

## Chapter 2

# Agile Software Development for Customizing ERPs

**Rogério Atem de Carvalho**  
*Fluminense Federal Institute, Brazil*

**Björn Johansson**  
*Copenhagen Business School, Denmark*

**Rodrigo Soares Manhães**  
*State University of North Fluminense, Brazil*

### ABSTRACT

*Customization of ERP systems is a complex task, and great part of this complexity is directly related to requirements management. In this context, a well-known problem is the misfit between the ERP functionalities and the business requirements. This problem comprises communication bottlenecks and difficulties on responding to changes. The proposals for minimizing these misfits are mostly focused on traditional, heavyweight waterfall-like approaches for software development. On the other side, the last decade has witnessed the rise and growth of Agile methods, which have both close communication and fast response to changes among their main values. This chapter maps some of the main agile practices to ERP customization processes, using, where applicable, practices from a real-world ERP project. Moreover, some limitations on the agile approach to ERP customization are presented and discussed.*

### INTRODUCTION

The last decades have witnessed two movements in industrial companies, namely Enterprise Resource Planning systems (ERPs) and Lean Manufacturing techniques. Lean techniques first appeared in the automobile industry within the so-called Toyota Production System, and afterwards were adapted and extended for application in practically all industry

segments, forming the basis of what is now called Lean Manufacturing. Lean techniques are based on the principle that activities not delivering value to the final product should be eliminated from the production process. The lean techniques also, instead of just trying to anticipate demand fluctuations, try to follow fluctuations, by the use of the “pull” effect, where instantaneous demand is propagated from clients to suppliers, not the opposite, as was the usual approach some decades ago.

DOI: 10.4018/978-1-61520-625-4.ch002

The term ERP was coined in the early 1990s by the Gartner Group (Wylie, 1990), and gained momentum in the years before the Y2K, considered the single event that signaled both the maturing of the ERP industry and the consolidation of large and small vendors (Jacobs & Weston, 2007). Blackstone and Cox (2005) define ERP as a “framework for organizing, defining, and standardizing the business processes necessary to effectively plan and control an organization so the organization can use its internal knowledge to seek external advantage.” ERP systems aim to realize integrated management, through the integration of business processes and the data they manipulate (Wier et al., 2007).

As was to be expected, some ERP vendors support Lean production planning and control techniques too, although some problems of this combination can be considered as still opened – since according to lean philosophy, some ERP features are too heavy (Seradex, 2009).

Development of ERP systems is an endeavor with a high level of complexity, and a great deal of this complexity is directly related to requirements management – being the misfit between ERP functionality and business requirements a “classical” problem (So, Kien & Tay-Yap, 2000). The problem of misfit means that there is a gap between functionality offered by the package and functionality required from the adopting organization. Askenäs & Westelius (2000) describe this in the following way: “Many people feel that the current ERP system has taken (or been given) a role that hinders or does not support the business processes to the extent desire”. Another way of describing this is as said by Bill Swanton, vice president at AMR research, that only 35 per cent of the organizations are satisfied with the ERP system they use at the moment, and the reason for this dissatisfaction is that the software does not map well with the business goals (Sleeper, 2004). It can be argued that the misfit results from deficiency in the requirement management process, in which business analysts and develop-

ers are supposed to agree on what functionality the ERP system should support (Johansson & Carvalho, 2009).

Johansson & Carvalho (2009) suggest that the misfit problem is related to the requirements management process used in ERP development. In fact, the communication between business people and the development teams to a high extent occurs in a waterfall-based software process, which creates a bureaucratic environment where communication is done by documents and not directly between people. Besides that, the ever-changing globalized business environment makes requirement changes could be seen as “natural” events. Therefore, it’s necessary to define a customization (and maintenance) process that simultaneously enhances communication between ERP stakeholders and copes with change. Another angle of this problem is that information systems, such as ERPs, deliver too much functionality, which overload users with information. The information overload results in that even if the system contains needed functionality, user may not know about it and have a hard time to find as well as understand how to use the specific functionality. This indicates that instead of talking about misfit or misalignment a better way of describing what is needed could be synchronization between stakeholders, or more specifically, business synchronization. From this it can be said that a focus on business synchronization in the form of describing how ERPs as well as organizations could be developed in tandem would be fruitful. This implies that a focus on how to achieve efficient communication between developers and users, and fast response to change is needed.

A way of both achieving an efficient communication and fast response to change is to deliver software in small and frequent increments, as advocated by Agile software development methods. Agile methods are a reflection of Lean Manufacturing techniques in the Software Engineering realm. Sutton (1996) pointed out that Lean can be a good middle ground between craft

18 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/agile-software-development-customizing-erps/42247](http://www.igi-global.com/chapter/agile-software-development-customizing-erps/42247)

## Related Content

---

### Enterprise Systems Outsourcing "Behind the Curtain": A Case Study Showing How Rational and Institutional Explanations Coexist and Complement Each Other

Per Svejvigand Jan Pries-Heje (2011). *International Journal of Enterprise Information Systems* (pp. 1-17).

[www.irma-international.org/article/enterprise-systems-outsourcing-behind-curtain/51614](http://www.irma-international.org/article/enterprise-systems-outsourcing-behind-curtain/51614)

### A Systematic Review for Predictive Models of IS Adoption

Rhouma Naceur, Yan Cimonand Robert Pellerin (2021). *International Journal of Enterprise Information Systems* (pp. 1-21).

[www.irma-international.org/article/a-systematic-review-for-predictive-models-of-is-adoption/268360](http://www.irma-international.org/article/a-systematic-review-for-predictive-models-of-is-adoption/268360)

### A Comparative Study of Different Approaches for Finding the Upper Boundary Points in Stochastic-Flow Networks

Seyed Mehdi Mansourzadeh, Seyed Hadi Nasser, Majid Forghani-elahabadand Ali Ebrahimnejad (2014). *International Journal of Enterprise Information Systems* (pp. 13-23).

[www.irma-international.org/article/a-comparative-study-of-different-approaches-for-finding-the-upper-boundary-points-in-stochastic-flow-networks/116764](http://www.irma-international.org/article/a-comparative-study-of-different-approaches-for-finding-the-upper-boundary-points-in-stochastic-flow-networks/116764)

### Service Computing for Design and Reconfiguration of Integrated E-Supply Chains

Mariagrazia Dotoli, Maria Pia Fanti, Carlo Meloniand Mengchu Zhou (2007). *Enterprise Service Computing: From Concept to Deployment* (pp. 322-355).

[www.irma-international.org/chapter/service-computing-design-reconfiguration-integrated/18488](http://www.irma-international.org/chapter/service-computing-design-reconfiguration-integrated/18488)

### Always-On Enterprise Information Systems with Service Oriented Architecture and Load Balancing

Serdal Bayram, Melih Kirlidogand Ozalp Vayvay (2010). *Always-On Enterprise Information Systems for Business Continuity: Technologies for Reliable and Scalable Operations* (pp. 91-108).

[www.irma-international.org/chapter/always-enterprise-information-systems-service/36593](http://www.irma-international.org/chapter/always-enterprise-information-systems-service/36593)