

Aktuelle Herausforderungen in verteilten RE-Prozessen

Andrea Herrmann¹, Timea Illes-Seifert¹, Michael Geisser², Tobias Hildenbrand²

¹ Universität Heidelberg, Institut für Informatik, Neuenheimer Feld 326,
69120 Heidelberg, {herrmann; illes}@informatik.uni-heidelberg.de

² Universität Mannheim, Lehrstuhl für ABWL und Wirtschaftsinformatik, Schloss,
68131 Mannheim, {geisser; hildenbrand}@uni-mannheim.de

Motivation und Einleitung

Zunehmende Globalisierung und steigende Komplexität von Software führen zu internationaler und nationaler Zusammenarbeit zwischen räumlich verteilten Organisationen, Standorten und Menschen. Dies trifft auch auf das Requirements Engineering (RE) zu. Mit einer Online-Umfrage untersuchten wir, wie verbreitet verteiltes RE aktuell ist, welche Probleme dabei auftreten und wie diese gelöst werden. In dieser 744 Fragebögen umfassenden Studie definieren wir verteilte Softwareentwicklung als: „Alle oder mindestens einige der Teilnehmer eines Softwareprojekts benutzen überwiegend verteilte Technologien für die Kommunikation innerhalb des Teams, z.B. weil dies aufgrund räumlicher Distanz nicht anders möglich ist“ (vgl. hierzu auch [1]).

Der Fragebogen besteht aus zwei Teilen: Der erste Teil ermittelt Teilnehmerprofile, Phasen und Werkzeuge. Der zweite Teil fragt nach Problemen, die in verteilten Projekten auftreten, deren Ursachen und Lösungen. Dieser Teil besteht aus vier Frageblöcken. Im ersten wurden die Teilnehmer zu allgemeinen Problemen befragt. „Allgemeine Probleme“ sind solche, die in allen Phasen gleichermaßen auftreten können. Die restlichen drei Blöcke des Fragebogens betreffen jeweils verteiltes RE, Softwareentwurf & Implementierung sowie Test. Der Fragebogen stand vom 2.-26. Februar 2007 drei Wochen lang online.

Ergebnisse: Teilnehmerprofile, verteilt

durchgeführte Phasen und verwendete Werkzeuge
Die Umfrage erbrachte 744 gültige Fragebögen, die von Praktikern mit Erfahrung in durchschnittlich 7,5 verteilten Softwareprojekten ausgefüllt wurden. 34% der verteilten Projekte werden in Unternehmen mit mehr als 10.000 Mitarbeitern durchgeführt, 38% in solchen mit weniger als 100. Der Rest verteilt sich auf Unternehmen mit 1.000-10.000 (13%) und 100-999 Mitarbeitern (15%). 33% der verteilten Projekte haben eine Größe von weniger als 10 Personenmonaten (PM) und weitere 22% eine Größe von 10 bis 20 PM. Die Projektphase, die am häufigsten verteilt durchgeführt wird, ist die Implementierung (92% der Teilnehmer), gefolgt von Testen (73%) und Architekturentwurf (62%). Am seltensten genannt wurden Betrieb und Wartung (46%) sowie RE (38%).

Die Teilnehmer der Studie verwenden E-Mail als wichtigstes Kommunikationsmedium (fast 95% der Befragten). Die wichtigsten synchronen Technologien sind das Telefon (77%) und die Konferenzschaltung (59%). Für den verteilten Informationsaustausch werden am häufigsten Versions- (88%) und Dokumentenverwaltungssysteme (68%) verwendet, CASE- und Projektmanagement-Werkzeuge nur von rund 40% der Befragten. Als weitere Informationsplattform werden Fehlerverwaltungssysteme genannt.

Ergebnisse: RE-Probleme beim verteilten RE

Mittels offener Fragen haben wir die Probleme, die beim verteilten RE auftreten, erhoben. Um die Antworten zu kodieren, verwendeten wir einerseits das durch sie bedrohte Qualitätsziel und andererseits die Problemursache. Die Qualitätsziele waren spezifisch für jede Softwareentwicklungsphase definiert. Für das RE sind das die Effizienz des Spezifikationsprozesses sowie die Qualitätsattribute des IEEE Standards 830-1998 [2]: Fehlerfreiheit, Eindeutigkeit, Vollständigkeit, Konsistenz, Nachprüfbarkeit, Bewertung nach Wichtigkeit und/oder Stabilität, Änderbarkeit, Nachverfolgbarkeit. Die Ursachen klassifizierten wir in fünf Kategorien:

- Kommunikationsbarrieren (z.B.: persönliche Kommunikation kann durch Technologie nicht ersetzt werden; asynchrone Kommunikation ist ineffizient; erschwertes Teambuilding)
- domänenspezifische Barrieren (unterschiedliche Ausbildung, unterschiedliche Erfahrung)
- kulturelle Barrieren (Sprachprobleme, verschiedene Motivation und Qualitätsverständnis)
- technische Barrieren (Werkzeuge sind ungeeignet oder weisen Qualitätsmängel auf)
- Prozessbarrieren (Verantwortlichkeiten und Prozesse sind schlecht definiert oder werden nicht gelebt, unangepasste oder unflexible Prozesse, erhöhter Kommunikationsbedarf)

Von den 122 RE-Problemen, die von den Teilnehmern angeführt werden, konnten sieben aufgrund vager Formulierung nicht kodiert werden. 47 beziehen sich auf die Zweideutigkeit der Anforderungsspezifikation, 44 auf die Prozesseffizienz und 12 auf die Vollständigkeit der Spezifikation. Nur

fünf beziehen sich auf die Änderbarkeit, drei auf Fehlerfreiheit, zwei auf Kontinuität, eine auf Nachprüfbarkeit, eine auf Priorisierung und keine auf Nachverfolgbarkeit. Tabelle 1 vergleicht die Anteile der Problemursachen, die zu einer zweideutigen Spezifikation führen, mit denen, die den RE-Prozess ineffizient machen, mit den Mittelwerten für das verteilte RE allgemein und die verteilte Softwareentwicklung insgesamt.

