

# Aktuelle Fragestellungen zum Zusammenspiel von BDD, MBT und KDT

Benedikt Eberhardinger<sup>1</sup>, David Farago<sup>2</sup>, Mario Friske<sup>3</sup>, Dehla Sokenou<sup>4</sup>

<sup>1</sup>Universität Augsburg, <sup>2</sup>QPR-Technologies, <sup>3</sup>Friske Consulting, <sup>4</sup>GEBIT Solutions

**Zusammenfassung.** Im Folgenden werden die Ergebnisse des letzten Treffens unseres Arbeitskreises „Testen objektorientierter Programme/Modellbasiertes Testen (TOOP/MBT)“ kurz dargestellt. Das Treffen mit 21 Teilnehmern fand im Rahmen des 39. Treffens der GI-Fachgruppe „Test, Analyse und Verifikation von Software (TAV)“ am 24.06.2016 in Bremen statt. Thema der inhaltlichen Diskussion war abermals, wie die drei Testmethoden Behavior-Driven Development (BDD), Model-Based Testing (MBT) und Keyword-Driven Testing (KDT) zusammenspielen. Anhand eines webbasierten Bankautomaten als Beispiel wurden die Vor- und Nachteile der drei Methoden betrachtet, sowie deren Zusammenspiel. Es wurden Hypothesen aufgestellt und diskutiert sowie zentrale Fragestellungen für die weitere inhaltliche Arbeit herausgearbeitet.

## 1 Hintergrund

Auf dem AK-Treffen während der TAV 37 in Friedrichshafen haben wir uns unter dem Einfluss der TAV-Vorträge entschieden, das Thema Model-Based Testing (MBT) erneut in den Fokus unserer inhaltlichen Arbeit im AK zu nehmen. Dabei stellte sich uns die Frage, warum in der Praxis derzeit ein so starkes Interesse an den Techniken Behavior-Driven Development (BDD) und Keyword-Driven Testing (KDT) besteht, jedoch oftmals weniger auf MBT-Techniken gesetzt wird.

Der Vergleich der drei Techniken wurde als neues inhaltliches Arbeitsthema mit sehr großem Interesse innerhalb des AK aufgenommen. Im Nachgang des Treffens wurde in Telefonkonferenzen ein einführender Artikel und ein Vortrag für die TAV 38 in Paderborn ausgearbeitet, siehe [1].

Nach der TAV 38 haben wir beschlossen, die drei Methoden mit verschiedenen Werkzeugen auf ein gemeinsames Beispiel anzuwenden und anschließend zu vergleichen. Dazu haben wir einen webbasierten Simulator eines Geldautomaten, engl. Automatic Teller Machine (ATM), implementiert. Dieser diente als System-under-Test (SUT). Mit den Werkzeugen Selenium [2], Wetator [3] und Integrity [4] (jeweils KDT), Cucumber [5] (BDD) und Tedeso [6] (MBT) haben wir jeweils den gleichen Ausschnitt der Funktionalität getestet. Dabei zeigten sich Überschneidungen und Integrationspotenziale. Diese praktischen Ergebnisse bildeten die Grundlage für die inhaltliche Arbeit auf der TAV 39. Die zugehörigen Folien sowie das ausführliche Protokoll des AK-Treffens sind auf der AK-Webseite [7] zu finden.

## 2 Diskussion

Die offene Diskussion fokussierte sich auf die Gemeinsamkeiten und Unterschiede der Techniken, mit vielen Beiträgen aus dem praktischen Einsatz der Teilnehmer des AKs. Im Laufe der Diskussion wurde die Grafik in Abbildung 1 erarbeitet. Sie zeigt primäre Nutzer der Techniken sowie Beziehungen zwischen den Techniken und daraus resultierende Fragestellungen. Auf diese einzelnen Aspekte gehen wir im Folgenden vertieft ein.

