

Jahresbericht 2012

Prof. Dr. Holger Giese
Fachgebiet Systemanalyse und Modellierung

Hasso-Plattner-Institut für
Softwaresystemtechnik
an der Universität Potsdam

Jahresbericht 2012

Fachgebiet Systemanalyse und Modellierung
Hasso-Plattner-Institut für Softwaresystemtechnik
Universität Potsdam



Fachgebiet *Systemanalyse und Modellierung*
Hasso-Plattner-Institut für Softwaresystemtechnik GmbH
Universität Potsdam
Prof.-Dr.-Helmert-Str. 2-3, D-14482 Potsdam
Leitung: Prof. Dr. Holger Giese

<http://www.hpi.uni-potsdam.de/giese>

Inhaltsverzeichnis

1 Personelle Zusammensetzung	1
2 Lehrveranstaltungen	3
2.1 Vorlesungen	3
2.2 Übungen/Projekte	3
2.3 Seminare	3
3 Betreuung von Studierenden und Dissertationen	4
3.1 Betreuung von Bachelorprojekten	4
3.1.1 Bachelorprojekte (abgeschlossen in 2012)	4
3.1.2 Laufende Bachelorprojekte (Abschluss in 2013)	4
3.2 Betreuung von Bachelorarbeiten	5
3.3 Betreuung von Masterprojekten	6
3.3.1 Laufende Masterprojekte (Abschluss in 2013)	6
3.4 Betreuung von Masterarbeiten	7
3.5 Betreuung von Dissertationen	8
4 Bearbeitete Forschungsthemen	9
4.1 Eine Modellgetriebene Infrastruktur für selbst-adaptive service-orientierte Systeme	9
4.2 Verifikation selbst-adaptiver service-orientierter Systeme	9
4.3 Effiziente Ausführung von Modell-Transformationen auf Basis von Tripel-Graph-Grammatiken	10
4.4 Effiziente Wartung von Modellen unter der Verwendung von Megamodellen und fortgeschrittene Anwendungen von Megamodellen	10
4.5 Megamodellierung der Entwicklung von Service Orientierten Enterprise Systemen	10
4.6 Erhebung und Validierung von Requirements durch Simulation und Animation . .	11
4.7 Modellierung, Analyse und Bewertung nicht-funktionaler Eigenschaften in komponentenbasierten eingebetteten Realzeitsystemen	11
4.8 Rekonfiguration und Adaption von Cyber-Physical Systems	11
4.9 Traceability zur Erfassung von Designrationalen und als Hilfsmittel für Versionierung	12
4.10 Quantitative Analyse von Service-orientierten Echtzeitsystemen	12
4.11 Graphtransformationssysteme und Invariant-Checking mit k-induktiven Invarianten	13
5 Drittmittelprojekte	14
5.1 DFG – Korrekte Modelltransormationen (KorMoran)	14
5.2 Hasso Plattner Design Thinking Research Program – Connecting Designing and Engineering Activities I / II	14
6 Forschungsk Kooperationen	16
6.1 Kooperationspartner aus der Wissenschaft	16
6.2 Kooperationspartner aus der Wirtschaft	16

7	Publikationen	17
7.1	Zeitschriftenartikel	17
7.2	Beiträge zu Büchern und Sammlungen	17
7.3	Begutachtete Konferenz- und Workshopartikel	18
7.4	Bücher und Tagungsbände	20
7.5	Technische Berichte	20
7.6	Miscellaneous	21
8	Vorträge	22
8.1	Eingeladene Vorträge	22
8.2	Vorträge auf Konferenzen und Workshops	22
9	Herausgeberschaft	25
9.1	Bücher und Tagungsbände	25
10	Web-Portale und -Services	25
10.1	Self-adaptive.org	25
10.2	MDELab.org	25
10.3	CPSLab.org	25
11	Mitgliedschaften, Programmkomitees und Gutachtertätigkeiten	26
11.1	Mitgliedschaften	26
11.2	Mitarbeit in Programmkomitees	27
11.3	Organisation von Tagungen und Workshops	28
11.4	Gutachtertätigkeiten	29
	11.4.1 Forschungsprojekte	29
	11.4.2 Zeitschriften und Magazine	29

1 Personelle Zusammensetzung



Leiter des Fachgebiets

Prof. Dr. Holger Giese

Sekretariat

Kerstin Miers

Postdocs

Dr. Christian Krause

Dr. Leen Lambers

Wissenschaftliche Mitarbeiter

Stephan Hildebrandt, M.Sc.

Dipl.-Inform. Stefan Neumann

PhD-Stipendiaten

Dipl.-Inform. Basil Becker

Thomas Beyhl, M.Sc.

Johannes Dyck, M.Sc.

Gregor Gabrysiak, M.Sc.

Regina Hebig, M.Sc.

Dipl.-Wirtsch.Inf. Thomas Vogel

Sebastian Wätzoldt, M.Sc.

Extern

Dipl.-Inform. Andreas Seibel

Studentische Hilfskräfte

Falk Benke
Josefine Harzmann
Jens Hildebrandt
Stefan Lehmann
Björn Müller
Dominic Petrick
Ingo Richter
Martin Zabel

Stefanie Birth
Peter Haucke
Sven Köhler
Alexander Lüders
Edgar Näther
Lukas Pirl
Christoph Sterz
Dmitry Zakharov

Johannes Dyck
Manuel Hegner
Christoph Kühnl
Christoph Matthies
Helena Niesche
Stephanie Platz
Henrik Steudel
Christian Zöllner

2 Lehrveranstaltungen

2.1 Vorlesungen

Sommersemester 2012

- Modellierung II
- Advanced Software Engineering for Embedded Systems

Wintersemester 2012/2013

- Modellgetriebene Softwareentwicklung
- Software Engineering for Embedded Systems

2.2 Übungen/Projekte

Sommersemester 2012

- Modellierung II
- Advanced Software Engineering for Embedded Systems

Wintersemester 2012/2013

- Modellgetriebene Softwareentwicklung
- Software Engineering for Embedded Systems

2.3 Seminare

Wintersemester 2012/2013

- Requirements Engineering: Elicitation and Negotiation
- Investigating Practical MDE-Approaches

3 Betreuung von Studierenden und Dissertationen

3.1 Betreuung von Bachelorprojekten

3.1.1 Bachelorprojekte (abgeschlossen in 2012)

- *Von der Spezifikation zum Strandeinsatz*

Betreuer: Prof. Dr. Holger Giese, Gregor Gabrysiak, Thomas Beyhl

Studenten: Matthias Bastian, Catharina Hahnfeld, Christoph Matthies, Lukas Pirl

Abstract: Im Projektseminar Requirements Engineering im Sommersemester 2011 wurden die Möglichkeiten zur Unterstützung der Wasserwacht des Berliner Roten Kreuz mit einem Softwaresystem zur Dokumentation in der Fernmeldeleitstelle am Wannsee erkundet. Auf Basis der in diesem Seminar erarbeiteten Ergebnisse, also konkreten Anforderungen befragter Bedarfsträger, soll nun im Rahmen dieses Bachelorprojekts ein umfassender Prototyp erstellt werden, der die gesammelten Anforderungen bestmöglich umsetzt. Da Anforderungen von Natur aus Änderungswünschen von Bedarfsträgern oder Änderungen aufgrund von externen Einflüssen unterliegen, besteht die Aufgabe nicht nur aus der einseitigen Umsetzung der gegebenen Anforderungen, sondern ebenso im Abgleich und notfalls in der Adaption derselben.

