



□ Joachim Herschmann

[E-Mail: joachim.herschmann@microfocus.com]

ist als Director of Product Management bei Micro Focus verantwortlich für die Weiterentwicklung der Silk Produktlinie, einer Suite von Testautomatisierungs- und -managementprodukten, die vor kurzem durch die Übernahme von Borland ins Portfolio aufgenommen wurde. Bei Borland war Herr Herschmann zuletzt für die Entwicklung der Lifecycle Quality Management (LQM) Produktsuite verantwortlich, nachdem er zuvor bereits als Solutions Marketing Manager LQM die EMEA Region betreut hatte. Herr Herschmann kam im April 2006 durch die Akquisition von Segue, einem führenden Anbieter von Testlösungen, zu Borland. Während seiner Zeit bei Segue war Herr Herschmann mehrere Jahre als Technical Account Manager und Technical Consultant im Bereich Software Testing und Quality Assurance tätig. Herr Herschmann hat mehr als 16 Jahre Erfahrung im Softwaregeschäft, davon mehr als die Hälfte im Bereich Consulting und Testen.

Borland[®]
(A MICRO FOCUS COMPANY)

Agile Testprozesse – mit großem „A“ wie Automatisierung

Agile Entwicklungsprozesse laufen den traditionellen Entwicklungsmethoden immer mehr den Rang ab. Eine 2008 durchgeführte Studie [Amb08] zeigte zum Beispiel, dass bereits 69 Prozent der Befragten in ihren Softwareprojekten einen agilen Ansatz verfolgen. Dieser Artikel informiert Sie darüber, wie agile Entwicklungsteams durch Automatisierung ihre Funktions- und Regressionstests effizienter gestalten und gleichzeitig Softwarequalität und Projekttermine sichern können.

Unter den Anhängern agiler Entwicklungsprozesse gibt es viele verschiedene Meinungen zum Sinn und Unsinn automatisierter Testprozesse. Während man theoretisch noch darüber diskutiert, ob Automatisierung und agile Entwicklung zusammenpassen, sorgt die Praxis in den Unternehmen – mit ihren geografisch verteilten Teams und begrenzten Ressourcen – für einen wesentlich pragmatischeren Ansatz. Dass Teams zur aktiven Risikovermeidung die Qualität ihrer Projekte steuern und sichern müssen, war seit jeher ein zentrales Konzept in der Softwareentwicklung. Da agile Entwicklungsmethoden die Selbstverantwortung im Team in den Vordergrund stellen, ist es keine große Überraschung, dass Unternehmen skeptisch sind, wie zuverlässig ein solcher Geschäftsprozess zu bewerten ist. Eine weitere Variable, die die Entscheidungsfindung in der Softwareentwicklung noch zusätzlich komplizieren kann, ist der Einsatz von Tools zur Automatisierung jener neu eingeführten agilen Prozesse.

Diese Ängste gilt es zu überwinden. Automatisierte Testprozesse, die direkt in den Entwicklungsprozess eingebunden sind, können entscheidend dazu beitragen, sowohl neue, als auch bestehende Prozesse in den Unternehmen zu unterstützen. Bei der Prüfung agiler Testmethoden können

Richtlinien und praktische Erfahrungen aus erster Hand helfen, Bedenken hinsichtlich automatisierter Testmethoden in verschiedenen großen Entwicklungsumgebungen auszuräumen. Testautomatisierung bedeutet mehr Flexibilität und Reproduzierbarkeit und erfüllt somit, was agile Entwicklung verspricht: nämlich eine Steigerung der Effizienz und der Qualität in der Softwareentwicklung.

Automatisierung und agile Softwareentwicklung

In der agilen Softwareentwicklung werden tageweise Anforderungen erhoben, geändert und neu priorisiert – und das in immer kürzer werdenden Zeitabständen. Funktionalitäten müssen immer schneller abgebildet und dabei komplett ausgetestet werden, um sicherzugehen, dass weder die Anforderungen der Endanwender, noch die des Unternehmens außer Acht geraten. In einer agilen Prozessumgebung sind fortlaufende Tests zwingend erforderlich, da eine Funktion erst dann als „erledigt“ gilt, wenn sie alle erforderlichen Testfälle erfolgreich durchläuft. Die Größe und Komplexität moderner Software lassen keinen Zweifel daran aufkommen, dass manuelle Prozesse nur eine geringe Chance haben, die schnellen und reproduzierbaren Ergebnisse zu

liefern, die man benötigt, um in die Vorzüge der agilen Entwicklung zu kommen. Es ist unvermeidbar, dass die Qualität leidet, wenn Tests nicht automatisiert und schnell wiederholt werden können, weil dann Tests eingeschränkt werden, um eine pünktliche Projektabgabe zu sichern.

