Situational Enterprise Services

Paul de Vrieze
Bournemouth University, UK

Lai Xu
Bournemouth University, UK

Li Xie
GuangDong Polytechnic Normal University, China

INTRODUCTION

The ability to rapidly find potential business partners as well as rapidly set up a collaborative business process is desirable in the face of market turbulence. Collaborative business processes are increasingly dependent on the integration of business information systems. Traditional linking of business processes has a large ad hoc character. Implementing situational enterprise services in an appropriate way will deliver the business more flexibility, adaptability and agility.

Service-oriented computing paradigm continues to transform traditional enterprise systems from close, centralized control systems into systems that feature dynamic information exchange and flexible business processes. Traditionally enterprise applications are defined as software designed to integrate all aspects of a company's operations and processes such as accounting, finance, human resources, inventory control, manufacturing, marketing, sales, and distribution, and resource planning. Advanced enterprise applications provide linkages with customers, business partners, and suppliers (Markus & Tanis, 2000). Normally enterprise applications are complex. These are mission critical applications which are developed, deployed and maintained by central IT with long development deployment cycles and dedicated IT budgets. A significant part of the value of these systems derives from links among systems. When these components were not initially designed for linking, this often makes the resulting systems brittle due to the workarounds used (de Vrieze et al. 2011).

At the same time, the current highly dynamic business environment provides increased pressure to build enterprise applications quickly in order to respond in an agile way to these situational needs of the business. Unfortunately, many of these applications (reflecting situational business needs) never get delivered because they are too difficult to write, too costly to implement, and too brittle to customize and maintain once deployed. As a result, many of the needs of the business are addressed by business people who have some knowledge on IT techniques and who devise ad-hoc partial solutions using tools like Excel, Access and Visual Basic. These solutions are often inadequate from many perspectives, including manual processes needed, maintenance and visibility.

With a growing number of services on the Web, these situational needs can now be satisfied more easily and effectively. These development and deployment services, combined with a "situational" mindset and methodology, can offer significant advantages. Unlike traditional enterprise applications, situational enterprise applications are relatively simple. They are not missioning critical for organizations. Lots of them can be developed at the point of need, with short development cycles, outside the

DOI: 10.4018/978-1-4666-9787-4.ch060

direct central IT control, and with little or no recognized budget. The centralized platform at the same time does allow for corporate monitoring.

These situational enterprise applications will not replace core business applications, such as ERP (Enterprise Resource Planning), SCM (Supply Chain Management), CRM (Customer Relationship Management) etc. They address different needs which are built for just handfuls of users, and may be used for only a few weeks or months, or to address a small piece of functionality. For example, within the perimeter ERP applications, departmental operational solutions, such as vacation scheduling, seminar and presentation management, purchase procedure management within a work unit, etc, normally are not included in an corporate ERP system. The functionality can however be desired by department staffs. From a corporate government perspective, the use of "free" external solutions is problematic, and as such this a typical situational applications for the department staff who manage those matters on a daily basis.

The types of situational enterprise applications can be divided into data-oriented applications and process-oriented applications. Enterprise widgets, gadgets, pipes and mash-ups belong to data-oriented applications. Lightweight process-oriented applications have been, and continue to be under research. The EU project SOA4All (http://projects.kmi.open.ac.uk/soa4all/) aims to provide a platform to build process-oriented applications for end users (non-technical users). The latest SOA4All software technologies can be found from http://technologies.kmi.open.ac.uk/soa4all-studio/.

The target audience for situational enterprise applications/services is an educated professional (e.g., accountant, HR personnel) with modest computer literacy (and interest) that mostly includes the Web and MS Office. These professionals have basic computer experience like using a wizard to generate something new; interacting with spreadsheets, documents, and forms; and using drag and drop to rearrange items on the screen.

These solutions on demand will help businesses slash expenses and reduce cycle times by more effectively supporting how people work, address challenges and make business decisions. Situational enterprise applications/services will allow also the business to be more innovative and competitive by supporting new processes more effectively, increasing overall productivity, and facilitating new ways for sharing information.

In this chapter, we introduce background information of situational enterprise applications, general description of Service-Oriented Architecture (SOA), Web services, and enterprise services in Section 2. The needs and benefits of situational enterprise services describe in Section 3. Design principles of situational enterprise services present in Section 4. Section 5 explains applying issues of situational enterprise services. Conclusions and future research direction are depicted in Section 6.

BACKGROUND

Relation between SOA, Web Services, and Enterprise Services

The Service-oriented architecture (SOA) paradigm is a design paradigm in which designers use loosely coupled services for building complex services or for incorporating them into systems. In the design of these systems, an ideal level of abstraction is required to align business needs and technical capabilities, and to create reusable, coarse-grained business functionality. SOA is not just an architecture of services seen from a technology perspective, but it includes the policies, practices, and frameworks by which it is ensured that the right services are provided and consumed (Rosen, Lublinsky, Smith & Balcer, 2012).