Problemursache	(a)	(b)	(c)	(d)
Kommunikationsbarriere	9%	27%	11%	41%
Domänen-spezifische B.	22%	33%	66%	5%
Kulturelle B.	23%	16%	21%	18%
Technische B.	16%	3%	0%	5%
Prozessbarriere	31%	21%	2%	32%

Tabelle 1: Problemursachen: (a) in verteilter Softwareentwicklung allgemein; (b) im verteilten RE; (c) im verteilten RE als Ursache für eine zweideutige Anforderungsspezifikation oder (d) als Ursache für einen ineffizienten RE-Prozess

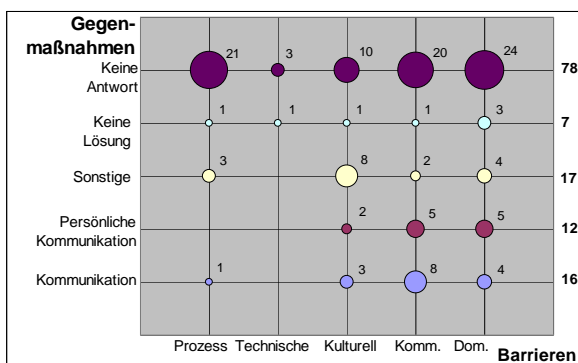


Abbildung 1. Gegenmaßnahmen pro Barriere beim verteilten RE – Anzahl der Nennungen (Komm. = Kommunikation, Dom.= Domänenspezifisch) [3]

Ergebnisse: Problemlösungen beim verteilten RE

Die Umfrageteilnehmer wurden auch gefragt, welche Gegenmaßnahmen sie gegen die genannten Probleme erfolgreich angewandt haben. Für nur 37 von 122 Problemen werden (insgesamt 45) Gegenmaßnahmen angegeben. Sieben Mal heißt es explizit, dass es (noch) keine erfolgreiche Gegenmaßnahme gibt. Bei den genannten Gegenmaßnahmen zeichneten sich drei Hauptgruppen ab: Kommunikation allgemein (16 Nennungen), persönliche Kommunikation (12) und sonstige (17). 28 von 45 Gegenmaßnahmen (d.h. 62%) sind also Kommunikationsmaßnahmen. Es wird empfohlen, die Kommunikation zu intensivieren, zu kommunizieren, sobald eine Frage auftaucht, formellen Regeln zu folgen, Werkzeugunterstützung zu nutzen (hier ein Wiki) und eine gemeinsame Technologie zu verwenden. Es ist bemerkenswert, dass im Vergleich

dazu bei der verteilten Entwicklung allgemein die persönliche Kommunikation nur vier Mal unter 55 Kommunikationsmaßnahmen genannt wird, d. h. 7% statt der 43% beim RE. Die weiteren (17) Gegenmaßnahmen sind Qualitätssicherung, Schulung, „Mehr arbeiten“ sowie RE-spezifische Maßnahmen (Glossar, Beispielanforderungen, frühe Testspezifikation, standardisierte Formate und Definition von Minimalanforderungen an Dokumente).

Zusammenfassung und Ausblick

Unsere Umfrage über verteilte Softwareentwicklung lieferte mit 744 gültigen Fragebögen eine große Menge Daten, die mit den Ergebnissen vereinzelter früherer Untersuchungen übereinstimmen, diesen jedoch eine quantitative Gewichtung zuweisen. Nicht überraschend, jedoch überzeugend war die enorme Bedeutung der Kommunikation als Problemlösungsstrategie für alle Arten von Problemen, ganz besonders im verteilten RE.

Weitere Ergebnisse dieser Umfrage und deren Vergleich mit ähnlichen Studien können in [3,4] nachgelesen werden. Die gewonnenen Daten werden zukünftig weitergehend ausgewertet, sowohl bezüglich den übrigen Phasen der Softwareentwicklung als auch auf der Suche nach Korrelationen, z.B. ob manche Probleme in großen Projekten häufiger auftreten als in kleineren.

Danksagungen

Bei den zahlreichen Teilnehmern der Studie wollen wir uns für ihre Zeit, ihr Vertrauen und ihre aussagekräftigen Antworten bedanken. Wir sind dankbar für die finanzielle und organisatorische Unterstützung der Umfrage durch die MFG. Wir bedanken uns auch bei Heise für die Veröffentlichung der Umfrageadresse in ihrem News-Ticker. Diese Arbeit wurde durch den Forschungsverbund PRIMIMUM gefördert (www.primium.org/).

Referenzen

- [1] T. Hildenbrand, F. Rothlauf, A. Heinzl: Ansätze zur kollaborativen Softwareerstellung. *Wirtschaftsinformatik* 49/ 2007, S. S72-S80
- [2] IEEE: Std. 830-1998: Recommended Practice for Software Requirements Specification. IEEE, 1998
- [3] M. Geisser, A. Herrmann, T. Hildenbrand, T. Illes-Seiffert: Verteilte Softwareentwicklung und Requirements Engineering: Ergebnisse einer Online-Umfrage. *OBJEKTSpektrum* Nr. 6, November/Dezember 2007, S. 40-51
- [4] T. Illes-Seiffert, A. Herrmann, M. Geisser, T. Hildenbrand: The Challenges of Distributed Software Engineering and Requirements Engineering: Results of an Online Survey. *GREW07: 1st International Global Requirements Engineering Workshop 2007*, München