

Electronic version of an article published as **Bleek, W.-G.; Raasch, J.; Züllighoven, H. (Hrsg): Software Engineering 2007. Fachtagung des GI-Fachbereichs Softwaretechnik. 27. - 30.03.2007 in Hamburg. LNI P-105, pp. 247-248, GI**

© [2007] Gesellschaft für Informatik e.V.

Die Originalpublikation ist unter folgendem Link verfügbar:

<http://www.gi-ev.de/service/publikationen/lni/gi-edition-lecture-notes-in-informatics-lni-p-105/>

# Entscheidungen im Testprozess

Lars Borner, Timea Illes, Barbara Paech

Lehrstuhl für Software Systeme  
Institut für Informatik  
Im Neuenheimer Feld 326  
69120 Heidelberg  
{borner, illes, paech}@informatik.uni-heidelberg.de

**Abstract:** Softwareprozessbeschreibungen konzentrieren sich auf Artefakte, Aktivitäten und Rollen, lassen aber die im Prozess zu treffenden Entscheidungen meist implizit. Dies beeinträchtigt die Qualität dieser Entscheidungen erheblich. In dieser Arbeit stellen wir eine Entscheidungshierarchie für den Testprozess vor. Sie umfasst die Entscheidungen des Testprozesses und spiegelt Abhängigkeiten zwischen diesen wieder.

## 1 Einleitung

Um festzustellen, in welchem Maße ein entwickeltes System die gestellten Anforderungen erfüllt, sind Testaktivitäten unverzichtbar. Da vollständiges Testen nicht möglich ist [Me79], müssen Entscheidungen darüber getroffen werden welche Teile des Systems wie zu testen sind. In den Testaktivitätsbeschreibungen, z.B. in [SLS06], werden Testentscheidungen allerdings nicht explizit fokussiert. Das Bewusstsein über die zu treffenden Entscheidungen kann die Qualität der Testprozessdurchführung erheblich erhöhen, da getroffene Entscheidungen die nachfolgenden Aktivitäten beeinflussen. Dieses Treffen von Entscheidungen setzt weiterhin (Test-)Aktivitäten und deren Artefakte voraus. In dieser Arbeit werden Entscheidungen identifiziert, die im Rahmen des Testprozesses getroffen werden. Dabei konzentrieren wir uns auf die Entscheidungen, die den *dynamischen* Test betreffen. Jede dieser Entscheidungen kann einer Entscheidungsebene zugeordnet werden, wodurch eine Entscheidungshierarchie entsteht.

## 2 Entscheidungshierarchie

Die Entscheidungshierarchie basiert auf Prozessbeschreibungen in [SLS06], [BSC00], [IEEE98], [MP02] und lässt sich in drei große Bereiche unterteilen. Auf den oberen Ebenen (Spezifikationsebene, Testzielebene) entscheiden Testdesigner und Testmanager, welche Teile des Softwaresystems getestet werden (*was*). Auf den drei nachfolgenden Ebenen wird vom Testdesigner, Tester und Testmanager festgelegt, *wie* diese Teile zu testen sind. Auf den abschließenden zwei Ebenen (Testlaufebene, Testauswertungsebene) wird entschieden, *welche Qualität* die zu prüfende Software hat. Abbildung 1 fasst die Entscheidungen im allgemeinen Testprozess zusammen. Die

Entscheidungen der ersten vier Ebenen sollten dabei parallel zu den „konstruktiven“ Phasen der Softwareentwicklung getroffen werden.

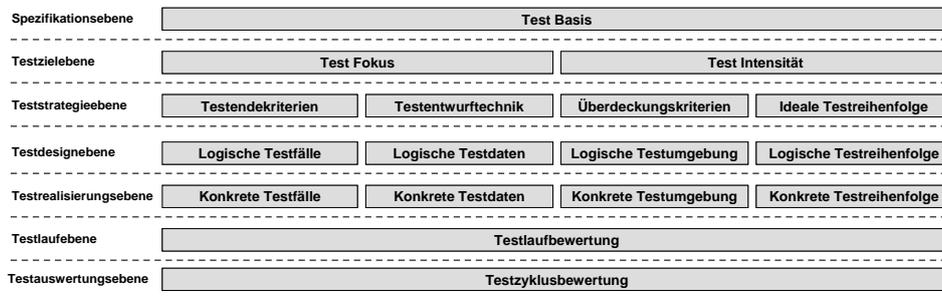


Abbildung 1: Entscheidungshierarchie des allgemeinen Testprozesses

### 3 Anwendungsmöglichkeiten

Unsere Entscheidungshierarchie kann in vielen Bereichen zum Thema Testen sowohl in der Forschung als auch in der Praxis angewendet werden. Sie diene uns als *Framework*, um die spezifischen Entscheidungen des System- und Integrationstestprozesses herauszuarbeiten und gegenüberzustellen sowie zur Kategorisierung von Testansätzen und -werkzeugen. Wir untersuchten, welche Entscheidungen durch den jeweiligen Testansatz bzw. durch das entsprechende Werkzeug unterstützt werden.

Die Entscheidungshierarchie kann weiterhin als Checkliste für Prozessanalyse und -bewertung sowie zur Erweiterung bestehender Testprozessmodelle, wie das TPI [KP99], um den Entscheidungsaspekt verwendet werden. Nicht zuletzt dient sie als Grundlage für Forschung im Bereich des Rationale Management für den Testprozess.

### Literaturverzeichnis

- [BSC00] Burnstein, I.; Suwannasart, T.; Carlson, C. R.: Developing a Testing Maturity Model for Software Test Process Evaluation and Improvement. In Proceedings of the IEEE International Test Conference on Test and Design Validity, 1996.
- [IEEE98] IEEE Std. 829-1998: Software Engineering Technical Committee of the IEEE Computer Society. IEEE standard for software test documentation, USA, 1998.
- [KP99] Koomen, T.; Pol, M.: Test Process Improvement: A step-by-step guide to structured testing. Addison-Wesley, 1999.
- [Me79] Meyers, G.J.: The Art of Software Testing, John Wiley & Sons, New York, 1979.
- [MP02] Mosley, D. J.; Posey, B. A.: Just Enough Software Test Automation, Prentice Hall, Juli, 2002.
- [SLS06] Spillner, A.; Linz, T.; Schaefer, H.: Software Testing Foundations - A Study Guide for the Certified Tester Exam - Foundation Level - ISTQB compliant. dpunkt.verlag, 2006.