

Towards Reuse-Oriented and Web-Based Collaborative Framework for e-Business Providers

Vladimirs Kotovs, Andrejs Lesovskis, and Leonids Novickis

Division of Applied System Software, Riga Technical University, Riga, Latvia
{vladimirs.kotovs, andrejs.lesovskis, leonids.novickis}@rtu.lv

Abstract. The need in systematic approach and effective software engineering methods that facilitate collaboration and allow reusing experience to address recurring problems successfully, is obvious and extremely important for e-business. The synergy of web-based and reuse-oriented methods for software solutions may create a new platform of e-business opportunities. This study describes the foundation of the framework for organizations that are moving towards a systematic reuse program and web-based information technology solutions paying special attention to the needs and the problems of e-business providers.

Keywords: e-business, software reuse, web-based collaboration

1 Introduction

Today e-business has emerged as a mainstream way to facilitate business transactions around the world. Web and Internet electronic business technologies promise to transform every aspect of the way companies conduct business and compete [1]. The Internet has fundamentally changed the economics of business transactions and provided new and different ways of conducting business. However, e-business segment of the particular domain a company is operating in is usually characterised by a very high competition. Large enterprises with their international networks have significant advantages over small and medium-sized companies. For individual small and medium enterprises (SME) it is difficult to solve the problems alone; however, by co-operation and efficient utilization of knowledge and resources, they can succeed and keep relatively big segment of the market.

The technologies enable people to communicate in new ways, provide new business models, permit businesses to operate more efficiently and take advantage of the new global network economy. Companies have come to realize that if they are going to respond rapidly and effectively to today's volatile marketplace, they need to do more than web-enable discrete systems, processes or business units. They need to pull together existing systems and integrate them securely with their core business activities [1].

At the same time in the current situation of financial crisis it is extremely important for applied technologies to provide some economic and organizational benefits by reducing development and operating costs, and by efficient utilization of development knowledge and corporate expertise. The need in systematic approach and effective software engineering methods, which allow reusing experience to address recurring problems successfully, is obvious and important for e-business providers.

With more than 40-year history reuse is recognized as an important mechanism to improve software quality and development productivity [2]. Today, we are just at the start of this challenge but we believe that the synergy of web-based and reuse-oriented methods for software solutions may create a new platform of e-business opportunities.

The paper presents the review of the ongoing joint research activities related to the introduction of systematic software reuse and web-based collaboration in the enterprise paying special attention to the needs and the problems of e-business providers. The following sections discuss current limitations and expected benefits of the research. It is followed by the review of proposed reuse-oriented and web-based collaborative frameworks. Finally, the conclusions are made and future work is discussed in the last section.

2 Expected benefits and the main drivers of the research

The following bottlenecks and limitations have been identified as the main drivers for our joint research activities:

- The existing IT solutions usually support separate stages of business processes and are not focused on integrated web-based solutions that provide an opportunity to manage core business activities within the whole company or across the entire value chain from customers to suppliers.
- Intermodal collaboration among e-business service providers usually isn't effectively organized.
- Existing efforts to attain the main objective of software reuse in the e-business domain rendered some successful stories, however, to benefit in more systematic and repeatable way, additional research is needed.
- Reuse requires up-front investments in infrastructure, methodology, training and tools without pay-offs being realized immediately and cannot be widely achieved in organizations without support of top-level management. However, reuse will only succeed, if it makes good business sense, and it will only be chosen, if a good case can be made that it is the best alternative choice for the use of capital [3].
- Reusing of large-scale system components is a problem too hard to be solved in general way. It is more effective when systematically applied in the context of families of related systems and thus is domain-dependent.

The expected advances and benefits of the research are the following:

- The reuse-oriented framework for organizations that are moving towards an effective reuse program composed of different aspects of software reuse and related practices. The final goal is to ingrain reuse into an organizations entire software production process [4]. A formalized process increases the chance that the project success can be repeated, facilitates adherence to the established best practices, standardization of practices across multiple projects, and helps less-expert developers to succeed via reliance on a standard process. Additionally, the framework expected to define effective measuring of economic benefits of the software reuse and development costs, where operational and strategic benefits are defined and quantified within the context of broader business strategy. However, each organization should analyse its own needs, define the key benefits, identify and remove impediments, and manage risks [5].
- The methodology for Internet-based Collaborative Framework (ICF) development is based on advanced web technologies. ICF provides the adaptation mechanism to distinct between different target groups (technology supplier, technology receiver, technology transfer facilitator). ICF is also expected to provide easy commercialization and adaptation of technology transfer process (Fig. 1).

