

The SOA Magazine

Feature Article



A Case Study on SOA and Process: Integrating E-Gov Travel Services with Federal Agency Financial Systems (Part II)

by Jian "Jeff" Zhong

Published: November 6, 2009 (SOA Magazine Issue XXXIII: October 2009)

[Download PDF](#)

[Digg This](#) • [De.licio.us](#) • [Slashdot](#) • [Technorati](#) • [StumbleUpon](#) • [Google Bookmark](#)

Abstract: In part I of the article, I have described the first four steps of an eight-step SOA development methodology with a focus on architecture, design and abstract thought process. In part II, I will continue to discuss the last four steps concentrating on Web service development, deployment, tests and management. I will then benchmark the case study against industry SOA principles, summarize lessons learned from this real world project and finally share with readers about my considerations for future SOA engagement and practices. After reading both parts, readers should have a complete understanding about SOA strategies, tactics and concrete implementations.

Service Component Development

The above four steps concluded the planning stage of the service development. Business problems were identified, abstract solutions were analyzed, technologies were selected, architectural decisions were made, and software components were identified. Our development effort was divided into two major categories:

1. All agency Web service end points, logging and exception handling, and reliable messaging were handled by TIBCO BusinessWorks and the following components were developed with a GUI tool called TIBCO Designer:

Inbound Lookup and Transactional Services This component provided the end points for all the agency developed Web services for E-Gov Travel Services-AEBS integration.

Outbound Traveler Profile and Transaction Notification Services This component calls E-Gov Travel Services and E-Gov Travel Services EAI Web services to update traveler profiles and send notifications.

2. All business logic, transactional features, and service-enabled portlets were handled by Oracle products, and the following components were developed with Oracle JDeveloper:

TIBCO Oracle Message Bridge This component enabled real-time patient travel.

Java EE Message Driven Beans This was the transactional subscriber to encapsulate all business logic related to accounts payable, accounts receivable, general ledger, and purchase orders.

Java EE eVoucher Service-Oriented Portlet This was a legacy Web application that provided user interface through the the agency portal. The migration maintained the same user interface and called E-Gov Travel Services Web services to retrieve and update data in E-Gov Travel Services.

Java EE eProfile Service-Oriented Portlet This was a new job aid to allow the agency users to update their personal information such as credit cards.

Service Component Assembly and Deployment

All components were assembled into Java EE EAR [REF-38] or TIBCO EAR files. A Java Enterprise ARchive, or EAR, is a file format used by Java EE for packaging one or more modules into a single archive so that the deployment of the various modules onto an application server occurs simultaneously and coherently. It also contains XML files called deployment descriptors which describe how to deploy the modules. A TIBCO EAR file contains process definitions and WSDLs, XML schemas, configuration files, and resource files. All software components in the E-Gov Travel Services-AEBS integration project were packaged as six EAR files:

1. TIBCO Inbound Services EAR
2. TIBCO Outbound Services EAR
3. Oracle Financials MDBs EAR
4. Oracle eVoucher EAR
5. Oracle eProfile EAR
6. Oracle-TIBCO Message Bridge EAR

TIBCO EARs were preconfigured with global variables and could be reconfigured using a TIBCO Administrator tool by the agency Integration Service Center support. Oracle EARs were preconfigured according to Java EE standards with Oracle Enterprise Manager by the AEBS infrastructure support team. This enabled the same set of code to be deployed into development, test, and production environments, saving significant time and resources.

It is worth mentioning that the development effort was divided into two major phases. Each phase completed the eight cycles: Phase I concentrated on infrastructure and framework development; and Phase II added more business functionality.

Verification and Validation

To obtain the best quality for the project, each component was unit tested thoroughly in a local development environment until the services were assembled and deployed. After service was deployed, and prior to the full integration, a functional expert completed a set of all test scenarios by interjecting XML data files into the workflow. When software deficiencies were identified, issues were addressed and corrected, and the software components were redeployed. After all AEBS generic business logic and E-Gov Travel Services specific business logic was tested and implemented correctly, the services were exposed to E-Gov Travel Services and E-Gov Travel Services EAI teams for consumption, further tests, and final integration. After E-Gov Travel Services and E-Gov Travel Services EAI had finished their development and all the moving parts were connected, functional teams conducted end-to-end independent tests with the predefined test scenarios. After integration tests and all critical bugs were corrected to the satisfaction of AEBS management, end-users conducted User Acceptance Tests.

