Semantically Integrated Business Applications for Enterprise Resource Planning Systems

Patricia Kraft and Rainer Thome

Julius-Maximilian-University, Chair of Business Information Systems,
Wuerzburg, Germany
{pkraft,thome}@wiinf.uni-wuerzburg.de

Abstract. The development of enterprise resource planning (ERP) software or enterprise information systems (EIS) has always been closely linked to the discussion on integration. Over time, integration of software solutions has improved considerably. This has been achieved through several changes made in the past, but mostly it has become possible thanks to cutting-edge technologies. A new approach is semantically integrated business applications that enable the evolution of standard end-to-end software.

Keywords: Business applications, software development, enterprise resource planning (ERP), integration, add-on, enterprise information systems.

1 Introduction

The development of solutions for supporting the processes and organization of companies with enterprise resource planning (ERP) software has always been accompanied by a call for more integration. The ERP systems are the target of a great many demands, all of which various advocates expect the systems to meet. Depending on the specific point-of-view, some demands seem more pressing than others. These include calls for more flexibility, because in adapting to the real world, organizations are subject to constant change. But software should support change rather than be so inflexible as to hinder it and reduce an organization's competitiveness. Costs are also an issue. Enterprise software is not a primary driver of profit. As a result, companies strive to keep the cost of their information system as low as possible. A third point of conflict is functionality. Organizations' user departments raise this point because to them, the system is only beneficial if it can model all relevant corporate processes. But additional functionality increases complexity, compromising a system's transparency and rendering it inflexible. What is more, functionality is the main cost driver. Integration receives the weakest representation in this discussion. It is often assigned the lowest priority because its advocates are few and have little influence. In an ERP system, integration (of technical aspects and of applications) is the key to the smooth operation and modeling of processes, decreased maintenance costs and consistent data pools. Integration can also include

M. Kurosu (Ed.): Human-Computer Interaction, Part III, HCII 2013, LNCS 8006, pp. 411–417, 2013.

the look and feel of the user interface, providing consistency for the user. But these points are not concrete, cannot be implemented immediately, are difficult to quantify, are not a priority for management and are therefore at a disadvantage when compared to the interests previously mentioned.

This paper will aim to answer the question: What changes will cloud-based solutions bring to ERP software and how will these changes influence integration of software development long-term? First, an overview of the research done on integration of ERP solutions will be presented and the most recent developments in cloud-based ERP systems described. Next, semantically integrated applications, an integrated method for developing business application software, will be introduced.

2 State of the Art

FINK and HANSEN/NEUMANN define the term enterprise resource planning (ERP) system as a multiple-component business application system for end-to-end information processing. These systems are used to model and execute processes in all user departments of an organization (e.g. Production, Procurement, Accounting, etc.). In addition to other manufacturers, such as Oracle and Microsoft, SAP is the worldwide market leader for business application software both for large companies and for small and mid-size businesses [1][2].

Development in the area of *integration of business application software* up to the present can be divided into phases or blocks along a timeline. HANSEN and NEUMANN created a breakdown (Table 1) that divides development into five distinct phases. These five phases reveal development toward increasingly open platforms, which can communicate with each other via the Internet and provide a network that extends beyond an organization's borders. In 2005, when this hypothesis was posed, cloud solutions were not prevalent for ERP software. This thought has only begun to push its way into the conversation since a range of offerings in this sector has been launched onto the market and gained widespread popularity.

Software as a service can be described as a software delivery and business model by which the software vendor enables customers use of the standard software applications via the Internet as a service. In contrast to traditional on-premises software applications, the customer does not acquire a software license, but subscribes to the service. The software itself is hosted by the software vendor and does not require implementation or deployment at the customer site. The specific pricing model for SaaS is commonly based on a usage metric, such as per user per month." [3]. Solutions also exist as part of the platform as a service model. In this case, the object of examination is not just software but a platform that unites multiple offerings and vendors selling their cloud solutions and services. In other words, the platform is actually the model of a partner ecosystem that delivers additional offerings for the solutions available [4].

Table 1. Integration phases of enterprise software [2]

	Phase 1 1960-1980	**Phase 2 1980-1990**	**Phase 3 1990-2000**	**Phase 4 2000-2010**	**Phase 5 2010-2020**
Area	Isolated functions	Application areas	Internal business processes	Cross-company processes	Extensive corporate networking
IT technology	Customer-specific programming, classic data organization for proprietary platform	One manufacturer's proprietary application software on a few proprietary platforms	One manufacturer's proprietary application software on many proprietary platforms	One manufacturer's proprietary application software on many proprietary or open platforms	Combination of several manufacturers' application software on open-source platforms
	Terminals connected to mainframe computers	Terminals connected to mainframe computers	Client-server system	Client-server system, Internet-based	SOA system, web services
	Text-oriented UI	Text-oriented UI	Graphic UI	Web-based UI (browsers)	Web-based UI (browsers)

