

Tutorial 2: Modellbasiertes Testen

Alexander Pretschner (Technische Universität München)

Inhalt des Tutorials ist eine Einführung in die fundamentalen technischen und methodischen Konzepte des modellbasierten Testens, mit einem Fokus auf reaktiven ereignisdiskreten, eingebetteten Systemen. Die zentralen Ideen und methodischen Überlegungen sind allerdings direkt auf andere Anwendungsdomänen übertragbar. Verhaltensmodelle dienen dabei eine Codierung des Sollverhaltens, das mit dem Istverhalten eines Systems verglichen wird. Modelle sind Abstraktionen oder Vereinfachungen; demnach müssen die Abstraktionsniveaus von Modell und Implementierung durch eine Treiberkomponente abgeglichen werden. Die Komplexität des Testproblems wird also auf intellektuell beherrschbare Modelle und die (ebenfalls beherrschbare) Treiberkomponente verteilt.

Die Auswahl relevanter Modellabläufe (Sollverhalten, also Testfälle) erfolgt anhand von Testfallspezifikationen, die Mengen von Testfällen charakterisieren. Aus den Testfallspezifikationen können dann mit automatischen Verfahren Testfälle berechnet werden. Testfallspezifikationen erscheinen als das zentrale Problem des modellbasierten Testens. In diesem Zusammenhang wird auch die Rolle von Modellen der Umwelt diskutiert.

Im Tutorium finden methodische und technische Aspekte des modellbasierten Tests Berücksichtigung. In methodischer Hinsicht werden insbesondere (1) die unterschiedliche Natur von Modellen zur Test- und Codeerzeugung und als Spezifikation sowie (2) verschiedene Szenarien zur Erstellung von Modellen und Code beleuchtet. Da ein zum Test verwendetes Modell valide sein muss, kann auch die Frage nicht ausgespart werden, ob die sorgfältige Qualitätssicherung einer Implementierung nicht kosten-effektiver als die Validierung eines Modells mit anschließender Testfallgenerierung ist. Kurz erwähnt werden auch Verfahren zur automatischen Testfallableitung wie Model Checking, deduktives Beweisen und dedizierte Verfahren für die Exploration des Zustandsraums.

Industrielle Beispiele (Chipkartenanwendungen und automobiler Multimedienetze) illustrieren die Konzepte. Der Test kontinuierlicher und nichtdeterministischer Systeme wird nur angerissen. Auf spezifische Werkzeuge wird nicht eingegangen.

Dr. Alexander Pretschner

Studium der Informatik an der RWTH Aachen (Dipl.-Inform.), der University of Kansas (M.S.) und der TU München (Dr.rer.nat). Arbeitsgebiet Systems Engineering, mit starker Konzentration auf modellbasierte Ansätze zur Entwicklung und Qualitätssicherung, mit klarem Fokus auf die Testthematik. Mitarbeit in Projekten zur verteilten und personalisierten Suche im Web, zur Entwicklung und Qualitätssicherung diskret-kontinuierlicher Systeme, zur modellbasierten Entwicklung sicherheitskritischer Systeme, und zum modellbasierten Test von Chipkarten und Netzwerkkomponenten. Organisation eines Testworkshops, eines Dagstuhl-Seminars zum modellbasierten Test und aktuell eines ICSE-Workshops zum automobilen SW-Engineering.