Operations and Management

Without effective management and monitoring, it would be difficult to diagnose the root cause when a problem occurred in the system. This was especially true for the near real-time financial integration with a long communication and component chain, which involved seven different organizations within the agency and the E-Gov Travel Services provider. The longest communication invoked 21 components to complete a single Web service request. Experience showed that most problems occurred on interfaces between enterprise boundaries and different vendor products.

To facilitate troubleshooting, a close collaboration effort was required and the participating groups often included network, proxy server, security, TIBCO servers, Oracle servers, and the DBA teams. A single command and control team with knowledge and working experience across all platforms and technologies directed the communication and workflow.

Benchmark against Industry SOA and the agency Enterprise Architecture Principles

Now that the solution development life cycle has been examined, I will show how the steps are related to industry SOA principles [REF-4-7] and the agency Enterprise Architecture principles [REF-33].

Standardized Service Contracts and Service Abstraction

The Service-Oriented Analysis step produced the abstract WSDL and associated XML schemas - a contract between service consumers and providers. For the AEBS integration project, the contract provided sufficient detail so that design and development from both the consumer and the provider could begin to work simultaneously. The abstract WSDL was platform independent so that the architecture could withstand technology evolutions. Most importantly, it provided a benchmark to measure vendor product features.

Service Loose Coupling

The service contract generated from the analysis steps decouples the service interface definition and the underlying implementation technologies. It also decouples the service provider and the consumer design and development effort. In the Solution Architecture step, AEBS adopted a publisher/subscriber design pattern so that publisher or subscriber could change without impacting the system so long as the contract was not changed.

On the subscriber side, a clear delineation of generic service logic from project specific plug and play components made the components and services reusable - yet fulfilled each project's unique requirements.

Service Reusability

As a result of the project, both the E-Gov Travel Services provider and the agency have developed a complete set of Web services to integrate E-Gov Travel Services and AEBS. E-Gov Travel Services is contractor-owned and contractor-operated software as a service (SaaS), while AEBS uses primarily Oracle COTS products with agency customizations. Since the agency was the first to use SOA to integrate with E-Gov Travel Services, the most usability would be achieved in a future project with a similar set of software and services. The lessons learned, processes followed, and components and services developed can all be reused with very little modification. This effort should be ideally coordinated by the E-Gov Travel Program Management Office in GSA.

For E-Gov Travel Services, the traveler profile services and eVoucher services could be used by all agencies. This would provide real-time integration over the currently dominant batch integration.

For future AEBS integration, the purchase order services, invoice services, general ledger services, and project lookup services can be shared and reused. At the agency, this was the largest SOA effort and first across-enterprise integration.

Service Autonomy and Stateless

The WSDL that was defined in the analysis steps has a complete set of data capture for all business requirements and logical processes to complete a single business transaction. It does not have any dependency on context data or session data. A complete set of data in a message can be transmitted from one component to another. It was difficult, if not impossible, to keep session data when messages passed into different layers of services and message systems.

Service Discoverability

Similar to any other integration project, each project has a registry for Web services, a WS-Inspection feature implemented by TIBCO BusinessWorks products.

Service Composability

Since AEBS services are mostly financial transactions, transactional messages are usually composed of entity objects and aggregated into a very high level. For example, a travel authorization is composed of up to four purchase orders, and each purchase order consists of vendor information, project, task, and expenditure

organizations. A voucher can be a mixture of accounts payable, accounts receivable and general ledger and be related to purchase orders. To compose a transactional service with transaction services posts some challenges. The following advanced features must be implemented to efficiently compose services from services.

- WS-AtomicTransaction - for short duration transactions
- WS-BusinessActivity - for long duration transactions
- WS-Coordination - to coordinate composing transactional services
- WS-ReliableMessaging - to guarantee a transaction is submitted exactly once in the event of system failure or application exceptions.
- Security infrastructure - to propagate security context from upstream to downstream service components

Unless the above features [REF-39-43] are supported by the technical infrastructure, the transactional services can only be composed of JMS, JDBC, and look-up read-only Web services.

