

Leif Geiger: Fehlersuche im Modell — Modellbasiertes Testen und Debuggen

Promotion: Universität Kassel, Fachbereich Elektrotechnik/Informatik

Erstgutachter: Prof. Dr. Albert Zündorf, Universität Kassel

Zweitgutachter: Prof. Dr. Wilhelm Schäfer, Universität Paderborn

Datum der Prüfung: 11. April 2011

Veröffentlichung: <https://kobra.bibliothek.uni-kassel.de/handle/urn:nbn:de:hebis:34-2011071338286>

Kurzfassung:

Kern der vorliegenden Arbeit ist die Erforschung von Methoden, Techniken und Werkzeugen zur Fehlersuche in modellbasierten Softwareentwicklungsprozessen. Hierzu wird zuerst ein von mir mitentwickelter, neuartiger und modellbasierter Softwareentwicklungsprozess, der sogenannte Fujaba Process, vorgestellt. Dieser Prozess wird von Usecase Szenarien getrieben, die durch spezielle Kollaborationsdiagramme formalisiert werden. Auch die weiteren Artefakte des Prozess bishin zur fertigen Applikation werden durch UML Diagrammarten modelliert. Es ist keine Programmierung im Quelltext nötig. Werkzeugunterstützung für den vorgestellte Prozess wird von dem Fujaba CASE Tool bereitgestellt. Große Teile der Werkzeugunterstützung für den Fujaba Process, darunter die Toolunterstützung für das Testen und Debuggen, wurden im Rahmen dieser Arbeit entwickelt. Im ersten Teil der Arbeit wird der Fujaba Process im Detail erklärt und unsere Erfahrungen mit dem Einsatz des Prozesses in Industrieprojekten sowie in der Lehre dargestellt. Der zweite Teil beschreibt die im Rahmen dieser Arbeit entwickelte Testgenerierung, die zu einem wichtigen Teil des Fujaba Process geworden ist. Hierbei werden aus den formalisierten Usecase Szenarien ausführbare Testfälle generiert. Es wird das zugrunde liegende Konzept, die konkrete technische Umsetzung und die Erfahrungen aus der Praxis mit der entwickelten Testgenerierung dargestellt. Der letzte Teil beschäftigt sich mit dem Debuggen im Fujaba Process. Es werden verschiedene im Rahmen dieser Arbeit entwickelte Konzepte und Techniken vorgestellt, die die Fehlersuche während der Applikationsentwicklung vereinfachen. Hierbei wurde darauf geachtet, dass das Debuggen, wie alle anderen Schritte im Fujaba Process, ausschließlich auf Modellebene passiert. Unter anderem werden Techniken zur schrittweisen Ausführung von Modellen, ein Objekt Browser und ein Debugger, der die rückwärtige Ausführung von Programmen erlaubt (back-in-time debugging), vorgestellt. Alle beschriebenen Konzepte wurden in dieser Arbeit als Plugins für die Eclipse Version von Fujaba, Fujaba4Eclipse, implementiert und erprobt. Bei der Implementierung der Plugins wurde auf eine enge Integration mit Fujaba zum

einen und mit Eclipse auf der anderen Seite geachtet. Zusammenfassend wird also ein Entwicklungsprozess vorgestellt, die Möglichkeit in diesem mit automatischen Tests Fehler zu identifizieren und diese Fehler dann mittels spezieller Debuggingtechniken im Programm zu lokalisieren und schließlich zu beheben. Dabei läuft der komplette Prozess auf Modellebene ab. Für die Test- und Debuggingtechniken wurden in dieser Arbeit Plugins für Fujaba4Eclipse entwickelt, die den Entwickler bestmöglich bei der zugehörigen Tätigkeit unterstützen.