

Aachener Informatik-Berichte, Software Engineering

herausgegeben von
Prof. Dr. rer. nat. Bernhard Rumpe
Software Engineering
RWTH Aachen University

Band 6

Christian Berger

**Automating Acceptance Tests for
Sensor- and Actuator-based Systems on
the Example of Autonomous Vehicles**

Shaker Verlag
Aachen 2010

Bibliographic information published by the Deutsche Nationalbibliothek

The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at <http://dnb.d-nb.de>.

Zugl.: D 82 (Diss. RWTH Aachen University, 2010)

Copyright Shaker Verlag 2010

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior permission of the publishers.

Printed in Germany.

ISBN 978-3-8322-9378-9

ISSN 1869-9170

Shaker Verlag GmbH • P.O. BOX 101818 • D-52018 Aachen

Phone: 0049/2407/9596-0 • Telefax: 0049/2407/9596-9

Internet: www.shaker.de • e-mail: info@shaker.de

Christian Berger

Automating Acceptance Tests for Sensor- and Actuator-based Systems on the Example of Autonomous Vehicles

Kurzfassung:

Die Software-Qualität der Datenverarbeitungskette Sensor- und Aktor-basierter Systeme zu beherrschen ist zunehmend komplexer und aufwändiger-beispielsweise bei autonomen Fahrzeugen für Autobahnen, städtische Umgebungen oder unbekannte Umgebungen. Ein Grund ist die bereits frühzeitige Abhängigkeit von den gesamten Sensorrohdaten, um die Datenverarbeitungskette auszulegen oder Teilkomponenten zu identifizieren. Bei Verwendung von Laserscannern oder Farbkamerasystemen können die Sensorrohdaten dabei sehr umfangreich ausfallen. Die Bereitstellung der Sensorrohdaten bedingt wiederum ein ebenfalls frühzeitig definiertes Sensorikkonzept, welches Montagepositionen und Kalibrierungsinformationen festlegt, um die erforderlichen Rohdaten über reale Situationen in der Systemumgebung zu sammeln. Diese Rohdaten werden dann bei der Integration einzelner Teilsysteme und für die Erprobung des Gesamtsystems eingesetzt.

Um diese Abhängigkeit zu reduzieren und damit Aufgaben, die auf dem kritischen Pfad eines Projekts liegen, zu entkoppeln, wird in dieser Arbeit ein Ansatz vorgestellt, der das Software Engineering für Sensor- und Aktor-basierte, autonome Systeme unterstützt. Der Ansatz basiert einerseits auf den Kundenanforderungen an das System im vorgesehenen Systemkontext und den damit zugeordneten Akzeptanzkriterien. Andererseits stützt sich der Ansatz auf einer strikten Entkopplung des eigentlichen Software Engineerings von der realen Systemhardware, um Akzeptanztests durch geeignete Systemsimulationen zu erlauben.

Hierfür werden zunächst basierend auf den Kundenanforderungen formal spezifizierte Szenarien über eine domänenspezifische Sprache abgeleitet, die Grundlage für die Bereitstellung von Umgebungsdaten und zugeordneten Situationen für die Sensor- und Aktor-basierten, autonomen Systeme sind. Ausgehend von diesen formal spezifizierten Umgebungsdaten werden die notwendigen Eingabedaten für die unterschiedlichen Schichten eines sensordatenverarbeitenden Systems abgeleitet, um die erforderlichen Aktionen und Reaktionen im Systemkontext auszulösen. Hierbei hängen auf unterster Ebene die Eingabedaten selbst wiederum von einem geeigneten Sensormodell ab, um die benötigten Sensorrohdaten zu berechnen. Unter anderem am Beispiel eines Laserscanners und eines Kamerasystems werden dazu Algorithmen konzipiert und realisiert, die hierfür moderne Grafikhardware einsetzen, um Sensorrohdaten selbst für komplexe Situationen zu berechnen.

