

## Protokoll des Treffens am 14. Juni 2013 bei GEBIT Solutions in Berlin

### Teilnehmerliste

Harald Cichos	TU Darmstadt
Hartmut Lackner	HU Berlin
Dehla Sokenou	GEBIT Solutions
Martin Thomas	Fraunhofer FOKUS
Florian Wartenberg	Fraunhofer FOKUS
Stephan Weißleder <sup>*)</sup>	Fraunhofer FOKUS

\*) Protokollführer

### Verteiler

Mitglieder des AK	info@toop.gi-ev.de
-------------------	--------------------

### Ablauf

- TOP 1) Begrüßung durch AK-Sprecher  
Vorstellung der Agenda und Tagesziele
- TOP 2) Vortrag "Akzeptanztest mit Integrity" (Dehla Sokenou, GEBIT Solutions)
- TOP 3) Zusammenfassung der vergangenen Arbeiten des AK
- TOP 4) Vortrag zu aktuellen Forschungsarbeiten zum modellbasierten Testen variantenreicher Systeme  
(Stephan Weißleder, Fraunhofer FOKUS)
- TOP 5) Planung des weiteren Vorgehens

### TOP 2) Vortrag "Akzeptanztest mit Integrity" (Dehla Sokenou, GEBIT Solutions)

Unser Treffen in Berlin wurde mit einem interaktiven Vortrag von Dehla Sokenou zu aktuellen Arbeiten bei der GEBIT Solutions eingeleitet.

Stichpunkte:

- Testausführungsprogramm als Alternative zu FitNesse wurde gesucht
- wichtigste Frage: Wie werden Tests spezifiziert?
  - Kunden verstehen Code oft nicht
  - natürliche Sprache ist hingegen relativ frei interpretierbar

- > Ziel: formalisierte Sprache, die nahe an natürlicher Sprache angelehnt ist
- Integrity ist Akzeptanz-Test-Framework nahe an FitNesse und Behavior-driven Testing
  - Open Source unter EPL (Einsatz möglichst wenig einschränken)
- Dehla demonstrierte das FitNesse-Framework
- FitNesse und Integrity sind an GEBIT-eigenes, modellbasiertes Entwicklungsframework "TREND" angebunden
- was fehlt: Testgenerierung aus Modell heraus
  - Form des Modells: Implementierungsmodell, Use Cases (mit dahinter hängenden Workflows), Activity-Diagramme, State Machines, Sequenz-Diagramme, ...
    - > Testgenerierung müsste im letzten Schritt über alle Modelle laufen können
    - > Einstiegspunkt sind Uses Cases, an denen das Verhalten hängt und Klassendiagramme, an denen die Struktur hängt
  - Testfälle derzeit manuell definiert -> Qualitätsmessung z.B. anhand von Transitionsüberdeckung
  - Entwicklermodelle reichen nicht aus, um Testorakel automatisch zu generieren
- Verweis: <http://www.integrity-tf.org>
  - inkl. Jenkins-Plugin zur Ergebnisauswertung im automatischen Build-Prozess.

### **TOP 3) Zusammenfassung der vergangenen Arbeiten des AK**

Stephan fasste die vergangenen Arbeiten des AK TOOP/MBT kurz zusammen. Eine wesentliche Aufgabe des AK bestand in den vergangenen Monaten in der Findung eines neuen Themas mitsamt Aufgaben und Zielen.

Die Arbeiten zum Thema „Traceability und Testen“ verliefen aus verschiedenen Gründen nicht sehr befriedigend. In diesem Treffen wurden die Themen also erneut kurz diskutiert. 3 Themen wurden andiskutiert:

- Thema 1: Traceability
  - Warum überhaupt mit Traceability beschäftigt?
  - Gibt es da überhaupt ein direktes Problem?
  - Polarion, etc. (auch Entwicklungsprozesse bei GEBIT) hat keine Probleme mit Anforderungsverknüpfung
  - > mögliches Vorgehen: Welche Probleme gibt es? Können diese bereits mit existierenden Tools gelöst werden?
- Thema 2: Wann ist man fertig mit Testgenerierung?
  - Zweckdienlichkeit von Modellüberdeckung?
  - wichtig ist aber Anforderungsüberdeckung
  - > also: wie sind Anforderungen im Modell repräsentiert?
  - > Wie sind Anforderungen beschrieben?
    - eher Thema für Requirements-Engineering-Gruppe?
- Thema 3: Variantenmgmt. + MBT
  - Präsentation von Stephan zu den Arbeiten bei FOKUS -> siehe TOP 4