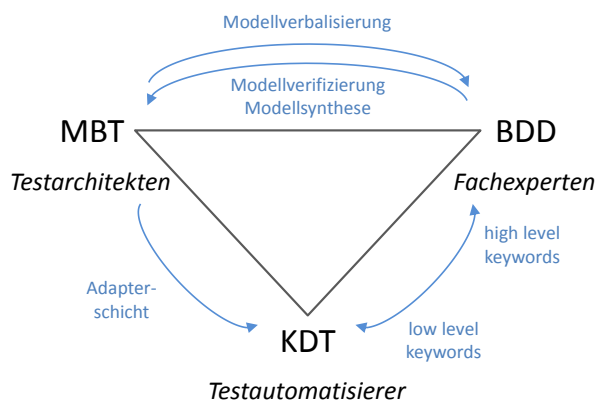


Abbildung 1: Zusammenhang zwischen MBT, BDD, KDT

Im Hinblick auf die primären Nutzer unterscheiden sich die Techniken wesentlich: BDD adressiert vorrangig Fachexperten, MBT wird aufgrund der Komplexität primär von Testarchitekten eingesetzt und KDT wird überwiegend von Testautomatisierern (Testdesignern) genutzt.

In der Diskussion wurde zunächst eine klare Verbindung zwischen KDT und MBT nochmal unterstrichen. KDT kann innerhalb von MBT dazu genutzt werden, um systematisch einzelne Schritte zu automatisieren. Mit anderen Worten: Mithilfe von KDT kann die zur Automatisierung von MBT notwendige Adapter-schicht realisiert werden. Dies soll in Kürze anhand unserer ATM-Fallstudie vertieft gezeigt werden.

Der Zusammenhang zwischen BDD und KDT ist hingegen nicht so offensichtlich. Beide Techniken haben die Testautomatisierung zum Ziel und unterstützen mehrere Abstraktionsebenen, unterscheiden sich jedoch im Fokus.

BDD hat einen klaren Fokus auf leichte und verständliche Darstellung (Modellierung) von Kundenanforderungen und kann somit gut zur Kommunikation mit Kunden und Fachexperten genutzt werden.

BDD ist aber nicht unbedingt immer auf der gleichen, feingranularen Ebene von KDT zu sehen, da sich KDT eher auf die Automatisierung implementierter Schnittstellen, beispielsweise Benutzeroberflächen, fokussiert und von diesen ausgeht. Eine Möglichkeit der Betrachtung ist es somit, BDD als Top-down-Ansatz (engl. von oben nach unten) zu sehen und KDT als Bottom-up-Ansatz (engl. von unten nach oben).

Sowohl bei BDD als auch bei KDT werden Keywords auf unterschiedlichen Abstraktionsstufen verwendet, d. h. High-Level Keywords und Low-Level Keywords, siehe auch [8]. Die sinnvolle Wahl der jeweiligen Abstraktionsebene ist Aufgabe des Anwenders. Als klassisches Beispiel sei an dieser Stelle auf die Anmeldung an unserem ATM-Simulator verwiesen. Das Login könnte sowohl mittels dreier separater Low-Level Keywords für Eingabe des Nutzernamens, Eingabe des Passworts sowie Druck auf den Bestätigungs-Button als auch mittels eines High-Level Keywords für das komplette Login mit den beiden Parametern Nutzernamen und Passwort realisiert werden.

Bei der noch verbleibenden Paarbeziehung zwischen BDD und MBT haben wir folgende Zusammenhänge identifiziert: Ein MBT-Modell kann mithilfe von BDD-Szenarien verbalisiert werden. Ein Modell kann aber auch mithilfe von Szenarien verifiziert werden. Zwischen MBT-Modell und BDD-Szenarien besteht eine 1-n-Beziehung. Es stellt sich die Frage, inwieweit ein MBT-Modell aus BDD-Szenarien synthetisiert werden kann.

### 3 Fragestellungen

Allein aus dem Vergleich der Methoden resultiert eine Vielzahl interessanter Fragestellungen, welche wir zum Teil schon in [1] adressiert haben: Was sind die Vor- und Nachteile der jeweiligen Methode? Wie hoch sind die jeweiligen Investitionskosten? Was ist der Return on Investment (ROI)? Wie erfüllt die jeweilige Methode in den unterschiedlichen Teststufen die Kernkriterien Wartbarkeit, Wiederverwendbarkeit und Testabdeckung?

