SOA Based Data and Application Integration-

An Enabler for Creative Interpretative Workflows

Clay Harter, OpenSpirit Corp., USA

Abstract

Virtually all geologic and geophysical interpretation relies upon the use of computer software to retrieve and display data and provide analytic and interpretive tools to model the subsurface. Most companies use a diverse set of software applications and data to perform these tasks. This paper describes some of the technical challenges faced when integrating these applications and data and describes an integration solution based upon a service oriented architecture (SOA). Several example workflows are then described that leverage a SOA integration framework.

Introduction

Since the widespread introduction of computer based interpretation in the 1980's most oil and gas exploration have come to depend upon computer software for the storage and retrieval of subsurface data and tools that allow geologists and geophysicists to process, analyze, and construct earth models that are used to make key E&P business decisions.

An individual earth scientist often must use a number of different applications to accomplish a task and use data from diverse sources. Even if an individual interpreter is able to do his/her job with one primary application, when the activities of the whole asset team are examined there normally are many diverse applications employed within the asset team.

Some technical problems inevitably arise because these applications come from diverse vendors that have made differing technical choices. Applications may operate on different computer operating systems (Windows, Linux, and Solaris are the most common in our business) and may be written in different computer languages (FORTRAN, C, C++, C#, Java, and Delphi are the common ones). This technical diversity means that these applications will not readily interoperate. Unlike a Microsoft Office application where a user expects to be able to cut and paste a chart, a table, an image, or text between applications, an interpreter cannot readily select a well or seismic section in one application and have it appear in another. This lack of application interoperability can even exist between applications coming from a single vendor as many vendors have acquired software products with differing technologies as part of corporate mergers and acquisitions.

Problems also arise due to the need to integrate data that may be stored in different application data stores, multi-user projects, or corporate repositories. It is difficult for a single application to connect to these diverse sources of data and access the required data because these different data stores have varying:

- Data models (e.g. how a well is described and represented varies between data stores)
- Units (e.g. all scientific measurements must indicate a unit of measure but varying data stores use different acronyms for the same unit and some data stores don't indicate units at all)
- Coordinate Systems (e.g. any spatial location must indicate the Coordinate Reference System (CRS) that it references but varying data stores use diverse descriptions of the CRS and many omit key parameters – some fail to indicate any CRS information at all)
- Data access mechanisms (e.g. how an application may read or write data to/from a data store varies. If the data store uses a relational database then there are standard software libraries like JDBC, ODBC, or ADO.NET that may be used. However, many data stores use propriety binary files that require proprietary software libraries to read and write data. These proprietary libraries are normally tied to a single programming language and must be run on a specific operating system.)

Figure 1 summarizes these integration challenges.

The net effect of this data and application diversity is that users can not readily choose an application and expect it to interoperate with their other applications or access data from their existing data stores. They must expend considerable effort to transfer data between data stores using manual and error prone export and import procedures. The steps required to make data that is stored in one application data store available to be used in another application is illustrated in Figure 2. As shown in this figure, there are several manual steps involved in this process and the problem of how to do coordinate and unit conversion may be particularly troublesome for some applications that have no means of doing these conversions.

Solution Approach

There have been many approaches taken to try to address the integration challenges raised in the introduction. In this paper we will describe an integration solution based on using a Service Oriented Architecture (SOA). SOA solutions have been widely used in many industries to provide a mechanism for loosely coupling diverse software packages and allowing them to work together. The basic tenet of an SOA approach is to define the services that are required by applications in a given business domain. A service definition defines how an application will interact with other software components. For example, in the broader IT arena, eBay defines a set of services that allow applications to find items for sale, find an item's bid status, and to find an item's shipping costs. A service definition defines how an application may use the service but shields the application using the service from any implementation details of the service.

In order to serve some key integration needs of geotechnical applications we propose the following services (illustrated in Figure 3):

Event

The event service allows diverse applications to share interaction messages without having to know about one another. An application may send an event (a message) to the event service and it will be delivered to any application that is run by the same user, anywhere on the network, if that application has registered interest in events of that type. The event service is also integrated with the desktop drag and drop mechanism so that messages may be delivered by this mechanism as well.