Up to now, the prevailing systems on the ERP market have been on-premises solutions. The customer installs, maintains and adds to these, if more functionality is desired. These add-ons are normally intended for a single customer, since each customer has his or her own requirements. Add-ons are either programmed by the customers themselves, or customers hire a software manufacturer to develop a solution. This gives rise to a number of problems: it compromises future ERP software releases, on the one hand, and on the other, customer-specific programming of the solution causes maintenance costs to skyrocket. All changes to the ERP system that go beyond those made via customization settings (adjustment of parameters) hamper flexibility and complicate maintainability. This knowledge leads to the demand for standard software and the adjustment of organization and processes to the ERP solution within its scope of performance. The continuation of this thought results in Continuous System Engineering (CSE) by THOME, the ongoing development of application software in interaction with the changes in corporate organization [5].

In the past decade, the issue of integration has been examined by a number of authors but never defined completely and conclusively. Many specialists in the field have attempted to quantify integration, and have illuminated it from various angles. MERTENS, for example, makes a distinction between the range, direction, object and scope of integration [6]. HAHN advocates a different approach. He distinguishes among data, model, and process integration. These levels can be closely or loosely

linked to one another [7]. JANSEN and HUFGARD introduce the business aspect. HUFGARD et al. maintain that the term business integration includes technical aspects of information processing, as well as organizational and marketing aspects [8]. JANSEN's "six target levels of integration" comprise strategic, organizational, human resources, cultural, operative and external levels of integration [9]. This diversity of definitions makes clear the complexity of the term integration and its far-reaching influence on an organization's processes. In sum, one can say that for integrated business processes, it is absolutely essential to be able to:

- access all relevant data
- save it without redundancy immediately after its creation,
- update software to state-of-the-art levels, and
- communicate digitally with other business partners across systems and company borders.

The user may not be impacted by changes in the system or in the look and feel of the user interface and yet, adjustments made necessary by organizational or business-related changes must be integrated into the system quickly and seamlessly.

3 Cloud ERP Systems Today

When comparing simple, cloud-based solutions to their locally installed competitors, one might wonder whether saving data and applications on external servers makes a difference with respect to integration. A cursory glance can lead to suspicions that cloud-based solutions are merely a marketing campaign and that the flashy new façade of cloud products conceal only externally-hosted variations of existing solutions. At first glance, the disadvantage of saving data in the cloud is that customers lose data sovereignty over their most valuable corporate information. But this argument can be refuted by the simple reflection that most organizations that purchase cloud applications are small and midsized businesses that do not have the necessary resources to operate their own high-security data centers. In this case, the data is probably safer in the hands of an external service provider than on the company's own premises. Another question has to do with workflow. When a web browser is used, it might seem as though users are more remote than when they use an in-house solution. But really, accessing an ERP system via the Internet is simpler and more secure than accessing an internal solution that is purposely blocked off from the external world and can be accessed only remotely via secure VPN entry points, if at all. The result is that, for example, sales reps can better access a cloud solution from a customer's premises than access a solution installed solely within their own company. SaaS improves networking and availability of all corporate data.

Regardless of the availability of and security issues concerning these solutions, the software structure holds the key to groundbreaking changes. Popular solutions on the market, such as NetSuite and Salesforce.com all offer the benefits of a web application, but behind their façades, they are nothing more than conventional software packages, i.e. software as a service [10]. But SAP's new Business ByDesign solution has a completely different structure; it is a whole new type of application.

At the heart of SAP Business ByDesign is Business Configuration, a catalog for setting all parameters of the ERP solution. It forms the basis for the application's configuration, the modeling of each customer's requirements. Users answer simple Yes or No questions to create, check and implement organization characteristics relating to system functionality. And then, depending on the answers to the questions, the application areas are parameterized with the help of a fine-tuning tool. It recognizes soft factors in the organization, migrates data and creates structures. This includes, for example, defining various models for employee working hours and migrating customer master data via an XML interface [11] [4].

Another unique component of ByDesign is the cloud-based development environment, Business ByDesign Studio. It can be used to develop add-ons that go beyond what is offered in the standard solution, to set up a link to external software, and to customize the standard software components. These two types of functionality differ very little from capabilities available in on-premises solutions. Customer-specific programming makes it possible to create each ERP solution in line with customer needs and to add functions. However, making changes in ByDesign via public solution models (PSM) is a completely different story. PSM contains a collection of data fields, methods and structures that programmers (even external programmers) can access via the ByDesign Studio development environment, in order to program additional functionality. PSM content adapts to future releases, which ensures that a one-off solution will still work after a release upgrade. Whenever a partner company develops an add-on for Business ByDesign, it automatically becomes part of the overall solution. The new components are sold in the SAP Store (a platform like an app store that offers add-on solutions as applications). When customers purchase a solution, it appears in their business configuration and they can make immediate use of it [4].

