, what, when, where, why, and how questions a potential end user may have about a dataset

Versioning – a database capability that enables users to create multiple copies of a dataset for certain use cases, such as enabling multiple concurrent editors of the data or handling data in stages like creation, quality control, production

2. Relational Database Management System Software

GIS software packages often use relational database management system (RDBMS) software for storing data, particularly for large-scale and long-lived projects (sometimes referred to as enterprise databases). For example, both ArcGIS and QGIS support usage of several RDBMS's, including the proprietary Microsoft SQL Server, Oracle, and SAP Hana, along with the open-source PostgreSQL and its spatial extension PostGIS.

All these RDBMS packages offer special data types for storing geometry. For example, both SQL Server and PostGIS support a geometry data type for representing data in a Euclidean coordinate space and a geography data type for representing data in a spherical coordinate space. Objects of these data types serve as inputs to a collection of functions written specifically for working with geographic data. A sampling of these functions is shown in Table 1.

Table 1. Commonly Used Functions in PostGIS, a Popular Open-Source Spatial RDBMS

Function	Description
ST_Length	Returns the length of a linear geometry
ST_Area	Returns the area of a polygonal geometry
ST_X, ST_Y	Returns the X and Y coordinates of a point
ST_Transform	Reprojects a geometry into a different spatial reference system
ST_Contains	Returns True if geometry B is inside geometry A
ST_Distance	Returns the distance between the two geometries
ST_Intersection	Returns the shared portion of geometries A & B
ST_Buffer	Creates a buffer polygon of a specified radius

Functions like these are invoked within SQL statements. Figure 1 illustrates the ST_Distance function being used in a query to identify features in a cities table within 100 km of some point.

```
SELECT cities.name
WHERE ST_Distance(cities.geom, ST_Point(-75.16, 39.95)) <= 100000;
```

Figure 1. An example of a SQL statement that invokes a spatial function. Source: author.

For some organizations, it is possible to build and maintain a GIS without dedicated GIS software, using only command-line SQL together with RDBMS spatial data types and functions like those described here.

3. Using Standards

Data standards exist to promote the ability of organizations to make accurate and efficient use of their data internally and to aid in the sharing of their data externally. Notable entities that have produced standards for the handling of geographic data include the US Federal Geographic Data Committee (FGDC) and the international Open Geospatial Consortium (OGC). Geospatial standards can be categorized as to whether they concern data storage, metadata, or services.

3.1 Data Storage Standards

This category of standards involves the initial creation and ongoing maintenance of a dataset. Managers of spatial databases should research standards that exist within their own organization (for example, a county department database manager taking into account rules and conventions that exist county-wide), but they should also consider adopting authoritative standards that have developed outside the organization. The search for standards should also include consideration of both the spatial and non-spatial components of the data. For example, an organization involved in measuring water quality would be wise to consult resources on the maintenance of water quality data.

Given the importance of street addresses in geo-locating data, the FGDC developed the United States Thoroughfare, Landmark, and Postal Address Data Standard (FGDC 2011). The major parts of this standard are concerned with the topics of address data content, classification, quality, and exchange. It offers a simple, yet instructive example of the benefit of consulting a standard. Virtually everyone is familiar with street addresses and many professionals may assume they have sufficient knowledge to define a set of data fields to store them. Looking at the standard, however, it is likely to reveal possibilities that hadn't been considered (e.g., the difference between situs and postal delivery addresses or that an address number may contain letters or fractions). Table 2 provides a non-exhaustive list of spatial data storage standards across a variety of disciplines.

Table 2. A Sampling of Geospatial Data Storage Standards in the United States

Standard	Managing Organization
Federal Information Processing Series (FIPS)	American National Standards Institute (ANSI)
Geographic Names Information System (GNIS)	American National Standards Institute (ANSI)

