A FRAMEWORK FOR MAINTAINING A MULTI-USER GEODATABASE: AN EMPIRICAL EXAMPLE

Muniru Ayotunde Owoola PhD

GIS Manager, Maltais Geomatics Inc. 17011 – 105 Avenue, NW Edmonton, AB T5P 4L4 Canada
tunde@maltaisgeomatics.com

KEY WORDS: Database, GIS, Versioning, Infrastructure, Updating

ABSTRACT:

Utility companies often collect a vast amount of geospatial data to support ongoing operations. Storing the datasets within a corporate database management system and using GIS technology to manage geospatial data is fast becoming the norm in this industry sector. However, as the size of corporate databases continues to grow, the need to develop data update and maintenance protocols that streamline corporate database management is becoming more evident. Software and hardware vendors are coming up with innovative solutions that attempt to resolve the issues involved, with varying degree of success. In this regard, increasing use is being made of hardware storage options, and software strategies such as versioning, replication, data integrity rules and metadata management.

This paper presents a framework for updating and maintaining multi-user geospatial databases with emphasis on electric utility data management. The paper argues that a combination of streamlining applicable internal workflow processes and selecting the appropriate technology solutions is often the panacea for developing a geospatial database update and maintenance strategy. The paper shows that keeping corporate database "evergreen" requires (i) investments in appropriate hardware and software solutions; (ii) developing geospatial database maintenance protocols using native software functionality where possible, and (iii) streamlining internal work-flow processes to seamlessly integrate with software process requirements.

1. INTRODUCTION

Data collection, update, retrieval and usage are critical operational components in most electric utilities. However, it is only in fairly recent times that full attention is being given to GIS as a technology that can provide an integrative platform for workflow processes related to:

- data gathering, integration and processing
- data storage and update requirements
- enterprise-wide/distributed data mining
- production of topographic products and their customized presentation, analysis and interpretation for decision making purposes

Data collected by a typical electric transmission company will normally include among others, information on Substations, Circuit, Overhead Structures, Transmission Towers, Airbreaks, Substations etc. Each of these data types has associated characteristics that need to be continuously accessed, updated, and maintained in a production environment to support ongoing operations. Moreover, a huge amount of resources is often expended on collecting and maintaining landbase data such as cadastral data, parcel ownership and so on; in addition to orthophoto imagery and similar raster data.

For utilities that have adopted GIS as a platform for spatial and attribute data management, there has been a gradual move away from compartmentalized and distributed model of data storage to a more centralized, enterprise-wide data storage. Although the advantages of centralized data repository far out-weight its disadvantages, it is becoming increasingly evident that careful considerations must be given to data management and update issues when adopting the centralized data storage option in this new multi-user, multi-edit environment.

This paper examines the key components that streamline data management tasks confronting utilities adopting enterprise GIS database architecture. Many of the same issues discussed in this paper can be extended to, and are indeed applicable to the vast majority of organizations utilizing GIS as a basis for storing, managing and analyzing spatial data.

The paper adopts a "cross-platform" approach to discuss the core requirements for developing a completely streamlined spatial database update and maintenance strategy. The approach effectively integrates hardware and software solutions with business driven processes. The ultimate objective is to show organizations how to address and resolve many of the potential frustrations likely to be experienced with using enterprise-wide consolidated data repository. This will in turn allow such organizations to maximize returns on investments in GIS technology for data management and application development purposes.

2. DATA MAINTENANCE REQUIREMENTS

Developing an effective data maintenance approach should start by first identifying factors that are most germane to those charged with day-to-day maintenance of corporate data infrastructures. These factors include, amongst others:

1. Eliminating error propagation within enterprise databases
2. Representing/structuring data maintenance strategies to replicate existing work-flow processes and flawlessly integrate with IT protocols
3. Providing multiple-editing, versioning, transactional, capabilities, and a highly secured environment for data sharing and update

4. Developing adequate metadata on enterprise-wide database to assist with making data currency and accuracy assessment analysis

A cursory examination of these four major factors will show that they are related and are best addressed by a combination of prudent technology (software) choices, hardware architecture and functional/effective organizational IT practices. The major components of this approach are discussed in the remaining sections of this paper. The technology component uses ESRI's ArcGIS 8x/ArcSDE 8x technology and Microsoft SQL 2000 RDBMS and Microsoft .NET suite of development environments. The hardware architecture component is flexible, and can be addressed on a case-by-case basis, while the overall IT best-practices are very critical to the success of the whole approach.