Im Bachelorprojekt soll auf Grundlage der zuvor übergebenen sowie eventuell zusätzlich erhobener Anforderungen eine prototypische Umsetzung erstellt werden. Beispiele der Anforderungen beinhalten unter Anderem die Prozessbegleitung (Unterstützung bei offenen Vorgängen oder Einsätzen), die Übersicht (Digitale Lagekarte zur Verwaltung der verfügbaren Einheiten) sowie die Technik (all dies unter Berücksichtigung des Schutzes kritischer Infrastrukturen).

Es soll ein Prototyp entwickelt und erprobt werden, der die erfassten Abläufe unterstützt und verfügbare Informationen (z.B. aus den neu eingeführten Digitalfunkgeräten) sammelt, aufbereitet und sinnvoll zu Verfügung stellt. Die übergebenen Anforderungen sind dabei als Problembeschreibung zu sehen – sie zeigen u.U. potenzielle Lösungen auf, schreiben sie jedoch nicht vor.

3.1.2 Laufende Bachelorprojekte (Abschluss in 2013)

- *From Creative Ideas to Well-Founded Engineering*

Betreuer: Prof. Dr. Holger Giese, Thomas Beyhl, Gregor Gabrysiak

Studenten: Anita Dieckhoff, Tom Bocklisch, Dominic Braeunlein, Tom Herold, Norman Rzepka, Thomas Werkmeister

Abstract: Softwareentwickler sind nicht dafür bekannt ihre Arbeitsergebnisse gerne zu dokumentieren. Bei Design Thinkern, die eine innovative Idee für neuartige Softwareprodukte mit Hilfe der Design Thinking Methode entwickelt haben, ist das ähnlich. Das Dokumentieren von Ideen, Anforderungen, Alternativen und Lösungen kostet Design Thinker Zeit, die sie selbst lieber in kreative Lösungen einfließen lassen. Jedoch ist gerade die Dokumentation der Design Thinking Ergebnisse als auch der Weg zur Lösung essentiell. Nur dadurch erfahren Softwareentwickler was sie lösen sollen, wie sie testen können, ob dies gelungen ist

und können im Fall von (techn.) Einschränkungen gut-informierte Entscheidungen treffen. Fehlende Dokumentation hingegen führt zum Verlust wertvoller Informationen bei der Übergabe von innovativen Ideen der Design Thinker an die Softwareentwickler. Zum Zeitpunkt der Übergabe treffen die informale Welt des Design Thinking und die formale Welt der Softwareentwicklung aufeinander. Diese Lücke gilt es zu verkleinern. Hinzu kommt, dass es oft besser ist Dinge sofort zu dokumentieren als erst Tage oder gar Wochen später. Daher muss eine Möglichkeit zur Dokumentation von Design Thinking Ergebnissen geschaffen werden, die den Design Thinking Prozess möglichst wenig beeinträchtigt und einen Mehrwert für Design Thinker selbst schafft, sodass sie gerne dokumentieren.

Design Thinker präsentieren oft Konzepte und Features in unterschiedlichsten Formen von Prototypen je nach dem was der Kunde wünscht. Die Adressaten sind dabei nicht die Ingenieure, die die Konzepte und Features umsetzen, sondern der Auftraggeber selbst. Die dahinterstehenden Anforderungen können durch fehlende Dokumentation verloren gehen. Diese Beobachtung ist Gegenstand unseres Forschungsprojekts Connecting Designing and Engineering Activities, das im Rahmen des Hasso Plattner Design Thinking Research Programms stattfindet. Unser Bachelorprojekt bettet sich in dieses Forschungsprojekt ein. Das Bachelorprojekt nimmt sich der Herausforderung an, Design Thinker zu motivieren und vor Augen zu führen warum die Dokumentation ihrer Ideen essentiell für die spätere Umsetzung und Markteinführung ist, sowie die Dokumentation zu erleichtern. Idealerweise entsteht die Dokumentation durch die Design Thinker als Nebenprodukt ihrer täglichen Arbeit und behindert die kreative Methodik des Design Thinking nicht. Durch Beobachtungen und Befragungen soll ermittelt werden wie eine entsprechende Softwarelösung zur Unterstützung der Dokumentation durch Design Thinker aussehen kann und welche Schritte hinsichtlich der Dokumentation der Design Thinking Ergebnisse zeitsparend automatisierbar sind. Die ermittelten Anforderungen und Use-Cases sollen als Ergebnis dieses Bachelorprojekts in ein System einfließen, das durch die Projektteilnehmer erstellt und evaluiert wird.

Erste Beobachtungen und Interviews mit, sowie Erfahrungen der HPI School of Design Thinking haben gezeigt, dass eine Art digitale Kommunikationsplattform, wie z.B. Social Networks, sowie dazugehörige mobile Apps eine mögliche Lösung der beschriebenen Herausforderung darstellen. Mit einer solchen Plattform wären Design Thinker in der Lage, entsprechende Dokumentation als Nebenprodukt täglicher Kommunikation zu sammeln und zusätzlich mit Meta-Informationen anzureichern. Durch zusätzliche Verarbeitungsschritte könnten die erfassten Daten verwendet werden, um Dinge wie Empfehlungen für Beiträge, Aussagen über die Vollständigkeit der Dokumentation sowie die Generierung einer finalen Dokumentation zu ermöglichen. Die Erstellung oder Adaption einer solchen Kommunikationsplattform steht entsprechend der durch die Projektteilnehmer ermittelten Anforderungen im Fokus des Projekts.

3.2 Betreuung von Bachelorarbeiten

[BA1] Matthias Bastian. Entwurf und Umsetzung eines Softwaresystems für die DRK Wasserwacht Berlin mit Fokus auf Architektur und Erweiterbarkeit. Bachelor's thesis, Hasso-Plattner-Institut für Softwaresystemtechnik, Universität Potsdam, 2012.

- [BA2] Robert Fruth. Gütebestimmung von Signalen: Design einer Fitnessfunktion für kontinuierliche Systeme im Bereich der evolutionären Testfallgenerierung. Bachelor's thesis, Hasso-Plattner-Institut für Softwaresystemtechnik, Universität Potsdam, 2012.
- [BA3] Catharina Hahnfeld. Analyse und Umsetzung sich ändernder Nutzeranforderungen eines Softwaresystems für die DRK Wasserwacht Berlin. Bachelor's thesis, Hasso-Plattner-Institut für Softwaresystemtechnik, Universität Potsdam, 2012.
- [BA4] Christoph Matthies. Analyse und Umsetzung von Anforderungen bezüglich der Verwaltung eines Softwaresystems für die DRK Wasserwacht Berlin. Bachelor's thesis, Hasso-Plattner-Institut für Softwaresystemtechnik, Universität Potsdam, 2012.
- [BA5] Lukas Pirl. Architektur und Entwurf der Datenhaltung mit nachvollziehbaren Änderungen und Löschungen von Daten für die DRK Wasserwacht Berlin. Bachelor's thesis, Hasso-Plattner-Institut für Softwaresystemtechnik, Universität Potsdam, 2012.

3.3 Betreuung von Masterprojekten

3.3.1 Laufende Masterprojekte (Abschluss in 2013)

- *Iterative Development and Execution of Consistency-Preserving Rule-Based Refactorings*
Betreuer: Prof. Dr. Holger Giese, Dr. Leen Lambers, Basil Becker, Johannes Dyck, Stephan Hildebrandt

Studenten: Ekaterina GavriloVA, Josefine Harzmann, Hauke Klement, Michael Wolowyk

Abstract: Refactorings sind eine weit verbreitete Technik zur Verbesserung der Lesbarkeit, Komplexität, Wartbarkeit oder Erweiterbarkeit eines Systems oder Modells durch strukturelle Änderungen. Dabei sollen diese Änderungen nicht nur das Verhalten, sondern insbesondere auch die Konsistenz des geänderten Systems bewahren. Insbesondere sollte ein Refactoring beispielsweise ein Softwaresystem nur derartig verändern, dass die Bedingungen der verwendeten Sprache zur Sicherstellung der Wohlgeformtheit auch nach der Ausführung des Refactorings gewahrt bleiben.