Ganz gleich, ob Ihr Entwicklungsteam noch darüber nachdenkt, agil zu arbeiten, sich in der Übergangsphase befindet oder bereits agil arbeitet: Es ist von entscheidender Wichtigkeit, darüber nachzudenken, wie Sie die neue Arbeitsweise technologisch unterstützen können – ganz besonders wenn es um Testprozesse geht. In allen wichtigen Testprozessen schlummern Optimierungspotenziale: in der Testautomatisierung, dem Testmanagement und im Anforderungsmanagement.

Auf diejenigen, die sich daran wagen, agile Testprozesse zu automatisieren, kommen etliche Herausforderungen zu – zum Beispiel agile und traditionelle Testmethoden parallel auszuführen, Tools zu finden, die für einen agilen Ansatz praktisch und flexibel genug sind, Teammitglieder mit und ohne technischen Hintergrund aktiv in den Testprozess einzubinden, die Integration von Open-Source-Produkten zu gewährleisten und eine test- bzw. modellgesteuerte Entwicklung zu ermöglichen. Überwältigende

Herausforderungen? Ja. Unüberwindbare? Ein klares Nein!

Wozu automatisierte Testprozesse in der agilen Entwicklung?

Werfen wir noch einen genaueren Blick auf die Gründe, die für die Automatisierung von Testprozessen in der agilen Entwicklung sprechen. Es sind Mitglieder des Projektteams, die an den Testprozessen beteiligt sind, und nicht etwa eine isolierte Gruppe, die als letzten Schritt vor der Freigabe den Code von den Entwicklern übernimmt. Idealerweise sollten Tester und Entwickler Hand in Hand arbeiten, wobei die Entwickler den Testern ihren Code frühzeitig und regelmäßig zum Abgleich der Akzeptanzkriterien zur Verfügung stellen sollten. Da Funktionen Schritt für Schritt entwickelt werden und Teams gleichbleibend zügig arbeiten müssen, müssen Code-Abschnitte schnell verifizierbar sein. Damit agile Ansätze wirklich funktionieren können, müssen Funktionstests schnell, iterativ und flexibel sein.

■ **Schnell, schneller, am schnellsten: Von der Programmierung direkt zum Testen** – In einem automatisierten Umfeld können Ihre Tester einfache Skripte erstellen, die sie immer wieder ausführen können. Das spart nicht nur Zeit, sondern bringt auch mehr Konsistenz in Ihre Testverfahren - und zwar über verschiedene User Stories bzw. Anforderungen innerhalb eines Projekts hinweg oder auch projektübergreifend. Um systematisch alle Funktionen auszutesten, können Sie Tests anhand von User Stories erstellen, die direkt und iterativ ausgeführt werden können. Die Automatisierung bringt Tempo in Ihre Testprozesse und erleichtert Ihren Testern maßgeblich die Arbeit. So gehören zermürbende Testmarathons weit nach Feierabend oder am Wochenende der Vergangenheit an.

■ **Gut, besser, reproduzierbar: Wiederverwendung der Tests und Skripte gegen die Acceptance Criteria** – In der agilen Entwicklung sollten nach jeder neuen Iteration Regressionstests ausgeführt werden. Bei Regressionstests müssen Sie darauf achten, dass Sie erstens immer die gleichen Tests ausführen, wenn Sie einen bestimmten Code-Abschnitt testen und zweitens, dass das betreffende Testskript die Akzeptanzkriterien der jeweiligen User Story abgleicht. Jedes Mal wenn

man Änderungen am Code vornimmt – bzw. ihn um neue Funktionalitäten erweitern – muss man bis auf die letzte Änderung für alle User Stories jegliche Funktionstests wiederholen, um unbeabsichtigte Auswirkungen auf andere User Stories auszuschließen.

