OO-Testwerkzeuge: Wunsch und Wirklichkeit

Stefan Jungmayr, Mario Winter
FernUniversität Hagen, Fachbereich Informatik, Prakt. Informatik III
{stefan.jungmayr, mario.winter}@fernuni-hagen.de

TAV 13, 4.-5.2.1999, Siemens AG, München

1 Einleitung

Der fortschreitende Einsatz der Objektorientierung in der Software-Entwicklung erfordert adäquate Werkzeugunterstützung beim Testen. Obwohl seit 1992 eine Vielzahl von auf die Objektorientierung zugeschnittenen Verfahren veröffentlicht wurde [1], mangelt es leider derzeit an

1. Testwerkzeugen, die alle wesentlichen Aspekte objekt-orientierter Systeme beherrschen und
2. einer Marktübersicht, die Werkzeuge nach den von ihnen unterstützten objektorientierten Aspekten des Tests gliedert.

Der Arbeitskreis "Testen objektorientierter Programme" der GI-Fachgruppe 2.1.7 hat sich daher zum Ziel gesetzt, Anwendern Unterstützung bei der Auswahl bzw. Vorselektion von geeigneten existierenden OO-Testwerkzeugen anzubieten und gleichzeitig Hersteller zu einer stärkeren Unterstützung des objektorientierten Paradigmas durch ihre Testwerkzeuge anzuregen.

Basierend auf den für den Test relevanten Eigenschaften objektorientierter Systeme haben wir zunächst Anforderungen an Testwerkzeuge identifiziert. Diese Anforderungen wurden dann anhand der Test-Arten (Methodentest, Klassentest, Integrationstest, Systemtest) kategorisiert und in Form eines (html-) Fragebogens gefaßt. Der Fragebogen wurde von uns an Werkzeughersteller gesendet, die mit Ihren Antworten darstellen können, welche Anforderungen ihre Produkte erfüllen.

2 Eigenschaften von objektorientierten Systemen mit Einfluß auf den Test

Wesentliche Eigenschaften von objektorientierten Systemen wie Kapselung, Vererbung, Polymorphismus, Botschaftsaustausch, und Zustandsverhalten beeinflussen den Test von Methoden, Basisklassen und Vererbungshierarchien sowie den Integrationstest und den Systemtest teilweise erheblich [2]. Beispielhaft skizzieren wir die Testproblematik für den Test von Vererbungshierarchien und den Integrationstest.

Die Wiederverwendbarkeit und Testbarkeit objektorientierter Software wird durch die Vererbung zum Teil vermindert, da geerbte und getestete Methoden im Kontext der Unterklasse zu neuen Fehlern führen können. Interaktionen der geerbten Methode mit in der Unterklasse re- oder neu definierten Methoden können neue Testfälle für diese geerbte Methode erforderlich machen.

Zudem treffen wir in objekt-orientierten Systemen meist keine hierarchische Struktur, sondern ein zeitvariantes Netzwerk von teils gleichberechtigten Objekten an, das durchaus Zyklen enthalten kann [3]. Für den Integrationstest bedeutet dies neue Herausforderungen: von der Definition einer geeigneten Intergrationsreihenfolge zur Einsparung von Implementierungsaufwand für Testtreiber über die Definition geeigneter Testfälle bis hin zur Berücksichtigung von dynamisch wechselnden Objektkonstellationen und Polymorphismus. Hierbei spielen detaillierte Entwurfsdokumente wie Kollaborationsdiagramme und Methodensequenzdiagramme eine wesentliche Rolle.

3 Anforderungen an OO-Testwerkzeuge

Mit Hinblick auf die objektorientierten Eigenschaften haben wir einen Katalog von Anforderungen in Form einer Checkliste bzw. eines Fragebogens zusammengestellt. Beispiele für darin enthaltene Fragen sind:

D1.1: Bietet das Werkzeug Information an, zu welchem Grad eine abgeleitete Klasse bereits im Rahmen des Basisklassentests getestet worden ist?

E1.1: Ist es möglich, Information aus dem Entwurf zur Generierung von Integrations-Testdaten einzusetzen?

E1.3: Wird das korrekte Löschen von Teilobjekten in einer Aggregationsbeziehung untersucht?

Der vollständige Katalog bzw. der Fragebogen ist im Internet unter folgender Adresse zugänglich: http://www.fernuni-hagen.de/inf/pi3/GI/CAST/

Hersteller	**Testwerkzeug**	**Art der Unterstützung**
IPL	Cantata	Testdurchführung*)
McCabe	Visual Toolset	Metriken
SCOPE	TESTSCOPE	Testdurchführung*)
SilverMark	Test Mentor	Testdurchführung*)
Testwell	CTA++	Testdurchführung*)
	CTC++	Testüberdeckung
	CMT++	Metriken

Tabelle 1: Testwerkzeuge

*) Unterstützung nicht auf Testdurchführung beschränkt

4 Auswertung der Antworten

Bis heute haben die in Tabelle 1 aufgelisteten fünf Hersteller den Fragebogen beantwortet und so die Erfüllung objektorientierter Anforderungen durch ihre Testwerkzeuge selbst bewertet. Aufgrund von Zeitschriftenartikeln und kommerziellen Marktübersichten schätzen wir, das zur Zeit etwa 15 bis 20 Hersteller Testwerkzeuge für den Test objektorientierter Programme anbieten, so daß die bisherigen Rückläufer — vorsichtig geschätzt — knapp 30% der verfügbaren Werkzeuge darstellen.