Um die vorgenannten Aspekte umzusetzen, ist darüber hinaus eine

Entwicklungsumgebung erforderlich, in der Werkzeuge zur Modellierung und Verarbeitung der domänenspezifischen Sprache integriert sind. Weiterhin ist ein Software-Framework zur Bereitstellung robuster Lösungen für wiederkehrende Anforderungen der Software-Entwicklung wie die Synchronisierung nebenläufiger Anwendungsfäden oder die Realisierung verschiedener Kommunikationskonzepte erforderlich. Daher wurde zur Umsetzung der Konzepte in dieser Arbeit ein echtzeitfähiges Software-Framework entworfen und komplett in portablen C++ realisiert, das sowohl die formale Modellierungssprache für Systemkontexte, als auch verschiedene Sensormodelle für die Simulation enthält. Dieses Framework bildet die Basis zur Unterstützung der Akzeptanztests über unbeaufsichtigte Systemsimulationen für Teilsysteme oder des Gesamtsystems.

Am Beispiel autonomer Fahrzeuge wird die Anwendbarkeit des Ansatzes und der im Software-Framework realisierten Konzepte demonstriert. Dazu wird ein Algorithmus zur Fahrzeugnavigation entworfen, der eine Digitalkarte verwendet, um eine optimale Route zu einem Zielpunkt von einem beliebigen Ausgangspunkt zu finden. Hierzu wurde der Algorithmus gemäß dem Test-First-Prinzip entwickelt und durch unbeaufsichtigte und automatische Software-Tests durch einen Integrationsserver evaluiert. Die Implementierung und die Evaluierung verwenden hierbei die vorgenannten Konzepte und Algorithmen, sowie das Software-Framework. Zur Umsetzung wurden der Systemkontext des Fahrzeugs sowie die verbauten Sensoren unter Verwendung der bereitgestellten Entwicklungswerkzeuge modelliert, um das Gesamtsystem vor der ersten Inbetriebnahme am Versuchsträger zu erproben und abzusichern.

Christian Berger

Automating Acceptance Tests for Sensor- and Actuator-based Systems on the Example of Autonomous Vehicles

Abstract:

In projects dealing with autonomous vehicles which are driving in different contexts like highways, urban environments, and rough areas, managing the software's quality for the entire data processing chain of sensor- and actuator-based autonomous systems is increasingly complex. One main reason is the early dependency on all sensor's raw data to setup the data processing chain and to identify subsystems. These sensors' data might be extensive, especially if laser scanners or color camera systems are continuously producing a vast number of raw data. Moreover, due to this dependency the sensors' setup including their respectively specified mounting positions and calibration information is also necessary to gather real input data from real surroundings' situations of the system. This is even more important before actually starting to integrate independently developed subsystems for carrying out tests for the entire data processing chain.

To reduce this dependency and therefore to decouple tasks from the project's critical path, an approach is outlined in this thesis which was developed to support the software engineering for sensor- and actuator-based autonomous systems. This approach relies on customer's requirements and corresponding customer's acceptance criteria as well as the decoupling of the software engineering from the real hardware environment to allow appropriate system simulations.

Based on the customer's requirements, formally specified scenarios using a domain specific language are derived which are used to provide surroundings and suitable situations for a sensor- and actuator-based autonomous system. From these formally specified surroundings, the required input data is derived for different layers of a sensor data processing system to generate actions within the system's context. This input data itself depends on a given sensor model to compute its specific raw data. Amongst others, on the example of laser scanners and camera systems, algorithms using modern graphical processing units are outlined to generate the required raw data even for complex situations.

To realize the aforementioned aspects, a development environment is necessary consisting of tools for modeling and working with instances of a domain specific language. Furthermore, a software framework is required which provides easily usable and mature solutions for common programming requirements like synchronization for concurrent threads or communication in a high-level point of view. For relying on a consistent and homogeneous software framework for implementing the concepts, a highly portable and real-time-capable software

framework for distributed applications was realized which was written entirely from scratch in strictly object-oriented C++. Moreover, this software framework also integrates the formally modeled input data derived from the specified requirements and the sensors' models to allow unattended system simulations to support the acceptance tests for subsystems or an entire system.

On the example of autonomous vehicles, the applicability of the approach and the software framework is demonstrated by implementing a vehicle navigation algorithm which uses a given digital map for finding the best route to a desired destination from an arbitrarily chosen starting point. This algorithm was developed considering the test-first-principle and is continuously evaluated by unattended and automatic software tests which are executed on a continuous integration system. Its implementation as well as its evaluation make use of the aforementioned concepts and algorithms. Therefore, the vehicle's surroundings were formally modeled together with its necessary sensors using the provided development tools and environment for performing and evaluating unattended system runs before the algorithm was put into operation on the real vehicle.