Ob ein konkretes Refactoring die Konsistenz bewahrt, kann zur Laufzeit für ein konkretes System festgestellt werden. Allerdings liegt die Entscheidung über den Umgang mit einer Konsistenzverletzung dann beim Benutzer, also Anwender des Refactorings, der diese Entscheidung möglicherweise nicht ohne genaue Kenntnis des Refactorings und dessen interner Funktionsweise treffen kann. Um stattdessen diese Entscheidung auf die Ebene der Refactoring-Entwickler zu verschieben, beschäftigt sich das Projekt mit einer Entwicklungsumgebung für Refactorings, in der Konsistenzverletzungen bereits zur Entwicklungszeit festgestellt und behoben werden können. Umgekehrt kann der Anwender dadurch Konsistenzverletzungen zur Laufzeit ausschließen.

Die im Projekt verwendeten Werkzeuge und Technologien basieren auf der Entwicklungsumgebung Eclipse und dem Eclipse Modeling Framework (EMF). Dabei werden die Bedingungen an die Wohlgeformtheit eines Systems mit Hilfe von EMF als Modelle auf der Basis von Metamodellen der jeweiligen Sprache dargestellt. Um Konsistenzbewahrung formal zu verifizieren, werden die Konzepte der Graphbedingungen und Graphregeln auf besagte Modelle

beziehungsweise auf die Refactorings angewendet. Ein Werkzeug zur Verifikation induktiver Invarianten, das auf Graphbedingungen und Graphregeln basiert, sichert dann die Konsistenzerhaltung zu oder liefert anderenfalls aussagekräftige Gegenbeispiele. Auf Basis dieser Gegenbeispiele ist der Entwickler in der Lage, durch iteratives Vorgehen schließlich ein konsistenzerhaltendes Refactoring zu entwickeln. Für ein solches Refactoring gilt dann, dass es für jedes konsistente Ausgangssystem bei Anwendung stets ein konsistentes Ergebnissystem liefert.

Zur Ausführung der Refactorings wie auch für eine zusätzliche Überprüfung der Konsistenz des Ausgangssystems zur Laufzeit werden Story-Diagramme verwendet. Sowohl für die Prüfung eines Refactorings auf Konsistenzerhaltung als auch für dessen Ausführung sind daher Modelltransformationen erforderlich. Zum einen müssen das Refactoring und die Wohlgeformtheitsbedingungen, die auf dem Metamodell der jeweiligen Sprache basieren, in ein für das Verifikationstool verständliches Format übertragen werden. Zum anderen ist eine Modelltransformation von einem Refactoring und zusätzlichen Konsistenzprüfungen in ein ausführbares Story-Diagramm erforderlich. Als Grundlage der in der Entwicklungsumgebung verwendeten Modelle ist weiterhin Metamodellierung ein zentraler Bestandteil des Projekts.

Abschließend soll eine Evaluierung als Teil des Projekts erfolgen. Insbesondere soll als konkretes Beispiel JaMoPP (Java Model Parser and Printer) verwendet werden, um aus Java-Programmen Modelle zu erzeugen, die auf einem durch JaMoPP bereitgestellten Metamodell für Java basieren. Dadurch können gängige und weit verbreitete Java-Refactorings auf Konsistenzbewahrung überprüft werden. Da der Ansatz allerdings generischer Natur ist, kann die zu entwickelnde Entwicklungsumgebung für alle Sprachen verwendet werden, sofern entsprechende Metamodelle der Sprache auf Basis von EMF zur Verfügung stehen und sich die Refactorings und Bedingungen an die Wohlgeformtheit als Graphregeln und Graphbedingungen darstellen lassen.

3.4 Betreuung von Masterarbeiten

- [MA1] Johannes Dyck. Increasing expressive power of graph rules and conditions and automatic verification with inductive invariants. Master's thesis, Hasso-Plattner-Institut für Softwaresystemtechnik, Universität Potsdam, 2012.
- [MA2] Daniel Eichler. Manipulation formaler Modelle anhand natürlichsprachlicher Repräsentationen. Master's thesis, Hasso-Plattner-Institut für Softwaresystemtechnik, Universität Potsdam, 2012.
- [MA3] Norman Kluge. Abbildung abstrakter AUTOSAR Architekturmodelle auf Timed Automata zur frühzeitigen Analyse von Realzeit-Eigenschaften. Master's thesis, Hasso-Plattner-Institut für Softwaresystemtechnik, Universität Potsdam, 2012.
- [MA4] Christoph Mecklenburg. Analyse, Einordnung und Evaluation des Design Led Innovation Ansatz: eine Fallstudie zum nutzerzentrierten Softwareentwurf. Master's thesis, Hasso-Plattner-Institut für Softwaresystemtechnik, Universität Potsdam, 2012.
- [MA5] Edgar Näther. Model-Checking zeitbehafteter Graphtransformationssysteme in GROOVE. Master's thesis, Hasso-Plattner-Institut für Softwaresystemtechnik, Universität Potsdam, 2012.

- [MA6] Steven Reinisch. Software Process Guidance in Model Driven Engineering,. Master's thesis, Hasso-Plattner-Institut für Softwaresystemtechnik, Universität Potsdam, 2012.
- [MA7] Henrik Steudel. An Integration Framework for Collaborative Model Management based on Distributed Megamodels. Master's thesis, Hasso-Plattner-Institut für Softwaresystemtechnik, Universität Potsdam, 2012.
- [MA8] Dmitry Zakharov. Design and Implementation of an Efficient Pattern Matcher and Debugger. Master's thesis, Hasso-Plattner-Institut für Softwaresystemtechnik, Universität Potsdam, 2012.

3.5 Betreuung von Dissertationen

Basil Becker: Modellierung und Verifikation selbst-adaptiver service-orientierter Systeme

Thomas Beyhl: Traceability zur Erfassung von Designrationalen und als Hilfsmittel für Versionierung

Johannes Dyck: Graphtransformationssysteme und Invariant-Checking mit k-induktiven Invarianten

Gregor Gabrysiak: Erhebung und Validierung von Requirements durch Simulation und Animation

Regina Hebig: Megamodeling the Development of Service Oriented Enterprise Systems

Stephan Hildebrandt: Effiziente Ausführung und Fehleranalyse von Modelltransformationen und -synchronisationen

Stefan Neumann: Modellierung, Analyse und Bewertung von Schnittstellen zur Beschreibung nicht funktionalen Verhaltens in komponentenbasierten Systemen

Andreas Seibel: Modellierung, Analyse und Bewertung von Verteilungs- und Architekturmodellen in Bezug auf Kompatibilität, Leistungsfähigkeit und Kosten

Thomas Vogel: Eine modellgetriebene Infrastruktur für selbst-adaptive service-orientierte Systeme

