

Chapter 2.6

Service Composition Based Software Solution Design: A Case Study in Automobile Supply Chain

Tong Mo

Harbin Institute of Technology, China

Jingmin Xu

China Research Laboratory, China

Zhongjie Wang

Harbin Institute of Technology, China

Yufei Ma

China Research Laboratory, China

Heyuan Huang

China Research Laboratory, China

Yuan Wang

Harbin Institute of Technology, China

Ying Liu

China Research Laboratory, China

Jun Zhu

China Research Laboratory, China

Xiaofei Xu

Harbin Institute of Technology, China

ABSTRACT

Service composition and Web mash-up are promising for meeting specific business requirements by integrating multiple distributed services and SaaS (Software as a Service) has become a popular way of software development and delivery paradigm. An application that meets the business challenges the customer faces in managing its supply chain by integrating two existing SaaS offerings into a newly developed technology presented to validate the service composition technologies. This paper contributes with a practice of a light weight approach of the problems that are inevitable in SaaS integration such as data synchronization, process control, and identification. The purpose of this paper tries to provide a reference for researchers and engineers in this domain.

DOI: 10.4018/978-1-61350-456-7.ch2.6

INTRODUCTION

SaaS (Software as a Service) is gaining a great deal of attractions today. More and more businesses are adopting SaaS for cost-effective software management solutions as well as business structure and process transformations (Koenig, 2006). The accelerating rate of SaaS adoption points clearly to the need for deeper integration with other enterprise applications, as well as other SaaS applications. This is one of the major IT concerns about SaaS viability. One potential solution for this emerging challenge is what Saugatuck calls the “SaaS Integration Platform” (West, 2006).

Some SaaS players are actually moving in the direction to various extents such as Jamcracker, nSite, Opsource, Salesforce.com, etc. The most notable one is the Salesforce’s unique multi-tenant platform, which provides different levels of integration capabilities based on its fundamental Web Services APIs (Salesforce.com). However, one common issue for these integration platforms is their self-centric design pays less attention to the integration with SaaS offerings from different providers. The only capabilities currently provided to connect to other external SaaS offering are Web Service calls. This is far from enough to address the issues including SSO (Single Sign On) and Web Service security caused by different identity management systems, service coordination for ensuring the integrity of business transactions across multiple SaaS offerings, etc.

Traditional middleware providers usually have a more complete view on integration, but the lacking of experiences in SaaS and lacking of industry standards and reference architectures currently available for SaaS integration make it difficult for them to get into the space. One of the challenges they are facing is how to radically simplify their products with significantly reduced footprint if they want to have their platform hosted as a service for a massive number of SMBs (Small and Medium Business).

This paper presents an SOA and Web 2.0 (Wiki) based lightweight approach for SaaS integration. It covers three levels of integration capabilities - presentation, function and data - with end to end security considerations. Most importantly, the approach allows SaaS providers to publish the integration specification of their SaaS offerings in a standard format. SaaS integrator can check out these standard integration elements and compose them together with drag-drop, point and click (DDPC) development tools. This makes the SaaS integration much easier than ever. The adoption of SOA makes the integrated solutions much more flexible and adaptive to changes, thus provides a level of business agility that was impossible to achieve before.

The paper presents a case study showing how we take the approach to compose two SaaS offerings using some of the key technologies we developed so far. Requirements of the case are picked up from supply chain management domain of a real-world automobile manufacturer, whose objective is to improve efficiency and quality of supply chain operation and delivery in its daily business, by importing a new SaaS named SNAP and integrating it with its existing SCM system.

The paper first gives an overview about the real business scenario and customer’s real requirements. Based on the analysis of this scenario, a tentative solution is proposed for the composition of two SaaS offerings. Then the issues surrounding the architecture design are discussed and related technologies and their implementation in the integrated system are introduced. At last, the case study is summarized and future works for improvement is briefed.

10 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/service-composition-based-software-solution/62447

Related Content

A Comparative Study on DNA-Based Cryptosystem

Thangavel M., Varalakshmi Pand Sindhuja R (2018). *Cyber Security and Threats: Concepts, Methodologies, Tools, and Applications* (pp. 1479-1510).

www.irma-international.org/chapter/a-comparative-study-on-dna-based-cryptosystem/203572

A Review of Literature About Models and Factors of Productivity in the Software Factory

Pedro S. Castañeda Vargas and David Mauricio (2021). *Research Anthology on Recent Trends, Tools, and Implications of Computer Programming* (pp. 1911-1939).

www.irma-international.org/chapter/a-review-of-literature-about-models-and-factors-of-productivity-in-the-software-factory/261109

Programming Languages as Mathematical Theories

Raymond Turner (2012). *Computer Engineering: Concepts, Methodologies, Tools and Applications* (pp. 1706-1722).

www.irma-international.org/chapter/programming-languages-mathematical-theories/62539

How to Use Information Technology Effectively to Achieve Business Objectives

Antonio Goncalves, Natália Serra, José Serra and Pedro Sousa (2012). *Computer Engineering: Concepts, Methodologies, Tools and Applications* (pp. 19-35).

www.irma-international.org/chapter/use-information-technology-effectively-achieve/62432

Impact Analysis of Intelligent Agents in Automatic Fault-Prone Components Prediction and Testing: Impact Analysis of Intelligent Agents in Test Automation

Jeya Mala Dharmalingam (2018). *Multidisciplinary Approaches to Service-Oriented Engineering* (pp. 287-316).

www.irma-international.org/chapter/impact-analysis-of-intelligent-agents-in-automatic-fault-prone-components-prediction-and-testing/205304