



□ Ernst Sikora

(E-Mail: Ernst.Sikora@sse.uni-due.de)

studierte Ingenieurinformatik an der TU Dortmund und promoviert an der Universität Duisburg-Essen im Bereich Software und Systems Engineering. Er ist Gruppenleiter in paluno – The Ruhr Institute for Software Technology. Sein Schwerpunkt ist das ziel- und szenariobasierte Requirements Engineering in Verbindung mit dem Architekturforschungswurf.

objektspektrum themenspecial: requirements engineering

Requirements Engineering – Auf dem Weg zu einer reifen Disziplin

Mit 15 Fachartikeln gibt Ihnen dieses Online-Themenspecial einen weit gefächerten Einblick in den Stand der Technik des Requirements Engineering und Management im Jahr 2010. Die Autoren haben für genau jenen Themenmix gesorgt, der eine gute Zeitschrift ausmacht. Nicht nur aktuellste Fachinformationen, sondern auch eine gehörige Portion Lesevergnügen sind garantiert. Und damit Sie sich bei der Erkundung dieser Ausgabe nicht verirren, gibt Ihnen dieses Editorial eine Orientierungshilfe an die Hand.

Winston W. Royce gab im Jahr 1970 in seinem Artikel über das Management von großen Softwareentwicklungsprojekten zu bedenken, dass System- und Softwareanforderungsanalyse nützliche Entwicklungsaktivitäten sind, auch wenn sie keine direkten Beiträge zur Erstellung des Endprodukts leisten. Heute, 40 Jahre später, wird kaum jemand abstreiten, dass Requirements Engineering eine erfolgskritische Disziplin für Unternehmen und deren Projekte ist. Doch welchen Stand hat das Requirements Engineering seit seinen Anfängen als Fachdisziplin heute erreicht – können wir von einer ausgereiften Disziplin „Requirements Engineering“ sprechen?

Klar definierte Konzepte und Begriffe

Ein Kennzeichen einer reifen Disziplin ist ein klares Verständnis der zentralen

Konzepte und Begriffe. Zwei Artikel leisten dazu einen Beitrag.

Karol Frühauf und seine Koautoren räumen mit Irrtümern und Mythen über zentrale Begriffe des Requirements Engineering auf, zum Beispiel mit der Vorstellung, dass eine Anforderungsspezifikation ausschließlich eine Problembeschreibung ist.

Der Beitrag des International Requirements Engineering Board (IREB) beschreibt das Zertifikat „Certified Professional for Requirements Engineering“, mit dem international einheitliche Standards für ein systematisches Requirements Engineering etabliert werden sollen.

Qualität von Anforderungen

Die Qualität von Anforderungen frühzeitig durch konstruktive und analytische Maßnahmen sicher zu stellen ist ein wichtiger Schritt im Reifeprozess des Requirements Engineering.

Jörg Möller setzt auf die Prinzipien des agilen Requirements Engineering, Kommunikation, intensive Einbindung von Stakeholdern und kurze Feedbackzyklen, um die Qualität von Anforderungen sicherzustellen. Der Beitrag zeigt, wie diese Prinzipien selbst unter schwierigen Bedingungen, zum Beispiel mit räumlich verteilten Teams, umgesetzt werden können.

Martin Kochloefl stellt anhand von empirischen Erkenntnissen die Bedeutung einer frühzeitigen Validierung von Anforderungen für den Entwicklungsprozess heraus und legt die Vorteile der Verwendung von Prototypen für die Validierung von Anforderungen dar.

Martin Heidenreich beschreibt, wie ein Softwarewerkzeug mittels objektiv bewertbarer Metriken die Prüfung von Anforderungen unterstützt und damit beispielsweise unerwünschte Formulierungen aufdeckt.

Interdisziplinäre Zusammenarbeit

Beim Hausbau ebenso wie bei der Systementwicklung gilt: um erfolgreich zu sein, müssen alle beteiligten Fachdisziplinen effektiv zusammenarbeiten. In einer reifen Disziplin weiß jede Hand, was sie zu tun hat und mit wem sie sich über was abstimmen muss.

Frank Besemer trennt die fachlichen Kernaktivitäten des Requirements Engineering von Querschnittsaktivitäten, die in ähnlicher Weise nicht nur im Requirements Engineering sondern auch in anderen Engineering-Prozessen stattfinden.