Sebastian Wätzoldt: Reconfiguration and Adaptation of Cyber-Physical Systems

4 Bearbeitete Forschungsthemen

4.1 Eine Modellgetriebene Infrastruktur für selbst-adaptive service-orientierte Systeme

Diese Arbeit verbindet die beiden Forschungsbereiche Model-Driven Engineering (MDE) und Selbst-adaptive Softwaresysteme, indem eine modellgetriebene Infrastruktur die Selbst-Adaption eines Systems unterstützten oder gar ermöglichen soll. Während der Fokus von MDE auf der Entwicklung und dem Deployment von Softwaresystemen liegt, können MDE Konzepte und Technologien auch für die Laufzeitverwaltung von Systemen hilfreich sein. Beispielsweise können Modelle verschiedene Sichten auf unterschiedlichen Abstraktionsebenen eines laufenden Systems bieten und damit eine reichhaltige semantische Grundlage für die Selbst-Adaption sein. Desweiteren soll der Einsatz von MDE Technologien für die Verwaltung und Evolution von Laufzeitmodellen untersucht werden und wie diese Laufzeitmodelle zu Modellen der Entwicklungsphase in Beziehung stehen können. Service-orientierte Systeme unterstützen aufgrund ihrer Modularität und losen Kopplung grundlegend die Adaption auf der Ebene der Architektur, aber ihre inherente Komplexität und Verteilung bringen Herausforderungen mit sich. Das Ziel dieser Arbeit sind Konzepte für eine generische modellgetriebene Infrastruktur, die die Laufzeitverwaltung und insbesondere die (Selbst-)Adaption von verteilten, service-orientierten Softwaresystemen ermöglicht. Diese Konzepte sollen durch einen Prototyp evaluiert werden.

Ansprechpartner: Thomas Vogel

4.2 Verifikation selbst-adaptiver service-orientierter Systeme

Service-orientierte Architekturen werden häufig eingesetzt, um die stetig steigende Komplexität moderner Softwaresysteme weiterhin beherrschen zu können. Eines der dabei eingesetzten Konzepte ist die lose Bindung der einzelnen Teile des Systems, was dazu führt, dass erst zum Zeitpunkt der Ausführung bekannt ist, welche Komponenten des Systems miteinander interagieren. Gleichzeitig erlaubt die lose Kopplung auf leichte Art und Weise neue Services zu einem System hinzuzufügen. Durch diese Dynamik stellen Service-orientierte Architekturen hohe Anforderungen an die Verifikationstechniken, die genutzt werden können, um diese Systeme zu verifizieren. In meiner Promotion entwickle ich Verfahren, die es durch den gezielten Einsatz von Abstraktion und Verfeinerung der Verhaltensspezifikationen erlauben, solche Systeme zu verifizieren. Das entwickelte Verfahren ist dabei robust gegenüber neu hinzugefügten Services, wodurch es sich deutlich von bestehenden Verifikationsverfahren unterscheidet. Robust meint in diesem Zusammenhang, dass sich die Schritte für die Verifikation von Änderungen des Systems auf die geänderten Teile beschränken.

Des Weiteren entwickle ich eine Verifikationsmethode, die es erlaubt, zeitbehaftete Systeme mit einem potentiell unendlichen Zustandsraum bezüglich Sicherheitseigenschaften (safety properties) zu verifizieren.

Ansprechpartner: Basil Becker

4.3 Effiziente Ausführung von Modell-Transformationen auf Basis von Tripel-Graph-Grammatiken

In der modell-getriebenen Softwareentwicklung spielen Modelltransformationen eine wichtige Rolle, um verschiedene Modelle eines Systems in einander zu transformieren und nach Änderungen konsistent zu halten. Dazu muss untersucht werden, wie Modelländerungen erfasst und verarbeitet werden können, und wie die Anzahl der Operationen auf dem Zielmodell einer Synchronisation minimiert werden können, um eine effiziente Ausführung zu gewährleisten.

Eine weitere Fragestellung ist, wie die Ausdrucksmächtigkeit von TGGs erhöht werden kann, ohne die Vorteile der formalen Absicherung von Graphtransformationen zu verlieren. Da TGGs im Rahmen des Modelltransformationssystems des Fachgebiets auf Story-Diagramme abgebildet werden, werden auch Möglichkeiten zur effizienten Ausführung solcher Story-Diagramme untersucht.

Ansprechpartner: Stephan Hildebrandt

4.4 Effiziente Wartung von Modellen unter der Verwendung von Megamodellen und fortgeschrittene Anwendungen von Megamodellen

Modellgetriebene Softwareentwicklung leidet unter der stetig steigenden Komplexität von heutigen Softwaresystemen. Wegen diesem Problem ist vor einigen Jahren der Forschungsbereich der Megamodellierung entstanden. Heutige Ansätze zur Megamodellierung sind schon weit fortgeschritten. Dennoch fehlt aktuellem State-of-the-Art noch Unterstützung zur automatisierten und effizienten Wartung von Modellen mittels Megamodellen. Zusätzlich wurde bis jetzt noch nicht alle Möglichkeiten durch die Anwendung von Megamodellen vergegenwärtigt. Diese Forschungsarbeit arbeitet an der Verbesserung der Automatisierung und effizienter Wartung sowie das Finden neuer Anwendungsgebiete für Megamodelle.

Ansprechpartner: Andreas Seibel

4.5 Megamodellierung der Entwicklung von Service Orientierten Enterprise Systemen

Für die Entwicklung von Software können vielfältige Ansätze verwendet werden. In den einfachsten Fällen werden einzelne Programmiersprachen genutzt, wobei ein Kompilierungsschritt genügt um ein lauffähiges Programm zu erhalten. In komplexeren Fällen, wie zum Beispiel bei Modell getriebener Entwicklung (MDE), werden in mehreren Entwicklungsschritten viele unterschiedliche Repräsentationen des Zielsystems genutzt.

Besonders die Entwicklung von Service Orientierten Architekturen (SOA), bei denen Programmodule implizit über sprachunabhängige Schnittstellen gekoppelt sind, führt zu komplexen und verschiedenartigen Entwicklungsabläufen. Das macht es schwierig sicherzustellen, dass die Entwicklung von Service Orientierten Architekturen so produktiv und effizient wie möglich ist.

Megamodelle können genutzt werden um die Beziehungen zwischen Modellen, Sprachen, Spezifikationen und Tools zu erfassen. Es ist Ziel der Promotion eine Megamodellierungsmethodik zu entwickeln, die es erlaubt Ansätze für die Entwicklung von Service Orientierten Architekturen

zu planen und zu dokumentieren. Auf Basis dieser Megamodelle und empirischen Studien soll es später möglich sein Best Practices zu identifizieren.

Ansprechpartner: Regina Hebig

4.6 Erhebung und Validierung von Requirements durch Simulation und Animation

Dieses Forschungsthema setzt sich mit dem Problem der Requirements Validierung auseinander. Während der Erhebung von Anforderungen haben Requirements Engineers die Wahl, auf welche Art und Weise sie die gewonnenen Erkenntnisse modellieren wollen. Obwohl es zahlreiche Notationen und Ansätze gibt, wird immer noch sehr häufig auf Natural Language (NL) zurückgegriffen. Obwohl NL zumeist nur aus einer Menge an einfachen Aussagen in Satzform besteht und daher nur schwer zu verwalten ist, zeichnet sich dieser Ansatz eine einfache Präsentation und hohe Verständlichkeit aus. Sobald komplexe Ansätze mit entsprechenden Notationen genutzt werden, können modellierte Erkenntnisse nur schwer mit den Endnutzern validiert werden, da diese die eingesetzten Notationen nicht kennen und daher den Inhalt nicht verstehen. Um dieses Problem zu umgehen, beschäftigt sich dieses Dissertationsthema mit der interaktiven Simulation und Animation solcher Modelle, um sie zum einen den Endnutzern verständlich zu präsentieren und zum anderen interaktiv zu erweitern, d. h. um neue Erkenntnisse zu bereichern.

