



MBTsuite

THE TESTING FRAMEWORK

Modell

Generierung

Test

[illegible]

Download der Testversion:
www.mbtsuite.de



sepp.med gmbh

Geschäftsführer

Dipl. Betriebsw. (FH) Barbara Bilen

Dipl. Inf. Florian Prester

Dipl. Kffr. Maria Prester

Handelsregister Fürth HRB B 6131

Umsatzsteuer-ID: DE 179 168 106

AFRA GmbH

Geschäftsführer

Dipl. Math. Peter Kreutzer

Dipl. Math. Bernhard Nuber

Handelsregister Fürth HR B 3797

Sitz der Gesellschaft: Erlangen

Umsatzsteuer-ID: DE 229 229 489

Impressum: Konzept, Text und Grafik: sepp.med gmbh & AFRA GmbH
Copyright by sepp.med gmbh & AFRA GmbH

Version 1.0

MBTsuite - Ihr Schlüssel zum Erfolg

Mit der **MBTsuite** wird der modellbasierte Test in einem neuen Format realisiert.
Reduzieren Sie in Ihrem Testprozess Kosten und steigern Sie gleichzeitig die Qualität!

Vorteile und Nutzen

Die **MBTsuite** erzeugt ausführbare Testfälle und Testdaten automatisch aus grafischen Modellen. Sie deckt alle Anforderungen aus dem Komponententest bis hin zum Systemtest perfekt ab.

Die erzeugten Testfälle sind plattformunabhängig und können in beliebigen Formaten ausgegeben werden. Für eine manuelle Testdurchführung können die Tests in Excel, Word, HP Quality Center etc. erzeugt werden. Aus den gleichen Modellen können aber auch Tests für SilkTest, QuickTestPro oder jede beliebige Scriptsprache entstehen.

In der agilen Entwicklung profitieren SCRUM-Teams von den Modellen als Entscheidungs- und Diskussionsgrundlage.

AFRA und sepp.med haben das modellbasierte Testen (MBT) konsequent zur Best Practice Methode **modellzentrierter Test (.mzT)** weiterentwickelt.

Durch die Kombination von **MBTsuite** und **modellzentriertem Test** lässt sich der Gewinn steigern.
Der Nutzer profitiert davon in vielfältiger Weise durch die:

- Steigerung der Qualität der Testspezifikation
- Standardisierung
- Systematik und Nachvollziehbarkeit
- Wartbarkeit und Wiederverwendbarkeit
- Effizienzsteigerung

Vorgehensweise



Die Methodik des **modellzentrierten Tests (.mzT)** stellt das Testdesign komplett in grafischer Form dar. Die **MBTsuite** generiert aus UML-Testmodellen ausführbare Testfälle. Damit bildet die **MBTsuite** die Brücke zwischen Modellierungs-, Testmanagement- und Testautomatisierungswerkzeugen.

Neben der Definition von Testabdeckungsstatistiken bietet die **MBTsuite** weitere Möglichkeiten zur Unterstützung des Testmanagements und der Testplanung, unter anderem die Testplanvisualisierung, Testpriorisierung, Anforderungsverfolgung und Fehlerrückverfolgung.