Project Staffing

In a competitive sourcing environment, resource allocation and cost estimating are very important for bidding, winning and successfully implementing any SOA project. Many people contributed and played essential roles in the E-Gov Travel Services-AEBS integration project. To fit into this technical discussion, only the technical team is mentioned here:

People:

- E-Gov Travel Services Integration SME (Phase I and II)
- SOA Architect (Phase I and II)
- Web Service Developer (Phase I and II)
- Web Service Developer (Phase II)
- Oracle Technical Expert (Phase I and II)

Domain Knowledge:

- E-Gov Travel Services, Federal Travel and Agency Travel Policies and Regulations
- Oracle Applications Workflow

Technical Skills:

- TIBCO Production Suite
- Oracle Production Suite
- https protocol, forward proxy server, reverse proxy server, and load balancer
- SSO, SAML
- PKI
- FIPS Compliant Encryption Modules
- XML Schema and WSDL Definitions
- Java EE: JMS, JCA, JDBC, AJAX, and Oracle RAC
- Application Server Administration
- Deployment
- Architecture Principles

- Design Patterns

Lessons Learned

SOA must be business driven and incrementally deliver business results. I have seen organizations that are heavily invested in SOA infrastructure and ambitious big-bang service design, deliver tons of architecture documentation yet fail to reach consensus or deliver concrete services and business value.

SOA is more than just Web services. I have seen a project develop Web services with the calling clients all batch processes. In another situation, a DBA developed a Web service to wrap a database table and in the same time frame also developed a Web service client to move the data into another table in the same domain. These were costly technical endeavors that resulted in little or no value-added benefit.

Do not rush into developing Web services without architecture and systems planning: you may discover, after a project is completed, that the services - which may be redundant - cannot be reused without significant modification. I have seen a case where the first project was copied into subsequent projects to claim reusability, which resulted in multiple versions of code and redundant Web services. The use of copy-and-paste is not SOA. Silo approaches may create redundant Web services; therefore, the architect must capture a holistic view of the enterprise.

Business Benefits Gained

Seamless near-real-time integration ensures that systems are always in sync with each other, reducing potential errors, and time and cost to correct those errors. Real-time integration also eliminated redundant data entry of tens of thousands of travelers' profiles at the agency, improving service performance over the legacy system.

It is a good return on investment (ROI) to utilize the enterprise-wide licenses, infrastructure, and future-proof SOA architecture and design to solve real business issues and provide great service to the user communities. The more projects that utilize the SOA infrastructure and architecture, the more the agency will reap a return on SOA investment and the benefits it offers.

Future Considerations

Industry Trend

In order to make a business case, services must be generic and reusable to accommodate a large enough customer base to sustain a profitable business. However, history in the last decade showed that when service is generic enough and the market is large, some of the giant gorillas will get you even when you have the best technology. Netscape, Sun Microsystems, WebMethods, and BEA are good examples that the best technology may not necessarily win the business war or even survive. Vendor consolidation will continue and after the recession is over, you may find only a very few dominant players left. In this sense, the Art of Services is the Art of War. It is critical for a customer to assess the long-term viability of a software vendor, foresee the changes, and upgrade when necessary.

Oracle Applications Upgrade

A frequently asked question about the AEBS SOA project is what it will look like when the agency upgrades to the newer Oracle Applications Release coupled with the Integrated SOA Gateway [REF-44-47]. The gateway offers the ability to expose AEBS functionality as SOAP-based Web services. While this is not something that can be answered in a short article, you will find that my solution is loosely coupled in many ways so that parts can be replaced without affecting the whole. The architecture has foreseen, and taken into consideration, future change and upgrades. The newer release will not diminish the unique business requirements and specific custom processing logic. However, it could create an opportunity to integrate the best features offered by Oracle into the current SOA architecture creating the most effective solution for supporting agency requirements.