Ansprechpartner: Gregor Gabrysiak

4.7 Modellierung, Analyse und Bewertung nicht-funktionaler Eigenschaften in komponentenbasierten eingebetteten Realzeitsystemen

Das Forschungsthema beschäftigt sich mit der Analyse von Eigenschaften verteilter, eingebetteter Realzeitsysteme, wie diese beispielsweise im Automobil zum Einsatz kommen. Dabei wird der Fokus auf die Analyse nicht-funktionaler Eigenschaften gelegt, wie etwa im Fall zeitlicher-Anforderungen bei eingebetteten Systemen. Die steigende Komplexität der Systeme macht es zunehmend schwieriger diese Systeme zu entwickeln, da gerade nicht-funktionale systemweite Eigenschaften mit den bisherigen Verfahren erst sehr spät im Entwicklungsprozess analysiert werden können. Es wird nach einem Ansatz gesucht, welcher es erlaubt, frühzeitig im Entwicklungsprozess Aussagen über entsprechende Eigenschaften des Gesamtsystems zu treffen.

Ansprechpartner: Stefan Neumann

4.8 Rekonfiguration und Adaption von Cyber-Physical Systems

Ein Großteil heutiger Software befindet sich in kleinen eingebetteten Systemen. Diese sind meist vor dem Nutzer verborgen und realisieren Steuerungs-, Überwachungs- oder Regelungsaufgaben. Zusätzlich sind diese Systeme durch eine eingeschränkte Ressourcennutzung und weiterer nicht funktionaler Eigenschaften, wie harte Echtzeit Anforderungen oder begrenzte Speicherkapazität, gekennzeichnet. Cyber-Physical Systems sind eine neue Generation dieser eingebetteten Systeme,

die durch eine immer stärkere Vernetzung und der Interaktion mit der physikalischen Umgebung entstehen.

Dieses Forschungsthema befasst sich mit der dynamischen Anpassung des Verhaltens von eingebetteten und Cyber-Physical Systems. Bestehende Ansätze zur Rekonfiguration oder Adaption des Verhaltens berücksichtigen die geforderten Echtzeiteigenschaften nur ungenügend oder fordern unrealistische Annahmen für die Software.

Eine Evaluierung wird anhand des AUTOSAR Standards vorgenommen, der größtenteils im Automobilbereich eingesetzt wird. Zusätzlich werden die Forschungsergebnisse anhand eines Roboterlabores evaluiert. Dieses besteht aus drei mobilen Robotern und einer variablen Umgebung, die ein Produktionsszenario simuliert.

Ansprechpartner: Sebastian Wätzoldt

4.9 Traceability zur Erfassung von Designrationalen und als Hilfsmittel für Versionierung

Traceability ist Gegenstand aktueller Forschung und ermöglicht durch die Erzeugung von Traceability Informationen u.a. Aussagen über den Einfluss von Artefaktänderungen, Erfüllung von Systemanforderungen, Entwicklungsfortschritt und Designrationalen. Um diese Aussagen zuverlässig treffen zu können müssen die Traceability Informationen komplett und korrekt sein. Ansätze zur Erzeugung dieser Traceability Informationen sind oft technologie- oder sprachen-spezifisch. Darüber hinaus ändern sich Traceability Informationen über die Zeit und sind daher Gegenstand von Wartung und Versionierung.

In diesem Forschungsthema geht es darum wie Traceability-Ansätze kombiniert werden können um deren Vollständigkeit und Korrektheit zu erhöhen, wie die erhaltenen Traceability Informationen versioniert werden können und wie sie im Kontext der Methodik des Design Thinking sowie der Modellgetriebenen Softwareentwicklung genutzt werden können.

Im Rahmen der Methodik des Design Thinking wird evaluiert wie Traceability Informationen und deren Versionierung zur Erfassung und Auffindung von Designrationalen genutzt werden können. Im Kontext der Modellgetriebenen Softwareentwicklung wird evaluiert wie Traceability Informationen kollaborative Entwicklungsszenarien unterstützen können.

Ansprechpartner: Thomas Beyhl

4.10 Quantitative Analyse von Service-orientierten Echtzeitsystemen

Eine der wichtigsten Herausforderungen in der Entwicklung von Service-orientierten Systemen ist die Vorhersage und die Zusicherung von nicht-funktionalen Eigenschaften, wie Ausfallsicherheit und Verfügbarkeit von zusammengesetzten, interorganisationellen Diensten. Diese Systeme sind oft charakterisiert durch eine Vielzahl von inhärenten Unsicherheiten, welche sowohl in der Modellierung als auch in der Analyse eine Rolle spielen. Ziel dieses Projektes ist es ein probabilistisches, Zeit-behaftetes Modell zu entwickeln, welches es ermöglicht quantitative Aussagen über nicht-funktionale Eigenschaften von Service-orientierten Echtzeitsystemen mittels formaler Methoden zu treffen. Als grundlegendes formales Modell für nicht-funktionale Eigenschaften werden sogenannte Interval Probabilistic Timed Automata (IPTA) benutzt. Dieses Modell besitzt sowohl ausreichende

Ausdrucksstärke für eine realistische und modulare Spezifikation als auch geeignete formale Methoden zur Bestimmung von quantitativen Sicherheits- und Zuverlässigkeitseigenschaften. Als technisches Mittel für die quantitative Analyse wird probabilistisches Model Checking, speziell probabilistische Zeit-beschränkte Erreichbarkeitsanalyse und Bestimmung von Erwartungswerten für Kosten und Vergütungen eingesetzt. Um die quantitative Analyse mittels probabilistischem Model Checking durchzuführen, wurde eine Erweiterung des PRISM-Werkzeuges zur Modellierung und Analyse von IPTA entwickelt.

In diesem Projekt wurde eine Zusammenarbeit mit der Gruppe von Prof. Marta Kwiatkowska in Oxford initiiert und ein Antrag auf Sachbeihilfe bei der Deutschen Forschungsgemeinschaft (DFG) verfasst.

Ansprechpartner: Christian Krause

4.11 Graphtransformationssysteme und Invariant-Checking mit k-induktiven Invarianten

Invariant-Checking ist eine statische Analyse-Technik, mit der auf Basis der Verhaltensspezifikation eines Systems die Gültigkeit oder Ungültigkeit bestimmter Eigenschaften des Systems formal nachgewiesen werden kann. Typische Beispiele für derartige Eigenschaften sind Sicherheits- und Lebendigkeitseigenschaften, die für die Korrektheit, Sicherheit und konstante Ausführbarkeit eines Systems eine wichtige Rolle spielen. Insbesondere für sicherheitskritische oder auch für selbstadaptive Systeme sind solche Eigenschaften und deren formale Verifikation interessant.

Der im konkreten Fall verfolgte Ansatz des Invariant-Checking basiert auf Graphtransformationen zur Verhaltensspezifikation und Graphbedingungen zur Darstellung der gewünschten Eigenschaften. Dabei kann festgestellt werden, ob eine solche Eigenschaft eine induktive Invariante ist, also ob sie für einen Übergang des Systems von einem Zustand in den nächsten in jedem Fall bewahrt bleibt.