4 Infrastructure for Business Application Development and Integration of Applications

The approach just described for developing add-ons for a cloud-based solution – fusing add-ons with the host's infrastructure and enabling them to become part of the overall solution – offers a higher degree of integration from the perspective of all stakeholders than the approach previously used to develop add-ons for on-premises solutions. For the developer, using SAP software development kits (SDK) closely links new functions to the overall solution. The development environment is based on the PSM settings and contains specific patterns for creating the user interface that correspond to the solution and its outward appearance. The user cannot tell which screen is part of the standard solution and which has been added. The development environment ensures that all requirements are met for an integrated, compatible solution that will also work with future release upgrades.

For the customer, this approach improves business flexibility; the system offers multiple setting options, and add-ons from the SAP Store can easily be downloaded to it. The vast number of customers it benefits also makes it less expensive than company-specific developments for on-premises solutions. And, for the first time it has

made possible an objective comparison of alternative business applications on the market and their respective providers. If a solution is not available, a customer can ask an SAP partner company to create it. If a solution becomes obsolete because, for example, an organization has changed its structure, the user can very simply deactivate the solution. As a result, an organization is not weighed down with superfluous coding from one release to the next.

The provider and platform hosts face higher infrastructure costs by providing the solution, and lower maintenance costs with respect to new releases. "The SaaS vendor provides multiple tenants, one tenant per customer, based on a shared technology stack including hardware infrastructure, database and application" [3]. Integration of infrastructure is seamless because the tenants share the same hardware. Because all users are forced to upgrade to the new release during the two-to-three-month transition phase, a maximum of two releases are maintained – the current and the future release. The entire infrastructure is state-of-the-art and no legacy systems need to be maintained. All stakeholders are in the same boat. Web services support communication with external programs. They unite the communications structures and ensure semantically-correct transmission of data. Communication between various different business partners, all with separate ERP systems, is done this way, ensuring seamless communication across company and system borders.

If one compares this to the degree to which ERP systems are integrated today, marked discrepancies become apparent. The transition from the on-premises world to that of cloud-based solutions with the option to add semantically integrated business applications is comparable to the step from terminal computers to desktop computers. A stand-alone, partially linked solution becomes a completely integrated standard solution that grows in proportion to user demands.

In sum, semantically integrated business applications offer the following advantages:

- Development takes place within an SDK provided by the manufacturer,
- The user cannot distinguish between the standard solution and the add-on, with respect to the look and feel of the user interface,
- The add-on is an integral part of the functionality within the basic ERP solution,
- The new function does not compromise future upgrades for the overall solution,
- The solution is activated and configured along with the default functions (e.g. via Business Configuration),
- All customers who have the default system can use the solution,
- The solution is sold at a marketplace that is equally accessible to all customers.

Systems that meet all of the above-mentioned criteria are considered semantically integrated business applications for cloud-based ERP solutions.

References

1. Fink, A., et al.: Grundlagen der Wirtschaftsinformatik, p. 207. Physica, Heidelberg (2005)
2. Hansen, H., Neumann, G.: Wirtschaftsinformatik 1, p. 529. Lucius & Lucius, Stuttgart (2005)

3. Bandulet, F., et al.: Software-as-a-Service as Disruptive Innovation in the Enterprise Application Market. In: Benlian, A., et al. (eds.) Software-as-a-Service. Anbieterstrategien, Kundenbedürfnisse und Wertschöftungsstrukturen, pp. 16–17. Gabler, Wiesbaden (2010)
4. Schneider, T.: SAP Business ByDesign Studio. Application Development, pp. 24–28. Galileo Press, Boston (2012)
5. Thome, R.: Grundzüge der Wirtschaftsinformatik, pp. 149–159. Pearson, Munich (2006)
6. Mertens, P.: Integrierte Informationsverarbeitung 1 - Operative Systeme in der Industrie, p. 2. Gabler, Wiesbaden (2007)
7. Hahn, A.: Integrationsumgebung für verteilte objektorientierte Ingenieursysteme, pp. 7–23. Universität Paderborn Heinz Nixdorf Inst., Paderborn (1998)
8. Hufgard, A., et al.: Business Integration mit SAP Lösungen, pp. 15–16. Springer, Berlin (2005)
9. Jansen, S.: Mergers & Aquisitions: Unternehmensakquisitionen und -kooperationen. Eine strategische, organisatorische und kapitalmarkttheoretische Einführung, p. 330. Gabler, Wiesbaden (2008)
10. Salesforce: Salesforce.com., `http://www.salesforce.com/company/`
11. Hufgard, A., Krueger, S.: SAP Business ByDesign: Geschäftsprozesse, Technologie und Implementierung anschaulich erklärt, pp. 154–177. SAP Press, Bonn (2012)