3. TECHNOLOGY SOLUTIONS FOR MAINTAINING SPATIAL DATABASE INFRASTRUCTURE

It is reasonable to expect that any adopted technology for developing an enterprise spatial database infrastructure should natively integrate functionalities for database maintenance. While this is often the case, the generic nature of most solutions makes it necessary for specific utilities to design and utilize protocols and procedures that best resolve their specific data maintenance requirements.

One of the lessons learnt in the empirical implementation discussed in this paper is the fact that buying into a technology solution is probably the easiest aspect of adopting an enterprise GIS; making the solution work for existing workflow processes that have evolved and have been standardized over time is the most difficult aspect of systems integration.

This paper is based on an empirical implementation of ESRI's geodatabase data model for developing an electric utility database and data maintenance infrastructure that addresses in varying degrees, the four factors earlier identified.

The geodatabase model supports the physical storage of spatial data inside a DBMS (e.g. SQL 2000) while also supporting transactional views of data (versioning), objects with attributes and behavior. These characteristics highlight the concept of intelligent GIS data, simplifying data management tasks, and allowing for more meaningful use of data.

With this architecture, multiple users can **access, share and edit** GIS data through the use of **versioning** and **long transactions** subject to DBMS permissions and GIS administration tools. Differences between versions of the database can be **reconciled** and the master version update with the reconciled version. Support for **disconnected editing** (data check-out and check-in) provides a platform for integrating field-based data collection and reconciliation with production data.

Key attributes of edited database objects are also automatically maintained e.g. shape area and shape length. Moreover, this model offers support for intelligent features, **rules and relationships**, allowing users to define topological and associative relationships/rules that determine how database objects interact, how they are edited and how referential integrity is preserved.

Data security and protection are also features of the software platforms being discussed, although security is best managed jointly with the hardware infrastructure. Nevertheless, at the software level, additional data protection is available by specifying validation rules, network connectivity rules, relationship rules and custom rules. Specifying these rules further enables the GIS systems administrator to maintain database integrity.

We will now discuss the features of the geodatabase model and how data maintenance tasks are streamlined using this model.

3.1 Multiple Data Access, Sharing and Updating

One of the most useful functionality of an enterprise geodatabase is the ability to support multiple users, sharing and editing the same data objects at the same time. Data sharing could involve a simple task as simply displaying the same data layer on multiple users' desktops, or more complex task like querying the same data object and retrieving records by multiple users.

Similarly, it is possible for multiple users to be editing the same data object (layer). These edits have to register in the database without the possibility of corrupting the database or making it unstable for other users who may not be necessarily carrying out edits to the database. This is where versioning and long transaction strategies come into play. Some practical scenarios are depicted in Figure 1 and Figure 2

3.2 Versioning and Long Transaction

One of the ways that a database can be effectively maintained is to grant specific editing privileges to certain users by the DBMS/ArcSDE administrator. This means that only users with appropriate read-write access can edit the database. Versioning is used to track multi-user editing, post edit conflicts and reconcile database versions. Un-registering the enterprise geodatabase as a versioned prevents all users from editing it through the clients application.

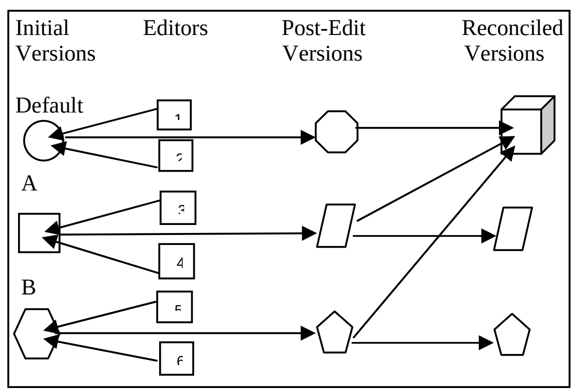

Figure 1. A Typical Versioning Scenario

Locking is also used to enforce data integrity during editing, and three types of locaks are available for this purpose. These are schema lock, share lock and exclusive lock. Schema lock ensures that the geodatabase structure cannot be modified when users are currently connected to the database. An exclusive lock is required to alter the structure (schema) of a geodatabase and once acquired, no share lock can be applied. Only the geodatabase owner can alter schema. Share edit locks are acquired when users are querying or editing a feature class or table.

