

TRENDTHEMA „QUALITÄTS- MANAGEMENT UND MODELLE“: WAS DIE EXPERTEN SAGEN

Qualität ist als Thema nicht besonders sexy und in der IT nahezu zeitlos. Trotzdem scheint es gerade ein Qualitäts-Revival zu geben, oder?

Gernot Starke: Ja, Sie sprechen mir aus der Seele. In unserem Vortrag auf der OOP 2011 zum Thema „Quality Driven Software Architecture“ haben Peter Hruschka und ich wieder einmal eine Lanze dafür gebrochen, den Fokus auf Qualitätsmerkmale zu richten: Wir müssen als Architekten die notwendigen und geforderten Qualitätsmerkmale bewusst und systematisch konstruieren. Alleine mit domänengetriebenen Ansätzen funktioniert das leider nicht – es gibt genügend Beispiele für korrekt funktionierende aber qualitativ mangelhafte Systeme. Das wird mit „Big Data“, Cloud und coolen UIs auch weiterhin sehr wichtig bleiben.

Johannes Mainusch: Qualität ist total in. Außerdem sind QA-Experten rar am Markt und teilweise gefragter als Entwickler. Eine zu meisternde Herausforderung ist die optimale Integration von Anforderungsmanagement, Entwicklung, QA und Betrieb in agilen Organisationen.

Wilhelm Schäfer: Qualität erlebt kein Revival, es ist ein Evergreen. Mich erschüttert es, wie wenig Qualitätsbewusstsein es in manchen IT-Disziplinen immer noch gibt. Es kann doch nicht sein, dass ich meinen PC so häufig booten muss, nur um weiterarbeiten zu können. Glücklicherweise gibt es andere Anwendungsgebiete mit höheren Qualitätsmaßstäben. In der Avionik, im Kraftwerksbau oder auch in der Automobilindustrie. Die Erfolge im Qualitätsmanagement in diesen Disziplinen werden für alle anderen Bereiche nutzbar sein. Den Rest wird der Markt erledigen, denn an fehlerhaften Systemen hat auf Dauer kein Anwender Freude.

Arne Pott: Qualität steht für uns von jeher im Mittelpunkt. Trotzdem gibt es hier einen weiteren Schub durch die Internet- und Portal-Technologie. Es ist halt ein Unterschied, ob Funktionen für einen großen, aber überschaubaren Kreis von Angestellten zur Verfügung gestellt werden oder ob sie im großem Stil Geschäftspartner und Kunden einbinden. Der betriebswirtschaftliche Schaden durch einen Ausfall kann hier erheblich größer sein. Hinzu kommt der Imageschaden. Schauen Sie sich doch nur einmal das Medienecho an, wenn ein Service auf einer der großen Internet-Plattformen nicht verfügbar ist, wie vor einigen Wochen. Durch eine Verbesserung des Qualitätsmanagements haben wir in den letzten Jahren auch messbare Fortschritte in der Qualität unserer Software gemacht. So haben wir die bereits geringe Anzahl notwendiger Änderungen an produktiven Systemen auf ein Minimum reduzieren können.

Das Thema Modellierung wirkt ja gegenüber der Agilität fast verstaubt. Welche Rolle spielt Ihrer Meinung nach in Zukunft die Modellierung und welche Modelle werden eine wesentliche

Rolle spielen für eine erfolgreiche Anwendungsentwicklung?

Pott: Einspruch: Modelle sind alles andere als verstaubt. Nehmen Sie doch einmal die weitere Vernetzung von Unternehmen mit Kunden, Geschäftspartnern, Behörden und anderen Organisationen. Ich denke, dass die inhärente Komplexität nur mit stärkerer übergreifender Standardisierung von Modellen in den einzelnen Fachdomänen und Branchen beherrschbar bleibt. Oder nehmen Sie die Architekturbeschreibungen eines Unternehmens. Wir haben noch keine befriedigenden Modelle, die das Zusammenspiel von Services und Komponenten in ähnlicher Transparenz darstellen wie die hergebrachten Bilder von Anwendungslandschaften. Woran ich hingegen aus wirtschaftlichen Gründen weniger glaube, ist, dass sich die Visionen im Bereich Model Driven Architecture im Mainstream der betriebswirtschaftlichen Softwareentwicklung durchsetzen werden. Generierende Ansätze in der Softwareentwicklung haben wir schon in der CASE-Phase in den 90er Jahren kommen und gehen sehen – da sehe ich gerade manche Parallelen.

Schäfer: Modellierung halte für einen der nachhaltigsten Trends überhaupt. Über die letzten Jahrzehnte hat es ja die verschiedensten Ausprägungen davon gegeben: 4GL und MDA sind da nur einige prominente Beispiele. Ich beobachte jetzt, dass die Modellierung in ganz neue Bereiche vordringt. Im Bereich der Embedded Systems spielen nun beispielsweise Modelle für zeitabhängiges Verhalten und Scheduling eine immer größere Rolle. Ich halte diese Entwicklung für richtig und wichtig. Und hier können wir von den klassischen Ingenieurdisziplinen noch eine ganze Menge lernen.

