



Faye Hall, Sr Software Engineer  
Enspira Solutions, Inc

# Geospatial Systems Integration Strategies

By: Faye Hall, Senior Software Engineer, Enspira Solutions, Inc

In recent years, enterprises have invested heavily in their computer assets either through investing in their existing infrastructure, or through acquiring assets due to mergers and acquisitions. The impact of these investments and acquisitions have been significant, though unfortunately, many of these acquired assets sit in solitude, incapable of sharing their information with the rest of the enterprise. Fortunately, system integration facilitates the communication between disparate systems for the purpose of information sharing and process enhancements.

Due to the variety of systems and integration requirements, many common strategies can be implemented to support communication between these systems. Integration requirements may dictate techniques as simple as file sharing or could involve intricate middleware packages managing the messages between systems. The type and complexity of the strategy employed is dependent upon the interface restrictions and applications involved.

## TWO LEVELS OF INTEGRATION

Within the realm of system integration, there are two general levels of integration: information-oriented and service-oriented. Information-oriented integration manages the exchange of information between systems. Service-oriented integration focuses on the sharing of business processes and methods to integrate applications.

## INFORMATION-ORIENTED INTEGRATION

For decades, information-oriented integration has been employed in enterprises to exchange simple data between a variety of systems. Generally, this involves exchanging data at the database level or through information producing interfaces such as integration brokers. Some of the more common techniques utilized with GIS applications are:

- Common Format/File Transfer
- Extract, Transform, and Load (ETL)
- Data Replication

Since the integration is typically performed at the lowest common denominator of a system, this approach benefits from the limited amount of modification to the source and target systems that will be required. As a result, information-oriented integration is considered to be one of the simplest integration frameworks. However, its lack of complexity can be misleading. Depending on the systems to be integrated and restrictions or limitations imposed by these systems, the integration solution can easily increase in complexity.

## COMMON FORMAT/FILE TRANSFER

As today's GIS applications become increasingly versed at reading a variety of file and data formats, it is becoming more and more common to have applications share data by simply utilizing data or files written directly from another application. The core of this strategy rests on the ability of one system to save or publish data to a format which can subsequently be consumed by the other system. For example, ESRI can currently consume AutoCAD's native DWG file format. Similarly, data stored by applications using the Oracle Spatial SDO geometry type can be accessed by many GIS and CAD applications.

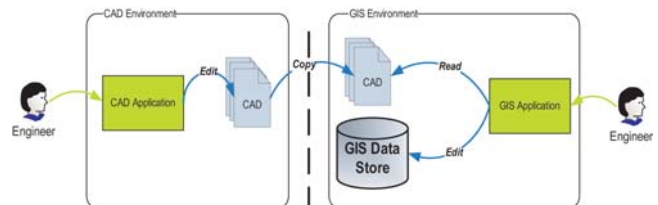


Figure 1. Common Format Example

The principal advantage to this strategy is its simplicity. In most cases, this technique requires very little, if any, additional technologies. If file transfers are required, the commands to copy files and create scheduled tasks are intrinsic operating system functions.

However, file size could be problematic. Files containing spatial data are often significantly larger than normal textual data thereby hampering network performance during the file transfer. Usage of common formats with direct access to the data does provide real-time data currency.

## EXTRACT, TRANSFORM, AND LOAD (ETL)

Should you prefer the simplicity of the previous strategy of transferring files, but your system is unable to consume the data in the format provided or you need the data schema to be altered prior to loading into the destination system, then an ETL approach may be appropriate.

Extract, Transform, and Load, otherwise known as ETL, will extract the data from its source format, transform the data, including data type, structure, or schema, and load the resulting data into a destination format or database. For instance, the attribution of a valve in an AutoCAD drawing may be separated in the GIS environment. An ETL scheme would have to separate the AutoCAD object data into the appropriate table and feature objects.

Another scenario may involve a SCADA system that utilizes AutoCAD files containing GIS data. Rather than maintain the same data in the GIS and the SCADA system, the data can be extracted from the GIS and manipulated to fit the AutoCAD template and layer definitions such that the SCADA system can utilize the resulting files.