Standard messages should include such items as:

- *Data Selection* – an event that contains references to data (datakeys) in a data store with a data connector. A receiving application may then use the data service to read any details of the referenced data item.
- *Data Change* – an event that contains references to data that has been created, updated, or deleted. A receiving application may register interest for different change types on specific data types or specific data items (e. g. "notify me if this well changes" or "notify me if any well is created in this project").
- *Spatial Events*

  All these events carry a definition of a CRS (and optional geodetic datum shift)

  - *GIS Feature* – a set of selected point, polyline, or polygon features that are associated with named layers. A receiving application would normally add these as layers to be displayed in a map or 3d view.
  - *Grid* - a grid of values (in a specified unit)
  - *AOI* – a 4 sided polygon defining an area of interest
  - *Cursor Tracking* – a continuous changing x,y,z location that tracks a mouse location in a 2d or 3d display window
  - *View Definition* - events that define a particular view in a well, section, map, or 3d viewing window so that a receiving application may synchronize its view with the sending application

Unit

This service allows conversion between any units that share a common base unit. Units are associated with a measurement (such as "length") that has a base unit (a meter in this case). Every unit then has a set of unit conversion factors that convert a unit to its base unit. Measurements are organized into a hierarchy so, for example, "small distance" is a sub class of the "length" measurement. A unit system is a set of measurements with their preferred units. So a metric unit system may have meter as its preferred length unit, seconds as its time unit, and kilogram as its unit of mass. A set of measurements, their allowable units, and predefined unit systems comprise a unit catalog. The default catalog should be based on the POSC/ISO unit definitions which contains more than 900 units.

Coordinate

The coordinate service provides for transformation between engineering (e.g. a seismic bin grid system), map projection and geographic coordinate reference systems as well as geodetic datum shifts. This service is based on the EPSG Geodetic Parameter Dataset (a standard set of coordinate system and transformations that are maintained by the OGP Surveying & Positioning Committee -see http://www.epsg.org). Any spatial data that is accessed via the data service is tagged with a coordinate reference system (CRS) and optionally a set of geodetic datum shifts. The coordinate service allows applications to find predefined EPSG coordinate systems and transformations based on a variety of preferences (e.g. find all map CRSs that are used in the UK) or to construct a custom CRS and associated geodetic datum shift. Many applications that utilize the framework may not be spatially aware so the coordinate service may be implicitly used in the data access service without having to be knowledgeable about the intricacies of geodesy.

Data

Applications may use the data access service to create, read, update, or delete data coming from any data store connected to the framework. The data service is implemented for a specific data store as a piece of software referred to as a data connector. A data connector uses whatever technology is required to access the underlying data store. In the case of a relational database, without any special data access API, the data connector will use SQL access but in the case of a data store with proprietary API the data connector will use the vendor's API to read and write this data. Regardless of where on the network the data is stored, how it is stored, and in what data model, an application connected to the framework may use the consistent API of the data access service to read and write data. The access control is handled by the underlying data store so that there is no added data administration overhead in using the data access service. An application may optionally assert a preferred unit system or coordinate reference system (CRS). If such an assertion is made the data service will internally use the unit and coordinate services to transform data on the fly from how it is stored in the data store into the desired units and CRS. If no unit or CRS assertion is made the data will be returned in the units and CRS as stored in the data store tagged with the appropriate unit and CRS. In addition, a client application may also assert whether it wants to access data via the common data model defined by the framework or the vendor specific data store data model (the "native" model).

The Data service also provides information on the capabilities of a given data store such as what data types may be stored, what are the required attributes, what are the length limits for string attributes.

Reference Value Service

The reference value service (RVS) is a service that is used by the data service when dealing with different sets of allowable values. For example, data store A may specify states or provinces in an abbreviated form (e.g. TX, OK, AB, etc…) whereas data store B may have these specified as the full name (e.g. Texas, Oklahoma, Alberta, etc…). A reference value set is a set of allowable values for a specified attribute for a given authority. So in this example we have two authorities - A & B and two reference value sets for the same attribute. In addition, the RVS allows for a reference value map which specifies how a value from one reference value set may be mapped into another set. In this example a map would make these associations TX->Texas, OK-> Oklahoma, and AB -> Alberta. The data access service then uses the RVS to perform transformations of designated attribute values in order to return the attribute values in a standard set to a requesting application.