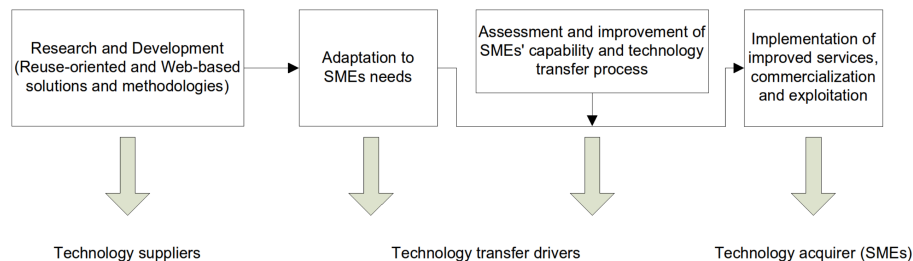


Fig. 1. General scheme of technology transfer process

3 Reuse-oriented framework for e-business providers

Software reuse is the process of creating software systems from existing software rather than building them from scratch, whereby an organization defines a set of systematic operating procedures to specify, produce, classify, retrieve, and adapt software artefacts with the purpose of using them in its development activities [6], so that similarities in requirements and/or architecture between applications can be exploited to achieve substantial benefits [7]. Our retrospective analysis of the origins and main contributions in the research area performed in [2] outlines

that software reuse is multidisciplinary and has deep and complex interactions with many areas of computer science. Table 1 consolidates the main benefits from [8] that can be achieved by the organizations with the maximization of the reuse level.

Table 1. Main benefits of reuse

Benefits of reuse	Description
Gains in quality	Quality of particular component could be improved because of accumulated error corrections
Gains in reliability	The use of a component in several systems increases the chance of errors to be detected and improves confidence in that component
Gains in productivity	Productivity could be achieved due to less design and code to be developed
Gains in performance	Extensive reuse can be worth the effort invested in optimizations, that may yield better performance of a reused component
Reduction of maintenance costs	Fewer defects and reduction of maintenance costs can be expected when proven quality components have been used
Reduction of product time to market	By using reusable artifacts organizations could reduce the product time to market which influences the success or failure of a software product
Rapid prototyping	Reusable components can provide an effective basis for quick building of a prototype of a software system

Directions and components for reuse-oriented framework comply with the following main conclusions that we have obtained through the survey [9] involving software organizations in Baltic regions, whose responses were analysed and used to relate the characteristics of organizations with their reuse experience:

- 72% participants claimed to succeed in projects by the means of software reuse in their organization, and small and medium software organizations presented higher reuse success rates than organizations of a large size.
- Organizations should focus on the development of product families if applicable in the operating business area.
- Attention should be spent on the introduction of process-driven systematic reuse in the organization.
- Organizations should consider using repository for storing and retrieving reusable assets, and a configuration management process should guarantee proper evolution of these assets.

The overall structure of the considered reuse-oriented framework is organized in economic (A), organizational (B), and process (C) dimensions as outlined on Figure 2.

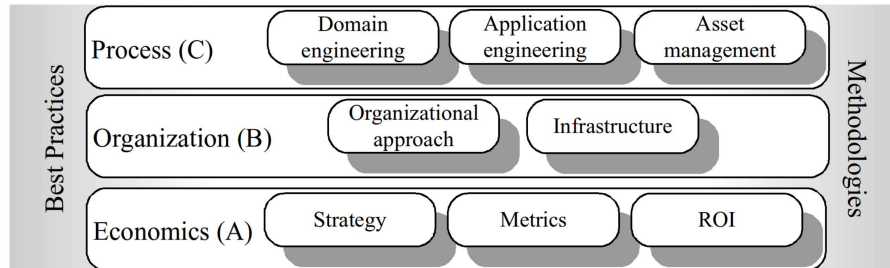


Fig. 2. The overall structure of reuse-oriented framework

(A) From the economic perspective implementing a reuse initiative in a corporate environment requires a decision about when and where capital investment for software reuse is to be made, and whether it will be proactive or reactive [3]. The measuring of economic benefits of the software reuse and development costs should be aligned with the following main principles from [10]:

- Economic value maximization drives reuse investment strategies for the business
- Strategy drives selection of reuse investments that are actively structured to maximize embedded strategic options.