Starke: Wir brauchen Abstraktionen vom reinen Source-Code. Ohne Modelle sind wir in mittleren bis großen Systemen verloren – ich persönlich glaube an Modellierung. Ob in UML oder domänenspezifischen Sprachen, ob in Visio oder mit (coolen) Modellierungswerkzeugen spielt dabei eine sekundäre Rolle. Ordentliche Dokumentation von Architekturen braucht Modelle – welcher Art auch immer. Ich predige seit Jahren pragmatische Dokumentation von Architekturen – und die basiert maßgeblich auf verschiedenen Modellen bzw. Sichten.

Mainusch: Ich bin kein Experte im Bereich der Modellierung, daher sollte meine Meinung dazu „cum grano salis“ gelesen werden. Ich habe in der Vergangenheit zum Beispiel UML oder MDA oft eher als Kommunikationshindernis zwischen Experten wahrgenommen, denn als Hilfe, Wissen weiterzugeben. Letztlich geht es darum, Informationen über Systeme zwischen Menschen auszutauschen oder – im Falle von Dokumentation – für die Personen zu sichern, die mit dem System zukünftig arbeiten werden. Da helfen auch gute cross-funktionale Zusammenarbeit, Clean Code, eine gute Testabdeckung, regelmäßige Refactorings, sinnvoll verwendete Frameworks, gute Anforderungsdokumentation, Betriebsmonitoring usw. ▶

QUALITÄTSSICHERUNG IM WANDEL: WAS SICH IN DEN LETZTEN 15 JAHREN GETAN HAT

Im Mai 2011 wurde im deutschsprachigen Raum eine anonyme Online-Umfrage zum Thema „Softwaretest in der Praxis“ durchgeführt. Ein Teil der im Mai gestellten Fragen wurden aus einer Umfrage zu „Prüf- und Testprozesse in der Softwareentwicklung“ aus dem Jahr 1997 übernommen. Mit Hilfe der beiden Umfragen lässt sich zeigen, was sich im Bereich der Qualitätssicherung bei der Softwareentwicklung verändert hat und ob aktuelle Trends sich bereits einen Platz in der Praxis erobert haben.

Die anonym durchgeführte Online-Umfrage hatte den Stand der Softwaretests in der Praxis zum Gegenstand und war rollenspezifisch aufgebaut. Die meisten Antworten bekamen wir von der Gruppe Projektleiter, QS-Beauftragte, Testmanager und Tester (1.008 Teilnehmende). Business-Analysten, Entwickler, Mitarbeiter aus Betrieb & Support und andere Mitarbeiter (394 Teilnehmende) haben sich ebenso wie das *executive* und mittlere Management (221 Teilnehmende) an der Umfrage beteiligt (Näheres zum Umfang der Beteiligung der einzelnen Gruppen, zur Verteilung auf Branchen usw. finden sich in [Umf]).

Im Fragenkatalog von 2011 haben wir einige Fragen aus einer Untersuchung aus dem Jahr 1997 von Uwe Müller u. a. (vgl. [Mül98], [Mül99]) übernommen, um eine direkte Vergleichbarkeit herstellen zu können. Damals wurden deutsche Softwareunternehmen papiergestützt befragt und es wurden 73 Fragebogen ausgewertet.

Auch wenn es zahlenmäßig zwischen beiden Umfragen in Bezug auf die Beteiligung einen erheblichen Unterschied gibt, glauben wir, dass ein Vergleich gerechtfertigt ist, da die „Basisangaben“ bei beiden Umfragen vergleichbar sind. Dies belegen die folgenden beiden Beispiele:

- Auf die Frage, für welche Branche die Software erstellt wird, gab es folgende Antworten: 1997: Versicherungen, Industrie, Handel, Banken, Telekommunikation; 2011: Automotiv, Telekommunikation, Banken, Versicherungen (dabei ist die häufigste Nennung jeweils zuerst aufgeführt).
- Die Antworten auf die Frage zur Anzahl der Mitarbeiter im Unternehmen fiel bei beiden Umfragen identisch aus: Schwerpunktmäßig 101-1000 bzw. 1001-10.000 Mitarbeiter.

Im Folgenden werden wir einige Aspekte der aktuellen Ergebnisse der Umfrage 2011 in Bezug zu der Untersuchung von 1997 setzen. Zeichnen sich Veränderungen ab? Gibt es neue Trends? Kurze Interpretation unsererseits schließen sich an jeden Fragenkomplex an. Dabei bleibt es dem Leser überlassen, aus dem angebotenen Zahlenmaterial seine Vergleiche zum eigenen Umfeld zu ziehen.

Verteilung der Qualitätssicherung auf die Entwicklungsphasen

Vorgehensmodelle in der Softwareentwicklung beschreiben grundlegende Strategien – auch für Qualitätssicherungsmaßnahmen. Wir stellten hierzu folgende Frage „Wie ordnen Sie Ihr Vorgehensmodell in der Softwareentwicklung ein, eher als ...“. Etwas über die Hälfte der Teilnehmenden gaben ein „phasenorientiertes Vorgehensmodell“ und etwa ein Drittel ein „agiles Vorgehensmodell“ an. Die restlichen Befragten entwickeln Software nach keinem expliziten Modell.