WSSE

One of the most challenging aspects in implementation of SOA in federal government agencies is Web Service Security (WSSE). Without a vigorous security review and enterprise security capability, the components and systems cannot be connected and integrated. AEBS has implemented WSSE with the support of agency security and IT personnel. AEBS also followed NIST Web Service Security Guideline (Pub 805), researched WSSE X509 and SAML authentication, Message Integrity and Confidentiality, and provided a technical proposal to agency management and to GSA. Government-wide WSSE implementation is useful to an individual agency as well as being very beneficial to all E-Gov initiatives, and therefore can reach much larger scale service reusability and realize bigger cost savings. The E-Authentication initiative, now called Federal Identity Management, could be expanded to cover Web applications as well as Web services [REF-48].

Enterprise SOA

There is still a long road from reusable to reusability where SOA is not implemented at the enterprise level. First, most projects are measured by fulfillment of business requirements. It is difficult to tell whether services can satisfy future business requirements and needs. In addition, other projects may not have the motivation or incentive to reuse existing Web services given that projects are often funded and managed by different organizations and sometimes supported by competing contractors. Therefore, to achieve the most business benefit, it is crucial for an organization to take a holistic approach to plan, execute, and manage SOA at the enterprise level. The affirmative experience from this effort will have a very positive impact on AEBS program's vision to become a world class service organization.

Conclusion

In summary, I have presented a step-by-step methodology to plan, design, and develop an effective technical solution for fulfilling a specific set of business requirements. The first four steps are focused on architectural planning and design, while the last four are focused on engineering and implementation. The SOA architecture, technical solution, and its implementation are driven by the business requirements and the needs of the organization, within the constraints of resources, costs, and the timeline. It must have sufficient detail and steps to provide the foundation for successful design, development, and deployment. It is my strong belief that with proper planning and the implementation of SOA, an organization may realize the long-term benefits of agility, reusability, and cost savings.

Acknowledgements

The author would like to thank Kate Burton for editing and proofreading this article and I'd like to thank the following leading industry experts and government executives for reviewing this article and providing valuable input:

- Jerry Zhou, CEO of Futrend Technology Inc., E-Gov Travel Integration Subject Matter Expert
- Venkatapathi Puvvada, CTO, Vice President and Managing Partner of Unisys, Former Chair of the Industry Advisory Council, Federal 100 Winner of 2003, 2005 and 2008
- John Butler, Chief Architect of Everware, Inc and Co-Chair of OMG Government Domain Task Force
- Miko Matsumura, Vice President, Product Marketing and Chief Strategist at Software AG, OASIS SOA Blueprints Adoption Committee Chair, author of "SOA Adoption for Dummies"
- Michael Poulin, Proof of Solution Manager at PADA, OASIS Reference Architecture Committee Member, and author of "Ladder to SOE"
- Dr. Jack Jones, NIH Chief Information Officer and Director of CIT
- Helen M. Schmitz, NIH Acting Chief Architect and Director of Enterprise Architecture
- Andrew Merriman, GovTrip eTravel Technical Director

The author would also like to thank our management and project team members for their outstanding work,

excellent support and assistance, and for providing a pleasant working environment.