.mzT - der modellzentrierte Test

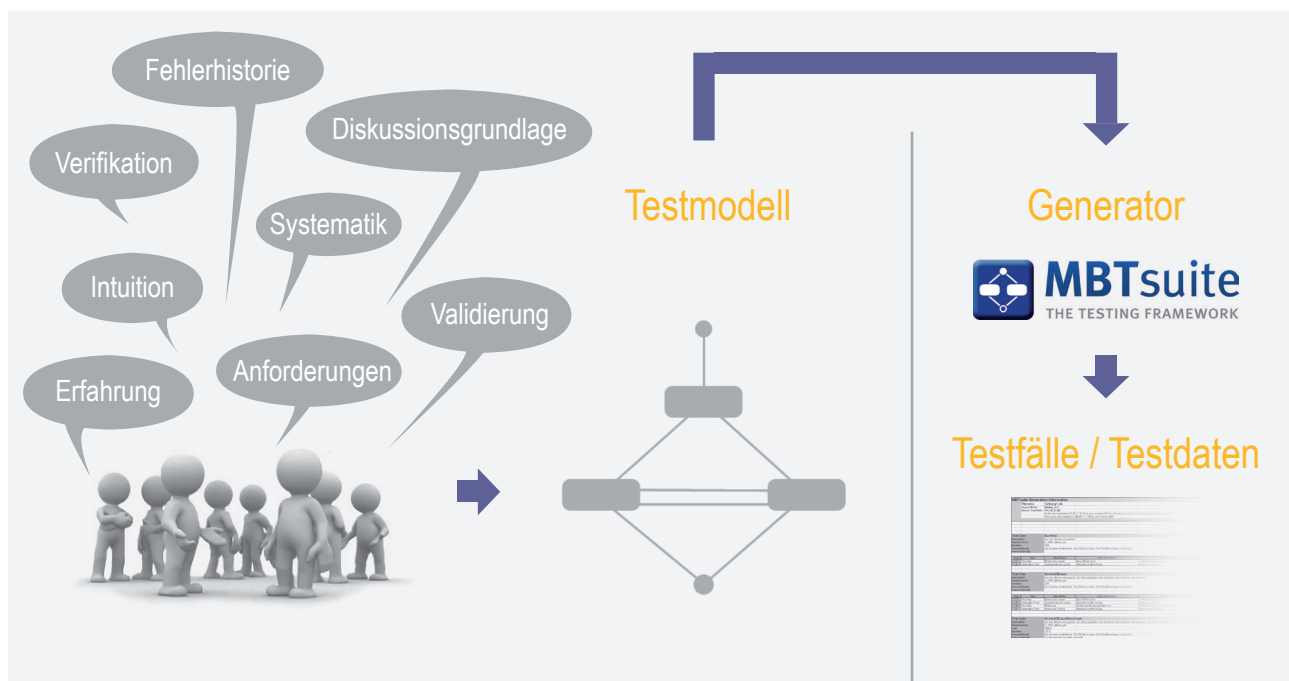
Die Methode **.mzT** definiert Testinhalte unter Verwendung von UML / BPMN -Werkzeugen. Sie fasst alle für den Test relevanten Informationen in einem Modell zusammen, aus dem die benötigten Testfälle generiert werden. Die Generierung der Testfälle lässt sich mit Hilfe der **MBTsuite** auf vielfältige Art und Weise steuern.

Die Modelle dienen sowohl der Konkretisierung der Anforderungen als auch der Sichtbarkeit von Abläufen und Zusammenhängen. Sie sind außerdem die Grundlage für eine systematische und automatische Testfallerstellung. Die Modelle können auch Informationen zur Steuerung der Testtiefe bzw. der Testabdeckung enthalten.

*„Bei der Methodendefinition ist unser oberstes Ziel, die Bedienung für den Testingenieur so intuitiv wie möglich zu gestalten. Wir streben höchste Flexibilität in der Anwendung an. So ist der Testingenieur mit **.mzT** in der Lage, vom Unit-Test bis zum Kundenabnahmetest alle Anforderungen an ein System zu beschreiben.“*

MBT-Experte Dr. Martin Beisser

Die Methode **.mzT** ermöglicht die systematische Erstellung von Testfällen durch Modelle und deren Wiederverwendbarkeit.



Systematischer Ablauf der Testfallerstellung

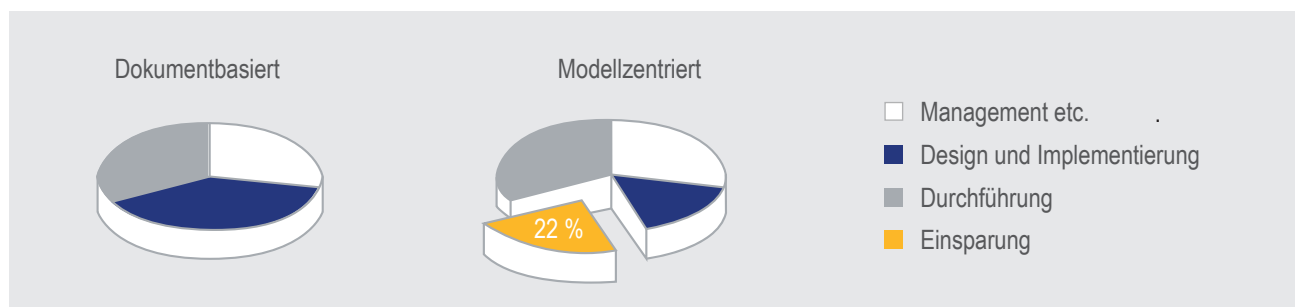
Die Methodik **.mzT** ist ohne große Vorkenntnisse erlernbar, da nur wenige Elemente der UML verwendet werden. sepp.med und AFRA bieten standardisierte Schulungen und Workshops rund um das Thema **.mzT** und die **MBTsuite** an. Aktuelle Termine entnehmen sie bitte der Webseite unter www.seppmed.de/sepptrain beziehungsweise www.afra.de/schulungen

