Enterprise Application Service Model

George Feuerlicht
University of Technology, Sydney, Australia

Jiri Vorisek
University of Economics, Prague, Czech Republic

INTRODUCTION

Following the recent changes in the global business environment, many organizations are reevaluating their approach to delivering enterprise applications and are looking for more effective ways to control IT costs. There is growing evidence of reluctance to fund large-scale implementation projects, and of tighter budgets forcing more careful cost-benefit analysis to justify IT investments. It is becoming increasingly clear that the traditional model for delivering enterprise applications that involves the implementation of licensed software such as ERP (enterprise resource planning) applications within end-user organizations is not suited to the fast-evolving business world of the 21st century. Almost invariably, situations in which organizations own and maintain their entire IT infrastructure lead to very high costs of ownership, and consequently high levels of IT spending, which can detract from the core business in which the organization is engaged. This has led to a situation in which some businesses doubt the benefits of IT (Carr, 2003), and some observers even contend that productivity improvements, once assumed to be the result of IT, are more likely to be the results of other factors such as longer working hours (Nevens, 2002). This backlash that followed the IT boom at the end of the last century has forced software vendors to seek more cost-effective models for the delivery of enterprise applications, and has led to the reemergence of the ASP (application service provider) model as an alternative to licensed software. Today, the ASP model (or software-as-a-service model) is a part of a more general trend toward utility computing, where the service provider delivers highly scalable application services to a large population of end-user organizations in a reliable and cost-effective manner, typically from a remote data center. Utility computing aims to supply application services on demand, similar to other utility services (e.g., gas or electricity), and relies on new technologies and architectures that enable the virtualization and sharing of resources across a large number of users in order to minimize costs and maximize utilization. The use of advanced service-oriented architectures (SOAs), grid computing, cluster technologies, and failure-resistant configurations enable the delivery of highly scalable application services in a reliable manner to a large population of users. These technological advances distinguish utility computing from the earlier ASP and outsourcing models, and will ultimately result in significant reduction in the costs of enterprise software solutions and wide adoption of the software-as-a-service model. Major IT vendors including IBM, Microsoft, Sun, Oracle, and HP are promoting utility computing, albeit under different names (e.g., on-demand computing, etc.), and are investing vast resources into the construction of data centers and related facilities (Abbas, 2003). Others, such as Salesforce.com, have been successful with providing hosted services for CRM (customer-relationship management) and other related types of applications, validating the ASP model and further confirming the trend toward utility computing.

As the enterprise application software market matures, major ERP vendors are changing their revenue model to decrease their reliance on new software licenses toward income generated from software-license upgrades and product support (Karpecki, 2004; Levy, 2004). This change combined with the fact that most organizations spend as much as 80% of software-related costs on software maintenance and related activities (Haber, 2004) creates a situation in which licensed software is de facto rented. It is precisely this high level of ongoing costs that motivate many organizations toward alternatives such as outsourcing and the ASP model.

In this article we first examine the business drivers for the ASP model and contrast the software-as-a-service model with the traditional software-as-a-license approach. We then discuss future enterprise computing trends, focusing on the reemergence of the ASP model for enterprise applications and the likely impact of the wide adoption of this model on the IT landscape. In conclusion, we summarize the main arguments in this article.

BACKGROUND

The economic downturn at the beginning of this decade resulted in organizations dramatically reducing IT bud-

gets, leading to scaling down existing projects and in some cases discontinuing projects altogether. In this section, we consider the background of these developments and the main business drivers that are forcing the transition to a new model for the delivery of enterprise applications as services.

High Cost of IT Projects

Problems of controlling the costs associated with IT projects are well documented. Notwithstanding the long experience that the IT industry has with the implementation of enterprise applications, costs of many projects significantly exceed their original budgets. According to a study of ERP implementation projects of 117 U.S. companies, 25% exceeded their budgets, 20% were abandoned before completion, and 40% failed to achieve business objectives (Cooke, Gelman, & Peterson, 2001). There have been other studies of this type that clearly demonstrate that the traditional model that involves the in-house implementation and maintenance of enterprise applications is associated with significant risks that are not being addressed by new, improved implementation methodologies and more technologically advanced software platforms. Equally, there is ample evidence that the outsourcing of the implementation and support of enterprise applications to a third party does not always bring the anticipated benefits (i.e., cost savings, improved responsiveness to new requirements, etc.); this is most likely because the implementation methodologies and technology architectures used by outsourcing organizations are essentially the same as those used by end-user organizations.

Fast Rate of Technology Change

Another significant risk factor associated with enterprise applications is the rapid development of underlying technologies, often necessitating the time-consuming reengineering of applications and costly upgrades. There is growing evidence that end-user organizations are no longer able to keep pace with new technological developments as delivered by IT vendors. New technology platforms and new versions of enterprise applications are often mandated by vendors who are reluctant to support older versions of their products, delivering no direct business benefit to end-user organizations.

High Demand on IT Skills

The traditional licensed-software model is associated with the high demand on IT skills needed to implement enterprise applications. Many small and medium-size enterprises (SMEs) cannot justify the cost of retaining their own internal IT staff with the appropriate expertise. Another contributing factor is that the expertise of IT specialists employed by end-user organizations and third-party consulting companies is not fully up to date and often significantly lags behind the expertise available from technology vendors. This factor leads to poor implementation results and is a major cause of the high rate of failure of IT projects.

Complexity of Enterprise Applications

Enterprise applications are becoming increasingly more sophisticated and complex. This complexity is particularly evident in large (horizontal) ERP application systems that provide comprehensive functionality designed to address requirements across a range of end-user organizations irrespective of the industry and the needs of individual businesses. While providing a complex and comprehensive solution, the actual utilization of the overall functionality of an ERP system by individual end users is relatively low. Customization to suit the needs of individual client organizations requires a high level of expertise and often involves the setting up of a large number of configuration parameters. Limited knowledge of the client's business processes by third-party consultants is another key factor, according to recent studies, that inhibits the successful implementation of ERP systems (Karpecki, 2002). ERP systems are characterized by high complexity of operation, even in situations in which the corresponding business process is relatively simple, and that leads to a high cost of end-user training. Because of the increasing size and complexity of new software versions, there is a corresponding growth in the demand on hardware resources. All these factors lead to increased implementation costs of enterprise applications that the client organizations are no longer willing to accept.

Globalization of the Business Environment

Globalization impacts enterprise applications in two important ways. First, as a result of globalization and the formation of regional economic blocks with standardized business practices and regulations, ERP applications can be used across a larger (international) user base without extensive customization to suit individual countries. For example, in the European Union, with the growing number of member countries using similar business laws and regulation, enterprise applications are becoming standardized across the entire region. Second, the global