Bei der Frage nach den konkreten Modellen, mussten die Umfrageteilnehmer sich entscheiden, welches Vorgehensmodell in der Regel eingesetzt wird. Bei phasenorientierten Modellen werden das allgemeine V-Modell oder ein eigenes angepasstes Phasenmodell favorisiert. Aber auch das W-Modell, das klassische Wasserfallmodell und das V-Modell XT kommen zum Einsatz (siehe Abbildung 1).

Bei agilen Modellen liegt mit 56,9 % eindeutig der Schwerpunkt auf Scrum, gegenüber 27 % für eigene angepasste Modelle. Alle anderen Modelle, wie beispielsweise *eXtreme Programming (XP)*, *Feature Driven Development (FDD)* oder *Kanban*, liegen bei maximal 5 % und spielen damit keine wesentliche Rolle. Gerade bei dieser Frage



Andreas Spillner

[E-Mail: andreas.spillner@hs-bremen.de]

ist Hochschullehrer für Software-Engineering mit Fokus auf Qualitätssicherung an der Hochschule Bremen. Er war Gründungsmitglied und ist nun Ehrenmitglied des German Testing Boards e. V. (GTB).



Karin Vosseberg

[E-Mail: karin.vosseberg@hs-bremerhaven.de]

ist Hochschullehrerin für Informatik und Wirtschaftsinformatik an der Hochschule Bremerhaven mit den Schwerpunkten Systemintegration und Qualitätssicherung.



Mario Winter

[E-Mail: winter@gm.fh-koeln.de]

ist Hochschullehrer an der Fachhochschule Köln für Software-Engineering und Qualitätssicherung. Er ist Mitglied im GTB und war Sprecher der Fachgruppe TAV der GI e.V.



Peter Haberl

[E-Mail: peter.haberl@anecon.com]

verantwortet bei der ANECON Software Design & Beratung G.m.B.H. als Geschäftsführer die Dienstleistungen und Kundenbeziehungen in Deutschland.

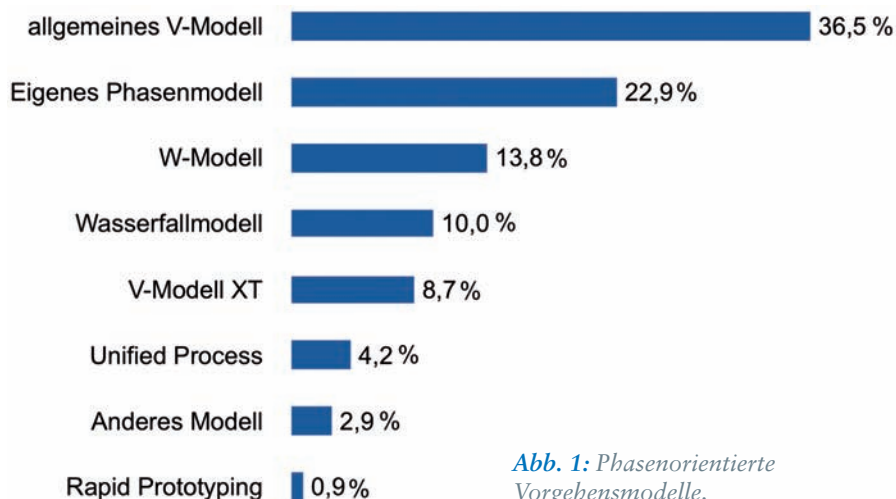


Abb. 1: Phasenorientierte Vorgehensmodelle.

wurde jedoch von einigen Teilnehmern eine Mehrfachnennung eingefordert, da anscheinend auch viele Mischformen der agilen Vorgehensweisen in der Praxis anzutreffen sind. 1997 waren agile Entwicklungsmodelle noch nicht verbreitet, sodass nicht explizit nach Vorgehensmodellen gefragt wurde. Ein direkter Vergleich ist somit nicht möglich. Interessant ist jedoch, ob das verwendete Vorgehensmodell 2011 einen signifikanten Einfluss auf Maßnahmen der Qualitätssicherung hat. Diese Frage werden wir im Folgenden weiter aufgreifen.

In beiden Umfragen wurde folgende Frage gestellt: „In welchen Phasen der Softwareentwicklung werden Maßnahmen

zur Qualitätssicherung (z. B. Reviews oder Testen) in Ihren Projekten durchgeführt?“

Unterschieden wurden die folgenden Phasen:

- Vorstudie (Projektdefinition, Konzeption)
- Fachkonzept (Anforderungsermittlung, Requirements, Analyse)
- Systementwurf (Grob-/Feinentwurf, technische Konzeption)
- Realisierung (Implementierung)
- Integration und Abnahme (nach Abnahme wurde nur 2011 gefragt)