Wirtschaftlichkeit

Effizienz

Neben allen bisher genannten Vorteilen des modellzentrierten Vorgehens im Testdesign, reduziert diese Methode den Aufwand für die Testfallerstellung deutlich. Dies wird mittlerweile durch mehrere unabhängige Studien belegt. Im Schnitt wird der Aufwand bei der Neuerstellung von Testfällen um 20% bis 40% verringert. Noch deutlichere Einsparungen bis zu 80% sind bei der Pflege und Wartung zu erreichen.

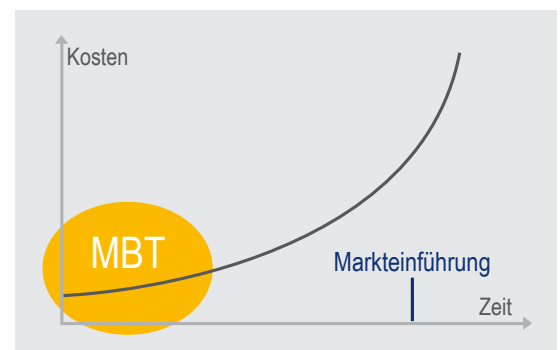
Die nachfolgende Grafik fasst die Ergebnisse des Vergleichs von modellbasierter mit dokumentenbasierter Testfallerstellung zusammen.



Benefits für das Requirements-Management

Durch die konsequente Visualisierung der Testanforderungen bietet .mzT eine Effektivitäts- und Effizienzsteigerung für das gesamte Projekt im Anfangsstadium: Es werden Unzulänglichkeiten in den Anforderungen aufgedeckt und damit dem Requirements-Engineering zugearbeitet. .mzT unterstützt die Aufdeckung von Inkonsistenzen zwischen Anforderung und Entwicklung.

Der Einsatz von .mzT/MBTsuite führt „immer“ dazu, dass Requirements noch einmal überarbeitet werden und „abgeschlossene“ Entwicklungen vor der Markteinführung überprüft werden; dies unterstützt den Test-First-Ansatz optimal.



Geringe Kosten bei früher Fehlererkennung

Time-to-Market

Die Methodik .mzT und die MBTsuite ermöglichen es bereits während einer frühen Phase in der Entwicklung mit der Testfallmodellierung zu beginnen. Im modellgestützten Testdesign sind Änderungen an den Anforderungen schnell und unkompliziert anpassbar.

Die Verwendung des modellbasierten Testdesigns mit .mzT bietet einen erheblichen Wettbewerbsvorteil gegenüber Marktbegleitern. Dies wirkt sich bei der Wiederverwendbarkeit der Testmodelle, insbesondere bei Produkten mit einem kurzen Produktlebenszyklus, aus.