Das Forschungsthema beschäftigt sich mit der Erweiterung des Konzepts der induktiven Invarianten auf k-induktive Invarianten, wobei nicht lediglich einschrittige Zustandsübergänge betrachtet werden. Vielmehr kann durch die Untersuchung eines Zustandspfades der Länge k eine detailliertere Aussage über die Gültigkeit der zu beweisenden Eigenschaften getroffen werden. Beispielsweise könnte eine Eigenschaft als induktive Invariante zurückgewiesen werden, weil die Eigenschaft nach einem Zustandsübergang aus einem Zustand verletzt wird, der wiederum nur aus einem anderen verbotenen Zustand erreichbar ist. Durch die Untersuchung eines längeren Pfades wird die Zahl der Gegenbeispiele, die auf derartigen nicht korrekt erreichbaren Zuständen basieren, reduziert. Ein weiterer Punkt im Rahmen des Themas ist die Ausdrucksmächtigkeit des Ansatzes und die potentielle Erweiterung derselben.

Ansprechpartner: Johannes Dyck

5 Drittmittelprojekte

5.1 DFG – Korrekte Modelltransformationen (KorMoran)

Gefördert: ab 10/2009

Drittmittelgeber: DFG

Bislang gibt es bis auf eigene Vorarbeiten keine Arbeiten, in denen Methoden für den formalen Nachweis der Korrektheit einer durch Modelltransformationen beschriebenen Transformation basierend auf Graphtransformationen vorgestellt werden. Ausgehend von den auf Graphtransformationssystemen basierenden Story Diagrammen und Triplegraphgrammatiken als Repräsentanten für operationale und relationale Modelltransformationsansätze wollen wir die Tatsache nutzen, dass Graphtransformationssysteme sich auch zur Spezifikation der Semantik von Modellen eignen, so dass wir das Problem der formalen Verifikation von Modelltransformationen mit einem einzigen formalen Modell angehen können.

Darauf aufbauend soll ein Ansatz für die systematische Entwicklung korrekter Modelltransformationen entwickelt und erprobt werden, der entsprechende Konzepte und Algorithmen für die formale Analyse und Verifikation der Modellsynchronisationen, Modelltransformationen und Modelltransformationsergebnisse enthält, die existierende Werkzeugunterstützung für Story Diagramme und Triplegraphgrammatiken soll um Werkzeuge für die formale Verifikation (automatisch und semi-automatisch) ergänzt werden, und es soll ein Vorgehen bzw. ein Prozess zur Verifikation aus Entwickler- und Benutzersicht ausgearbeitet werden. Anhand von zwei Fallstudien (aus dem Automotive-Bereich und dem Maschinenbau) soll die Praxistauglichkeit der entwickelten Methoden nachgewiesen werden

Ansprechpartner: Holger Giese, Leen Lambers

5.2 Hasso Plattner Design Thinking Research Program – Connecting Designing and Engineering Activities I / II

Gefördert: ab 10/2011

Drittmittelgeber: Hasso Plattner Design Thinking Research Program (HPDTRP)

Dies ist ein Projekt aus dem HPI - Stanford Design Thinking Research Program, einem Kooperationsprogramm zwischen der Stanford University School of Engineering und des Hasso-Plattner-Instituts.

Die verschiedenen Design Thinking Aktivitäten resultieren in einer Vielzahl analoger als auch digitaler Artefakte, die Arbeitszustände widerspiegeln und als Medium für die Kommunikation verwendet werden. Dabei enthalten sie Designentscheidungen, Beobachtungen und Erkenntnisse. Wenn es zum Engineering kommt und manchmal auch wenn Design Thinking Aktivitäten erneut durchgeführt werden, sind die Informationen, die durch die Artefakte mitgeliefert oder verwaltet werden nicht genug. Hinzu kommt, dass frühere Artefakte, deren Kontext, Abhängigkeiten zwischen den

Artefakten, die Design Rationale und viele andere Details, die das Artefakt selbst nicht speichert, benötigt werden. Diese Informationen sind oft nur schwer oder gar nicht wiederherzustellen. Im ersten Teil des Research Projekts stellen wir vor wie Design-Artefakte und deren Abhängigkeiten in einer kosten-effizienten Art und Weise organisiert werden sollten um das Extrahieren der benötigten Informationen für das Engineering als auch Design Thinking zu ermöglichen. Im zweiten Teil des Research Projekts untersuchen wir wie Traceability Informationen mit möglichst geringem Aufwand aber großen Nutzen erfasst werden können. Dazu untersuchen wir welche Anforderungen eine geeignete Dokumentationsplattform ermöglichen sollte.

Ansprechpartner: Holger Giese, Thomas Beyhl

6 Forschungsk Kooperationen

6.1 Kooperationspartner aus der Wissenschaft

Sabine Glesner (TU Berlin)

Verifikation von Code-Generierung und Modelltransformationen

Paola Inverardi und Henry Muccini (Universität L'Aquila, Italien)

Analyse von Softwarearchitekturen

Wilhelm Schäfer (Universität Paderborn)

Mechatronic UML

Tingting Han und Marta Kwiatkowska (Universität Oxford)

Modellierung von zeitbehaftetem probabilistischem Verhalten

Jesper Andersson (Linnaeus University, Schweden), Luciano Baresi (Politecnico di Milano, Italien), Nelly Bencomo (INRIA, Frankreich), Rogério de Lemos (University of Kent, UK), Alessandra Gorla (University of Lugano, Schweiz), Paola Inverardi (Universität L'Aquila, Italien)

Software Engineering Processes for Self-adaptive Systems

6.2 Kooperationspartner aus der Wirtschaft

D-LABS GmbH, Potsdam

Design Consulting für Softwareprodukte

dSpace GmbH, Paderborn

Automotives Software Engineering, Sicherheitsanalysen, Verifikation von Echtzeitverhalten

Hella KG Hueck & Co., Lippstadt

Automotives Software Engineering, Sicherheitsanalysen

SAP AG

Model-driven Language Engineering

SAP Deutschland AG & Co. KG, Walldorf

Erfassung von modelgetriebenen Entwicklungsansätzen in der Praxis

Capgemini

Erfassung von modelgetriebenen Entwicklungsansätzen in der Praxis

Ableton AG

Erfassung von modelgetriebenen Entwicklungsansätzen in der Praxis

VCat Consulting GmbH

Erfassung von modelgetriebenen Entwicklungsansätzen in der Praxis

Wasserwacht Berlin vom Deutschen Roten Kreuz

Anforderungsanalyse

7 Publikationen

7.1 Zeitschriftenartikel

- [A1] Hartmut Ehrig, Ulrike Golas, Annegret Habel, Leen Lambers, and Fernando Orejas. M-Adhesive Transformation Systems with Nested Application Conditions, Part 1: Parallelism, Concurrency and Amalgamation. *Mathematical Structures in Computer Science*, 2012. to appear.
- [A2] Hartmut Ehrig, Ulrike Golas, Annegret Habel, Leen Lambers, and Fernando Orejas. M-Adhesive Transformation Systems with Nested Application Conditions, Part 2: Embedding, Critical Pairs and Local Confluence. *Fundamenta Informaticae*, 118(1-2):35–63, 2012.
- [A3] Holger Giese, Stephan Hildebrandt, and Leen Lambers. Bridging the gap between formal semantics and implementation of triple graph grammars. *Software and Systems Modeling*, pages 1–27, April 2012. (Published online: 28 April 2012).
- [A4] Ulrike Golas, Leen Lambers, Hartmut Ehrig, and Fernando Orejas. Attributed graph transformation with inheritance: Efficient conflict detection and local confluence analysis using abstract critical pairs. *Theoretical Computer Science*, 424:46 – 68, 2012.
- [A5] Natallia Kokash, Christian Krause, and E. P. de Vink. Reo+mCRL2: A Framework for Model-checking Dataflow in Service Compositions. *Formal Aspects of Computing*, 24(2):187–216, 2012.
- [A6] Christian Krause, Holger Giese, and Erik de Vink. Compositional and behavior-preserving reconfiguration of component connectors in Reo. *Journal of Visual Languages & Computing*, 2012.
- [A7] Fernando Orejas and Leen Lambers. Lazy Graph Transformation. *Fundamenta Informaticae*, 118(1-2):65–96, 2012.