Economic aspects of software reuse are divided into the following broad classes by [6]:

- Metrics, which reflect attributes that increase the market value of an asset
- Reuse cost estimation techniques and return-on-investment (ROI) models to quantify reuse related decisions.

(B) Several approaches exist for establishing a reuse program from the organizational perspective:

- Centralized, with an organizational unit dedicated to developing, distributing, maintaining, and providing training on reusable assets.
- Distributed asset development, where reuse is implemented collaboratively by projects in the same product line.

Reuse environment on the organizational level assumes development of

- Managerial infrastructure in the form of a set of functions, responsibilities, reporting requirements, and reward, which are required to ensure operation of reuse processes
- Technological infrastructure that includes a configuration management and quality assurance functions to support reuse operations, as well as enforcement of testing, verification and asset certification standards.

(C) Process-driven means that the software development is done in accordance with well defined processes that are enforced through management policies. A software process could be defined as a set of activities that lead to the production of a software product [11], and it is important in order to ensure efficiency, reproducibility, homogeneity, and predictable time and effort constraints. As it was noted already, a key concept of systematic reuse is the domain, which may be defined as an application area or, more formally, a set of systems that possess similar functionality and share design decisions. By domain-specific reuse we assume, that the reusable assets, the development processes, and the supporting technologies are appropriate to the application domain for which the software is being developed.

It is common to identify three main stakeholders in the process of software reuse within an organization: corporate management, domain engineering and application engineering teams. By reviewing the tasks of each of these stakeholders, the following top-level reuse related activities can be identified:

- Asset production - identification, development and classification of reusable artefacts
- Asset usage - locating and evaluating assets, and achieving their actual reuse by integrating them into applications being developed
- Asset management - asset storage, repository management, and asset dissemination
- Maintenance and support - support for asset usage, methodological support, and corrective and evolutionary maintenance of assets
- Reuse management - introducing and monitoring reuse within a company or department.

To support reuse, software processes have to consider two facets: developing for reuse and developing with reuse. As a result comparing to conventional monolithic set of tasks organizations are responsible for providing and maintaining software systems, software reuse introduces a differentiation between the tasks related to the production of reusable assets and the tasks related to the production of end-user applications [12]. Such two-life-cycle approach with generalized activities schematically displayed on Figure 3 is commonly referred as domain and application engineering, which address development for reuse and development with reuse respectively.

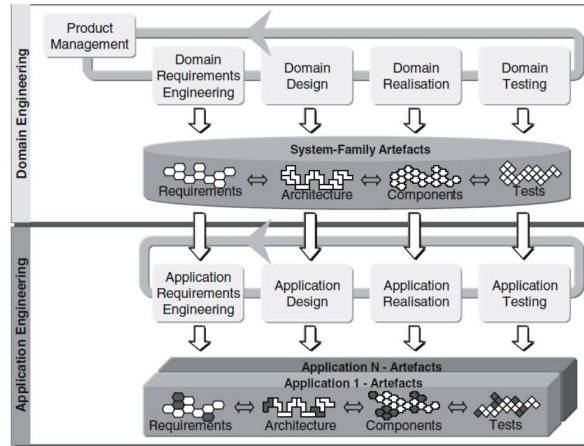


Fig. 3. The two-life-cycle model of domain and application engineering [13]

From the technical point of view the e-business provider development within the proposed framework should rely on the best practices and appropriate supporting methodologies facilitating global visibility, integration and collaboration as noted in section 1. The Internet-based collaborative framework (ICF) as discussed in the next section aimed to provide a facade for several advanced IT solutions in order to support e-business.

4 Internet-Based collaborative framework for e-business providers

E-business denotes using computer-mediated networks to perform a wide range of commercial transactions, to automate and transform business processes at the firm level, and to manage more effectively value chains across firms. In addition to e-commerce, customer service and support applications, e-business can include human-resource functions, sales-force automation, supply-chain management and other applications to automate firms' central internal processes [14].

