

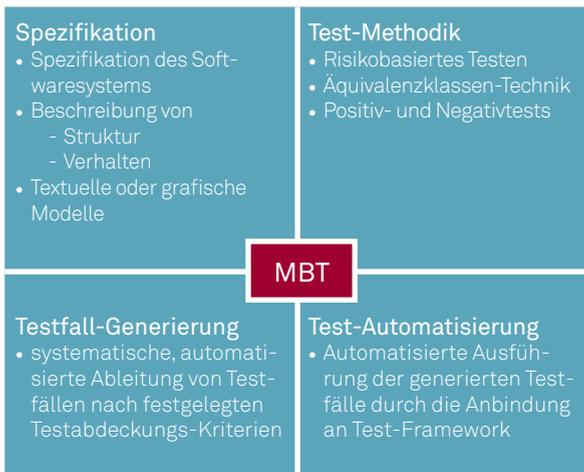
Modellbasiertes Testen (MBT)

Anwendungen effizient und automatisiert testen

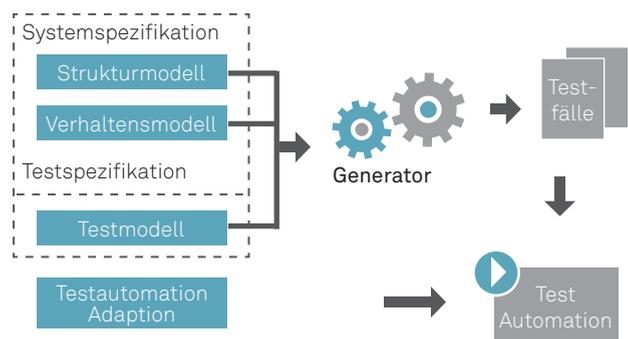
Bei modellbasiertem Testen werden die Testfälle zur Absicherung eines Software-Systems systematisch nach festgelegten Abdeckungskriterien aus Modellen abgeleitet. Dies ermöglicht eine sehr effiziente Erstellung von qualitativ hochwertigen Testfällen sowie einen hohen Grad der Testautomatisierung, da die generierten Testfälle in der Regel leicht an Testautomatisierungswerkzeuge angebunden werden können.

Definition

Die Entwicklung von Testfällen für die Absicherung von Enterprise-Anwendungen ist mit beträchtlichem Aufwand verbunden. Oft leitet das Testteam die Testfälle nach mehr oder weniger willkürlichen Kriterien aus den Spezifikationsdokumenten ab.



Das Ziel von modellbasiertem Testen ist, die Durchführung der Tests und deren Erstellung zu automatisieren. Durch die systematische Ableitung von Tests aus Spezifikationsmodellen wird die Testfallerstellung transparenter und besser steuerbar. Dies führt zu höherer Wirtschaftlichkeit im Testprozess, sowie einer personenunabhängigen Qualität der Testfälle. Als Basis für die Generierung von Testfällen dienen die System- und Testspezifikation in Form von Modellen, wobei folgende Modelltypen unterschieden werden:

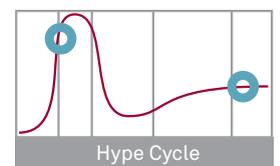


- Ein Strukturmodell ist, ebenso wie ein Verhaltensmodell, Teil der Systemspezifikation und enthält keine testspezifischen Informationen. In diesen beiden Modelltypen werden Aufbau und Funktionalität des Systems beschrieben.
- Ein Testmodell reichert die Modelle der Systemspezifikation mit zusätzlichen, Test-relevanten Informationen wie Prüfschritte, Risikobewertung und Testabdeckungskriterien an.

Die Testfälle werden systematisch mit Hilfe eines Testfall-Generators aus diesen Modellen abgeleitet. Bei ausreichender Detailtiefe der Modelle können die Testfälle zur automatischen Ausführung an ein Test-Framework angebunden werden. Da durch die Generierung potentiell sehr viele Testfälle entstehen, ist dies fast eine Notwendigkeit.

Reifegrad

Textuelle Modelle erfreuen sich in den letzten Jahren immer größerer Beliebtheit. Durch ihre Ausdrucks-



stärke und Mächtigkeit verhelfen sie auch dem modellbasierten Test zu einem neuen Boom. Die verfügbaren Frameworks sind mittlerweile reif für den produktiven Einsatz. Besonders durch die Popularität von UML um die Jahrtausendwende sind bereits viele Ansätze für modellbasiertes Testen entstanden. Die UML-basierten Produktsuiten besitzen deshalb inzwischen einen hohen Reifegrad. Im Enterprise-Software-Segment spielt UML zur detaillierten Spezifikation von Verhalten jedoch keine nennenswerte Rolle mehr.

Marktübersicht



Unter den UML-basierten Tools sind die Conformiq ToolSuite, Expecco der Firma eXept Software AG sowie Telelogic Rhapsody zu nennen. Auch Siemens hat ein UML-basiertes Tool TDE/UML hervorgebracht. Microsoft hat ein modellbasiertes Test-Tool namens SpecExplorer entwickelt, das auf den eigenen Modellierungssprachen Spec# und AsmL basiert. Die auf textuellen Modellen basierenden Tools sind hauptsächlich im Umfeld von Behavior-Driven-Development (BDD) entstanden. Prominente Vertreter sind JBehave, Cucumber sowie Jnario.

Alternativen

Als Alternative zum modellbasierten Ansatz bleibt nur die herkömmliche, manuelle Ableitung der Testfälle nach klassischen Test-Methodiken. Hier unterliegt die Qualität der Testfälle maßgeblich dem Know-how des Test-Designers.

msg systems ag

Robert-Bürkle-Straße 1 | 85737 Ismaning/München
 Telefon: +49 89 96101-0 | Fax: +49 89 96101-1113
www.msg-systems.com | info@msg-systems.com

Referenzszenario

Modellbasiertes Testen ist vor allem in Verbindung mit modellgetriebener Entwicklung sinnvoll, da die benötigten Modelle schon vorhanden sind und nicht eigens für den Test erstellt werden müssen. Aber auch ohne bereits vorhandene Modelle kann modellbasiertes Testen sinnvoll eingesetzt werden. Die Modelle müssen zwar explizit für den Test erstellt werden, können aber auf die für den Test relevanten Informationen beschränkt werden.

Business Impact

Die auf textuellen Modellen basierenden Testsprachen bieten großes Potential für eine effiziente Erstellung qualitativ hochwertiger Tests mit weitreichenden Möglichkeiten zu deren automatisierter Ausführung. Mit ihrem Einsatz kann eine hohe Wirtschaftlichkeit bei der Testfall-Erstellung sowie -Durchführung erreicht werden.

Pro	Contra
Effiziente Testfall-Erstellung	Einarbeitung in Modellierungs-Toolkette nötig
Transparenz und Steuerbarkeit der Testfallentstehung	Nur in Verbindung mit Testautomatisierung sinnvoll
Testqualität personenunabhängig	
Anbindung an Testautomatisierung	

