

BIOGRAPHICAL INFORMATION

Mark Hing
Principal Consultant
GE Energy

Specific Responsibilities

Prior to joining GE Smallworld, as part of a GE acquisition, Mark worked with Geodata Solutions on a variety of projects involving Electrical, Municipal, Gas, and Telecommunications systems. Involved in all aspects from design to implementation.

Also designed Web based systems and completed a number of integration projects between GE's Smallworld™ GIS and other external systems. As a certified Smallworld™ Training Instructor, taught a variety of courses covering all aspects of GIS at various client sites and educational institutions.

Past Experience

- Designed and implemented an EAI solution to interface GE's Smallworld Physical Network Inventory™ GIS product to multiple Siebel clients for Manquehue.Net in Santiago, Chile.
- Designed and implemented an EAI solution for BC Gas, using Mercator and MQSeries, to interface their WMS with GE's Smallworld Design Manager™ product.
- Designed and implemented the Energy CIS to AM/FM and AM/FM to DCRS interfaces for BC Gas.
- Designed AltaGas's CIS to Smallworld™ GIS interface.
- Designed and implemented Westcoast Energy's CEPA to GIS/Smallworld PowerOn® interface.
- Designed and implemented a call taking application and associated GIS/Smallworld PowerOn® interface for CERJ in Rio de Janeiro, Brazil.
- Provided technical expertise for Nova Scotia Power's GIS/Smallworld PowerOn® RFP.
- Provided design services for Ontario Hydro's CSS to Smallworld PowerOn® interface.
- Designed and implemented Metronet's Butterfly Diagram functionality in GE's Smallworld Physical Network Inventory™ software product.
- Designed an interface between BC Hydro's Customer Information System and Smallworld PowerOn® product using MQSeries.
- Taught Smallworld™ courses at New York City's Hunter College, the British Columbia Institute of Technology (BCIT) and at various client sites.

Educational Information

B.S. Computing Science, Minor in Economics, Simon Fraser University, B.C., Canada.

AUGMENTING EXISTING EAI WITH WEB SERVICES TO REDUCE GIS INTEGRATION COSTS

Mark Hing
GE Energy, Suite 100
8525 Baxter Place
Burnaby, BC, Canada, V5A 4V7
E-mail: mark.hing@ps.ge.com

ABSTRACT

With the proliferation of EAI vendors, corporate mergers and takeovers, enterprises are discovering that they must now integrate multiple EAI solutions.

By establishing an integration standard, Web Services are in the unique position to create an environment where GIS applications can seamlessly interoperate with other software while still leveraging traditional EAI technologies' robustness, reliability and speed.

This paper discusses how the strengths inherent in both architectures can be combined to achieve solutions that are more flexible, robust, efficient and less costly.

The salesman in the expensive suit at the client's headquarters was dancing like a preacher at a revival meeting, hands moving to and fro, eyes lit up and hypnotic words flowing from his mouth. A projector, humming softly in the background, displayed a synchronized Microsoft® PowerPoint® presentation extolling the virtues of this particular vendor's EAI solution. A young sales associate with a disheveled look hurriedly counted out slick sales brochures as the presentation approached its end.

And then came the message. "Buy our EAI offering and you'll never have to worry about integrating your applications again!" True? Probably not.

In the IT world, integration is big business. With a multitude of legacy systems, new applications and a constant need to provide better customer service and use existing data more efficiently, EAI is poised to grow at a phenomenal rate.

However, all is not as clear-cut as some would have you believe. Seamless integration is a nascent technology fraught with inconsistencies, emerging standards and dubious claims. But it is getting better.

WEB SERVICES ADVANTAGES

With the advent of Web Services, companies can now use generally accepted standards that fill the, often substantial, gaps in traditional EAI offerings. This means that Web Services and EAI can complement one another to provide a synergistic solution that is stronger than either solution on its own.

It's All About Standards

The main advantage of Web Services is in its ability to provide a standard. If everyone agreed to use one EAI vendors' solution worldwide, then that vendor's claims of using its product to easily integrate with everything and anything would have some validity. For obvious reasons, however, a single vendor does not control one hundred percent of the EAI space.

Rather multiple vendors with different proprietary solutions abound and customers who are victims of corporate takeovers, or those merging once independent departments, are beginning to realize that they now have to integrate EAI solutions in much the same way they once tried to integrate individual applications – point to point and inefficiently. In essence, they now need middleware to integrate their middlewares.

The reason? No standards. And that's where Web Services shine. They provide a common framework that can be universally used to allow seamless application-to-application communication devoid of problems associated with differing operating systems and languages.

The result is that these standards will be everywhere. Major application vendors will, and do, support such standards as SOAP, WSDL, UDDI and a number of soon to be available technologies that address security, work flow and transaction handling. And Web Services are built on top of the ubiquitous XML standard that most, if not all, computer systems can read and write. This provides a flexibility not historically available with proprietary EAI solutions.

Scalability

Scalability is another significant advantage. Traditionally, because it was a costly undertaking, EAI was relegated to very large corporations connecting small subsets of applications. Web Services changed that. The same technologies that are used to connect multi-million dollar applications can now be used to connect inexpensive applications running on devices such as cell phones, PDAs and mobile GIS field information systems. This allows a company to leverage its existing infrastructure and extend it beyond its corporate boundaries. Because of this, an entirely new class of uses will begin to emerge thus enabling corporations to share data and functionality in ways they would have never dreamed possible. Efficiency and lower costs will surely follow.

Web Services also tend to be easier to develop and maintain. With thousands of developers familiar with the technology, enterprises have more choice in staffing their integration projects. When compared with the relatively few developers that have expertise in a proprietary EAI's technology, it doesn't take an economics genius to realize the end result.

