

Effizienzsteigerung im Systemtest von Steuergeräten

Eine empirische Untersuchung

Matthias Wiemann, Jörg Gericke

Siemens AG
Corporate Research & Technologies
CT PP 6
Otto-Hahn-Ring 6
81730 München
joerg.gericke@siemens.com

Abstract: Die Softwarequalität in Body-Controllern kann im Systemtest durch verhaltensorientierte Testmethodiken erhöht werden. In der Literatur wurden hierzu bereits zahlreiche Verfahren vorgeschlagen. Die vorliegende Arbeit stellt Ergebnisse eines empirischen Vergleichs einer herkömmlichen, intuitiven mit einer verhaltensorientierten Testmethodik vor. Die Analyse der Anzahl identifizierter Fehler bestätigt, dass durch die verhaltensorientierte Testmethodik eine deutlich höhere Effizienz und Effektivität bzgl. der Fehleridentifikation erzielt werden kann. Bereits das Aufstellen der erforderlichen Verhaltensmodelle ermöglicht die Fehleridentifikation in frühen Phasen. Der Aufwand zur Modellerstellung des verhaltensorientierten Tests wird von den Testmanagern als geringer eingestuft als der Aufwand für den intuitiven Test.

1 Motivation

In der Automobilindustrie werden heute 80-90% der Innovationen durch Software realisiert [DV05]. Dies beinhaltet ein exponentielles Anwachsen des Softwarevolumens in Automobilen [DK04] mit erheblicher Komplexitätssteigerung der entwickelten Systeme [Be06] und dadurch induzierte Qualitätseinbußen [Be06, Dp03, Eq07, OS04]. Fahrzeugherstellerübergreifend wuchs die Zahl der Fahrzeugpannen aufgrund von mangelbehafteter Elektronik stark an [Dp03, Eq07, OS04]. Funktions- und verhaltensorientierte Testverfahren sollen dieser Entwicklung entgegenwirken. Diese scheinen nur zögerlich Eingang in die Praxis zu finden. Ein möglicher Grund mag darin liegen, dass eine Überlegenheit hinsichtlich der Effektivität in der Fehleridentifikation bei vertretbarem Aufwand in der Praxis bezweifelt wird. In dieser Arbeit wird daher diese Effektivität durch eine funktions- und verhaltensorientierte Teststrategie im Kontext der Automobilindustrie empirisch untersucht und einer herkömmlichen, intuitiven Testmethodik vergleichend gegenübergestellt.

2 Konzept der durchgeführten Untersuchung

2.1 Iterative Systementwicklung

Untersucht wurden die Systementwicklungen von zwölf *Steuergeräten* (engl. *Body-Controller*). Die Steuergeräte wurden nach einer iterativen Anwendung des *V-Modells* entwickelt. In jeder Iteration wurde der Funktionsumfang erweitert und getestet. Betrachtet werden vier V-Modell-Iterationen mit Anwendung einer herkömmlichen, intuitiven Testmethodik. Vier Steuergeräte wurde zur Qualitätssteigerung in der dritten Iteration mit einer speziell für den Systemtest von Steuergeräten entwickelten, funktions- und verhaltensorientierten Testmethodik validiert.

2.2 Funktions- und verhaltensorientierte Testmethodik

Ausgangsbasis war eine intuitive Testfallgenerierung. Diese basierte auf der Erfahrung der Tester sowie auf der Vorgabe, dass für jede Anforderung mindestens ein Testfall auszuführen ist. Im Vergleich zur intuitiven Testmethodik soll eine funktions- und verhaltensorientierte Testmethodik die Qualität der Steuergeräte durch Verringerung der Restfehleranzahl erhöhen. Diese Methodik fokussiert das *Systemverhalten*, welches durch hierarchische *Verhaltensmodelle* beschrieben wird. Dieses Systemverhalten war in textuellen Spezifikationen beschrieben und wurde in Verhaltensmodelle, z.B. Zustandsautomaten, überführt.