Da agile Modelle keine expliziten Phasen unterscheiden, wurde diese Frage nur

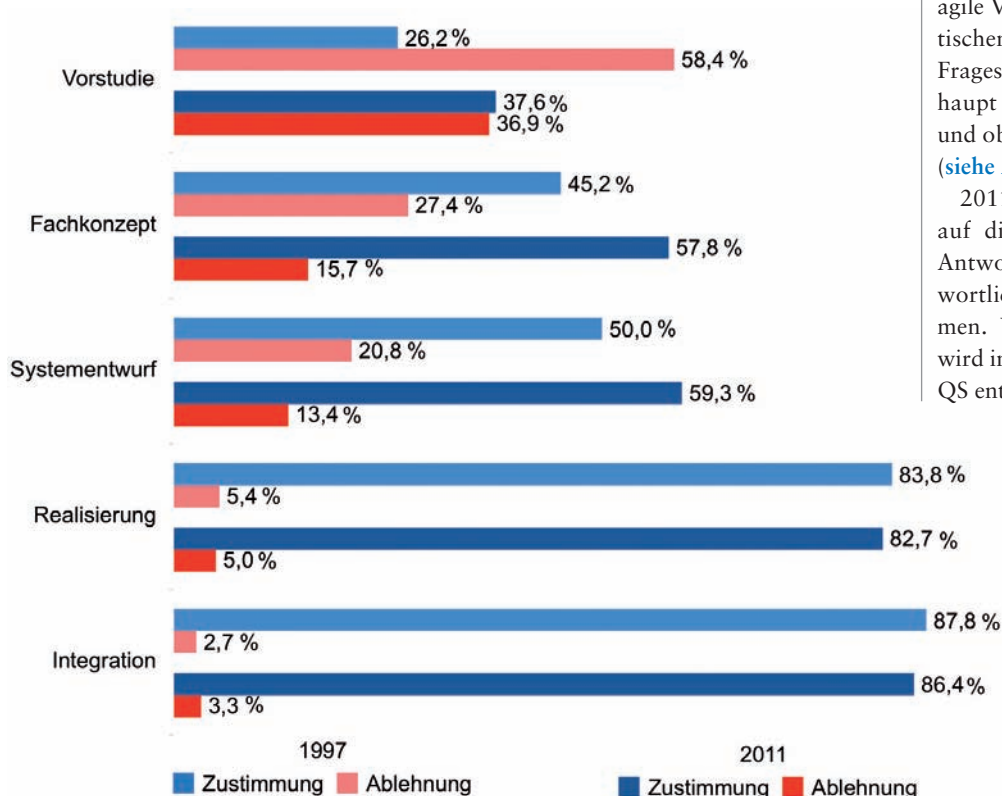


Abb. 2: Qualitätssicherung in den Phasen der Softwareentwicklung (Mehrfachnennungen möglich).

Teilnehmern mit phasenorientierten Vorgehensmodellen gestellt. Die Teilnehmer konnten angeben, ob „immer“, „meist“, „teils/teils“, „selten“ oder „nie“ bei den einzelnen Phasen Qualitätssicherungsmaßnahmen durchgeführt werden. Zur besseren Übersicht werden die Antworten „immer“ und „meist“ als Zustimmung sowie „selten“ und „nie“ als Ablehnung zusammengefasst. „Teils/teils“ wird bei den Abbildungen nicht berücksichtigt und als neutral angesehen, um die Unterschiede besser hervorheben zu können. **Abbildung 2** verdeutlicht die Zustimmung und Ablehnung zu den einzelnen Phasen und lässt die Unterschiede zwischen 1997 und 2011 erkennen.

Unsere Interpretation

Eine leichte Verschiebung hin zu den frühen Phasen der Softwareentwicklung wird deutlich, allerdings ist auch 2011 noch keine wünschenswerte Gleichverteilung über alle Phasen erreicht.

Durchführung der Qualitätssicherung

Häufig entfacht sich eine Diskussion an der Frage, wie genau die *Qualitätssicherung (QS)* geregelt und vorgeschrieben werden muss, damit sie „richtig“ durchgeführt wird. Agile Ansätze setzen da ganz bewusst auf mehr Eigenverantwortung. 1997 waren agile Vorgehensweisen noch nicht im praktischen Einsatz. Der Fokus in der Fragestellung lag daher darauf, ob es überhaupt Richtlinien und Vorschriften gibt und ob diese zwingend eingehalten werden (*siehe Abbildung 3*).

2011 haben wir in der Umfrage bewusst auf die explizite Entscheidung für eine Antwort gesetzt und die Eigenverantwortlichkeit der Projekte mit aufgenommen. Wir formulierten die Frage: „Wie wird in der Regel über die Maßnahmen zur QS entschieden?“ (*siehe Abbildung 4*).



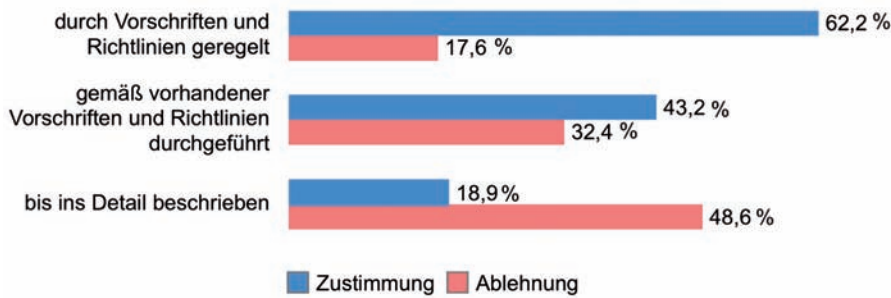


Abb. 3: Regelung der Qualitätssicherung von 1997 (Mehrfachnennungen möglich).