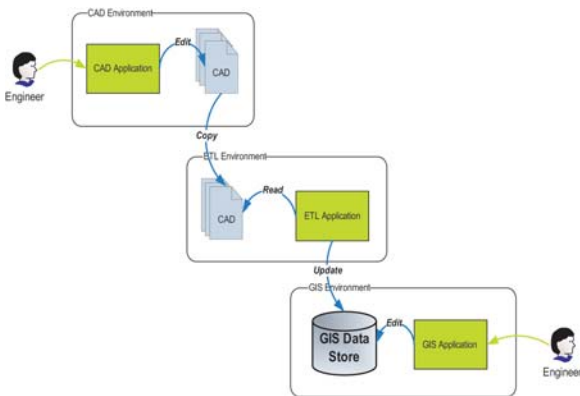


Figure 2. ETL Example

ETL provides more control and customization than a file transfer strategy because it allows data to be transformed and validated prior to being consumed by the destination system. However, another technology or product must exist (e.g. Safe Software's Feature Manipulation Engine) to perform the transformation. ETL is also plagued by network performance issues due to file sizes.

#### DATA REPLICATION

Replication is a mechanism for copying and distributing data from one database to another database while maintaining the consistency of the data between all databases. It allows for the distribution of data to a variety of locations such as remote or mobile users via local and wide area networks, wireless connections, and dial-up connections.

The type of replication chosen is dependant on several factors including the databases involved, the network over which the data will be transported, the minimum latency requirements, and the type and amount of data to be replicated. Each replication type has its own pros and cons to be considered.

Snapshot replication essentially truncates and replaces all existing data in the subscriber database with an exact copy from the publishing database. Though simple, this style is not preferable in situations where data is required at or near real time due to the possibility of overwhelming network demands.

Transactional replication minimizes the peak network demands by tracking and transmitting only the data that has been modified since the last broadcast. Due to the reduced amount of data being transferred, transactional replication is ideal for low latency requirements.

Merge or multi-master replication extends transactional replication by permitting each subscriber to update the replicated data. This type of replication allows many sites to work autonomously while maintaining a single enterprise view of the data. This method of replication possesses many of the advantages of transactional replication; however, it suffers from heavy management and control

requirements and necessitates methods to manage conflict resolution.

Each replication type relies upon actions performed at the database level, thus consideration must be given to the location of the business logic. Should the business logic reside on the desktop or exist in code modules, it may be difficult for the replication scheme to process the data and invoke the business logic validation. Further, should the information being replicated fail validation, a mechanism for resolving errors will have to be incorporated. These schemes are therefore best suited for situations where the data can be consumed without having to pass validation tests at the subscriber.

In the scenario where asset data is maintained in an external application (e.g., Hansen Asset Management), yet the GIS would benefit from the existence of the data in a specific feature class in its database, replication could be used to copy the data from the asset management application database to the GIS. In this case, the asset management application has performed all the necessary business logic to deem the attribution valid. Therefore, the GIS must simply store the data related to the feature.

Another example is an environment where the asset data and the GIS data are both maintained in SQL Server databases, and the GIS data is ESRI version managed. A transactional replication scheme could be developed to transfer attribute modifications to a specific version of ESRI's multi-versioned views within the GIS. Once the edits are

**HI-RISK RAINWEAR**

**FLASH FIRE AND ARC RESISTANT RAINWEAR**

from Ranpro offer the protection that workers in high-risk job applications need, no matter what the job hazard. These suits, job-fit for applications ranging from electrical utility to petro chemical, are rugged and watertight. They are designed to ensure that workers are comfortable in extreme work environments.

**EXTREME CONDITION PROTECTIVE APPAREL**

**RANPRO**

Ranpro Inc.  
620 Ireland Road  
Simcoe, Ontario N3Y 4L6  
Tel: (519) 426-1094  
Fax: (519) 426-5313  
E-mail: ranpro@ranpro.com  
www.ranpro.com

received into the default version, GIS editors would have the most recent data and be able to attach geometries to the new records.