Funktionalitäten

Modellierung

Generierung

Tools



MID
the modeling company
Innovator



Microsoft®
Visio

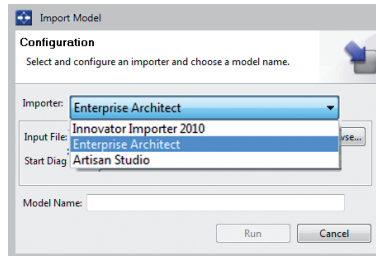
Requirements

HP
Quality Center

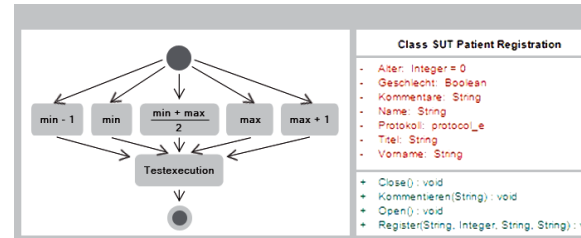
Rational
RequisitePro

IBM
Rational Doors

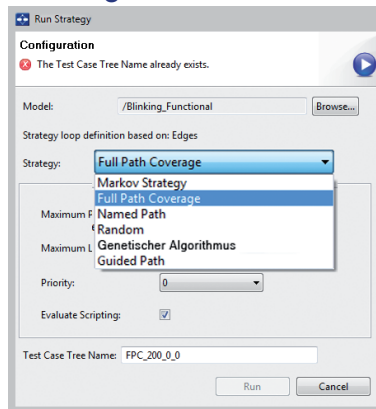
Import



Testdaten



Strategien



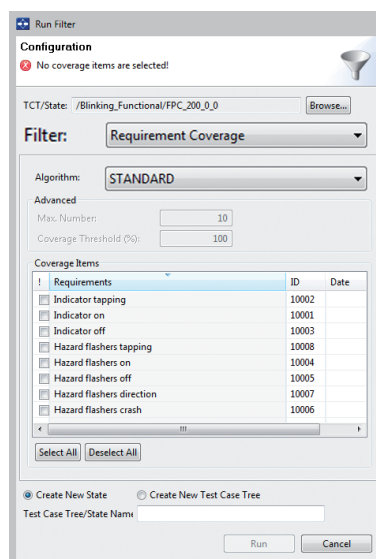
Test Case Tree

- ActivityInitial-Initialization
- ActivityInitial-Precondition
- Precondition-Initialization variables
- Initialization variables-ActivityFinal
- Initialization-Directional blinking
- Right blinking
- Right indicator tapping
- Check short right blinking
- Right blinking
- Directional blinking-ActivityFinal
- Directional blinking-Hazard flash
- Crash simulation
- Crash simulation-Crash
- Hazard flashers crash
- Check hazard
- Crash-Post
- Post c
- Hazard flash
- Left
- Left
- Activate hazard flashers
- ActivityInitial-Activate h
- Activate hazard flash
- Check hazard flash
- Activate hazard
- Hazard flash
- Hazard flash
- Left
- Left

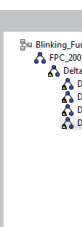
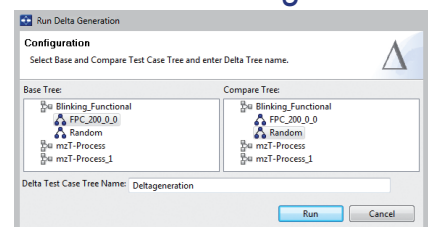
Statistics

Property	Value
Used Strategy...	Full Path Coverage
Scripting Used	true
Path Length	200
Loops	0
Priority	0
Number of Tes...	56
Generation Time	0.054s
Covered Edges	64 (92%)
Available Edges	69
Covered Nodes	58 (100%)
Available Nodes	58
Average Test C...	21.89
Maximum Test...	29
Minimum Test ...	11
Loops based on	Edges
Requirements ...	0
Test Cases with...	0
Elements with ...	0
Generation Date	16.09.11 08:59
User	seppmed_user
Requirements r...	8 (100%)

Filter



Delta Generierung



So profitieren Sie von .mzT und der MBTsuite

Die Methode **.mzT** in Kombination mit der **MBTsuite** wird seit Jahren in unterschiedlichen Projekten eingesetzt. Das standardisierte und dokumentierte Verfahren bringt umfangreiche Vorteile mit sich:

Qualität der Spezifikation

- Die Requirements der zu testenden Systeme werden in Modellen dargestellt, die leicht und schnell zu erfassen sind. **.mzT** Modelle sind eindeutig und konkret und bieten damit eine ideale Kommunikationsgrundlage für alle Beteiligten. Umfangreichere Testbeschreibungen und -spezifikationen in Textform entfallen, was den Aufwand für Analysen und Reviews reduziert.
- Mit hierarchisch strukturierten Diagrammen lassen sich komplexe Systeme aus einer groben Beschreibung heraus entwickeln und verfeinern. Die Darstellung der einzelnen Modelle bleibt so immer kompakt und übersichtlich, enthält aber dennoch genügend Detailinformationen, um das System umfassend und genau zu beschreiben.
- Während der Testmodellierung wird zugleich die Qualität der Anforderungsbeschreibung überprüft. Fehlerhafte und unzureichende Anforderungen lassen sich somit bereits vor der Testausführung aufdecken. Die Modellierung ist der erste Schritt der Qualitätssicherung.
- Aus den Modellen lassen sich Testbeschreibungen in unterschiedlichen Formaten ableiten. Zum Beispiel kann parallel zur Generierung von automatischen Tests ein Dokument in MS Word als Testspezifikation erzeugt werden, wenn das zur Einhaltung der Regularien benötigt wird.