Über den verhaltensorientierten Systemtest hinausgehend wurde die Steuergerätesoftware zusätzlich mit dem funktionsorientierten Testverfahren der *Äquivalenzklassenanalyse* [My79] geprüft. Hierbei wurden die Wertebereiche der Eingabevariablen unter Zuhilfenahme der *Klassifikationsbaummethode* [GG93] in Äquivalenzklassen unterteilt. Die Grenzen der Äquivalenzklassen wurden durch eine *Grenzwertanalyse* [My79] validiert. In der Praxis erwies sich die resultierende Testfallanzahl jedoch als zu hoch. Daher wurde die Testfallanzahl mit der bewährten *paarweisen Überdeckung von Parameterwerten* [Wi00] reduziert.

2.3 Manuelle Verhaltensmodellierung und automatisierte Testfallgenerierung

Steuergeräte eins bis vier wurden in der dritten Iteration mit der beschriebenen verhaltensorientierten Testmethodik validiert. Die zur Testfallgenerierung notwendigen Verhaltensmodelle wurden von den Testern manuell aufgestellt. Die Tester wurden bezüglich der Modellerstellung und Testfallgenerierung vor Durchführung der empirischen Untersuchung durch eine zweitägige Schulung befähigt, unter Coaching die notwendigen Verhaltensmodelle aufzustellen. Die Verhaltensmodellierung sowie die automatisierte Testfallgenerierung wurden durch die bei der Siemens AG, CT PP entwickelte Werkzeuglandschaft *GATE – Generic and Automated Test Environment* – unterstützt. Dieses generiert u.a. auf den Verhaltensmodellen basierend entsprechend des *Wp-Pfadüberdeckungskriteriums* [FBK91] automatisiert Testfälle.

2.4 Bewertungskriterien zur Effektivität der Testmethodiken

Die Effektivität einer Testmethodik soll anhand der Anzahl gefundener Fehler pro Iteration bewertet werden. Zum geeigneten Vergleich zwölf untersuchter Steuergeräteprojekte werden anstelle absoluter Fehleranzahlen relative Anzahlen der insgesamt gefundenen Fehler angegeben. Die Gesamtanzahl der in den betrachteten vier Iterationen identifizierten Fehler wird pro Steuergerätegruppe (s. Kapitel 3.1) als 100% gesetzt.

3. Ergebnisse

3.1 Effektivere Fehleridentifikation

Die zwölf untersuchten Steuergeräte sind in zwei Gruppen unterteilt: Steuergeräte 1-4 wurden in Iteration drei mit der verhaltensorientierten Testmethodik validiert. Steuergeräte 5-12 wurden ausschließlich unter Anwendung der herkömmlichen, intuitiven Testmethodik getestet. Abbildung 1 zeigt die durchschnittliche relative Anzahl gefundener Fehler für die untersuchten Steuergerätegruppen nach Iterationen aufgeschlüsselt.

In den ersten beiden Iterationen wurden alle zwölf Steuergeräte entsprechend der herkömmlichen Teststrategie validiert. Da die Werte sich nur in Nachkommastellen unterscheiden und Projekteigenschaften wie Komplexität, Systemgröße und Erfahrung der Entwickler und Tester vergleichbar sind, werden die Steuergeräte auch hinsichtlich anderer testrelevanter Eigenschaften wie Testbarkeit als vergleichbar angesehen.

In der dritten Iteration wurde der Funktionsumfang erheblich erweitert, was sich in allen zwölf Steuergeräten durch eine deutlich erhöhte Anzahl identifizierter Fehler widerspiegelt. Zur Verbesserung der Qualität wurden Steuergeräte 1-4 mit der vorgestellten funktions- und verhaltensorientierten Testmethodik getestet. Referenzsteuergeräte 5-12 wurden weiterhin auf herkömmliche Weise (s. Kapitel 2.2) getestet. Mit Ausnahme der Testmethodik blieben restliche Aspekte wie Programmier- und Testteams sowie Automatisierungen konstant. Steuergeräte 1-4 weisen aufgrund einer etwa 1,5-mal (s. Abbildung 1) so hohen relativen Anzahl identifizierter Fehler eine deutliche Steigerung der Effizienz und Effektivität im Vergleich zu den Steuergeräten 5-12 auf.