School Locations & Geoassignments	Dept of Education
National Geological Map Database	US Geological Survey (USGS)
National Shoreline Data Content Standard	Nat Oceanic & Atmospheric Administration (NOAA)
Cultural Resource Spatial Data Transfer Standard Data Model	National Park Service (NPS)
Wetlands Mapping Standard	Fish & Wildlife Service (FWS)
National Vegetation Classification Standard	US Forest Service (USFS)
Earth Science Data Systems	National Air & Space Administration (NASA)
Geospatial Intelligence Standards	Department of Defense (DoD)
Spatial Data Standards for Facilities, Infrastructure, and Environment	Department of Defense (DoD)
Geog Info Framework Data Standard: Part 1, Cadastral	Bureau of Land Management (BLM)
Geog Info Framework Data Standard: Part 2, Orthoimagery	US Geological Survey (USGS)
Geog Info Framework Data Standard: Part 3, Elevation	US Geological Survey (USGS)
Geog Info Framework Data Standard: Part 4, Geodetic Control	National Geodetic Survey (GDS)
Geog Info Framework Data Standard: Part 5, Govt Unit Boundaries	US Census Bureau
Geog Info Framework Data Standard: Part 6, Hydrography	US Geological Survey (USGS)
Geog Info Framework Data Standard: Part 7, Transportation	Dept of Transportation (DoT)

3.2 Geospatial Service Standards

The discussion above focused on the storage of data for internal purposes. However, today's cloud and server-based application architectures require a separate set of standards related to the configuration of geospatial services. These standards promote the idea of sharing data outside the organization, so that it can be integrated with other data sources across a range of platforms more easily (i.e., the idea of interoperability).

A good example of such a standard is the General Transit Feed Specification (GTFS), which was initially developed by a Google engineer in collaboration with IT managers from the City of Portland, Oregon, out of a need for data to support trip planning by modes beyond just cars. Due to its widespread adoption by transit agencies around the world, trip planning application development and analyses can be carried out using a single solution rather than separate solutions for each agency. Table 3 provides a non-exhaustive list of geospatial service standards.

Table 3. A Sampling of Geospatial Service Standards.

Standard	Managing Organization
Web Map Service (WMS)	Open Geospatial Consortium (OGC)
Web Feature Service (WFS)	Open Geospatial Consortium (OGC)
Web Map Tile Service (WMTS)	Open Geospatial Consortium (OGC)
Keyhole Markup Language (KML)	Open Geospatial Consortium (OGC)
Geography Markup Language (GML)	Open Geospatial Consortium (OGC)
Geographic Javascript Object Notation (GeoJSON)	Internet Engineering Task Force (IETF)
GeoPackage	Open Geospatial Consortium (OGC)
Network Common Data Form (netCDF)	Open Geospatial Consortium (OGC)

CityGML	Open Geospatial Consortium (OGC)
WaterML	Open Geospatial Consortium (OGC)
GeoSciML	Open Geospatial Consortium (OGC)
IndoorML	Open Geospatial Consortium (OGC)
SensorML	Open Geospatial Consortium (OGC)
PipelineML	Open Geospatial Consortium (OGC)
Mobile Data Specification	Open Mobility Foundation
Indoor Mapping Data Format (IMDF)	Apple / OGC

4. Data Access Processes in a Spatial Database

When spatial data are stored in a RDBMS (e.g., in an Esri enterprise geodatabase or a PostGIS database), some concepts are shared in common with regards to configuring user access. Figure 2 depicts an imaginary municipal government with two departments, each staffed by different sets of database users:

Planning	Water
UserA	UserC
UserB	UserD

Figure 2. Database users in an imaginary municipal government. Source: author.

Separate schemas might be created within the municipal database to store the data maintained by each department, as shown in Figure 3.

```
CREATE SCHEMA dept_planning;
CREATE SCHEMA dept_water;
```

Figure 3. SQL code for creating schemas for each government department. Source: author.

While database privileges can be assigned on a user basis, it is often more convenient to define roles and assign privileges based on role. In Figure 4, new roles are created for each department with each role being given read/write privileges for data in their own department's schema and read-only privileges for data in the other department's schema.

```
CREATE ROLE role_planning;
CREATE ROLE role_police;
GRANT USAGE ON SCHEMA dept_planning, dept_water TO role_planning,
role_water;
GRANT SELECT ON ALL TABLES IN SCHEMA dept_planning TO role_water;
GRANT SELECT ON ALL TABLES IN SCHEMA dept_water TO role_planning;
GRANT SELECT, INSERT, UPDATE, DELETE ON ALL TABLES IN SCHEMA dept_planning
to role_planning;
GRANT SELECT, INSERT, UPDATE, DELETE ON ALL TABLES IN SCHEMA dept_water to
role_water;
```

Figure 4. SQL code for establishing roles with read/write permissions for data in users' own departments and read-only permissions for data maintained by other departments. Source: author.

Users can then be assigned to roles (see Figure 5) and will inherit the privileges belonging to their roles:

```
GRANT role_planning TO UserA, UserB;
GRANT role_water TO UserC, UserD;
```

Figure 5. SQL code for assigning database users to roles giving them appropriate data access. Source: author.