Standardisierung

- Mit Hilfe der festgelegten Methodik (**.mzT**) in Verbindung mit einer einheitlichen Modellierungssprache wird die Testbeschreibung standardisiert und über Abteilungs- und Domänengrenzen hinweg verständlich und somit austauschbar.
- Testfälle werden in Modellen einheitlich und ohne Redundanzen durch abweichende Formulierungen dargestellt. Identische Zustände oder Aktivitäten im Testfall werden auch identisch beschrieben.
- Durch die Generierung ist es möglich, verschiedene Domain Specific Languages (DSL) automatisiert zu erzeugen und damit verschiedene Testumgebungen gleichzeitig aus einem Modell anzusprechen.

Agilität

- Der Top-Down- Ansatz ermöglicht es, frühzeitig mit dem Testdesign zu beginnen, selbst wenn Teile des Systems noch nicht vollständig spezifiziert sind.
- Für das Verifizieren der in Sprints erzeugten Inkremente können Testmodelle entsprechend schnell und leicht modifiziert werden.
- Die Modelle eignen sich hervorragend, um in SCRUM-Teams Absprachen zu treffen und Abhängigkeiten sichtbar zu machen.

Systematik und Nachvollziehbarkeit

- Die Modellierung verschafft den vollständigen Überblick über das zu testende System. Dadurch lässt sich die Testüberdeckung auf Funktions- oder Workflow-Ebene definieren und abschätzen.
- Pfade, die zu Fehlern geführt haben, lassen sich im Modell anzeigen. Dies erleichtert die Impact-Analyse und ermöglicht es, genau die Teile des Systems zu identifizieren, die von einer Änderung nach der Fehlerbehebung beeinflusst wurden.
- Die Anforderungen (Requirements) können an beliebigen Stellen in den Modellen hinterlegt werden. Bei der Abarbeitung durch die **MBTsuite** werden sie bis in die einzelnen Testschritte weitergegeben. So lassen sich im Fall eines Fehlers ganz einfach und exakt die betroffenen Anforderungen identifizieren.
- Modelle erleichtern das Testmanagement z. B. durch die Priorisierung von Testfällen innerhalb des Modells oder durch die Erstellung von Metriken zur Testabdeckung.
- Durch die automatische Testfallerstellung können Tests neuen oder veränderten Situationen angepasst werden. Das Testmanagement ist nicht auf einen einmal erstellten Testfall angewiesen.

Wartbarkeit und Wiederverwendbarkeit

- Aus Diagrammen und Sequenzen können komplette Testbibliotheken aufgebaut werden. Bereits erstellte Modelle lassen sich wiederverwenden, wenn Aktivitäten innerhalb einer Testspezifikation oder zwischen Testspezifikationen verschiedener Testphasen wiederkehren.
- Die Verringerung der Zahl der Artefakte reduziert den Wartungsaufwand deutlich. Durch eine klare Strukturierung und Abstrahierung lassen sich die Modelle leichter anpassen und optimieren als in einer Testfalldatenbank.
- Das Konfigurationsmanagement wird vereinfacht; statt unzähliger Testfälle müssen nur noch wenige Modelle archiviert und versioniert werden.

Testabdeckung

- Auf Modellebene und auf der Basis der generierten Testfälle lässt sich eine Aussage über die tatsächliche Testabdeckung bezüglich der Anforderungen treffen.
- Gegen die Modelle lassen sich konkrete, messbare Abdeckungsmaße definieren, vergleichbar der Abdeckungsmaße beim White-Box-Test.