7.2 Beiträge zu Büchern und Sammlungen

- [S1] Gregor Gabrysiak, Holger Giese, and Thomas Beyhl. Virtual Multi-User Software Prototypes III. In Hasso Plattner, Christoph Meinel, and Larry Leifer, editors, *Design Thinking Research - Measuring Performance in Context*, Understanding Innovation, pages 263–284. Springer Berlin Heidelberg, 2012.
- [S2] Gregor Gabrysiak, Holger Giese, and Andreas Seibel. Towards Next-Generation Design Thinking II: Virtual Multi-User Software Prototypes. In Hasso Plattner, Christoph Meinel, and Larry Leifer, editors, *Design Thinking Research*, Understanding Innovation, pages 107–126. Springer Berlin Heidelberg, 2012.
- [S3] Holger Giese, Leen Lambers, Basil Becker, Stephan Hildebrandt, Stefan Neumann, Thomas Vogel, and Sebastian Wätzoldt. Graph Transformations for MDE, Adaptation, and Models at Runtime. In Marco Bernardo, Vittorio Cortellessa, and Alfonso Pierantonio, editors,

Formal Methods for Model-Driven Engineering, volume 7320 of *Lecture Notes in Computer Science (LNCS)*, pages 137–191. Springer Berlin / Heidelberg, June 2012.

- [S4] Holger Giese, Bernhard Rumpe, Bernhard Schätz, and Janos Sztipanovits. Science and Engineering of Cyber-Physical Systems (Dagstuhl Seminar 11441). *Dagstuhl Reports*, 1(11):1–22, 2012.
- [S5] Andreas Seibel, Regina Hebig, and Holger Giese. Traceability in Model-Driven Engineering: Efficient and Scalable Traceability Maintenance. In Jane Cleland-Huang, Orlena Gotel, and Andrea Zisman, editors, *Software and Systems Traceability*, pages 215–240. Springer London, 2012.
- [S6] Thomas Vogel and Holger Giese. Requirements and Assessment of Languages and Frameworks for Adaptation Models. In Jörg Kienzle, editor, *Models in Software Engineering, Workshops and Symposia at MoDELS 2011, Wellington, New Zealand, October 16-21, 2011, Reports and Revised Selected Papers*, volume 7167 of *Lecture Notes in Computer Science (LNCS)*, pages 167–182. Springer-Verlag, April 2012.

7.3 Begutachtete Konferenz- und Workshopartikel

- [K1] Moussa Amrani, Juergen Dingel, Leen Lambers, Levi Lúcio, Gehan Selim, Eugene Syriani, and Manuel Wimmer. Towards a Model Transformation Intent Catalog. In Juergen Dingel, Levi Lúcio, Hans Vangheluwe, and Daniel Varró, editors, *MoDELS Workshop on Analysis of Model Transformation*, pages 3–8. ACM, 2012.
- [K2] Gregor Gabrysiak, Markus Guentert, Regina Hebig, and Holger Giese. Teaching Requirements Engineering with Authentic Stakeholders: Towards a Scalable Course Setting. In *Proc. of ICSE 2012 Workshop on Software Engineering Education based on Real-World Experiences*, Zurich, Switzerland, June 2012.
- [K3] Gregor Gabrysiak, Regina Hebig, and Holger Giese. Decoupled Model-Based Elicitation of Stakeholder Scenarios. In *Proc. of the Seventh International Conference on Software Engineering Advances*, Lisbon, Portugal, November 2012. IARIA.
- [K4] Gregor Gabrysiak, Regina Hebig, and Holger Giese. Simulation-Assisted Elicitation and Validation of Behavioral Specifications for Multiple Stakeholders. In Sumitra Reddy and Khalil Drira, editors, *21st IEEE International WETICE conference (WETICE-2012)*, pages 220–225, Toulouse, France, June 2012. IEEE.
- [K5] Holger Giese and Leen Lambers. Towards Automatic Verification of Behavior Preservation for Model Transformation via Invariant Checking. In *Proceedings of Intern. Conf. on Graph Transformation (ICGT' 12)*, volume 7562 of *LNCS*, pages 249–263. Springer, 2012.
- [K6] Ulrike Golas, Leen Lambers, Hartmut Ehrig, and Holger Giese. Toward Bridging the Gap between Formal Foundations and Current Practice for Triple Graph Grammars - Flexible Relations between Source and Target Elements. In *Proceedings of Intern. Conf. on Graph Transformation (ICGT' 12)*, volume 7562 of *LNCS*, pages 141–155. Springer, 2012.

- [K7] Regina Hebig, Gregor Gabrysiak, and Holger Giese. Towards Patterns for MDE-Related Processes to Detect and Handle Changeability Risks. In *2012 International Conference on Software and Systems Process*, 2012.
- [K8] Stephan Hildebrandt, Leen Lambers, Basil Becker, and Holger Giese. Integration of Triple Graph Grammars and Constraints. In Christian Krause and Bernhard Westfechtel, editors, *Proceedings of the 7th International Workshop on Graph Based Tools (GraBaTs 2012)*, volume 52, pages 1–12. EC-EASST, 2012.
- [K9] Stephan Hildebrandt, Leen Lambers, and Holger Giese. The MDELab Tool Framework for the Development of Correct Model Transformations with Triple Graph Grammars. In Juergen Dingel, Levi Lúcio, Hans Vangheluwe, and Daniel Varró, editors, *MoDELS Workshop on Analysis of Model Transformation*, pages 33–34. ACM, 2012.
- [K10] Stephan Hildebrandt, Leen Lambers, Holger Giese, Dominic Petrick, and Ingo Richter. Automatic Conformance Testing of Optimized Triple Graph Grammar Implementations. In Andy Schürr, Daniel Varró, and Gergely Varró, editors, *Applications of graph Transformation with Industrial Relevance, 4th International Symposium, (AGTIVE 2011)*, volume 7233 of *LNCS*, pages 238–253. Springer, 2012.
- [K11] Alexander Krasnogolowy, Stephan Hildebrandt, and Sebastian Wätzoldt. Flexible Debugging of Behavior Models. In *2012 IEEE International Conference on Industrial Technology (ICIT)*, pages 331–336, March 2012.
- [K12] Christian Krause and Holger Giese. Probabilistic Graph Transformation Systems. In *Proceedings of Intern. Conf. on Graph Transformation (ICGT' 12)*, volume 7562 of *Lecture Notes in Computer Science*, pages 311–325. Springer-Verlag, 2012.
- [K13] Leen Lambers, Stephan Hildebrandt, Holger Giese, and Fernando Orejas. Attribute Handling for Bidirectional Model Transformations: The Triple Graph Grammar Case. In Frank Hermann and Janis Voigtländer, editors, *Proceedings of the First International Workshop on Bidirectional Transformations*, volume 49, pages 1–16. EC-EASST, 2012.
- [K14] A. Marconi, Antonio Bucchiarone, K. Bratanis, Antonio Brogi, Javier Camara, D. Dranidis, Holger Giese, R. Kazhamiakink, Rogério de Lemos, C.C. Marquezan, and Andreas Metzger. Research challenges on multi-layer and mixed-initiative monitoring and adaptation for service-based systems. In *Workshop on European Software Services and Systems Research - Results and Challenges (S-Cube)*, pages 40–46. IEEE Computer Society, June 2012.
- [K15] Stefan Neumann, Norman Kluge, and Sebastian Wätzoldt. Automatic Transformation of Abstract AUTOSAR Architectures to Timed Automata. In *Proceedings of the 5th International Workshop on Model Based Architecting and Construction of Embedded Systems, ACES-MB 12*, pages 55–60, New York, NY, USA, 2012. ACM.
- [K16] Henrik Steudel, Regina Hebig, and Holger Giese. A Build Server for Model-Driven Engineering. In *6th International Workshop on Multi-Paradigm Modeling (MPM 2012)*. ACM, 2012.
- [K17] Thomas Vogel and Holger Giese. A Language for Feedback Loops in Self-Adaptive Systems: Executable Runtime Megamodels. In *Proceedings of the 7th International Symposo-*

sium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS 2012), pages 129–138. IEEE Computer Society, June 2012.