GIS database environments often make replication difficult due to the spatial data storage format and the versioning schemes employed by the GIS. Some replication mechanisms, such as Oracle Streams, do not support the replication of objects with LOB attributes, object types that use type inheritance or binary objects, which are commonly used for spatial data storage. Another common issue is that the versioning mechanism employed by the GIS impedes the use of transactional replication mechanisms.

### SERVICE-ORIENTED INTEGRATION

Service-oriented integration integrates applications by providing access to functions or services that are normally only available in specific applications. While this form of integration has existed for many years utilizing remote procedure calls and application programming interfaces, it has recently jumped onto the radar with the advent of web services and XML. In general, service-oriented integration wraps existing application functionality through exposed interfaces to create aggregate applications. By leveraging the behavior of other applications, data validation and business rules are enforced during the transactions. A service-oriented approach will provide all modifiable systems with access to the published services.

Service oriented integration techniques include:

- Remote Procedure Call Integration/Application Programming Interface (RPC/API)
- Web Services

### REMOTE PROCEDURE CALL INTEGRATION/APPLICATION PROGRAMMING INTERFACE (RPC/API)

For interfaces that must invoke existing business rules, it may be necessary to rely on Remote Procedure Calls (RPC) or the Application Programming Interfaces (API). RPCs and APIs are often provided by an installed application, and can be utilized for communication between applications. For instance, when ESRI's ArcGIS software is installed, a very extensive set of objects and controls are loaded to aid development that will enrich the users experience and ESRI functionality. These objects allow for the development of integration tools to interact with the GIS.

Unlike previous integration strategies, which focused on publishing data to be interpreted by the destination applications, utilizing the API allows for leveraging the behaviors already implemented in the systems. It provides more flexibility for an enterprise than data publication since it is generally a custom tool or utility developed specifically for the environment. However, network to network limitations, the amount of development, resources, and skill sets required to build and maintain the integration software can be a concern for some enterprises.

### WEB SERVICES

Should the environment and architecture provide for it, another viable integration alternative for leveraging remote methods and behaviors are web services. Web services refer to technologies that provide a standardized method for integrating applications over a network, independent of platform. They allow systems to communicate through fire-

walls, between networks, or the internet, without intimate knowledge of the underlying data structure or systems. Such is the case with Google Maps and their use of AJAX technology.

In an enterprise with disparate systems and operating systems, web services offer a means by which many systems can communicate, including the GIS. For instance, if human resources needed to know if a person lived within the city limits, a web service could be developed to perform a spatial query in the GIS with a person's address which would then return a yes or no response. Similarly, a capital improvements project portal could request maps or data from the GIS related to a buffered point, such as all addresses within a 5-mile radius. Web services provide reusability as all applications would be able to utilize the same web server to acquire the functionality. This also helps to further utilize capital investment in existing applications. Another feature of web services is their ability to work through firewalls to connect to disparate networks. As well, additional security features within web services can avail any concerns regarding corporate security policies.

### INTEGRATING YOUR ENTERPRISE

While each integration technique has been described in isolation, it is rare to find an enterprise that has met its needs without taking on a multi-faceted integration approach. Figure 3 illustrates a scenario with the following combination of integration techniques:

- Web services to initiate work orders in work management applications
- Custom interfaces using APIs to integrate the work management/inspection & maintenance applications with GIS
- Data replication and ETL from external agencies and offices