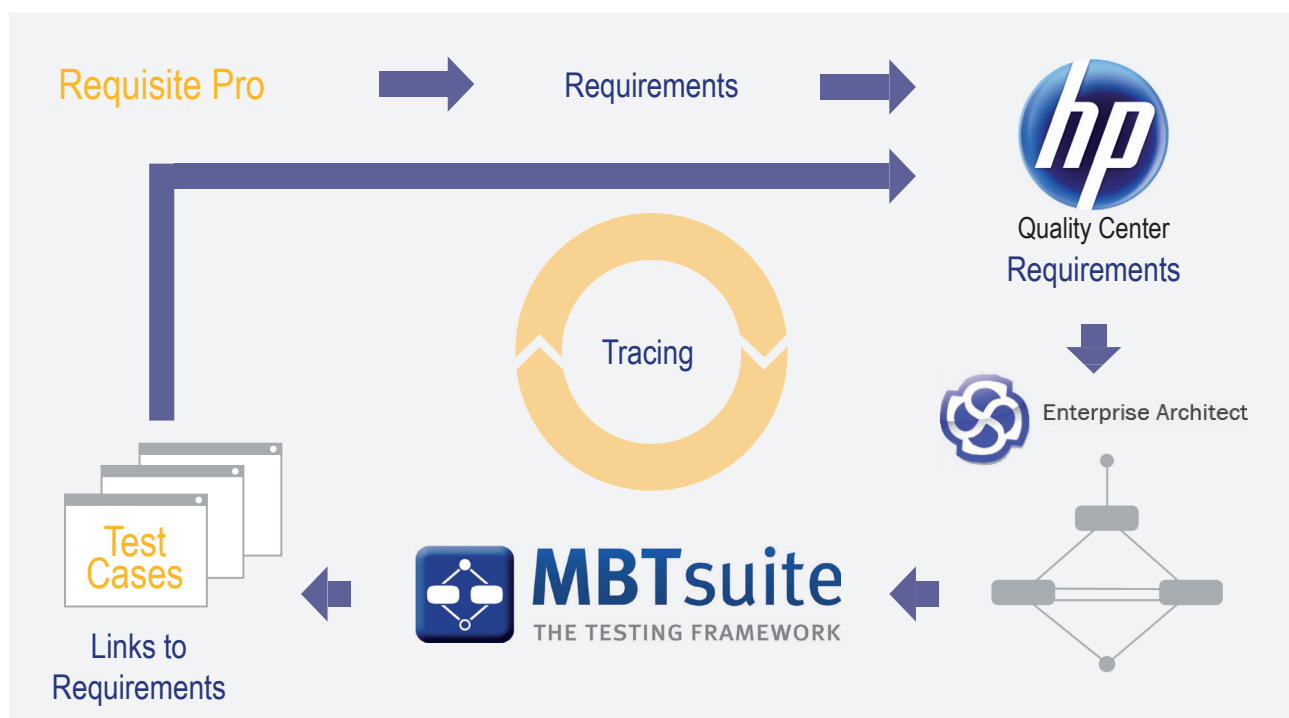
Erfahrungsbericht - MBTsuite im industriellen Einsatz

In einem gemeinsamen Projekt der Universität Erlangen-Nürnberg, Siemens Healthcare, AFRA und sepp.med wurde die **MBTsuite** implementiert und in eine Kette von existierenden Werkzeugen integriert:

Die Software „syngo.via“ von Siemens ist eine proprietäre Plattform für bildgebende Verfahren und vereinigt verschiedene Funktionalitäten dieser Geschäftsbereiche sowie multiple Kunden- und Marktanforderungen. Darüber hinaus ist sie ein Medizinprodukt und muss gemäß den Vorgaben und Normen entsprechend entwickelt werden. Unter anderem gilt, dass alle Anforderungen lückenlos durch die Entwicklung und den Test verfolgbar sein müssen, sowie weitere Forderungen z.B. hinsichtlich der Sicherheit oder der Gebrauchstauglichkeit. Es handelt sich also um eine mehrdimensionale und sehr komplexe Anforderungslandschaft.

Die Beschreibung der Testfälle war sehr umfangreich und wenig änderungsfreundlich. Gleichzeitig haben sich die Anforderungen an die Software sehr oft und noch kurz vor der geplanten Fertigstellung geändert. Unter Mitwirkung der Firmen AFRA GmbH aus Erlangen und sepp.med gmbh aus Röttenbach wurde 2007 der modellbasierte Test nahtlos mit folgender Werkzeugkette in den Prozess eingeführt:

1. Automatischer Abgleich von in Requisite Pro verwalteten Anforderungen mit dem HP Quality Center
2. Import der Anforderungen in das UML-Modell und Verlinkung im Modell
3. Import erstellter Modelle aus Enterprise Architect in die **MBTsuite**, daraus Generierung von Testfällen mit zugehörigen Testdaten.
4. Automatische Übertragung dieser Testfälle ins HP Quality Center, diese stehen mit Verlinkung zu den Anforderungen für die Testplanung und -durchführung zur Verfügung.
5. Die Verbindung zwischen Requisite Pro und dem HP Quality Center ermöglicht die geforderte Zuordnung von Testfällen und deren Durchführung mit den Testergebnissen zu den ursprünglichen Anforderungen.