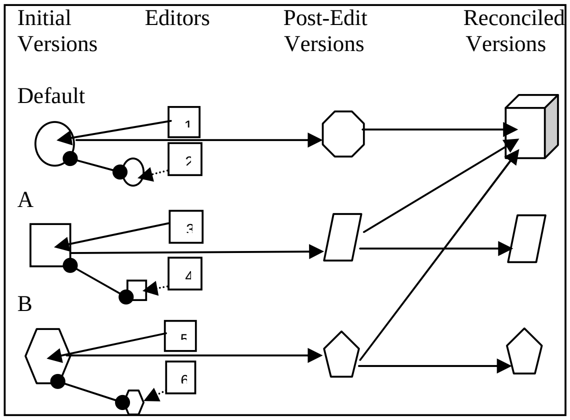

Figure 2. A Disconnected Editing Versioning Scenario

Versioning strategies have to be well thought-out, and should closely replicate quality assurance/quality control (QC/QA) procedures within an organization.

3.3 Disconnected Editing

Another feature of database maintenance in the empirical implementation is the use of disconnected editing functionality. A corporate database is constantly changing in a production environment. In the utility sector, field crews and office staff are constantly collecting new data and updating database contents respectively. To ensure that the database update process is as seamless as possible, three factors are addressed in our implementation:

1. The new data must be structured to the same specifications as the data it replaces.
2. The links to unchanged elements must not be broken.
3. Links to new or modified features should be automated as much as possible to avoid error and maintain data integrity.

To address these factors, disconnected editing approach was adopted. Disconnected editing is a new concept in spatial database editing and updating. It pre-supposes that a "snapshot" of a database can be taken to the field and edits and changes made to the snapshot in a remote location(s). In this scenario, it is also possible for the production database to be undergoing changes at the same time as a result of editing from other users (Figure 2). With prudent versioning architecture, changes made at remote location(s) can be reconciled with changes made to the production database. Conflicts are assessed and addressed by the database administrator or whoever has the final authority to make the decision. Once conflicts have been resolved, changes to the disconnected database are posted to the production database.

3.4 Rules and Relationships

While data update generally refers to actions that are executed to input new data, delete existing data or change database schema, in contrast, rules are often system specific, and are designed to impose constraints on certain actions when the database is being modified by the users.

Rules and relationships between database objects were specified apriori at the database development and population phase. Rules help governs how database objects will interact with each other and also impose restrictions on how changes can be made to the objects and what happens when those changes are made.

Besides, custom database functionalities are largely dependent on established rules. For instance, triggering built-in-relationships may be determined by the sequence of actions that an editor performs. Moreover, rules may govern feature class/table relationships in an edit mode (Table 1).

Rules, when used in conjunction with other data integrity routines, especially a geometric network, provide powerful tools for establishing data integrity especially to ensure that positioning of facility data and spatial associations between them are not compromised when features are being edited.

A geometric network is a one-dimensional non-planner graph primarily designed to store connectivity information between spatial features in a geodatabase. Once in place, a geometric network preserves the database and eliminates data corruption associated with data editing operations. In developing the geometric network for the database (Table 2), careful consideration is given to the following factors:

- The network rules must support the database without any major change(s) to existing schema
- The rules should be simple enough to understand, and should reflect accepted standards for modeling typical electric networks
- The geometric network must support in-built front-end functionalities

ARCSDE OBJECT	**FIELDS**	**MAINTENANCE RULE**
	FEATURE CLASS	
Circuit	Pri_Circuit#	Null or empty value not permitted if a line is added to the GIS
	Sec_Circuit#	Null or empty values permitted for single circuit lines, otherwise not permitted
	TABLE	
Secondary Circuit Info	Sec_Circuit#	Null or empty values not permitted if a double circuit line is added to the database
	Circ_OID	This is the foreign field for the S***econdary Circuit Info*** relationship. Auto Concatenate PRI_CIRCUIT# and SEC_CIRCUIT# fields to populate this field. If DC segments are > 1, add A, B, C etc after the line number to differentiate each segment

Table 1: Maintenance Rules for Feature Classes and Tables

Role/Rule	Network Feature	Properties
Complex edge	Circuit	
Simple Junctions	Structures Airbreaks	
Snapping	Circuit Structures Airbreaks	Only Circuit feature class can move and snap to other features during the network build process
Snapping tolerance	Circuit Structures Airbreaks	An optimal snapping tolerance is required for the build process
Sources and Sinks	Structures	Sources and Sinks ancillary roles are applied to all identified structures based on the Str. #. Str. # = 0 are sources, Str. # = xxxxS are sinks.
Enabled	Circuit Structures Airbreaks	Set to true for all network features. AIRBREAKS may be reset as barriers i.e. enabled is false, depending on network implementation
Weight	Circuit	Feature length is a used as a weight constraint

Table 2. Geometric Network Parameters

3.5 Metadata

Creating and managing metadata for database objects is an important component of GIS implementation, and has profound impact on tracking data update. Metadata provides complete information on GIS data, including data properties (usually derived from the data source) and documentation (often user-input information).