Bei einer Differenzierung der Ergebnisse von 2011 nach den Vorgehensmodellen ist bei agilen Projekten eine deutliche Verschiebung in Richtung Eigenverantwortung der Projektbeteiligten erkennbar. Weniger als 9 % der Teilnehmer aus agilen Projekten führen eine QS zwingend nach Vorschriften bzw. Richtlinien durch, während dieses Vorgehen bei knapp 20 % der Teilnehmer aus phasenorientierten Projekten der Fall ist. Der Anteil der bis ins Detail vorgeschriebenen Handlungsanweisungen ist bei beiden Welten der Vorgehensmodelle ähnlich niedrig. Werden keine expliziten Vorgehensmodelle für die Softwareentwicklung eingesetzt, geben sogar 23 % der Befragten an, gar keine Regelungen für die QS zu haben.

Unsere Interpretation

Die Umfrage 2011 zeigt, dass nach wie vor in vielen Bereichen die QS durch Richtlinien und Vorschriften geregelt ist; eher noch weniger als 1997 fordern eine zwingende Durchsetzung der Richtlinien ein. Jedoch setzt ein Großteil der Projekte auf Eigenverantwortlichkeit. Dieser Trend ist in agilen Projekten verstärkt zu beobachten. Offensichtlich hat das Vorgehensmodell Einfluss auf das Verständnis und den Einsatz von Regelungen der QS.

Artefakte, die einer QS unterzogen werden

Welche Bedeutung der QS beigemessen wird, lässt sich auch daran ablesen, welche Dokumente einer QS unterzogen werden. Die Auswertung beschränkt sich auf die frühen Dokumente, da hier eine Verschiebung sehr deutlich wird (siehe Abbildung 5).

Während 1997 danach gefragt wurde, welche Dinge im Unternehmen Prüf- und Testobjekte sind, lautete 2011 die entsprechende Frage: „Welche Dokumente werden in Ihren Projekten in der Regel einem Review unterzogen?“

Beim differenzierten Blick entlang der Vorgehensmodelle wird deutlich, dass phasenorientierte Vorgehensmodelle den Schwerpunkt auf die Anforderungen mit 67 % Zustimmung und 21 % Ablehnung legen und dass die Architekturspezifikation etwas an Bedeutung abnimmt. Die Designspezifikation wird geringfügig gestärkt, kommt aber nicht an die Werte von 1997. Bei agilen Projekten ist die Zustimmung zu Reviews insgesamt niedriger. Aber wenn Reviews durchgeführt werden, ist auch hier eine Verschiebung in Richtung Anforderungen klar zu erkennen. Designspezifikationen nehmen an Bedeutung jedoch stark ab. Wird nicht nach einem Vorgehensmodelle entwickelt, dann werden kaum Reviews durchgeführt. Hier liegt die ablehnende Haltung gegenüber allen drei Dokumenten bei über 60 %.

Unsere Interpretation

Eine leichte Verschiebung zu Gunsten einer QS in den frühen Phasen der Softwareentwicklung war bereits in einer vorherigen Frage zu erkennen. Dies wird mit der Betrachtung der Artefakte aus den frühen Phasen bestätigt. Es scheint jedoch, dass Designspezifikationen an Bedeutung verlieren. Reviews haben bei den phasenorientierten Vorgehensmodellen in der Softwareent-

wicklung einen weit höheren Stellenwert als bei den agilen Vorgehensweisen.

Zuständigkeit für die Durchführung der QS

1997 wurde die Frage nach der Zuständigkeit für die Durchführung von Prüfungen und Tests gestellt (siehe Abbildung 6). 2011 haben wir die Zuständigkeiten viel differenzierter für die einzelnen Aktivitäten betrachtet. Dabei waren wie 1997 Mehrfachnennungen möglich. Wir haben jedoch keine Einordnung in einer Werteskala von „immer“ bis „nie“ gefordert. Für folgende drei Aufgabenbereiche wurden die durchführenden Personengruppen erfragt:

- „Folgende Personengruppen führen die Reviews in den Projekten durch:“
- „Folgende Personengruppen führen die Tests der Teststufen in den Projekten durch:“
- „Folgende Personengruppen führen die Tests der Testarten (z. B. Zuverlässigkeitstest, Regressionstest, Benutzbarkeitstest) in den Projekten durch:“

Auch den Personenkreis haben wir etwas anders kategorisiert (siehe Abbildung 7). So wollten wir wissen, ob beispielsweise Reviews von speziell für diese Aufgabe ausgebildetem Personal durchgeführt werden. Wie 1997 haben wir eine Unterscheidung zwischen Mitarbeitern aus den Fachabteilungen und den zukünftigen Anwendern getroffen. Außerdem interessierte uns, ob die Durchführung der QS als eine Dienstleistung gesehen wird, die an externe Dienstleister ausgliedert wird. Auch wenn die Fragen nicht direkt vergleichbar sind, lassen sich doch Tendenzen beobachten.

Wie vermutet, zeigt sich bei genauerer Betrachtung, dass wir in agilen Projekten eine leichte Verschiebung der Zuständigkeiten zu den Entwicklern beobachten können.

