

Trends and Issues in Integrating Enterprises and other Associated Systems using Web Services

S. Justin Samuel
a research scholar of Sathyabama
University,
Chennai – 600 119

Dr. T. Sasipraba
Professor and Head
Dept. of IT, Sathyabama University
Chennai - 600 119

ABSTRACT

A Web service is defined as an autonomous unit of application logic that provides either some business functionality or information to other applications through an Internet connection. Web services are based on a set of XML standards such as Simple Object Access Protocol (SOAP), Universal Description, Discovery and Integration (UDDI) and Web Services Description Language (WSDL). The notorious success that the application of the Web service technology has achieved in B2B e-Commerce has also lead to consider it as a promising technology for designing and building effective business collaboration in supply chains. The key challenges for enterprises nowadays are, how efficiently to integrate business functions and associated systems using web services. The business process automation has generated the need for integrating the available web services across different systems. To achieve this, a better understanding of current methodologies for service integrations is required. In this paper, we investigate the current trends and issues in integrating businesses and other associated systems using web services.

Keywords

Web services and business integration

1. INTRODUCTION

One of the key challenges for different size enterprises is how to efficiently integrate the different business functions using the information technology. Few years back EAI (Enterprise Application Integration) was found as the suit for business integration where EAI is the process of creating an integrated infrastructure for linking disparate systems, applications, and data sources across the corporate enterprise. Then the B2B(Business to Business) integration was extended to integrate applications outside the boundaries of the enterprise. ERP (Enterprise Resource Planning) supports a system that maintains a single database where the data needed for a variety of business functions such as Manufacturing, Supply Chain Management, Financials, Projects, Human Resources and Customer Relationship Management are kept. The common database can allow every department of a business to store and retrieve information in real-time.

This information should be reliable, accessible, and easily shared. But they are often seen as too rigid and too difficult to adapt to the specific workflow and business process of some companies—this is cited as one of the main causes of their failure. Once a system is established, switching costs are very high for any one of the partners. There are frequent compatibility problems with the various legacy systems of the partners. Nowadays, Web services offer an enabling step towards the integration technology. They play a major role in integrating business processes. Deploying Web services reduces the integration costs and brings in the required infrastructure for business automation. Traditional web based business data and transactions can now be analyzed, extracted and modeled to discover new business rules and to form new business strategies, let alone mining the business data in order to classify customers or products.

In this paper, we investigate the trends and issues in integrating the businesses and other associated systems using web services. We have described many service integration issues but the main focus is on the service composition since the service-oriented applications are pushing traditional software engineering problems - distribution, componentization, composition, requirements, specification, verification, and evolution - to their extreme. Services are in fact distributed and may be composed to form new added-value services. Moreover, they are developed, deployed, and evolved by different organizations. Services are exposed for possible use, and a faithful specification of their functional and non-functional properties must be made available for other parties who may search and discover them. Systems, implemented as composite services, must be designed to fulfill requirements that very often evolve continuously, and thus require adaptation of the solution. Furthermore, such systems depend on their combined services, which, in turn, may evolve autonomously. The key challenges for enterprises nowadays are, how efficiently to integrate business functions and associated systems using web services. The business process automation has generated the need for integrating the available web services across different enterprise systems. To achieve this, a better understanding of current methodologies for service integrations is required. In this paper the current methodologies are analyzed and some issues related to web service composition and processes have been enlightened. Since web services and integration of service-oriented systems are the key focus area of nowadays research, it is very important to

understand the various trends in integration of systems and how to use web services to make it more effective and know the related issues in it.

This paper is structured as follows: Section 2 describes the various integration techniques and a brief about web services integration and in section 3 the various issues regarding web services integration have been analyzed. Section 4 concludes the paper and gives an outlook for the work to be carried out.

2. INTEGRATION TECHNIQUES

Features such as common intake and ‘seamless’ service delivery characterize integration, where the client may receive a range of services from different programs without repeated registration procedures, waiting periods, or other administrative barriers. The terms “integration” and “integrated” refer to the activity of combining things from multiple sources to present a single collection of thing to the user. The ‘thing’ may be data or business process. It can be simply defined as the act or process of making whole or entire. We have systems integration, Applications integration, Process integration and Services integration. So integration is a general term. The various types of integration listed above refer to different service components or levels of the service. The specifics of what is to be integrated and the mechanisms by which integration is accomplished will, of course, be different for different service components and at different levels of usage. The primary focus of integration is to identify and manage the resources and to provide the single point service to the user. Though there are different kinds of integration techniques available, we are not going to discuss about all, but about a few which are related to our work.