The online environment that supports dynamic e-business should provide the following capabilities:

- Open application interfaces that provide interaction between business partners, service providers, and other participants;
- Open standards that define business transaction processes and would improve automation of business to business (B2B) interactions;
- Internet-based communication methods with opportunity to extend to other public or private systems;
- Easy adaptation to market condition changes.

Web services enable many of these key prerequisites for dynamic e-business solutions. They define a standard way to describe and invoke a Web service interface so that the business application can easily connect to it. In the report entitled Executive Survey: SOA Implementation Satisfaction, Hurwitz & Associates analysts point out that the top reasons for SOA adoption include the expectation of greater reuse in existing and newly-built Web services, business flexibility, ease of integration and speed of integration with nearly 90% of respondents pointing to service reuse as their number one concern [16].

According to World Wide Web Consortium (W3C), there are two major classes of Web services:

- REST-compliant Web services, in which the primary purpose of the service is to manipulate XML representations of Web resources using a uniform set of "stateless" operations; and
- Arbitrary Web services, in which the service may expose an arbitrary set of operations (most of the SOAP-based Web services) [15].

REST (Representation State Transfer) provides four familiar operations GET, POST, PUT, and DELETE of the HTTP standard, and make the resource addressing handle the variability. SOAP (Simple Object Access Protocol), on the other hand, permits quite a complex set of XML formatted commands and data transmission options. In contrast with REST which requires HTTP, SOAP messages can be moved by any transport method which can handle Unicode text. But HTTP transport is preferred by many developers because it is more familiar and SOAP with HTTP can tunnel easily over existing firewalls and proxies, without modifications to the SOAP protocol, and can use the existing infrastructure.

Technologies like SOA, Web services, Semantic Interoperable Services, XML, SOAP, and AJAX Web development techniques can be used as a technological platform for the integration. Using these technologies together with the existing enterprise web platforms and portal frameworks (like Liferay, GateIn, WebSphere Portal, JetSpeed, etc), it is possible to design the integrated portals that will provide seamless integration of the various business-related services and data sources. For instance, separate portals that support different logistics business processes along the selected freight route (www.elogmar-m.org, ww.containeronlineshop.de, www.e-56.com.cn) are united into one framework with a single entry point (Fig. 4) [17].

An enterprise portal, built with a robust portal product, provides an off-the-shelf framework for developing and deploying service-oriented applications. It can serve as a ready-made Web services consumer platform, and enables you to build composite applications, deploy syndicated content from other portals through remote portlets, replace/augment legacy interfacing applications, create common views of data, and facilitate access by mobile and wireless devices [18].

Most of the mentioned above portal frameworks also have personalization features. Personalization provides end-users with an opportunity to customize their version of a portal; thereby, greatly improving end-user performance and satisfaction. Using personalization mechanism, portal administrators can define

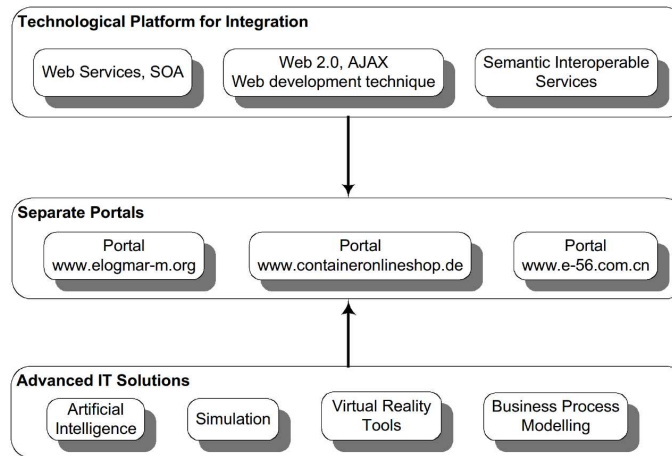


Fig. 4. Integration in ICF

specific groups which may represent portal users real-life roles (operator, manager, CEO, etc). Depending on their roles, users will have access to the features and content designed for their group.

