

Arbeitskreis "Testen objektorientierter Programme"
der GI-Fachgruppe TAV

Protokoll des Treffs am 5. Mai 2006 im Physikzentrum Bad Honnef

Teilnehmerliste

Jens Calame	FernUni Hagen	Timea Illes	Uni Heidelberg
Jens Doll	Uni Heidelberg	Dr. Stefan Jungmayr	Teradyne DS
Dr. Matthias Hamburg	Sogeti	Prof. Rainer Koschke	Uni Bremen

*) Protokollverfasser

Ablauf

- Vortrag von Rainer Koschke
- Diskussion des Entwurfs zum „Certified Tester – Expert Level Module: Object-Oriented Software Tester“ von Mario Winter.
- Diskussion des nächsten Schwerpunkt-Themas

Reengineering und Testen

u.A.:

- Gliederung nach

Nächstes Schwerpunktthema

Vorschläge für Themen waren:

- Bewertung der Testmethoden bzgl. ihrer Eignung, bestimmte Fehlerkategorien zu entdecken (experimentell)
- Welche Tests können aus dem Code generiert werden, welche aus den Anforderungen?
- Bewertung von anforderungsbasierten Testmethoden
- domänenspezifische Sprachen und Testen
- wie können sprachliche Anforderungen formalisiert werden
- Vorbereitung der Kursinhalte für Mario Winter's Proposal des Certified Testers, Advanced Level)
- Testen von SOA-Architekturen
- Literatursammlung (offline)
- Fehlerkategorisierung mit dem AK „Test von eingebetteten Systemen“ abstimmen

Vorschlag:

1. Bewertung der Fehlerkategorien, ob sie durch statische Analyse abgedeckt werden können. Bestimmung per Email.
2. Experimentelle Untersuchung im Anschluss.
3. Literatursammlung (parallel dazu):
4. Ableitung von Tests aus Anforderungen (insbesondere natürlichsprachlichen Anforderungen) als Thema eines Treffens
 - Klären, wo diese stattfinden soll: TOOP-Seite, TAV-Seite
 - Format? BibTeX
 - Suchmöglichkeit
 - Editierbarkeit
 - Vollständigkeit oder „Lieblingsbücher“ mit Rezensionen

Abdeckung der Fehlerkategorien durch Reengineering-Verfahren

Fehlerkategorie	durch Reengineering-Verfahren abdeckbar	Kommentar
Anforderungsfehler	-	
Konfigurationsfehler	-	
Fehler im Datenfluss/Objektzugriff	++	Datenflußanalyse
Schnittstellenfehler	+	Objekt-Prozess-Graph
Kontrollflussfehler (falsch in Abh. von Eingabewerten)	?	
Berechnungsfehler („immer falsch“)	-	
Fehler in Klassen oder Datentypen	-	

Allgemeine Konklusio:

- Auf Unit-Testebene und mit Code als Basis sind formale bzw. automatische Ansätze teilweise möglich.
- Auf Systemtestebene mit natürlichsprachlichen Anforderungen sind automatische Ansätze nicht anwendbar.

Frage:

- Wie unterschiedlich sind Testfälle im Reengineering und im Testen?

Infos

Artikel zu empirisc

Nächstes Treffen des Arbeitskreises

- Zwischentreffen im Herbst 2007 wird über Newsgroup angekündigt
- **TAV 25**, Februar 2007

Weitere Infos

web: <http://toop.gi-ev.de/>

email: info@toop.gi-ev.de

news: <http://lists.gi-ev.de/mailman/listinfo/toop>