References

- [REF-1] http://en.wikipedia.org/wiki/The_Art_of_War
- [REF-2] http://en.wikipedia.org/wiki/Sun_Tzu
- [REF-3] http://en.wikipedia.org/wiki/Thirty-Six_Strategies
- [REF-4] <http://www.soamethodology.com/>
- [REF-5] An SOA Vendor Evaluation Methodology; <http://www.soamag.com/l20/0708-3.asp>
- [REF-6] SOA and EDA: Using Events to Bridge Decoupled Service Boundaries; <http://www.soamag.com/l4/0207-2.asp>
- [REF-7] Service-Oriented architecture (SOA) was first described by Gartner in 1996; http://www.gartner.com/DisplayDocument?doc_cd=114358
- [REF-8] <http://www.opengroup.org/projects/soa/>
- [REF-9] <http://soa.omg.org/>
- [REF-10] http://www.oasis-open.org/committees/tc_cat.php?cat=soa
- [REF-11] Navigating the SOA Open Standards Landscape Around Architecture; A paper edited by: Heather Kreger, IBM Jeff Estefan, NASA/Jet Propulsion Laboratory July 2009; <https://www.opengroup.org/projects/soa/uploads/40/20044/W096.pdf>
- [REF-12] <http://www-01.ibm.com/software/solutions/soa/>
- [REF-13] <http://www.ibm.com/developerworks/webservices/library/ws-soaintro2/>
- [REF-14] <http://www.microsoft.com/soa/>
- [REF-15] <http://www.oracle.com/technologies/soa/index.html>
- [REF-16] <http://www.tibco.com/software/soa/default.jsp>
- [REF-17] <http://www.softwareag.com/corporate/products/wm/default.asp>
- [REF-18] <http://www.jboss.com/resources/soa/>
- [REF-19] <http://www.sap.com/platform/soa/index.epx>
- [REF-20] <http://www.linkedin.com/pub/anne-thomas-manes/0/1/71>
- [REF-21] <http://blogs.oracle.com/davidchappell/>
- [REF-22] <http://booksbysandy.com/>
- [REF-23] <http://www.thomaserl.com/>
- [REF-24] From Stove-piped Projects to Unified Enterprise Architecture: Strategic considerations for E-Authentication service development; <http://www.javaworld.com/javaworld/jw-03-2003/jw-0321-ss0.html>
- [REF-25] U.S. Department of Energy Signs on to J2EE: Create a secure single sign-on Web service for multiple n-tier Web applications. JavaWorld May 2002; <http://www.javaworld.com/javaworld/jw-05-2002/jw-0524-signon.html>
- [REF-26] A Java Case Study: The power of J2EE How the Energy Information Administration secured a complete J2EE solution in less than five months. JavaWorld January 2002; <http://www.javaworld.com/javaworld/jw-01-2002/jw-0118-j2ee.html>
- [REF-27] Step into the J2EE Architecture and Process: Develop complete J2EE solutions with an eight-step cycle. JavaWorld September 2001; <http://www.javaworld.com/javaworld/jw-09-2001/jw-0928-rup.html>
- [REF-28] Working with SOA and RUP by Solmaz Boroumand; <http://www.soamag.com/l16/0308-1.asp>
- [REF-29] Event-Driven SOA; http://elementallinks.typepad.com/bmichelson/2006/02/eventdriven_arc.html
- [REF-30] <http://www.gsa.gov/etravel>
- [REF-31] GSA Solicitation FBGT-CD-030001-N available at <https://www.fbo.gov/notices/993abc7c2c6f22ef8bfef6f94c8861de>
- [REF-32] <http://www.nih.gov/>
- [REF-33] <http://enterprisearchitecture.nih.gov/ArchLib/AT/IA/Integration/IntegrationPrinciples.htm>
- [REF-34] <http://www.gsa.gov/fttr>
- [REF-35] <http://acquisition.gov/far/>
- [REF-36] <http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf>
- [REF-37] <http://enterprisearchitecture.nih.gov>
- [REF-38] <http://java.sun.com/javaee/>
- [REF-39] <http://www.ibm.com/developerworks/library/specification/ws-rm/>

- [REF-40] <https://www.ibm.com/developerworks/webservices/library/ws-wslover/>
- [REF-41] <http://www.ibm.com/developerworks/library/specification/ws-rm/>
- [REF-42] <http://www.ibm.com/developerworks/library/specification/ws-tx/>
- [REF-43] <http://www.ibm.com/developerworks/library/ar-archtemp/>
- [REF-44] http://download.oracle.com/docs/cd/B34956_01/current/html/docset.html
- [REF-45] http://download.oracle.com/technology/tech/soa/soa_best_practices_1013x_drop3.pdf
- [REF-46] <http://convergence.satpathy.org/>
- [REF-47] http://www.oracle.com/technology/products/applications/events/oow-2008/S298465_VeshaalSingh-SOA-EBS.pdf
- [REF-48] <http://www.idmanagement.gov/>

THE PRENTICE HALL SERVICE-ORIENTED COMPUTING SERIES FROM THOMAS ERL



[Home](#) [Past Issues](#) [Contributors](#) [What is SOA?](#) [SOA Glossary](#) [SOA School](#) [SOA Books](#) [About/RSS](#)

Copyright © 2006-2009
SOA Systems Inc.

[Legal](#)