2.1 Business Integration

Business Integration is referring to the low friction and usually automated linkage of business processes within or across departments or organizations thereof using internet (IP) and associated technologies. It is to realize certain business objectives by bringing together different business services using business processes and internet based IT services.[6] Therefore, it is to compose an appropriate set of business and IT services to meet certain business objectives.

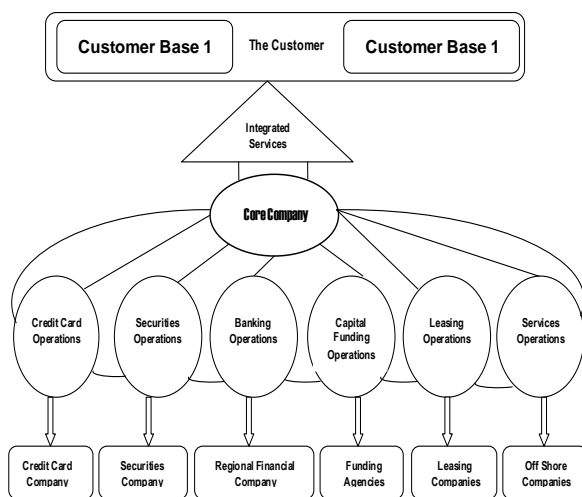


Figure 1. Schematic view of integrated businesses

In Fig. 1, it shows the various automated linkages of business processes across different business organizations and provides the integrated services to its customer.

2.2 Enterprise application integration (EAI)

Enterprise application integration is translating data and commands from the format of one application into the format of another across an enterprise. It is a continual conversation between two or more incompatible systems made possible through networking for the purpose of sharing enterprise data and streamlining functionality.

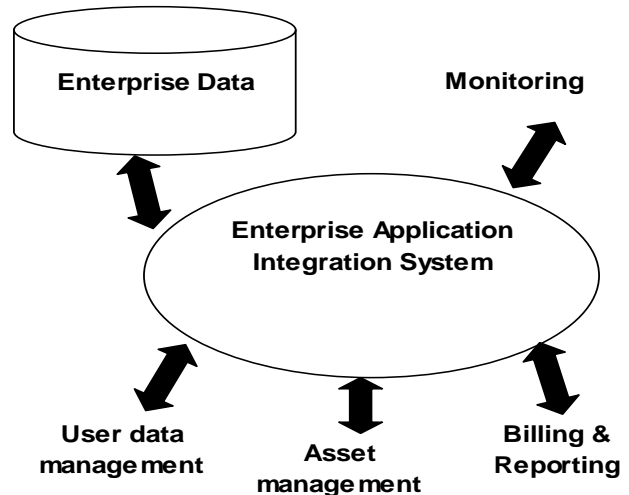


Figure 2. Applications Integration in an Enterprise

Fig.2 depicts the different applications of an enterprise and how they are integrated in the application level. This application integration is called as the EAI (Enterprise Application Integration) and these EAI system’s applications are made available to the customers through application integration.

The most common approaches in EAI are the following. They are,

2.2.1 Data-level integration

This approach works best when the objective is simple information sharing. Systems that need only share customer information or sales data are good candidates for data-level integration.

2.2.2 Method-level integration

This includes concepts of brokering or message-oriented exchange. The method-level approach lets you share data as well as the business rules, restraints, and real-time transaction information. This is also called as web application integration.

2.2.3 Process integration

Business-process integration adds capabilities such as business-process modeling and workflow simulation. Using this, users can define potential solutions, simulate solutions to analyze the flow of information and optimize the business process prior to solution implementation. Using process modeling, companies can focus on process optimization and easily change the process (or implement a new one) with minimal programming or coding. This method, which is also called the process-automation approach, enables movement among application outputs through “intelligent” routing.

2.3 Web Service Integration

Service integration is intended to make it easier for clients to access services by providing them with a single point of entry. The foundation of web service integration is collaborating with other web services. The purpose of integrating web services is, a business can select the modules they need, mix and match modules from different vendors, and add new modules of their own to improve business performance. Since ERP and other COM technologies failed in integrating the business processes of different platform applications with various sizes, now web services integration is the only alternative for making the solution. Seamless integration of web services has enormous potential in the B2B or Enterprise Application Integration. Now, enterprises can deploy their processes as different services, so that others can make use of them. In [11], the various business application logics are provided as web services and are integrated at the enterprise level to meet the business requirements.