As with other operations, software vendors often provide point-and-click user interfaces for carrying out the tasks outlined in this section.

5. Data Validation and Verification

Spatial database administrators have a number of options for improving data integrity and usability:

- Default values can be assigned to data columns where appropriate.
- Domains/constraints can be used to limit the entries made in columns. Numeric columns can be limited to values between a specified minimum and maximum; textual columns can be limited to a list of allowed values.
- Subtypes can be used to model differences across categories of real-world features. For example, a roads dataset might have subtypes such as highway, arterial, local, etc. Each subtype can have its own set of default values and domains allowing for differences in attributes like number of travel lanes, flow directions, pavement types, etc.
- For administrators in organizations that perform a lot of data creation work, it is wise to consider a technology stack that offers support for versioning. This allows editors working simultaneously on the same layer to create their own copies of the data that are isolated from their colleagues'. An administrator can then periodically import edits

from child versions into the parent version through a “post” operation. In the event that a post would result in replacement of an already existing feature, the administrator has the ability to compare the two conflicting features (both their geometries and attributes) and decide which to keep in the layer and which to discard. The editors can also ensure that their child versions are kept in sync with the base by completing a “reconcile” operation. In addition to its benefits in concurrent editing scenarios, versioning also offers organizations a convenient means of conducting quality control on new edits prior to exposing the data to internal business processes or to the external world.

6. Metadata

An important, if not particularly sexy, task in database maintenance is compiling metadata, a documentation of key aspects of the data. The purpose of metadata is to answer the who, what, when, where, why, and how questions a potential end user of a dataset might have. Effective metadata greatly improves the likelihood that a dataset can be discovered through clearinghouse and portal search tools and ultimately put to use. Compiling metadata also protects an organization’s investment in data (e.g., in the event of a loss of key personnel).

The Content Standard for Digital Geospatial Metadata (CSDGM, FGDC 1998), a widely used metadata standard authored and endorsed by the FGDC in the 1990s, was one of the first major metadata standardization initiatives. The CSDGM organizes metadata information into 7 main categories, shown in Table 4.

Table 4. Sections of the Content Standard for Digital Geospatial Metadata

Section	Information Covered
Identification Information	creator, area covered, thematic topic, currency
Data Quality Information	spatial / non-spatial accuracy, completeness
Spatial Data Organization Information	how location is conveyed (e.g., vector, raster, street addresses)
Spatial Reference Information	spherical/planar coordinates, projection, datum
Entity and Attribute Information	which real-world objects are depicted, using what attributes
Distribution Information	how the data can be obtained
Metadata Reference Information	who compiled the metadata and when

The International Standards Organization (ISO) developed its own geospatial metadata standard in the 2000s, ISO 19115 (ISO 2014). This standard is not very different from the CSDGM, but represents the consensus of the international community rather than of US federal agencies. It has evolved since its original publication into a multi-part suite of standards. ISO 19115-1, Geographic information - Metadata - Part 1: Fundamentals, is the base standard. It was built upon by ISO 19115-2, Geographic information - Metadata – Part 2: Extensions for Imagery and Gridded Data, and later, ISO 19115-3, Geographic information - Metadata - Part 3: XML schema implementation of metadata.

Given its more recent development, the ISO standard reflects advances in geospatial technology, supporting documentation of a wider array of resources, including feature and

map services, sensors, data collection methods, application schemas, and more. The FGDC has endorsed ISO 19115, recommends implementing it as the preferred standard for new databases, and recommends translating CSDGM metadata into ISO for existing databases where feasible. Most GIS and image processing applications offer tools for maintaining metadata. Many other metadata authoring tools may be found online.

7. Database Security

Spatial databases, like non-spatial ones, are vulnerable to a host of potential threats. These include access by unauthorized parties, misuse by authorized parties, physical damage to servers due to causes like fires/floods/lightning, and accidental data corruption.

Network security measures maintained by system administrators such as firewalls and intruder detection systems provide one layer of protection against some of the threats noted above. Database security measures – user identification, authentication, and privilege management (discussed in the Data Access section above) – provide an additional layer of protection. Database administrators should consider conducting regular security vulnerability assessments, especially when handling highly sensitive data or data that would be difficult to replace.

Regular replication of the database (i.e., creating a backup) is a smart practice since even the best security practices sometimes fail to prevent corruption. Storing backups in a different building or geographical region is advised. Automation is possible for both vulnerability testing and backup production.