MetaModel

Lastly, the metamodel service provides the ability for applications to discover the data model of the common model or any of the underlying native models. The metamodel service provides a uniform API to discover what data models are known to the framework and for a given version of data model what entities, attributes and relationships are defined.

To see how these services may improve the integration between applications and facilitate the access to data let's revisit the simple workflow example that was shown in Figure 2. In Figure 4 we show how applications may use the services in the integration framework to readily accomplish the goal of accessing one application's data store from another application. A user simply selects the desired data in application A and either broadcasts a data selection event (a reference to the data) or drags and drops the data selection onto application B. In either case, application B receives the reference to the data and then requests that data, via the data service, in its desired coordinate and unit system. In this improved workflow there is no intermediate file format and no manual transfer of data and any coordinate or unit conversions are automatically done (even if neither application has the ability to do conversions on import or export)

Workflow Examples

Let us now examine how this integration framework, based upon the described services, may aid in improving more complete workflows commonly used by geoscientists and data managers. The following scenarios illustrate some realistic workflows.

Subsurface G&G Interpretation

Often times a user will use one application to perform seismic interpretation and another to construct a 3d earth model. If these applications are both integrated via a SOA framework then a user may run both applications simultaneously and share event interactions and data in order to quickly iterate through various interpretation options. This concurrent iteration is much more natural and productive than a series of seismic interpretation- > earth modeling and earth modeling -> seismic interpretation manual data transfer steps. In addition to being quicker and less error prone this concurrent iteration also allows the geoscientist to be more creative and spend time focusing on the geology rather than the mechanics of data transfer.

Petrophysical Analysis

Petrophysicists commonly use a specialized application to perform well log processing and analysis that may not be well integrated with the geologic and geophysical applications used by the rest of the company. If the petrophysical analysis application is integrated with a SOA framework then the petrophysicist may readily access existing well data from the geoscientists' working projects, analyze new log information, and write back final interpreted results into the geologists' projects without dealing with file imports and exports.

Well Planning
Geologists and geophysicists may use one application to do their subsurface interpretation and define a series of drilling targets. These targets are then typically exported in a file format and sent to a drilling engineer who uses other specialized software to design a drillable well. Utilizing a SOA framework allows the driller and geoscientist to share drilling targets and planed wellbore paths seamlessly without file transfer and without worry of errors in coordinate or unit conversion.

GIS Related
Companies that have invested significant time and money in deploying either web based or desktop based GIS solutions can derive more value from these by ensuring that these GIS solutions tie into the SOA framework in order to send data and GIS selection events. This will allow the GIS map to be the focal point for finding data and then sending it to any other geosciences application that ties into the SOA framework. The same can be done in reverse – the GIS application may receive G&G data and make it available for analysis and display in the GIS map.

Data Management
There are many tasks that need to be performed by data managers or geotechs in oil and gas companies in order to manage the tremendous amount of data used in our industry. These include:

Data Browsing
By using a set of simple data viewers and text based data browsing tools that tie into a SOA framework one can use the same set of tools regardless of the underlying data store. Often these SOA enabled viewers can be easier to use and more fit for purpose than the viewing tools than come with the native data store.

Data QC
By using QC tools that perform analysis of the completeness, consistency, and quality of data that exploit the SOA framework, one can employ the same QC tools across diverse data stores and run these QC tools on the computer platform of choice regardless where the data may be stored. Because the SOA framework is based on standard JDBC and ADO.NET data access APIs it is also easy to use generic QC tools that have come from outside of our industry to analyze E&P specific technical data that would otherwise be inaccessible to these QC tools.

Data Loading and Editing
By using the metamodel and capabilities services associated with the data access service one can use common data loading and editing tools independent of the underlying data store and data model. Since the data service automatically uses the coordinate and unit services, as required, the development of such data loaders and editors is vastly simplified.

Data Transfer
Data transfer tools that exploit the common model exposed by the data access service may be used to transfer data between any data stores with a data connector. Instead of writing N2 different data transfer solutions for N different data stores one can use the same data transfer application that exploits the framework's data access service. By harnessing the capabilities and metamodel services this common data transfer solution can even deal with data store specific limitations.

