

# 1 Introduction

*Now from memory experience is produced in men;  
for the several memories of the same thing  
produce finally the capacity for a single experience.*

– Aristotle

Originally retrospectives, a special kind of a facilitated workshop for gaining feedback (German: Feedbackmoderation, cf. Graeßner, 2008, p. 156-168), have been designed to mainly look back at a finished (software) project and learn from these past experiences.

The originator of retrospectives, Norm Kerth clarified that:

*“[r]etrospective rituals are more than just a review of the past. They also provide a chance to look forward, to plot the next project, and to plan explicitly what will be approached differently next time.” (Kerth, 2001, p. 5)*

Even with this forward looking perspective, retrospectives were still used as a ritual for finalizing a project and preparing for the next one. The usage of retrospectives changed with a new approach in software development: Agile software development, as

defined by the Agile Manifesto is based on a value system and twelve principles which emphasize the importance of collaboration, frequent feedback and continuous learning.<sup>1</sup> The last principle of the Agile Manifesto even requests the following:<sup>2</sup>

*“At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.”*

Thus, with the iterative approach of this kind of development it soon became common to not only reflect on the lessons learned at the end of a project, but both earlier and more regularly during the project. As a consequence, retrospectives are meanwhile well accepted in agile software development. The motivation for reflecting more frequently was according to Esther Derby and Diana Larsen that:

*“[...] teams no longer wait until the end of a long project to inspect and adapt. They look for ways to improve at the end of every iteration.” (Derby & Larsen, 2006, p. xx)*

Retrospectives are nowadays mainly used on a frequent basis in order to leverage the learning throughout a project. Most teams define this frequency as two weeks, some as one week others as one month. Yet still, the core idea of all these more or less frequent retrospectives is to reflect on a past common experience over a fixed period of time, to learn from this and to come up with ideas on what to do differently in the upcoming timeframe. Retrospec-

---

<sup>1</sup> The definition of the value system and the twelve principles can be found at: <http://agilemanifesto.org>.

<sup>2</sup> See <http://agilemanifesto.org/principles.html>.

tives allow a group to address changing needs through continuous learning and adaptation.

However, a group implementing this retrospective approach does not work in isolation. It is embedded in a greater whole – the organization. Yet, the organization that provides this environment faces many different challenges these days. Current trends suggest that approaches that enabled organizational changes in the past will not be successful in the future (cf. Scharmer, 2009, pos. 81). This suggests the need for organizations to look for change methods that address the current challenges, such as globalization or the growing demand in serious participation of everyone involved in the change (cf. Whitmore, 1996, p. 24).