8. Database Performance

An important task for spatial database administrators is ensuring that the database performs at an acceptable level. This is especially the case for databases that serve data used by public facing applications.

At the hardware/software level, database performance can be improved through keeping up to date on software versions and patches, reallocating the server's memory reserves, and deploying database clusters. At the level of the database itself, several administrative strategies exist for optimizing performance. Like indexes in a book aid in locating topics, database indexes can shorten the time needed to locate records requested by a query.

Spatial databases can have both attribute indexes and spatial indexes. Consider a database table with a column that holds the names of cities and a query that searches for a particular one (e.g., having a SQL WHERE clause of "WHERE city = 'Philadelphia'"). Building an index on the city column allows the RDBMS to begin its search at a particular row of the table rather than starting at row 1 and doing a full table scan.

Spatial indexes similarly work by eliminating some features from consideration when carrying out spatial queries. A commonly used spatial indexing method is the grid method, which is analogous to the map index found in road atlases. A grid of equal-sized cells is laid over the layer's geometries and each row and column of the grid is assigned an identifier.

Geometries in the layer are compared to this grid and a list of grid cells intersected by each geometry is produced. A query like “find all cities in the state of Pennsylvania” can begin with the software looking up the grid cells intersected by the Pennsylvania polygon. It can then throw out any cities points that don't intersect those same grid cells. It only needs to test for containment on points that share grid cells with the Pennsylvania polygon.

In addition to the grid indexing method, other spatial indexing methods implemented in geospatial software include the R-tree, B-tree, and Generalized Search Tree (GiST) methods. Database administrators grappling with poor spatial query performance should consider experimenting with different spatial indexing methods from what the software implements by default.

Query optimizers built into DBMS’s use database statistics to determine the most efficient way to obtain the requested results. These database statistics include information like the number of rows in a table, number of distinct values in its columns, distribution of values, etc. As a table's data changes over time, the statistics will become out of date and less helpful in optimizing performance. Thus, an important job for administrators is to ensure that database statistics are recalculated, at least after major edits are carried out. One option to consider is automating the calculation of database statistics so that it happens on a regular basis (along with the re-building of indexes, which can also become out of date over time).

9. Beyond Relational Data Models

The traditional relational database has served the geospatial industry well for most use cases. However, developments such as the advent of big data and the increased growth in mobile apps have revealed shortcomings in the relational data model. Many geospatial applications now require faster query responses, greater scalability, and more easily changeable data schemas.

NoSQL (Not only SQL) databases developed in the broader IT world in response to these challenges. These databases offer simpler designs, which translate to easier scaling to clusters of machines. The data structures used in NoSQL databases offer greater schema flexibility and make some operations faster. Notable compromises are that NoSQL databases are generally not as well suited to modeling complex relationships among multiple entities and do not offer a standard query language for data retrieval like SQL.

The major categories of NoSQL databases include key-value store, document store, wide column store, and graph. A key-value store represents data as a collection of key-value pairs, such that each key appears no more than once in the collection. A document store encodes data in some standard format such as XML or JSON. A wide column store uses tables, rows, and columns, but the columns are dynamic, varying from row to row in the same table. Finally, a graph database is based on graph theory and prioritizes the relationships among data items. The data items are referred to as nodes, while the relationships are termed edges. A few of the more popular NoSQL databases are MongoDB, Cassandra, and HBase. Guo and Onstein (2020) provide an informative review of NoSQL databases applied in a geospatial context.

One concept that is worth keeping in mind with regard to database administration is that of

polyglot persistence. The idea is that administrators should not limit themselves to choosing between an RDBMS and NoSQL database, and should instead consider employing multiple data storage technologies simultaneously, using each to solve whatever problem it is best at to achieve the organization's goals.

References

Federal Geographic Data Committee (FGDC). (1998). Content Standard for Digital Geospatial Metadata (FGDC-STD-001-1998). FGDC: Washington, D.C.

Federal Geographic Data Committee (FGDC). (2011). United States Thoroughfare, Landmark, and Postal Address Data Standard. FGDC Document Number FGDC-STD-016-2011.

Guo, D., & Onstein, E. (2020). State-of-the-Art Geospatial Information Processing in NoSQL Databases. International Journal of Geo-Information, 9 (5), 331.

ISO (International Standards Organization) (2014). ISO 19115-1:2014 – Geographic information – Metadata – Part 1: Fundamentals. Accessed Aug 8, 2022.