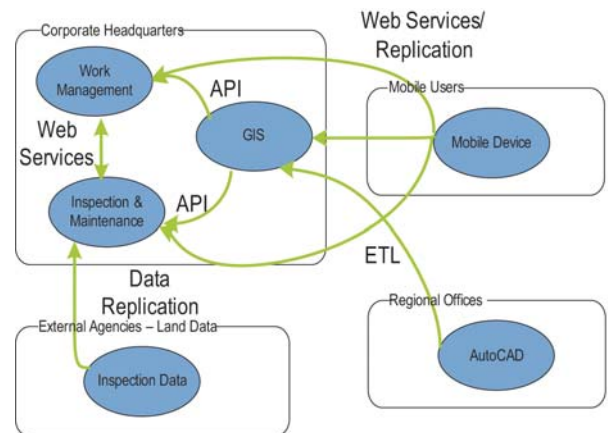


Figure 3. Example of Multiple Integration Techniques

A true enterprise integration framework can be achieved using an enterprise service bus to manage the requests for services to the various systems (Figure 4). This approach provides a bridge to allow functional older applications to interact with newer environments seamlessly. By leveraging off the shelf Service Oriented Architecture middleware, Business Intelligence product suites and adapters, Enterprise Oriented Architecture provides ease of integration, faster deployment windows, reusability, and reduced integration, implementation, and maintenance costs.

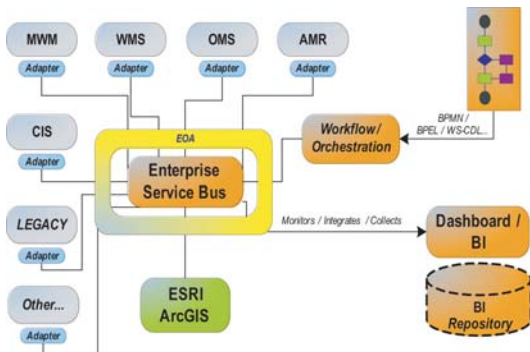


Figure 4. Enterprise Integration Framework

Rather than having various integration techniques randomly used to access diverse applications as in Figure 3, the Enterprise Oriented Architecture provides a central bus through which all communications are directed (Figure 5). Many diverse systems can utilize the same services thus increasing reusability and reducing implementation and maintenance costs. It allows an enterprise to introduce new technologies and immediately take advantage of the existing infrastructure and services.

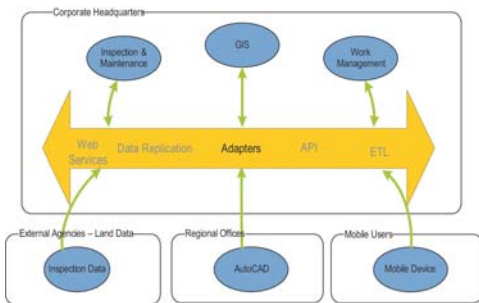


Figure 5. Enterprise Oriented Architecture

### OTHER INTEGRATION CONSIDERATIONS

Every integration strategy will present some limitations and restrictions depending on the environment, the applications, and the latency requirements just to name a few. Some additional aspects commonly considered are:

- Application coupling versus cohesion
- Interface intrusiveness
- Manageability

Application coupling and cohesion refers to the level of dependency on each application's structure and interface that an integration strategy relies upon. A tightly coupled integration strategy is heavily reliant on the interfaces into the applications. As applications evolve, and interfaces are modified, the integration will need to be updated.

Another consideration is the intrusiveness of the interface that is being deployed. If the integration strategy being proposed requires alterations to the workflow of the business, how receptive will the business be to the solution?

The manageability of an integration solution should also be considered. Utilizing an Enterprise Oriented Framework which provides existing middleware solutions eases the maintenance burden that many integration techniques present.

Including these additional considerations when determining an integration solution will ensure a proper solution will be implemented for your enterprise.

### SUMMARY

Systems integration allows information to be shared amongst disparate systems and enables your enterprise assets to realize their full potential. By taking an enterprise approach with your integration techniques, such as an Enterprise Oriented Architecture, you get all the advantages of the integration techniques along with increased reusability, reduced maintenance, and improved information accessibility. Its modular approach to systems integration, through the use of adapters and middleware, enables you to choose the degree of systems integration for your enterprise now, and in the future. ■

### About the Author

**Faye Hall** is a Senior Software Engineer with Enspira Solutions, Inc. Faye designs and develops software applications to integrate Geographic Information Systems (GIS) and legacy systems for the utility industry and for local governments. She holds a Bachelors of Applied Science degree in Systems Design Engineering.