Weitere und aus unserer Sicht wesentlich interessantere Fragestellungen ergeben sich, wenn wir die Kombinationsmöglichkeiten der Techniken betrachten. Die Diskussion im AK hat gezeigt, dass einige dieser Fragen durchaus noch offen sind.

Zunächst ergeben sich Fragen zur technischen Seite der jeweiligen Verknüpfung: Wie effektiv können die Methoden verknüpft werden? Inwieweit gibt es für die Verbindung schon Tool-Support? Welche Ansätze gibt es bereits in der Forschung?

Darüber hinaus ergeben sich natürlich insbesondere bei der Kombination von Methoden viele Fragen zur Komplexität, zum erforderlichen Lernaufwand, zur Effektivität und zur Wirtschaftlichkeit, siehe oben. Als Beispiel sei hier nur folgende Frage genannt: Vergrößert die Verbindung der Techniken nicht noch ein-

mal die Einstiegshürde, durch noch größeren technischen Aufwand und noch höhere Anforderungen an die Mitarbeiter?

Zusätzlich ergeben sich auch Fragen zum Einsatz und zur Rollenverteilung: Wie integrieren sich kombinierte Ansätze in eine Test-First-Strategie? Wie ist das Zusammenspiel mit Acceptance-Test Driven Development (ATDD)? Wenn das Zusammenspiel der Techniken so eng gestaltet werden kann, wachsen dann die Rollen Testarchitekt, Fachexperte und Testautomatisierer enger oder gar vollständig zusammen?

Abschließend ergeben sich natürlich Fragen zum Potenzial der Kombinationsmöglichkeiten. Es ist zu klären, ob evtl. Zusammenhänge über mehrere Abstraktionsstufen besser sichtbar und nutzbar gemacht werden können. Inwieweit kann die Kombination der drei Testtechniken BDD, MBT und KDT dazu beitragen, auch in Zukunft die Qualität immer komplexerer Projekte zu sichern?

### 4 Ausblick

Den Fokus unserer zukünftigen Arbeiten möchten wir auf die Verbindungen zwischen den Technologien legen, d. h. auf die Kanten in Abbildung 1. Anhand unseres ATM-Simulators wollen wir die aufgedeckten Beziehungen konkret am Beispiel weiter ausarbeiten und weitere der obigen Fragestellungen bearbeiten.

Zunächst planen wir, an einem kleinen Ausschnitt aus unserem Beispiel die Verknüpfungsmöglichkeiten technisch im Detail ausarbeiten. Wir erwarten, dass wir dabei neben Erkenntnissen zur Realisierbarkeit auch schon erste Antworten auf Fragen zur Wirtschaftlichkeit erhalten. Im Laufe dieser Arbeiten werden sich sicherlich noch viele weitere, hier bisher nicht genannte interessante Fragestellungen ergeben.

Am Thema interessierte Leser sind gerne eingeladen am AK-Treffen im Rahmen der nächsten TAV aktiv teilzunehmen. Kontaktmöglichkeiten finden Sie auf unserer AK-Webseite [7].

### Literatur

- [1] Christian Brandes, Benedikt Eberhardinger, David Faragó, Mario Friske, Baris Güldali, and Andrej Pietscher. Drei Methoden, ein Ziel: Testautomatisierung mit BDD, MBT und KDT im Vergleich. *Softwaretechnik-Trends*, 35(3), 2015.
- [2] Selenium. <https://seleniumhq.org>.
- [3] Wetator. <http://www.wetator.org/>.
- [4] Integrity. <http://www.integrity-tf.org>.
- [5] Cucumber. <https://cucumber.io/>.
- [6] Tedeso. <https://www.imbus.de/testbench/zusatzkomponenten/managed-model-based-testing/>.
- [7] Webseite des AK TOOP/MBT. <http://ak-toop.gi.de/>.
- [8] ISO/IEC/IEEE International Standard 29119-5 Software and systems engineering - Software testing - Part 5: Keyword-driven testing (draft).