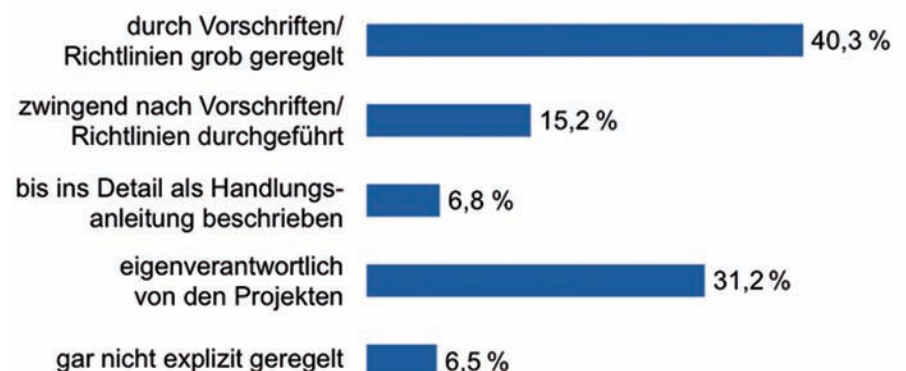


Abb. 4: Regelung der Qualitätssicherung von 2011 (Einfachnennung).

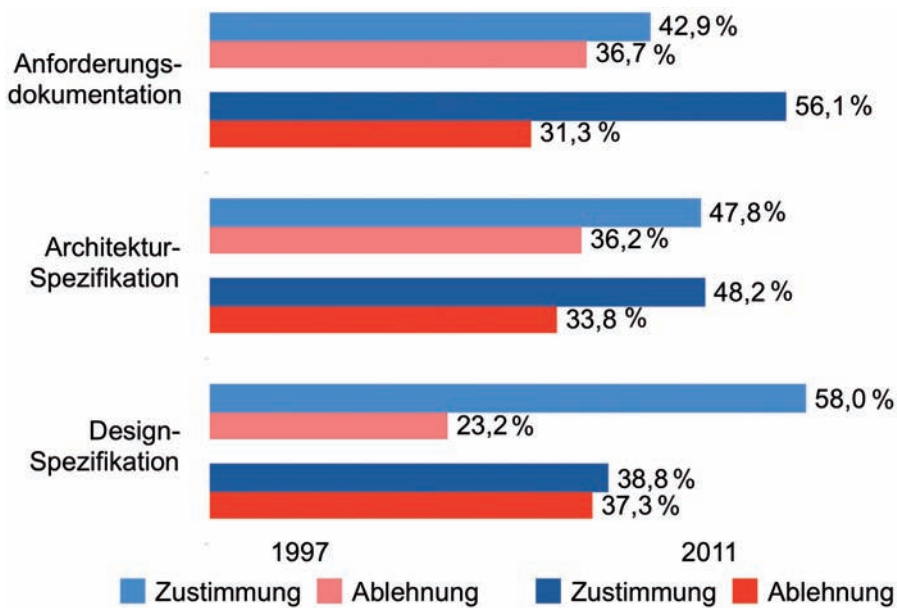


Abb. 5: Dokumente, die geprüft werden (Mehrfachnennungen möglich).

Mitarbeiter aus den Fachabteilungen sind aber deutlich weniger involviert als in klassischen phasenorientierten Projekten. In beiden Vorgehensmodellen der Softwareentwicklung wird beim Testen der verschiedenen Testarten verstärkt auf ausgebildete Tester gesetzt. Während für diese Tests ein Viertel der Befragten in phasenorientierten Projekten angibt, externe Dienstleister einzusetzen, ist es in agilen Projekten ca. ein Fünftel.

Unsere Interpretation

Die Hauptlast der verschiedenen QS-Maßnahmen tragen immer noch die Entwickler, besonders in agilen Projekten. Die Bedeutung ausgebildeter Tester hat aber bei allen Vorgehensmodellen deutlich zugenommen. Externe Dienstleister werden für die Durchführung der QS-Maßnahmen wenig eingesetzt. Wenn es jedoch um Spezialwissen geht, wird stärker auf ihre Expertise zurückgegriffen, insbesondere in Projekten mit phasenorientierten Vorgehensmodellen.

Aktuelle Trends

Mit der Umfrage 2011 wollten wir neben dem Vergleich zu 1997 auch untersuchen, ob bereits aktuelle Trends – beispielsweise testgetriebene Softwareentwicklung oder modellbasiertes Testen – in der Praxis ihren Platz gefunden haben.

Testgetriebene Softwareentwicklung

Auch wenn *testgetriebene Softwareentwicklung (TDD)* nicht zwingend mit agilen Vorgehensweisen verbunden ist, wird TDD in der Diskussion häufig als ein wesentliches Merkmal agiler Praktiken genannt. Auf die

Frage, „Welche Praktiken agiler Vorgehensmodelle haben eine hohe Bedeutung für Sie in Hinblick auf QS?“, nennen 49,2 % aus einem agilen Projektumfeld das Konzept der TDD. Dieser Aspekt steht hinter einer gemeinsamen Aufwandsschätzung (60,2 %), den Story-Cards (50,4 %) und regelmäßigen Stand-Up-Meetings (50,4 %) an vierter Stelle. Ein anderes Indiz dafür, dass TDD noch nicht in dem Maße in der Praxis angekommen ist wie vielleicht gewünscht, gibt uns die Untersuchung zum Grad der Testautomatisierung.