In der vierten Iteration wurden nur wenige Funktionen hinzugefügt. Sämtliche zwölf Steuergeräte wurden wieder entsprechend der herkömmlichen, intuitiven Testmethodik getestet. Die relative Anzahl identifizierter Fehler sank für die vorher mit der verhaltensorientierten Testmethodik getesteten Steuergeräte 1-4 deutlicher, als für die restlichen Steuergeräte 5-12. Diese Beobachtung ist auf zwei miteinander in Beziehung stehende Ursachen zurückzuführen: a) die herkömmliche Testmethodik identifiziert weniger Fehler als die funktions- und verhaltensorientierte Testmethodik und b) in den Steuergeräten 5-12 wurden Fehler erst jetzt identifiziert, die gemäß der Annahme der vergleichbaren Fehlerstruktur aller Steuergeräte bereits in der dritten Iteration hätten identifiziert werden können bzw. sollen. Die verhaltensorientierte Testmethodik führte folglich zu der gewünschten guten und früheren Fehlererkennung. Dieses führt entsprechend dem Kos-

tenmodell von Boehm [Bo81] zu einer 11%-igen Gesamtprojektkosteneinsparung.

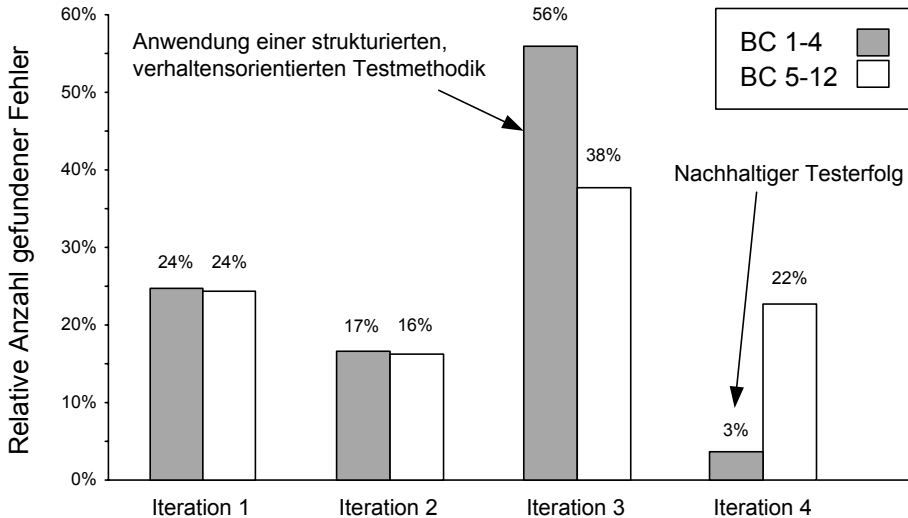


Abbildung 1. Relative Anzahl identifizierter Fehler nach Iteration

3.2 Effizientere Testfallerstellung

Die Erstellung der Verhaltensmodelle für den Systemtest wird in der Praxis von Testmanagern häufig als zu aufwendig kritisiert. Die Erstellung der verhaltensorientierten Testfälle erwies sich jedoch in allen Fällen innerhalb des ohnehin knapp kalkulierten Zeitrahmens als möglich. Die Testmanager beurteilten den Zeitaufwand der Verhaltensmodellierung mit einfachen Werkzeugen wie folgt als wirtschaftlich positiv:

- Unmittelbar vor der Testfallerstellung in der Testmethodik geschulte Tester benötigten vier Tage zur Testfallerstellung, wofür ursprünglich erfahrungsgemäß acht Tage mit bisheriger Testmethodik eingeplant waren. Die Zeitersparnis betrug in diesem Fall etwa 50%.
- Für die Testfallerstellung eines weiteren untersuchten Steuergerätes benötigte ein hinzugezogener, erfahrener Tester zwei der auf Basis der Erfahrung eingeplanten zehn Tage. Die Zeitersparnis betrug hier etwa 80%.
- Die Verhaltensmodelle eines der Steuergeräte konnten für ein parallel entwickeltes Fahrzeug wiederverwendet werden. Die Wiederverwendung war möglich, da sich das Verhalten beider Systeme nur marginal unterscheidet. Nach lediglich geringen Anpassungen lag das benötigte Verhaltensmodell vor. Somit entfiel fast der komplette Aufwand zur Erstellung eines Verhaltensmodells und Testgrunddaten.