Monika Vetterling legt dar, wie Business Analysten, die der Fachabteilung zugerechnet werden, und Requirements Ingenieure, die der IT-Abteilung angehören, zusammenarbeiten können, sodass Verständnisprobleme, die zu fehlerhaften Anforderungen führen, vermieden werden.

Martin Hilgers beschreibt die Rollen und Aufgabenteilung innerhalb eines Requirements Engineering-Teams, das für die Durchführung des Requirements Engineering-Teilprojekts in einem größeren Entwicklungsprojekt verantwortlich ist.

Vom Textverarbeitungsprogramm zur integrierten Werkzeugkette

Die Verfügbarkeit von Werkzeugen und der Grad der Integration der verfügbaren Werkzeuge in eine durchgängige Werkzeugkette sind weitere Indizien für die Reife einer Disziplin.

Jürgen Gladigau legt die Gründe dar, die für den Einsatz eines professionellen Werkzeugs für das Requirements Management sprechen. Requirements Management ohne ein solches Werkzeug wird mit der Arbeit einer kaufmännischen Abteilung

ohne Unterstützung durch eine Buchhaltungssoftware verglichen.

Joachim Schulz erläutert die wesentlichen Schritte bei der Einführung eines professionellen Requirements Management-Werkzeugs. Der Beitrag zeigt den Weg von der Analyse der Schnittstellen des Requirements Management zu anderen Prozessen über die Werkzeugauswahl, Werkzeuganpassung und Datenmigration bis hin zum Pilotprojekt.

Sven Hubert stellt eine Lösung zur Realisierung einer durchgängigen Werkzeugkette auf Basis des Microsoft Team Foundation Servers vor. Der Schwerpunkt des Beitrags liegt auf der Durchgängigkeit zwischen Anforderungsdefinition, Projektfortschrittsüberwachung und Abnahmetest.

Jürgen Pilz legt dar, wie das HP Quality Center die Disziplinen Requirements Management, Test Management und Fehlermanagement in einem Werkzeug vereint, um das Qualitätsmanagement in Softwareprojekten zu unterstützen.

Von natürlichsprachlichen Anforderungen zu Modellen

Eine reife Disziplin nutzt fachspezifische Modellierungstechniken, um die Konzeption und Analyse und später auch die Konstruktion zu unterstützen. Von jeher problematisch ist dabei der Übergang zwischen textuellen Anforderungen und Modellen des Systems oder der Software.

Andreas Krauß und **Lutz Schwärzler** stellen eine Werkzeuglösung vor, mit der textuelle Anforderungen und Modellelemente miteinander verknüpft werden können, um eine durchgängige Nachvollziehbarkeit zwischen textuellen Anforderungen und Softwaremodellen zu erreichen.

Jörg Holtmann beschreibt einen Ansatz, der im Software Quality Lab entwickelt

wurde, um mithilfe von Satzmustern den Übergang zwischen textuellen Anforderungen und Systemmodellen zu automatisieren und damit Aufwand und Fehleranfälligkeit zu reduzieren.

Andreas Lux stellt einen modellgetriebenen Requirements Engineering-Ansatz vor, bei dem fachliche Anforderungen in einer eigens für die jeweilige Fachdomäne entwickelten Sprache spezifiziert werden und die Software mithilfe eines domänenspezifischen Software-Frameworks aus dem Anforderungsmodell abgeleitet wird.

Kommen wir zurück zu der Ausgangsfrage, ob das Requirements Engineering eine reife Disziplin ist? Eine Antwort könnte lauten: Ja, denn es gibt, wie die Beiträge dieses Themenspecials zeigen, für viele zentrale Herausforderungen praxiserprobte Lösungen. In Anlehnung an das Thema des Keynote-Vortrags von Ian Alexander bei der International Working Conference on Requirements Engineering (REFSQ 2010) ist dem jedoch entgegen zu stellen, dass die heutige Situation immer noch einem Puzzle gleicht, dessen Einzelteile noch zu einem ganzheitlichen Ansatz zusammengefügt werden müssen!

Trotz aller bisherigen und zukünftigen Fortschritte wird sich eines jedoch kaum ändern: Kreativität und Fachwissen über die Anwendungsdomäne sind die wesentlichen Voraussetzungen, um die richtigen Anforderungen zu (er)finden. Erfolgreiche Requirements Engineering-Methoden und -Werkzeuge versuchen nicht, das kreative Denken zu ersetzen, sondern es zu unterstützen.

Viel Vergnügen beim Lesen dieser Ausgabe!

Ernst Sikora