TDD wird in der Diskussion sehr eng mit dem Thema Testautomatisierung verbunden, insbesondere auf der Ebene der Unit-Tests. Betrachten wir die Auswertung näher, wird deutlich, dass ungefähr die Hälfte der Teilnehmenden in phasenorien-

tierten Projekten angeben, bei einem Automatisierungsgrad größer gleich 70 % auf der Teststufe der Unit-Tests zu sein. In einem agilen Projektumfeld sagen etwas mehr als zwei Drittel, dass sie 70 % und mehr ihrer Unit-Tests automatisiert haben. Die Zahlen belegen, dass TDD – in dem Sinne, dass vor der eigentlichen Umsetzung ein Testfall implementiert und damit automatisiert wird – selbst in agilen Projekten noch nicht vollständig umgesetzt ist.

Modellbasiertes Testen

Modellbasiertes Testen ist in der Forschung eines der aktuellsten Themen im QS-Bereich. Doch wie findet dieses Thema Einzug in die Praxis? 2011 haben wir die Teilnehmer gefragt, ob sie ihre Testfälle mit graphischen Modellierungssprachen beschreiben oder aus Modellen generieren.

Die Antworten waren sehr klar: Nur ca. 5 % der Befragten stimmten zu, ihre Testfälle mit graphischen Modellierungssprachen zu beschreiben oder aus Modellen zu generieren. 75 % lehnten ein solches Vorgehen sogar ab. Hieraus lässt sich schließen, dass Wissen über modellbasiertes Testen noch nicht in der Praxis etabliert ist.

Diejenigen, die das Ausbildungsschema des „ISTQB Certified Tester“ kennen (vgl. [IST]) kennen, haben wir gefragt, ob sie modellbasiertes Testen als Thema für den *Expert Level* sehen. Lediglich 50 % der Befragten sehen modellbasiertes Testen als Thema für den *Expert Level*. Darüber hinaus gab es in der Umfrage 2011 zu jedem Fragenkomplex die Möglichkeit, Kommentare abzugeben. Betrachten wir einzelne Kommentare der Teilnehmer zum modellbasierten Testen, verstärkt sich das ambivalente Interesse. Exemplarisch macht dies der folgende Kommentar deutlich: „Modell-

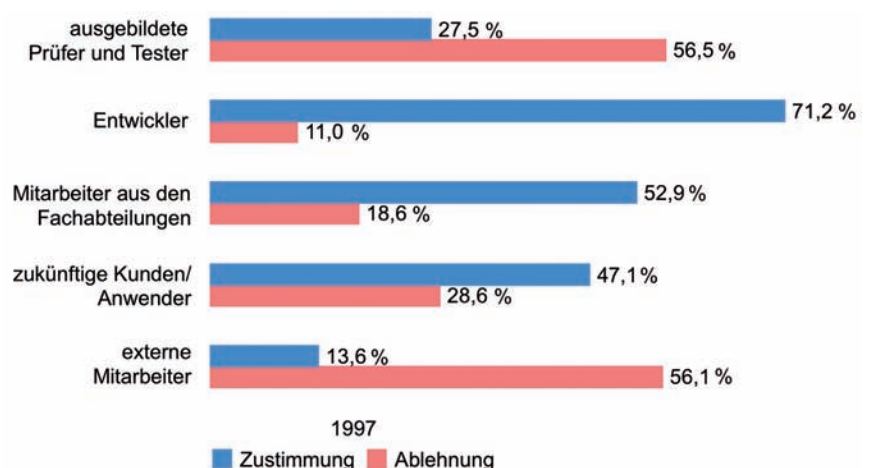


Abb. 6: Personen, die 1997 QS-Maßnahmen durchführten (Mehrfachnennungen möglich).

