

Evaluierung von Werkzeugen für Distributed Pair Programming: Eine Fallstudie

Dietmar Winkler Stefan Biffli

Technische Universität Wien, Institut für Softwaretechnik und Interaktive Systeme,
Favoritenstrasse 9/188, A-1040 Wien.
{Dietmar.Winkler, Stefan.Biffli}@tuwien.ac.at

Ein zentrales Ziel der industriellen Software Entwicklung ist die Herstellung qualitativ hochwertiger Produkte innerhalb definierter Zeit- und Kostengrenzen (möglichst kurze Entwicklungszeiten bei möglichst kostengünstiger Herstellung) sowie eine schnelle Reaktionsfähigkeit auf geänderte Kundenanforderungen. Dieser Trend führt dazu, (a) Vorteile agiler Techniken in der Softwareentwicklung nutzen und (b) die Softwareentwicklung zu „globalisieren“ und „zu verteilen“. Diese zunehmende Globalisierung in der Softwareentwicklung erfordert eine effiziente und flexible Vorgehensweise, eine strukturierte Planung und eine enge Zusammenarbeit aller Mitglieder eines Entwicklungsteams. Agile Prozesse, wie beispielsweise Scrum oder eXtreme Programming (XP), bieten diese Flexibilität, in dem durch engen Kundenkontakt rasch auf geänderte Anforderungen reagiert werden kann. Agile Praktiken, wie beispielsweise Pair Programming (PP), ermöglichen eine effektive und effiziente Entwicklung von qualitativ hochwertigen Softwareprodukten. Ein Pair, bestehend einem Driver und einem Observer, arbeitet gleichzeitig am selben Produkt und nutzt dieselbe Entwicklungsumgebung. Während der Driver die Entwicklungsarbeit, z.B. eine Implementierungsaufgabe, ausführt, wird er durch den Observer unterstützt, der z.B. Qualitätssicherungsaufgaben, wie die Durchführung kontinuierlicher Reviews, wahrnimmt. Diese Rollenzuteilung kann zyklisch wechseln. Vorteile von PP sind beispielsweise (a) gesteigerte Produktivität, (b) verbesserte Qualität der Produkte und (c) erhöhte Flexibilität durch die Einbindung in einen agilen Prozess. Weiters bietet PP eine effiziente Möglichkeit, neue Teammitglieder in ein Projekt einzubinden (PP als Lernmethodik).

Um diese Vorteile agiler Prozesse und Praktiken in einem globalen und verteilten Umfeld nutzen zu können, sind effektive und effiziente Methoden und Werkzeuge erforderlich, die einerseits die Stärken der „Agilität“ unterstützen und andererseits die räumliche Trennung der Teammitglieder quasi aufheben. In der Praxis existieren zahlreiche Kollaborations-Werkzeuge, die unterschiedliche (Teil-)Aufgaben, wie z.B. Tools für die Unterstützung der Kommunikation oder Document Sharing, unterstützen können. Die Auswahl geeigneter Tools im jeweiligen Unternehmens- oder Projektkontext erfolgt in der Regel durch Ausprobieren oder aus Erfahrungsberichten. Um passende Werkzeuge zu finden, ist eine systematische Evaluierung der in Frage kommenden Werkzeuge erforderlich.

Der Einsatz geeigneter Tools muss die Bedürfnisse der Anwender – in diesem Fall ein

global verteiltes Pair – angepasst sein. Daher ist es notwendig, (a) einen Überblick über die erforderlichen Anforderungen an das Werkzeug im jeweiligen Anwendungskontext zu ermitteln, (b) ein effizientes Evaluierungsframework für die systematische Bewertung der Werkzeuge zu erstellen und dieses (c) auf eine Auswahl möglicher Werkzeuge anzuwenden. Die Anwendung dieses Evaluierungsframeworks ermöglicht (a) eine systematische Untersuchung über die Eignung ausgewählter Werkzeuge und (b) eine Unterstützung bei der Auswahl eines möglichst passenden Werkzeugs im jeweiligen konkreten Anwendungsfall.

Diese Präsentation richtet sich an (a) Projektleiter zur Entscheidungsunterstützung bei der Auswahl geeigneter Werkzeuge in einem konkreten Anwendungsfall, (b) an Werkzeughersteller zur systematischen Sammlung von Anforderungen an eine Werkzeuglösung in einem konkreten Anwendungsbereich und kann somit als Ausgangsbasis für die Verbesserung und Erweiterung existierender Lösungen dienen.

Ziele der Präsentation sind daher:

- Vorstellung einer *generischen Vorgehensweise* zur systematischen Evaluierung von Werkzeugen in einem definierten Kontext.
- Anwendung einer systematischen Werkzeug-Evaluierung am Fallbeispiel von Distributed Pair Programming (DPP):
 - Sammlung und Klassifikation von Anforderungen an ein Werkzeug zur Unterstützung verteilter Entwicklungen.
 - Erstellung eines konkreten Evaluierungsframeworks.
 - Durchführung der Evaluierung von möglichen Werkzeugen für die Unterstützung von Distributed Pair Programming.

Ergebnisse der Arbeit sind eine grundlegende Vorgehensweise, wie bei der Evaluierung von Werkzeugen vorgegangen werden kann sowie eine mögliche Unterstützung bei der verteilten Entwicklung von Softwareprodukten durch Distributed Pair Programming. Durch die Ergebnisse wird mögliches Verbesserungspotenzial von bestehenden Werkzeugen aufgezeigt, die zur Weiterentwicklung der untersuchten Tools verwendet werden können um Anforderungen der globalen und verteilten Entwicklung adressieren zu können.