Das Aufstellen der Verhaltensmodelle identifizierte grundsätzliche Spezifikationsfehler, die in vorangegangenen Iterationen nicht identifiziert wurden. Die Ursache liegt in der

expliziten Betrachtung von Abhängigkeiten zwischen Anforderungen. Folglich ist allein das Aufstellen – ohne Ausführung – von verhaltensorientierten Testfällen bereits ein Mittel zur Identifikation von Spezifikationsfehlern, welches in frühen Projektphasen parallel zur Implementierung angewendet werden kann.

3.3 Validität der Ergebnisse

Die untersuchten Steuergeräte hätten zur Maximierung der Aussagekraft der Untersuchung parallel einmal mit der intuitiven und einmal mit der funktions- und verhaltensorientierten Testmethodik validiert werden müssen. Eine solche Versuchsdurchführung war im Hinblick auf Ressourcen und Kosten leider nicht möglich. Die Vergleichbarkeit der zufällig gebildeten Steuergerätegruppen ist nicht bewiesen, dennoch sprechen zahlreiche Indizien (s. Kapitel 3.1) für diese Annahme.

4. Schlussfolgerung

In dieser empirischen Gegenüberstellung einer intuitiven und einer funktions- und verhaltensorientierten Testmethodik wird ersichtlich, dass eine Anwendung einer funktions- und verhaltensorientierten Testmethodik Fehler deutlich effektiver und früher identifiziert. Ein erhöhter Aufwand zur Modellerstellung wurde nicht beobachtet. Vielmehr wurde von Kosteneinsparungen im ersten Einsatz berichtet. Daher sollte der Praxiseinsatz einer verhaltensorientierten Testmethodik verstärkt durchgesetzt werden.

Literaturverzeichnis

- [Be06] K. Bender. Embedded Systems-qualitätsorientierte Entwicklung. Springer. p. 1-10. 2006
- [Bo81] W. Boehm, Software Engineering Economics. Prentice Hall PTR, USA. 1981
- [DK04] J. Dannenberg, C. Kleinhans. The Coming Age of Collaboration in the Automotive Industry. Mercer Management Journal, 17:88–94. 2004
- [Dp03] dpa Meldung vom 30.09.2003 zur 33. Jahrestagung der GI, zitiert nach <http://www.heute.t-online.de/ZDFheute/artikel/21/0,1367,COMP-0-2070133,00.html>
- [DV05] W. Damm, A. Votintseva, A. Metzner, B. Josko, T. Peikenkamp. Boosting Re-use of Embedded Automotive Applications through Rich Components. Elsevier Science. 2005
- [Eq07] Methoden zur Unterstützung der entwicklungsbegleitenden Qualitätssicherung von eingebetteter Software (EQUAL). <http://www.embedded-quality.de/>
- [FBK91] S. Fujiwara, G. Bochmann, F. Khendek, M. Amalou. Test selection based on finite state models. IEEE Transactions on Software Engineering, Vol. 17, No. 6, pp. 591-603. 1991
- [GG93] M. Grochtmann, K. Grimm. Classification Trees for Partition Testing. Software Testing, Verification & Reliability, Vol. 3, No. 2, pp. 63-82. 1993
- [My79] G. Myers. The Art of Software Testing. J. Wilsey & Sons, pp. 147-174. 1979
- [OS04] R. Otterbach, F. Schütte. Effiziente Funktions- und Software-Entwicklung für mechatronische Systeme im Automobil. Dspace. 2004
- [Wi00] A. Williams. Determination of test configurations for pair-wise interaction coverage. Proceedings of TestCom 2000, Ottawa, Canada, pp. 59-74. 2000