Conclusions

In this paper we discussed many of the technical challenges that are faced when users need to use several applications and integrate data from diverse sources. We introduced the concept of a SOA (Service Oriented Architecture) integration framework that allows applications to seamlessly access data from any connected data store and to share user interactions with other applications. We illustrated how this type of integration can benefit many types of geotechnical workflows and allow users the freedom to put together applications and data in ways not anticipated by the original developers of the applications. By using a SOA integration framework users can independently choose the best application and data management solutions and more readily "mix-and match" vendors' applications to assemble their own custom solutions that best support their creative ideas.

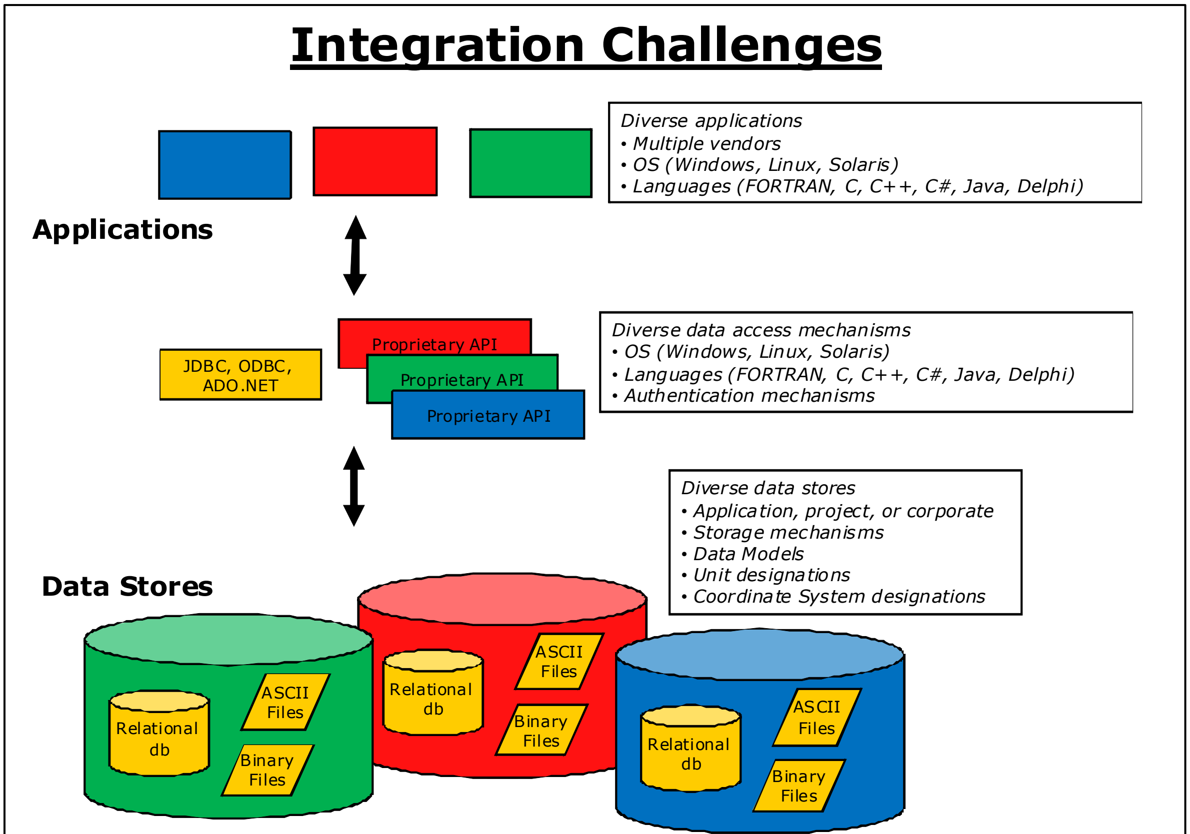

Figure 1

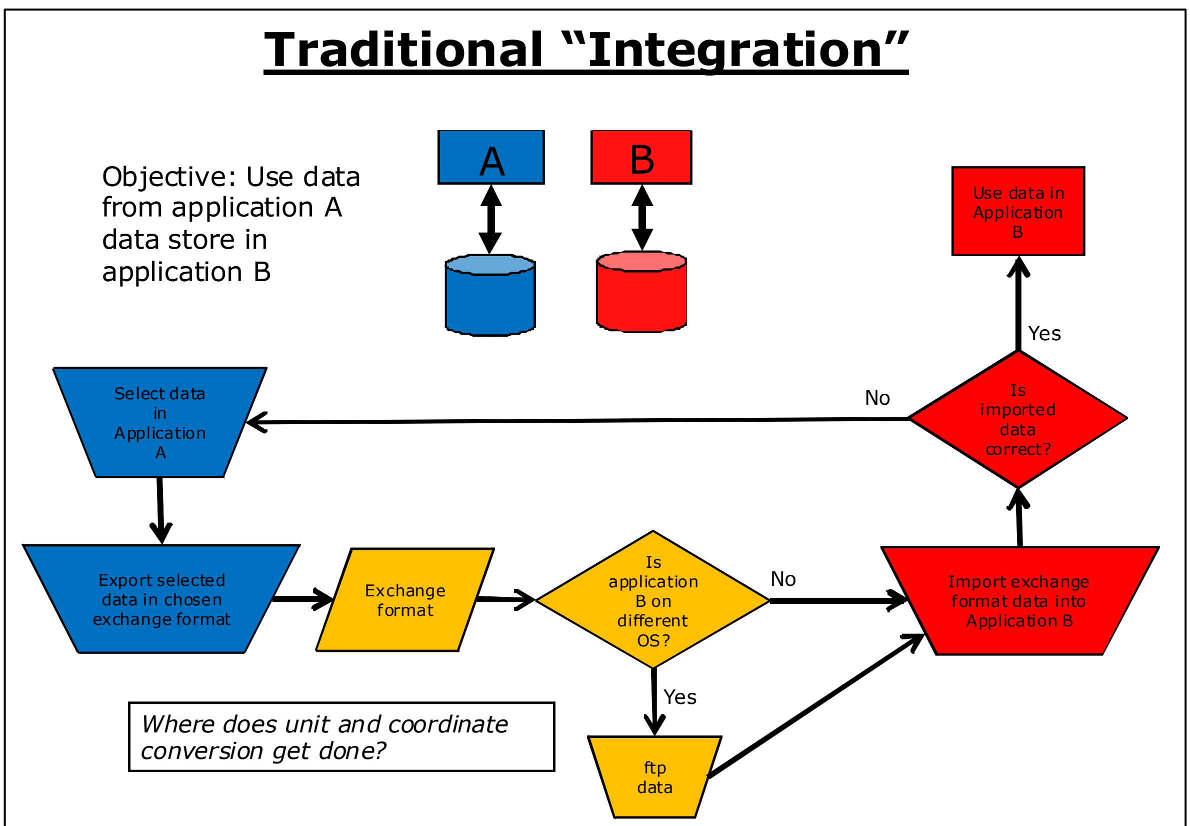

Figure 2

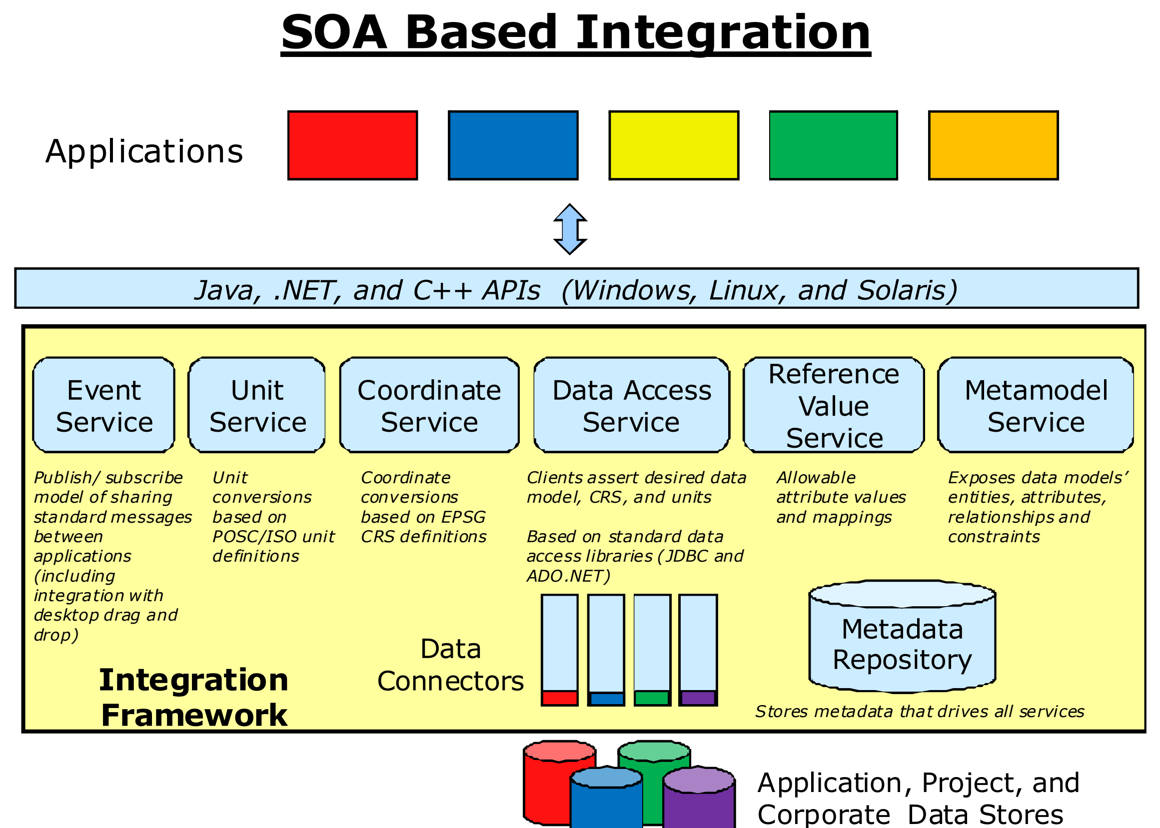

Figure 3

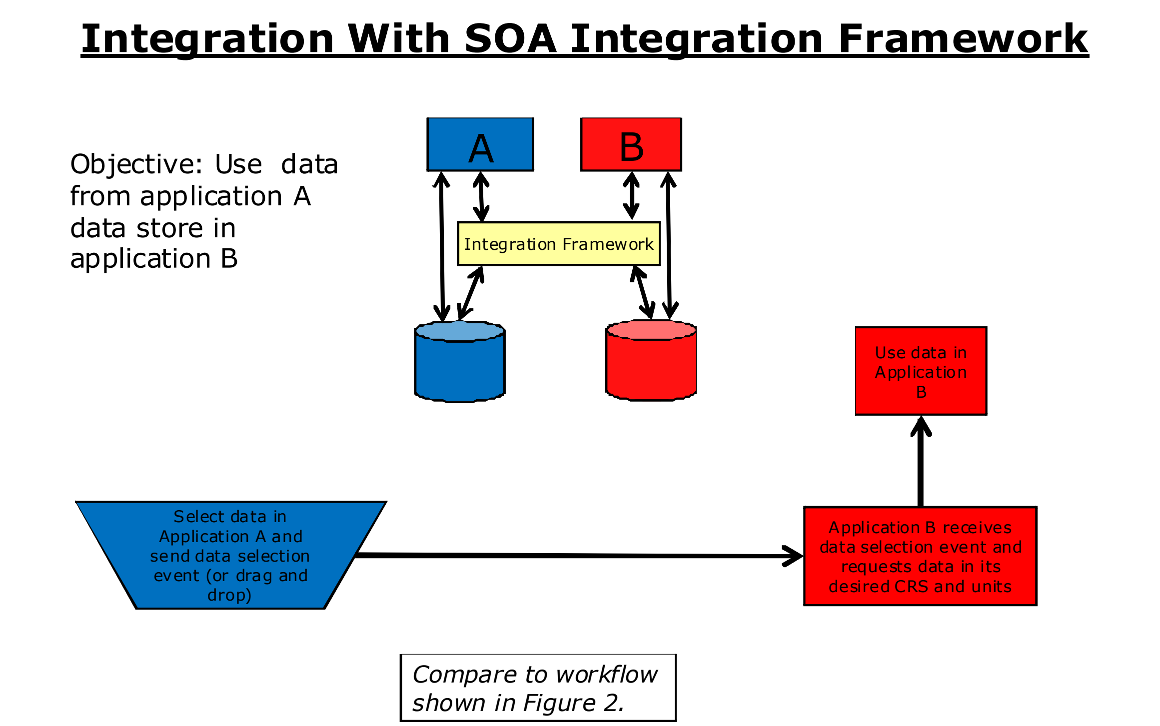

Figure 4