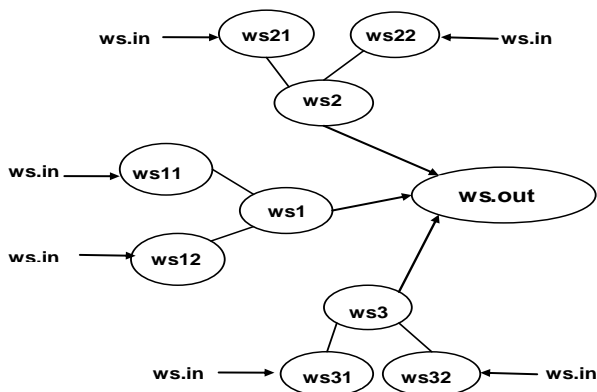


Figure 3. Web services Integration

Fig.3 shows the different web services and their integration in order to get the potential solution. w.in refers the input for a service. ws1, ws2, ws3 refer the integrated services. ws11, ws12, ws21, ws22, ws31, ws32 are the services of enterprises or other domain services. ws.out refers the potential result of the integration.

3. ISSUES IN WEB SERVICES INTEGRATION

Web Service is a piece of software that makes it available over the web and uses a standardized XML messaging system [1]. The web service methods provide various supporting facilities to describe the software for sharing and reuse. There are two issues significant for a web based application development. First is, service description and selection of software components. For this, Web Service Description Language (WSDL) and Universal Description, Discovery and Integration (UDDI) are the proposed standards for software component description. The purpose of description is for the users to select the right and suitable software components. The second issue is service integration or composition of the selected software components. [2]

3.1 WS – Transactions

A key requirement for the success of Web service based integration of business applications is to ensure a correct and reliable execution of the composed process with regards to partners' transactional requirements. Web services technology provides standard interface like WSDL and XML SOAP messaging for the

integration of enterprise applications. However, do not fully support transactional management yet. The web service providers cannot lock the resources for a long time, since it is a time consuming one. So the distributed transaction control has to be carefully planned and implemented for web service based systems. [3] Also, the transactional consistency of workflows and database systems has been an active research topic. Composite Web services indeed introduce new requirements for transactional systems such as dynamicity, semantic description and relaxed atomicity. Existing transactional models for advanced applications [12] are lacking of flexibility to integrate these requirements [13] as for instance they are not designed to support the execution of dynamically generated collaboration of services.

3.2 WS – Negotiation

Negotiation is a process to reach an agreement that is beneficial to the players involved in a business transaction through information discovery and compromises. When the negotiations performed well business partners are encouraged to find creative solution. But the negotiation mechanism for conducting the agreement successfully is a research issue. The research on 'Automated Negotiation' has been started only recently. [4]

3.3 WS – Data

In the Web services domain too little attention has been given to the importance of data and process issues. While transmitting data gets a lot of attention in the web services domain, data storage is hardly being dealt with. The web services paradigm has been characterized by a steady advance in the realization of direct message transmissions for the purpose of integrating systems. Web service standards address only direct point-to-point transmissions for B2B processes. They do not address the fact that what should be done if there are some intermediaries (like shared data storage) are introduced. MapForce is a tool which allows to connect and use existing Web services through data mappings. Also it supports the ability to connect to existing Web services and integrate their functionality into the mappings of XML, databases, flat file, EDI, Excel 2007, and XBRL data. [10] Using Mapforce existing Web services can be used as components for building new Web services and can aggregate two or more Web services to create a "mash up" that integrates functionality from multiple services.

3.4 WS – Security

Data gets a lot of attention in the web services domain, but the web service standards do not talk much about securing storage spaces. Web Services standards do not discuss the communication of decisions with respect to how the storage spaces should be secured. For example, they do not address what to be done to manage the passwords, how long the passwords to be kept valid and how to secure them. [5] Moreover, a few efforts have been made so far to secure a Web Service's availability. Existing security standards like e.g. WS-Security only address message integrity and confidentiality, and user authentication and authorization. There is no specific mechanism to overcome the denial of service attacks in web services so far, where a denial of service (DoS) attack is an incident in which a user or organization is deprived of the services of a resource they would normally expect to have. In a distributed denial-of-service, large numbers of compromised systems attack a single target. Largest concern today is on securing Web services i.e. means of secure credential transfer, message integrity, message confidentiality. There is no clear answer to the questions

concerning security and the legal responsibility to the clients resulting from the execution of a service. [11]