Zur Auswertung der Antworten gliedern wir die Anforderungen nach der Unterstützung der unterschiedlichen Testphasen sowie nach den objektorientierten Eigenschaften. Pro Testphase bilden wir zunächst den Quotient aus der Anzahl erfüllter Anforderungen und ihrer Gesamtanzahl. Eine zusammenfassende Darstellung für alle Testwerkzeuge zeigt Abb. 1:.

Obwohl die hier vorgestellte Auswertung vorläufig und sicherlich abhängig von der Anzahl der Anforderungen pro Testphase und ihrer Art ist, drängt sich folgende Bewertung auf: Die Werkzeugunterstützung für das Testen objektorientierter Programme nimmt vom Methodentest hin zum Integrationstest tendenziell stetig ab. Dies läßt sich dadurch interpretieren, daß der Methodentest noch die größte Ähnlichkeit zum Test in konventionellen Systemen besitzt (wie auch der Systemtest, der gleichfalls besser unterstützt ist).

Abb. 2 zeigt, ob das Testwerkzeug zumindest eine der mit einer bestimmten objektorientierten Eigenschaft verknüpften Anforderungen erfüllt. Diese zweite Auswertung ist nun objektiver, da keine Abhängigkeit zur Anzahl der Anforderungen bzw. Fragen besteht und auch die Art der Frage keinen wesentlichen Einfluß ausübt. Auffällig sind die schwache Unterstützung von Alias-Betrachtungen, Kollaborationen, Methodenredefinition/Subtypbildung und Zustandsverhalten.

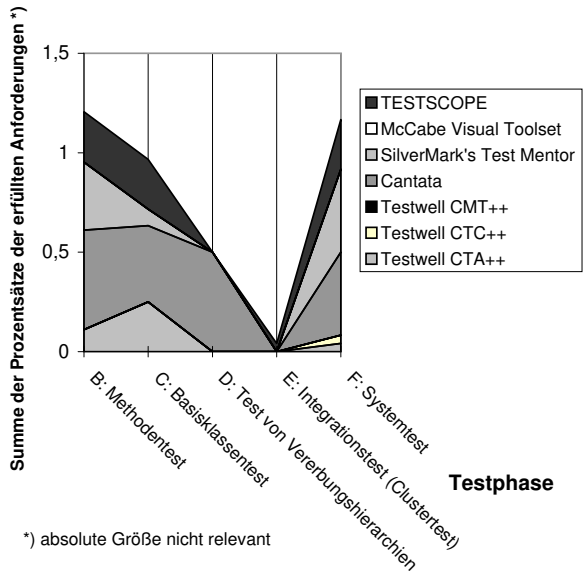

Abb. 1: Unterstützung der Testphasen

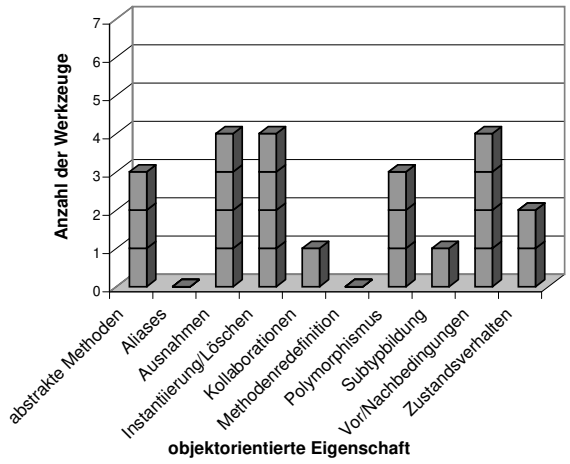

Abb. 2 : Unterstützung objektorientierter Eigenschaften

5 Zusammenfassung

Kommerzielle Testwerkzeuge unterstützen leider die bisher im Bereich des objektorientierten Testens veröffentlichten Methoden und Techniken nur wenig. Die von Mitwirkenden des GI-TAV Arbeitskreises „Testen von objektorientierter Programme“ (Stefan Jungmayr, Franz Maurer, Harry Sneed, Andreas Spillner, Stephanie Ulrich, Mario Winter) zusammengestellten Anforderungen geben Anwendern nun Hilfestellung zum Vergleich und zur Bewertung existierender OO-Testwerkzeuge (wobei wir hier keine Konkurrenz zu kommerziellen Werkzeug-Evaluierungen, sondern eine Ergänzung derselben beabsichtigen). Gleichzeitig hoffen wir, mit den im Fragebogen angerissenen Themen Testwerkzeugherstellern eine — in absehbarer Zukunft — stärke Unterstützung der OO-Spezifika ans Herz legen zu können.

Die vollständige Auswertung der Umfrageergebnisse ist unter der gleichen Internetadresse wie der Fragebogen zugänglich. Eine Fortführung der Untersuchung im Jahr 1999 ist geplant.

6 Literatur

[1] Jörn Münzel, Mario Winter. *GI-TAV AK Test Objektorientierter Programme — Annotierte Literaturliste.* 1998. http://www.fernuni-hagen.de/inf/pi3/GI/akOOT.html.

[2] GI-Fachgruppe 2.1.7, Arbeitskreis "Testen objektorientierter Programme". *Testbarkeit objekt-orientierter Programme*. http://www.fernuni-hagen.de/inf/pi3/GI/OOTestbarkeit.doc2.html.

[3] Mario Winter. *Managing Object-Oriented Integration and Regression Testing*. Proceedings of the 6th euroSTAR 98, 30. Nov - 4. Dez., München, 1998, S. 189-200.