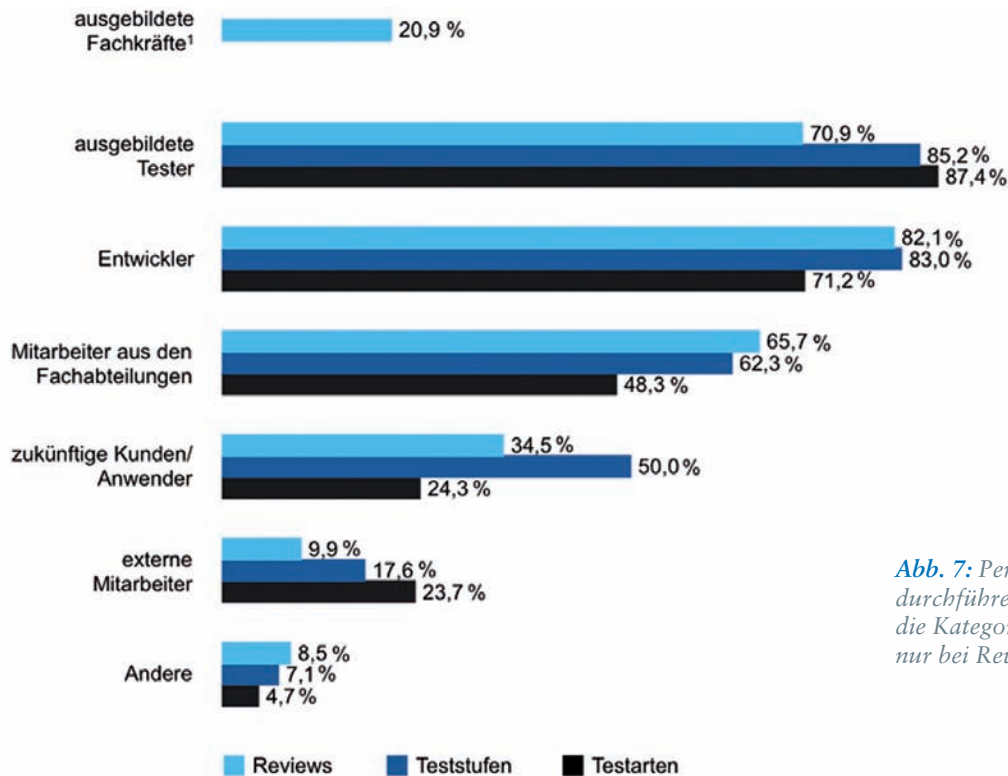


Abb. 7: Personen, die 2011 QS-Maßnahmen durchführen (Mehrfachnennungen möglich; die Kategorie „ausgebildete Fachkräfte“ wurde nur bei Reviews erfragt).

basiertes Testen: 1) Alter Wein in neuen Schläuchen. 2) Lässt viele bewährte Bereiche außer Acht. 3) Modeerscheinung => Auf keinen Fall zum Thema machen.“

Resümee und Ausblick

Für diesen Artikel haben wir einen kleinen Ausschnitt aus dem sehr umfangreichen Fragenkatalog ausgewählt, um den Aspekt der Veränderungen in den letzten 15 Jahren etwas näher zu beleuchten. Weitere Auswertungen werden folgen. Insgesamt stellen wir fest, dass sich in der QS eine Entwicklung zum Positiven in den letzten Jahren bestätigt. Aber es ist weiterhin noch ausreichend genug Potenzial für Verbesserungen vorhanden, sodass ein „sich Zurücklehnen“ noch nicht angesagt ist.

Wo sehen die Teilnehmer einen besonderen Bedarf an Forschung und Weiterentwicklung? Zu jedem Fragenkomplex hatten diese die Möglichkeit, sich in Form von Freitexten zu äußern. Diese Möglichkeit wurde erfreulicherweise vielfältig

genutzt, sodass hier wertvolle Hinweise zu finden sind. Aus vielen Kommentaren wird deutlich, dass die Vorteile aus den verschiedenen Welten der Vorgehensmodelle in der Softwareentwicklung genutzt werden sollen. Für viele ist jedoch der Weg einer „Migration V-Modell zu agil“ noch unklar:

- „Testspezifika für agile Vorgehensmodelle, Rolle des Testers im agilen Vorgehensmodell, Standards/Empfehlungen für Dokumentation im agilen Vorgehensmodell“.
- „Testmethodik bei Integration agiler Entwicklung in ein phasengesteuertes Entwicklungsmodell“.
- „Paralleler Umgang im Test von Software Modulen aus Scrum und aus V-Modell“.
- „Zusammenführung der besten Eigenschaften von V-Modell & agilen Methoden (Zusammenführung von Phasen & iterativem Vorgehen), insbesondere organisatorisch hinsichtlich

Wissenschaftliche Leitung

- Prof. Dr. Andreas Spillner, Hochschule Bremen
- Prof. Dr. Karin Vosseberg, Hochschule Bremerhaven
- Prof. Dr. Mario Winter, FH Köln

Förderer des Projekts

- ANECON Software Design und Beratung G.m.b.H.
- German Testing Board e. V.
- Swiss Testing Board

Unterstützer des Projekts

- ASQF e. V.
- Austrian Testing Board
- dpunkt.verlag GmbH
- Softwareforen Leipzig GmbH
- GI-Fachgruppe TAV

Kasten 1: Hintergrundinformation der Umfrage 2011 – Softwaretest in der Praxis.

getrennten Organisationen von Entwicklung & Validation“.

Dies sind nur wenige Beispiele aus den sehr vielfältigen Kommentaren. Es bleibt also noch einiges zu tun.

Unter [Umf] finden Sie weitere Informationen zur Umfrage 2011 und Ergebnisse. Dort finden Sie auch Hinweise auf weitere Vorträge sowie Artikel und Berichte zur Umfrage mit jeweils anderen Schwerpunkten in der Auswertung. ■

Literatur & Links

[IST] German Testing Board e. V., ISTQB Certified Tester, siehe: http://www.german-testing-board.de/de/certified_tester.shtm

[Mül98] U. Müller, T. Wiegmann, O. Avci, „State of the Practice“ der Prüf- und Testprozesse in der Softwareentwicklung, Ergebnisse einer empirischen Untersuchung bei deutschen Softwareunternehmen, in: W. Mellis, et al (Hrsg.), Studien zur Systementwicklung, Band 16, 1998, siehe: <http://systementwicklung-archiv.bibliothek.informatik.uni-koeln.de/28//>

[Mül99] U. Müller, Prüf- und Testprozesse in der Softwareentwicklung, Shaker Verlag 1999

[Umf] Umfrage 2011 – Softwaretest in der Praxis, siehe: <http://www.softwaretest-umfrage.de/>