Metadata on geographic data properties will include data source, scale, resolution, accuracy, projection, and its reliability. Additional documentation may be designed to track data update cycles, to provide information on projects for which the data was created, and to provide details on data completeness.

4. HARDWARE INFRASTRUCTURE

The hardware infrastructure and the adopted technology solution complement one another, and both are jointly employed to provide support for database maintenance with regards to data security.

The first line of security is usually provided by the storage solutions. Storage vendors have standardized on RAID storage solutions for data protection. Optimal RAID configurations are available that balances data access performance with data protection.

The second line of security is managed by the software architecture in the empirical implementation, and this involves two security layers:

1. SQL 2000 has in-built security protocols that ensure that data stored in the RDBMS are protected against unauthorized access. These involve protocols for creating database users, roles and access to specific databases and tables within the SQL database.
2. The second security layer is provided by ArcSDE in-built functionality. ArcSDE tools are available to create database users and grant different roles to authorized users in order to edit, update, query or manipulate the geodatabase from clients' applications.

5. IT BEST PRACTICES

Until very recently, one often ignored factor in corporate database maintenance is risk management, including software disaster management and recovery all of which are essential components of modern business processes.

In this implementation, we recognized that implementing a new business-critical application requires the identification of business continuance strategies and the implementation of technologies and procedures that (i) ensure that corporate data is maintained in a secure, reliable state, and (ii) maximize the database availability to users to support their defined roles

Effective business continuance strategy is addressed by focusing on two factors - (i) the maximum level of downtime, if any, that is acceptable for maintaining business operations; and (ii) the minimum level of employee interaction with the system that is mandatory for maintaining business operations.

After determining these requirements with respect to acceptable levels of downtime and employee-system interaction in the event of a failure, the hardware and software requirements for meeting those expectations are identified. Existing on-site hardware and software are assessed and tied into the new system wherever applicable.

Low-Cost Strategy: If some downtime is acceptable, a relatively basic hardware solution can be put into place utilizing a simple tape backup device as the interface for archiving snapshots of the data. In this scenario, the data and application can be restored from the tapes after a hardware or software failure has been resolved. This is a relatively inexpensive approach, but it can take some time to restore the application and data.

Mid-Cost Strategy: If being "up and running" as quickly as possible is important, a more advanced hardware/software configuration and redundant hardware devices may be required. Then, if the main system fails, a secondary device (of equal or lesser capability) containing a real-time copy of the application and its data comes online to meet the client's minimal requirements until the main system is restored. In addition, tape backups would be run to keep a relatively current copy of the data offsite. This is a somewhat more expensive and slightly more complicated approach, but business continuance is maintained at a higher level.

High-Cost Strategy: If no downtime is acceptable, a much more robust system can be created using SAN technology to store data and provide multiple levels of redundancy despite storage device failure. All application servers would be set up in redundant clusters so that if one fails, another immediately takes over, resulting in zero downtime and full application capability in the event of a single server failure. Data would also be stored in a real-time remote storage facility utilizing an external high speed network. In the event of a catastrophic failure, all data would be safe and completely up-to-date because of the remote facility's distance from the application site. This is a very expensive approach, but supplies the utmost in reliability, minimizes data loss, and reduces the amount of time required to bring the system back up in the event of a catastrophic failure.

6. CONCLUSION

Integrating prudent geo-spatial data management strategies within the corporate IT infrastructure is becoming increasingly critical in reducing bottlenecks associated with the multi-user, multi-edit and multi-platform environments that utilities operate in. However, such integration must be intuitive, and should be structured to replicate existing workflow processes without compromising data security. A careful mix of spatial data maintenance solutions, IT best practices and optimal hardware configuration will meet this objective and provide the database administrator with the generic framework required for supporting business driven operations.