### **TOP 4) Vortrag zu aktuellen Forschungsarbeiten zum modellbasierten Testen variantenreicher Systeme (Stephan Weißleder, Fraunhofer FOKUS)**

Stephan stellte kurz die aktuellen Arbeiten zum Thema Testdesignautomatisierung für variantenreiche Systeme beim Fraunhofer FOKUS vor. Kernpunkt ist, dass es verschiedene Ansätze für die Testgenerierung gibt und dass sie unterschiedliche Vor- und Nachteile haben. Auf dieser Basis hat sich eine Diskussion um die verschiedenen Vorgehensweisen entsponnen. Unter anderem hat Harald seinen Ansatz für die Testgenerierung für variantenreiche Systeme vorgestellt und Florian hat die Werkzeugunterstützung bei FOKUS kurz demonstriert.

Die wesentlichen Ansätze waren:

- 1) „Product-centered“ approach
- 2) „Product line-centered“ approach (Inject FM into transition)
  - > Suchanfragen können über Feature-Belegung gesteuert werden
  - > Feature-Belegungen teilweise (!) vorgeben / Teile des Feature-Modells ignorieren
  - > in Haralds Szenario: Zusammenlegung von Testfällen
- 3) „Product line-centered“ approach (inject FM into initial transition -> Festlegung der Variante vor Testgenerierung)
- 4) Mittelweg zwischen Produkt-zentriertem und Produktlinien-zentriertem Verfahren
  - 4.a) einzelne Features vorselektieren und produktlinien-zentriertem Ansatz auf dem Rest ausführen
    - > einzelne Variablen werden im injizierten 150%-Modell vorbelegt
  - 4.b) wichtige Produktvarianten testen und ungetestete Features/Modellteile mit 4.a testen
    - > hierbei die dem bereits getesteten Produkt entsprechenden Features und deren Abhängigkeiten untereinander im 150%-Modell vorbelegen
- 5) Haralds Ansatz: auf allen (!) produktspezifischen Testmodellen wird eine vorgegebene Überdeckung erreicht
  - > daraus wird eine Menge an Testfällen erzeugt, die allumfassend ist in dem Sinne, dass für jede Produktvariante Testfälle daraus selektiert werden können, sodass immer 100% Transitionsüberdeckung erreicht wird

Daraus ergaben sich die folgenden Fragen:

- > Welcher Ansatz ist zu bevorzugen? Warum?
- > Wie ist der Aspekt der Wiederverwendung von Testfällen zu berücksichtigen?
  - insb. im Vergleich zur automatischen Testgenerierung
- Vorgehen zur Modellierung: 150%-Modell vs. Deltas
  - Vorteile der Methoden?
- Gibt es Produktfamilien bei der GEBIT?
  - Dehla: Kassensystem ist auch eine Produktvariante
  - > Implementierung in OSGi -> einzelne bundles -> Bindungszeitpunkt ist die Laufzeit
  - > Test: produktzentriert mit manuell selektierten Varianten
    - Variantenselektion ist kundenspezifisch

## TOP 5) Planung des weiteren Vorgehens

Auf dem Treffen wurde das Thema „Testgenerierung für variantenreiche Systeme“ intensive behandelt, sodass der AK sich auch diesem Thema verschreibt. Das im vorherigen Treffen definierte Thema „Traceability“ bleibt weiterhin auf der Tagesordnung und wird auch zukünftig berücksichtigt werden.

Für die Arbeiten in Richtung „Testgenerierung für variantenreiche Systeme“ wurden die folgenden Unterpunkte im AK beschlossen:

- Sammlung von Testgenerierungsansätzen für Produktlinien
- Publikation über die verschiedenen Ansätze anstreben
  - > Beispiele definieren und für Testgenerierung nutzen
  - > allgemeine Diskussion dazu
  - > Implementierung + Mutationsanalyse ??? -> dieser Punkt eher für wissenschaftliche Publikation!

-> Spielbeispiele vs. richtige Beispiele: Body Comfort System

-> Fraunhofer FOKUS baut an Integrationsumgebung „SPLTestbench“ für existierende Modellierungs- und Testgenerierungswerkzeuge

-> TU Darmstadt baut gerade an Toolsuite „Imotep“ für Testgenerierung für variantenreiche Systeme

- Delta-Ansatz vs. 150%-Modell

- Sind die vergleichbar? ... ineinander überführbar?

- Woher kommen die Deltas? oder das 150%-Modell? -> Prozess