One of the goals in our test case is the integration of eLogmar portal services with a newly designed portal. We are using NuSOAP Toolkit in eLogmar portal to provide SOAP-based access to system functionality (shipping lines services, freight cost calculation, etc).

Conclusion

The need in systematic approach and effective software engineering methods that facilitate global visibility and collaboration and allow reusing experience to address recurring problems successfully, is obvious and extremely important for e-business. The study presents the foundation of the framework for organizations moving towards a reuse program that is organized considering economic, organizational and process aspects of software reuse. Although the work might be still considered to be in its early stages, the study defines the basis, the components and directions for further research activities. Based on our experience in portal development and integration projects, we are currently working on consolidation of best practices and general recommendations for e-business providers. Another important issue concerns maturity of reuse processes without gaps in reuse activities covering steps of domain engineering, domain analysis, design and implementation. The planned future work is related to the formalization of the processual dimension of the proposed framework and methodological support of ICF. More attention is planned to be paid to the processes of identification of domain parts reasonable for automation and to systematic software reuse opportunities in the context of Model-Driven Development.

Acknowledgments. This work is partly financed by the European Union (European Regional Development Fund) within the Baltic Sea Region Programme 2007-2013 project BONITA (Baltic Organisation and Network of Innovation Transfer Associations). The present activity is partly funded by the ERDF project "Support of FP7 ICT STREP project "Simulation Highway"". This work has been supported by the European Social Fund within the project "Support for the implementation of doctoral studies at Riga Technical University".

References

1. Westermeier, J., Plave, L.J.: E-Business: The E-Business Legal Survival Kit (2004)
2. Kotov, V.: Forty years of software reuse. In: Scientific Journal of RTU. 5. series. Computer Science, vol. 38., pp. 153-160. RTU, Riga (2009)
3. Frakes, W. B., Kang, K. C.: Software Reuse Research: Status and Future. In: IEEE Transactions on Software Engineering, vol. 31., no. 07., pp. 529-536 (2005)
4. Frakes, W. B.: Software Reuse and Domain Engineering, <http://wfrakes.wordpress.com/category/introduction>
5. Frakes, W. B., Isoda, S.: Success Factors of Systematic Reuse. In: IEEE Softw., vol. 11., pp. 14-19 (1994)
6. Mili, H., Mili, A., Yacoub, S., Addy, E.: Reuse-based Software Engineering: Techniques, Organization, and Measurement. John Wiley & Sons Inc. (2002)
7. Ezran, M., Morisio, M., Tully, C.: Practical Software Reuse. Springer (2002)
8. Sametinger, J.: Software Engineering with Reusable Components. Springer-Verlag (1997)
9. Kotov, V.: Reuse In Software Development Organizations In Latvia. In: Scientific Journal of RTU. 5. series. Computer Science. vol. 43., pp. 90-96. RTU, Riga (2010)
10. Favaro, J., Favaro, K., and Favaro, P.: Value Based Software Reuse Investment. In: Annals of Software Engineering. vol. 5., pp. 5-52 (1998)
11. Sommerville, I.: Software engineering. 6th. Addison-Wesley (2001)
12. Ezran, M., Morisio, M., Tully, C.: Practical Software Reuse. Springer (2002)
13. Linden, Frank, Schmid, Klaus, Rommes, Eelco: Software Product Lines in Action: The Best Industrial Practice in Product Line Engineering. Springer (2007)
14. Canadian e-Business Initiative, 2004. The Digital Economy in Canada: The e-Business Environment, <http://www.ic.gc.ca/eic/site/ecic-ceac.nsf/eng/gv00501.html>
15. W3C Working Group, 2004. Web Services Architecture, <http://www.w3.org/TR/ws-arch/>
16. Baroudi, C., Halper, F.: 2006. Executive Survey: SOA Implementation Satisfaction, http://www.cio.co.uk/cmsdata/whitepapers/4474/110206_Mindreef_Report%5B1%5D%5B1%5D.pdf
17. Bluemel, E., Strassburger, S., Novitsky, L.: IT&T Solutions in Logistics and Maritime Applications. JUMI (2006)
18. Phifer, G.: A Portal May Be Your First Step to Leverage SOA, http://www.gartner.com/DisplayDocument?doc_cd=13014