3.5 WS – Composition

Web Service composition is a specification of how to invoke a series of web service operations. There are many web service composition algorithms, but less importance given for the fault-tolerance. In [7], optimization of composed web services has been specified using heuristic approach. Incremental based composition algorithm has been proposed by Carlos and Jose [8], which has not mentioned about the selection of alternative method to handle, in case of a service failure. The management of the purchase order process is a business integration problem and can be realized through the composition of several relative business services [6]. Tree based composition is effective method [9] for composition but failure or about the irresponsible web services have not been addressed. Nowadays GIS web services play a major role in the IT industry. But it has got less attention from the research community. Web service compositions along with GIS services are having the great potentiality to handle the crisis management because it relies upon geospatial information to depict geographic distribution of events, their causes, affected people and infrastructure and available resources. But GIS web services composition is rarely utilized in real time crisis response. There is a great need for GIS web services composition to integrate different emergency services which are scattered (For example, Police, Fire and Medical). There is no efficient emergency response system with the GIS web services composition to save lives by providing timely help. By building an efficient emergency Geo-database, the various inter-related web services can be composed and used by the requesting system. For example, if there is a standard emergency Geo-database managed by the emergency response system, then it's statistical reports can be used by the third party services like Insurance (verification and claim), police (to identify the places more vulnerable for accidents, identifying high crime prone areas), media, etc., There is no perfect composition algorithm for managing such kind of Geo-database and for the integration of the associated systems using web services.

4. CONCLUSION AND THE PROPOSED WORK

In this paper, we have discussed about the various integration techniques and the recent trends in web service integration. Also, we have analyzed the various issues the research community has to focus on the web services integration research area. Our next focus is to analyze the various web service composition techniques and to propose an enhanced efficient web service composition technique for the integration of inter-related GIS web services.

5. REFERENCES

- [1] E. Cerami, Web Services Essentials – Distributed Applications with XML-RPC, SOAP, UDDI & WSDL., The O'Reilly Network, February 2002.
- [2] William Song, Deren Chen, Jen-Yao Chung, "Business Integration Models in the context of web services" e-Business Engineering, IEEE 2005
- [3] Mohammed Alrifai, Peter Dolog, Wolf-Tilo, Wolfgang Nejd , "Nonblocking Scheduling for Web Service Transactions", IEEE, European Conference On Web Services , p.213-222, 2007
- [4] Jin Back Kim, Arie Segav, Ajit Patanker, Min gi Cho "Web Services and BPEL\$WS for Dynamic eBusiness Negotiation Processes, CEMC/NGI 000104, 2004
- [5] Frank G. Goethals, Monique Snoeck, Wilfred Lemahieu, Jacques Vandenbulcke "Considering (de) centralization in a web services world" IEEE, International Conference on Internet and Web Applications and Services, 2007
- [6] Ying Huang, Jen-Yao Chung, Kuo-Ming Chao "A stochastic Service Composition Model for Business Integration" IEEE, International Conference on Next Generation Web Services Practices, NweSP'05.
- [7] Kwangcheol Shin, Sang Yong Han " Efficient Web Services Composition and Optimization Techniques" IEEE, International Conference on Web Services (ICWS 2007)
- [8] Carlos, Jose and Michael Gould "Incremental Composition of Geographic Web Services : An Emergency Management Context" 7th AGILE Conference on Geographic Information Science, 2004
- [9] Haiming Tang, Farong Zhong and Chang Yang " A Tree-based Method of Web Service Composition" IEEE, International Conference on Web Services, 2008
- [10] Web Services Mapping available at: http://www.altova.com/using_web_services.html
- [11] Witold Abramowicz, Konstanty Haniewicz, Monika Kaczmarek, Dominik Zyskowski "Automatic Web services interactions - requirements, challenges and limits from the F-WebS system perspective", IEEE, International Conference on Next Generation Web Services Practices, 2006
- [12] A. K. Elmagarmid. Database Transaction Models for Advanced Applications. Morgan Kaufmann, 1992.
- [13] G. Alonso, D. Agrawal, A. E. Abbadi, M. Kamath, R. Gnthr, and C. Mohan. Advanced transaction models in workflow contexts. In Proc. 12th International Conference on Data Engineering, New Orleans, February 1996.