Durch Fehlerquellen wie menschliches Versagen, Schwankungen und mangelnde Konsistenz ist Reproduzierbarkeit beim manuellen Testen so gut wie unmöglich. Beim besten Willen ist es Mitarbeitern nicht möglich, sich zu erinnern, welchen Test sie im letzten Iterationszyklus für einen ganz bestimmten Code-Abschnitt ausgeführt haben. Schon ein einziger Fehler kann zu Problemen im Endergebnis führen. Ganz zu schweigen davon, dass bei einer großen Code-Basis das manuelle Testen in aller Regel den für die Iteration bemessenen Zeitrahmen bei weitem sprengt. Im Gegensatz dazu kann man automatisierte, reproduzierbare Funktions- und Regressionstests, die konsistente Ergebnisse liefern, jederzeit ausführen.

Die Automatisierung verschafft Ihnen Vorteile, die mit manuellen Testverfahren unerreichbar sind. Beispielsweise sorgen schnell ausführbare, automatisierte Testskripte dafür, dass Sie von der Programmierung direkt zum Testen übergehen können und so die Prozesskette maßgeblich verkürzen. Auch werden Ihre Tests durch Automatisierung reproduzierbar und ermöglichen Regressionstests in jedem Sprint bzw. jeder Iteration. Durch ein gleichermaßen stabiles wie flexibles Testmanagement gestalten Sie Ihre Testprozesse nicht nur effizienter, sondern können darüber hinaus auch typische Fehlerquellen manueller Testprozesse ausschließen.

■ **Die richtigen Tools zur Testautomatisierung**
Das A und O in der agilen Softwareentwicklung ist die richtige Mischung von Menschen, Prozessen und Tools. Damit Sie tatsächlich von den mit agilen Ansätzen verbundenen Effizienz- und Qualitätssteigerungen profitieren können, ist es von entscheidender Wichtigkeit, die richtige Testsoftware einzusetzen. Dies ist allerdings eine zweiseitige Angelegenheit: Auf der einen Seite machen die manuellen Testprozesse, die heute in den meisten Unternehmen noch Gang und Gäbe sind, die Implementierung und das erfolgreiche Arbeiten mit agilen

Entwicklungsprozessen schwer, wenn nicht unmöglich. Und auf der anderen wurden automatisierte Testtools speziell für den Einsatz in traditionellen Entwicklungsumgebungen entwickelt, in denen Tester bestimmten Entwicklungspraktiken folgen, um exakte Testergebnisse zu erhalten. So ist es kein Wunder, dass viele Unternehmen vor dem Einsatz automatisierter Testtools zurückschrecken.

Es gibt Automatisierungstools, die sowohl traditionelle als auch agile Entwicklungsmethoden bestmöglich unterstützen. Das Testautomatisierungstool Silk4J beispielsweise bindet sich nahtlos in die agile Umgebung ein durch:

- Verwendung derselben Programmiersprache wie das Team
- Einfache Einbindung in jegliches agiles Testframework (wie Fitnesse, JUnit, Robot Framework u. a.)
- Integration in den Build-Prozess
- Flexible Objekterkennung ermöglicht das GUI-Testen von Anfang an und jederzeit.

Fazit

Immer mehr Unternehmen entdecken agile Softwareentwicklungsprozesse für sich und damit auch die verbundenen Herausforderungen. Die Arbeitsbedingungen in den Unternehmen – geografisch verteilte Teams, Kostensenkungen, immer kürzer werdende Abgabefristen und beschränkte Ressourcen – bedingen eine pragmatische Herangehensweise an das Thema. Moderne automatisierte Testlösungen spielen eine zentrale Rolle in dieser Entwicklung, da sie agile Teams in die Lage versetzen, Teststrategien zu etablieren, die flexibel und reproduzierbar sind und darüber hinaus präzise Ergebnisse liefern. Zusammenfassend kann man sagen, dass Unternehmen durch agile Testautomatisierung das Risiko minimieren können, Software zu liefern, die qualitative Mängel hat oder nicht sicher läuft. ■

Literatur

[Amb08] Ambler, Scott: "Has Agile Peaked? Lets Look at the Numbers" ("Große Zuwächse bei agilen Entwicklungsmethoden? Zahlen und Fakten") Dr. Dobbs Journal, Mai 2008. <http://www.ddj.com/architect/207600615?pgno=1>.