Available online at www.sciencedirect.com
ScienceDirect
The Journal of Systems and Software 81 (2008) 1899–1919
The Journal of Systems and Software
www.elsevier.com/locate/jss

A framework to support the evaluation, adoption and improvement of agile methods in practice

A. Qumer, B. Henderson-Sellers *

Faculty of Information Technology, University of Technology, Sydney, P.O. Box 123, Broadway 2007, Australia

Received 7 May 2007; received in revised form 12 December 2007; accepted 27 December 2007
Available online 12 January 2008

Abstract

Agile methods are often seen as providing ways to avoid overheads typically perceived as being imposed by traditional software development environments. However, few organizations are psychologically or technically able to take on an agile approach rapidly and effectively. Here, we describe a number of approaches to assist in such a transition. The Agile Software Solution Framework (ASSF) provides an overall context for the exploration of agile methods, knowledge and governance and contains an Agile Toolkit for quantifying part of the agile process. These link to the business aspects of software development so that the business value and agile process are well aligned. Finally, we describe how these theories are applied in practice with two industry case studies using the Agile Adoption and Improvement Model (AAIM).

Keywords: Agile methodologies; Framework; Transition to agile; Industry case studies

1. Introduction

Agile methods are often welcomed by both managers and programmers as providing a much needed release from the overheads typically perceived as being imposed by traditional software development approaches. Created in the context of small, greenfield projects, agile methods are often seen as unable to scale to larger situations (Greenfield and Short, 2004, page 123). Their adoption seems to need an all-or-nothing approach, suggesting that "being agile" is binary.

In practice, few organizations are able, psychologically or technically, to take on agile development approaches immediately and adopt them successfully over a short period – a full transition often taking a few years. Furthermore, it may be inappropriate for them to be fully agile in all aspects of development, perhaps retaining well-known and trusted elements of a more traditional approach within an overall agile project. One way to do this is by the use of situational method engineering (Henderson-Sellers, 2002, 2003). But even then, the method engineer and the software development manager may be unsure how to identify how to adopt agile methods incrementally, which bits to choose as most appropriate for their situation, how to engender enthusiasm in team members (Syed-Abdullah et al., 2007), how to ensure that their adopted method can mature and grow as the development team's skills mature and how to ensure that the whole of the development team don't succumb to the inherent desire of humankind to "resist change" (Henderson-Sellers and Serour, 2005).

In this paper, we introduce a complete framework to assist managers in assessing the degree of agility they require and how to identify appropriate ways to introduce this agility into their organization, illustrated with some industry case studies. Section 2 describes the Agile Software Solution Framework (ASSF). A major element of the ASSF is the Agile Toolkit, which is discussed in Section 3. A second, and as yet unexplored, element of the ASSF is governance – the topic of Section 4. Section 5 discusses issues relating to the adoption of an agile process in

* Corresponding author. Tel.: +61 2 9514 1687; fax: +61 2 9514 4533.
E-mail address: brian@it.uts.edu.au (B. Henderson-Sellers).

0164-1212/$ - see front matter © 2008 Elsevier Inc. All rights reserved.
doi:10.1016/j.jss.2007.12.806

industry situations and how the ASSF can help, linking the approach to software process improvement concepts. In Section 6, we illustrate how the ideas propounded in Section 5 regarding the Agile Adoption and Improvement Model (AAIM) are enacted on two real industry case studies; before concluding in Section 7.

2. Agile software solution framework

With the above questions in mind, we have developed an Agile Software Solution Framework (ASSF). Fig. 1, at a very abstract level, represents the components of the ASSF and their relationships. The elements of the ASSF can be classified in terms of agile conceptual aspect model (agile as a characterizing-noun) and tools. The agile conceptual aspect model represents the aspects of knowledge, governance and method core; these three being linked to business via a business-agile alignment bridge or business value. This bridge has an impact on governance, which in turn shapes an agile software development method (construction and application), in terms of the business value it delivers. Business represents the software development organization. The remaining 'abstraction' element of the ASSF represents the type of abstraction in use (e.g. object, agent, service) or the associated software technology. The Agile Toolkit (Qumer and Henderson-Sellers, 2007c) provides an application of these ideas in practice (construction or tailoring of software processes), whereas the embedded analytical tool (4-DAT; four-dimensional analysis tool) is specially used as a quality evaluation measure to evaluate the degree of agility in software development practices. The 4-DAT (as a quality evaluation measure) has already been tested, documented and published, and the details can be found in Qumer and Henderson-Sellers (2007a).

Furthermore, in Fig. 1, Method Core represents the different but related core aspects of an agile method (a.k.a. methodology): agility, people, process, product and tools (agile workspace), which can be combined by using a method engineering approach for the construction of agile situation-specific methods to achieve the desired business value. The purpose of the ASSF (a vision-guiding model) is to guide the behaviour of self-organizing and empowered agile teams with a cohesive set of shared information in large and complex project development environments. The ASSF is also a first step towards the future development of a meta-model for agile software development methodologies – although there are a number of meta-models (OPF: Firesmith and Henderson-Sellers, 2002; Standards Australia, 2004; ISO/IEC, 2007) for traditional approaches to software development, they do not explicitly discuss the new aspect of agility, which is an essential aspect for an agile software development method. According to our survey and focus groups, it is also found that existing standards do not explicitly discuss the aspect of abstraction either – a critical aspect of any software development methodology (Qumer and Henderson-Sellers, 2006d). The following sub-sections precisely describe the main parts of the ASSF, two of which (the toolkit and governance), being the least well documented in the literature, are described in more detail in Sections 3 and 4.

2.1. Method core

The absence of a shared or common vision is one of the main factors of software project failures. As noted above, the "method core" and abstraction elements of ASSF (Fig. 1) represent six aspects of an agile software development methodology: agility, process, people, product, tools and abstraction. This set of aspects attempts to provide a

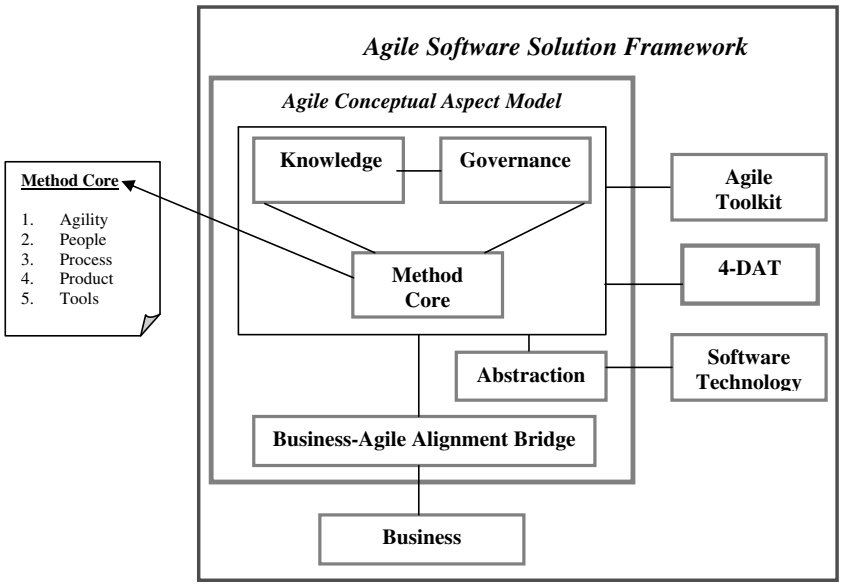

Fig. 1. The main components of the agile software solution framework.