Basic economic theory states that when supply is greater than demand, prices usually fall (the opposite is also true) and thus projects based on open standards that are familiar to many tend to be less expensive to develop. However, it goes further than that. Since various studies have confirmed that between 70 and 80 percent of a system's cost is spent in the maintenance and support phase of the lifecycle, having comparatively lower-priced resources maintaining a system will reap significant cost savings down the road.

One such study was completed by R.S. Pressman, in "*Software Engineering: A Practitioner's Approach*," who wrote that, "80% of software lifecycle costs occur during the maintenance phase." A Standish Group study came to a similar conclusion and reported that, "typically about 70% of the total lifecycle cost of software occurs after initial acquisition. Much of this is functional enhancement and product upgrade rather than defect removal."

This plays right into Web Services' strengths because the related technologies lend themselves nicely to implementing loosely coupled architectures that make it comparatively easier to enhance and upgrade existing systems.

Web Services' Technologies

Web Services depend on three primary XML-based technologies.

SOAP (Simple Object Access Protocol) is like an envelope that contains a letter. In the Web Services universe, the letter is the data needing to be exchanged and the SOAP envelope contains the delivery details. SOAP servers handle routing, processing delivery instructions and passing the encapsulated data (the letter) to the appropriate recipient.

And since SOAP can be used over a variety of transport protocols, including HTTP and messaging queues, it allows organizations to integrate internally first (EAI) and then, with very little additional effort, expose data and functionality to external parties (Inter-enterprise Integration or IEI).

WSDL (Web Services Description Language) is, like SOAP, a specific XML vocabulary designed to allow applications to determine what a service can do and describes how to use it. Using WSDL, other applications can quickly discover what they need to do in order to use a Web Service of interest.

UDDI (Universal Description, Discovery and Integration) is similar to a telephone book for Web Services, listing various WSDL documents so external entities can see what's available. Existing third-party tools can easily interface with UDDI repositories and select the services they need.

By using UDDI, WSDL and SOAP, applications can quickly and easily connect to one another without the laborious, and sometimes cost- and time- prohibitive, methods required by proprietary EAI solutions.

On the other hand, Web Services have their problems too. Contrary to the evangelists, Web Services are not a panacea for all integration problems. Traditional EAI still shines in areas such as robustness, reliability and speed.

TRADITIONAL EAI SOLUTIONS

The EAI industry has been around for decades, and in that time it has been tested and proven on many fronts. Mission critical applications have been entrusted to these technologies and major bugs and quirks have long ago been found and eliminated. This is not something that can be said of current Web Services implementations.

Robustness and Reliability

Robust solutions can only be attained when given enough time and a wide variety of problems to handle. This has happened with the major EAI technologies. Although proprietary, they have withstood long years of real-world use and have had to solve a variety of difficult problems. Through constant upgrading and redesigning, these technologies are now at a point where they can handle almost any type of traditional integration problem. In other words, they are field tested and robust.

As a consequence, they've also attained new heights in reliability. EAI vendors have long had two quality-of-service message delivery levels. Although each vendor calls them something different, they basically come down to probable delivery and guaranteed delivery. The trade-off is between speed and safety.

For data where speed is of the essence, and where time to live is short, probable delivery is usually selected. For data that positively, absolutely has to get there, and where speed is a secondary concern, guaranteed delivery is an absolute must. A good example can be seen with a power outage management system connected to a GIS.

Probable delivery is often good enough for a call processing application that interfaces to customers through an IVR (Interactive Voice Recognition) system. If a customer calls in for the latest estimated time to restore his power and doesn't receive it, he will most likely call back or repeat his request through the IVR's menu system. In addition, estimates have a relatively short time to live as they are constantly being updated.

However, guaranteed delivery is required when actually determining what pieces of equipment have malfunctioned and their current statuses – as you would not want to dispatch a crew, in an ice storm, to fix a transformer that was just fixed by another crew.

Performance

Another strength of EAI technologies is the fact that they are fast. Most enterprise-level systems can handle thousands of transactions per second because the proprietary layers are tuned to work together efficiently. They don't have to worry about conforming to less efficient standards that can add layers of inefficient processing to a system. Message formats can be designed to be compact and relatively small.

Thus, when high throughput is required, EAI technologies will generally outperform their Web Services brethren.

WEB SERVICES AND EAI COMBINED: THE SYNERGISTIC SOLUTION

It should be clear, then, that Web Services and traditional EAI technologies both have their own strengths and weaknesses. The good news is that there are many areas where one's strength plays into the other's weakness and vice versa. Therefore, there is a distinct opportunity to combine the two integration methods and reap a synergistic reward.

There are times when using Web Services to integrate certain applications will be far more effective than using traditional EAI solutions. These times generally arise when connecting to disparate applications, possibly within different companies, take precedence. On the other hand, when speed, reliability and security is of prime importance, traditional EAI technologies have the advantage.

Another caveat is that Web Services are an evolving standard. That means that something implemented today might have to change in a few years. This could require significant update effort or, in the worst case, a complete rewrite.

On the other hand, as Web Services mature, more and more vendors will support them out of the box. The same cannot be said about proprietary EAI solutions. In the future, Web Services will most likely attain most of the characteristics that currently make EAI technologies so desirable. When that happens, it will sound the death knell of traditional EAI as we know it today. Unless the EAI vendors quickly adapt to bring their products in line with the future standards, they won't remain as going concerns.

However, that time is not yet upon us and that means there are still many benefits to using an intelligent mix of both traditional EAI and Web Services in an integration project.

By leveraging each solution's strength and minimizing its weaknesses, organizations can implement effective integration solutions to achieve greater benefits, reduced risks and lower costs. A true win-win scenario.