Der Paradigmenwechsel in der Testfallspezifikation von textbasiertem Verfahren zur Modellierung wurde reibungslos und gezielt vollzogen.

Um eine laufende Software-Entwicklung und die Freigabe gemäß einem verabschiedeten Projektplan nicht zu gefährden, wurde mit der Schulung eines Teammitgliedes begonnen. Dieses Teammitglied modellierte den Test eines abgegrenzten Teilprojektes. In regelmäßigen Teambesprechungen wurde der Fortschritt berichtet und die gesammelten Erfahrungen diskutiert, um die Wiederholung von Fehlern zu vermeiden.

Neben den bisherigen Teambesprechungen zum allgemeinen Projektfortschritt wurden spezielle Meetings für die Einführung und Umsetzung der neuen Methode eingeführt. Dabei konnten „Lessons Learned“ ebenso klar angesprochen werden wie die Planung des weiteren „Roll-Outs“ und die Unterstützung durch die Leitung.

Am Ende der Einführung arbeiteten fünf der 13 Teammitglieder an der Modellierung und hatten erkannt, dass die Methode viele Vorteile bringt. Eine wesentliche Voraussetzung für die erfolgreiche Einführung des modellbasierten Tests ist die Akzeptanz aller Beteiligten. Für das erfolgreich durchgeführte Projekt wurden 29 Modelle erstellt und damit 50% aller Testfälle automatisch erzeugt.

Die folgenden Erfahrungen konnten bei der Pilotierung und beim anschließenden Roll-Out gemacht werden:

- Die visuelle Darstellung ermöglicht es, schon lange vor der Testausführung Fehler in der Spezifikation zu erkennen und sich innerhalb eines weltweit kooperierenden Teams sowie mit den Entwicklern besser zu verständigen.
- Die erzeugten Modelle sind anpassbar und dadurch nach einer Änderung schneller wieder verwendbar.
- Anstatt Reviews von Hunderten von Testfällen durchzuführen werden die Testmodelle für das Review herangezogen.
- Eine geforderte Testabdeckung kann auf Modellebene durch den Einsatz der **MBTsuite** nachgewiesen werden.
- Nach Änderungen der Requirements sind die Modelle sehr schnell anpassbar und die Testfälle damit schnell aktualisiert.

Der oben beschriebene Informationsaustausch wird über die Einrichtung eines „Wikis“ institutionalisiert. Erfahrungen und wichtige Neuigkeiten rund um das Thema „Modellbasierter Test@Magnetresonanz“ können ins Intranet gestellt und dort auch gelesen werden. In einem nächsten Schritt wird die Testautomatisierung mit Hilfe der Technik des „Keyword Driven Tests“ über MBT ermöglicht.

Eine Erweiterung ist durch die Anwendung der **MBTsuite** in der agilen Entwicklung vorgesehen. Die Entwicklung und die Testmodellierung können für Sprints gleichzeitig gestartet werden. Testfälle werden für jedes Inkrement und jeden Sprint schnell erzeugt.

Mit einer geplanten Einführung der **MBTsuite** können nicht nur die Ergebnisse hinsichtlich einer Nachverfolgbarkeit (also der Absicherung und Steigerung der Effektivität) sondern auch der Einsatz von Ressourcen dafür (die Effizienz) gesteigert werden. Von entscheidender Bedeutung für die Einführung ist die gleichzeitige konsequente Einbindung aller involvierten Mitarbeiter und die Gewährleistung eines permanenten Kommunikationsprozesses durch das Management. So wird die Anerkennung und Mitwirkung aller Beteiligten gesichert und die Methode und das Werkzeug werden zum „Selbstläufer“.



sepp.med gmbh
Gewerbering 9
91341 Röttenbach

Fon: +49 (0) 91 95 - 9 31 - 0
Fax: +49 (0) 91 95 - 9 31 - 300

E-Mail: info@seppmed.de
Web: www.seppmed.de



AFRA GmbH
Henkestraße 77
91052 Erlangen

Fon: +49 (0) 91 31 - 82611 - 0
Fax: +49 (0) 91 31 - 82611 - 40

E-Mail: info@afra.de
Web: www.afra.de