- [K18] Sebastian Wätzoldt, Stefan Neumann, Falk Benke, and Holger Giese. Integrated Software Development for Embedded Robotic Systems. In Itsuki Noda, Noriaki Ando, Davide Brugali, and James Kuffner, editors, *Proceedings of the 3rd International Conference on Simulation, Modeling, and Programming for Autonomous Robots (SIMPAN)*, volume 7628 of *Lecture Notes in Computer Science*, pages 335–348. Springer Berlin Heidelberg, October 2012.

7.4 Bücher und Tagungsbände

- [B1] Andrew Fish and Leen Lambers, editors. *Proceedings of the 11th International Workshop on Graph Transformation and Visual Modeling Techniques (GTVMT 2012)*, volume 47 of *EC-EASST*, 2012.
- [B2] Holger Giese and Grigore Rosu, editors. *Formal Techniques for Distributed Systems: Joint 14th IFIP WG 6.1 International Conference, FMOODS 2012 and 32nd IFIP WG 6.1 International Conference, FORTE 2012, Stockholm, Sweden, June 13-16, 2012, Proceedings*, volume 7273 of *Lecture Notes in Computer Science (LNCS)*. Springer-Verlag, 2012.

7.5 Technische Berichte

- [TR1] Basil Becker and Holger Giese. Cyber-Physical Systems with Dynamic Structure: Towards Modeling and Verification of Inductive Invariants. Technical Report 64, Hasso Plattner Institute at the University of Potsdam, Germany, 2012.
- [TR2] Holger Giese, Stephan Hildebrandt, Stefan Neumann, and Sebastian Wätzoldt. Industrial Case Study on the Integration of SysML and AUTOSAR with Triple Graph Grammars. Technical Report 57, Hasso Plattner Institute at the University of Potsdam, September 2012.
- [TR3] Regina Hebig and Holger Giese. MDE Settings in SAP. A Descriptive Field Study. Technical Report 58, Hasso-Plattner Institut at University of Potsdam, 2012. (without cover page).
- [TR4] Markus von Detten, Christian Heinzemann, M. Platenius, Jan Rieke, J. Suck, Dietrich Travkin, and Stephan Hildebrandt. Story Diagrams - Syntax and Semantics. Technical report, Software Engineering Group, Heinz Nixdorf Institute, 2012.
- [TR5] Markus von Detten, Christian Heinzemann, M. Platenius, Jan Rieke, Dietrich Travkin, and Stephan Hildebrandt. Story Diagrams - Syntax and Semantics. Technical report, Software Engineering Group, Heinz Nixdorf Institute, 2012. Version 0.2.
- [TR6] Markus von Detten, Christian Heinzemann, M. Platenius, Jan Rieke, Dietrich Travkin, and Stephan Hildebrandt. Story Diagrams - Syntax and Semantics. Technical report, Software Engineering Group, Heinz Nixdorf Institute, 2012.

7.6 Miscellaneous

- [M1] Thomas Vogel. Models at Runtime for Self-Adaptive Software. Technical report, Proceedings of the Joint Workshop of the German Research Training Groups in Computer Science: Dagstuhl 2012, Books on Demand, Norderstedt, June 2012. (Extended Abstract).

8 Vorträge

8.1 Eingeladene Vorträge

Prof. Dr. Holger Giese

June 2012 *Graph Transformations for MDE, Adaptation, and Models at Runtime*. 12th International School on Formal Methods for the Design of Computer, Communication and Software Systems: Model-Driven Engineering (SFM-12:MDE), Bertinoro, Italy, June 18-23, 2012.

June 2012 *Model-Based Integration*. Volvo Cars, Torslanda, Sweden, June 13, 2012.

May 2012 *On Models, Model Operations, Mega Models, and Runtime Models*. Chalmers University of Technology, Gotheburg, Sweden, May 07, 2012.

Thomas Vogel

May 2012 *Engineering Self-Adaptive Software Systems with Runtime Models*. GI Dagstuhl Seminar on Quality-of-Service Attributes in Service- and Cloud-based Systems: Specification, Modelling, Monitoring, Prediction, and Optimisation, Seminar 12211, Schloss Dagstuhl, Wadern, Germany, May 20-25, 2012.

8.2 Vorträge auf Konferenzen und Workshops

Thomas Beyhl

February 2012 *Connecting Designing and Engineering Activities*. 8. HPDTRP Community Building Workshop, Stanford University, CA, USA, February 22, 2012.

June 2012 *Connecting Designing and Engineering Activities*. 9. HPDTRP Community Building Workshop, Hasso Plattner Institute, Potsdam, Deutschland, June 28, 2012.

September 2012 *Connecting Designing and Engineering Activities*. d.confestival 2012, Research Track, Hasso Plattner Institute, Potsdam, Deutschland, September 21, 2012.

Gregor Gabrysiak

November 2012 *Decoupled Model-Based Elicitation of Stakeholder Scenarios*. 7th International Conference on Software Engineering Advances (ICSEA'12), Lisbon, Portugal, November 19, 2012.

Regina Hebig

- April 2012 *Towards Guiding Evolution of Model-Driven Engineering with Focus on Changeability*. HPI Research School on Service-Oriented Systems Engineering Ph.D. Retreat, Potsdam, Germany, April 20, 2012.
- June 2012 *Towards Patterns for MDE-Related Processes to Detect and Handle Changeability Risks*. International Conference on Software and Systems Process, Zurich, Switzerland, June 02, 2012.
- June 2012 *Teaching Requirements Engineering with Authentic Stakeholders: Towards a Scalable Course Setting*. ICSE 2012 Workshop on Software Engineering Education based on Real-World Experiences, Zurich, Switzerland, June 09, 2012.
- June 2012 *Simulation-Assisted Elicitation and Validation of Behavioral Specifications for Multiple Stakeholders*. 3rd IEEE Track on Collaborative Modeling & Simulation (CoMetS'12) at 21st IEEE International WETICE conference (WETICE-2012), Toulouse, France, June 26, 2012.
- October 2012 *A Build Server for Model-Driven Engineering*. 6th International Workshop on Multi-Paradigm Modeling (MPM), Innsbruck, Austria, October 01, 2012. (Poster)
- October 2012 *On the Complex Nature of MDE Evolution - A Meta Study*. HPI Research School on Service-Oriented Systems Engineering Ph.D. Retreat, Rheinsberg, Germany, October 18, 2012.

Stephan Hildebrandt

- March 2012 *Flexible Debugging of Behavior Models*. 2012 IEEE International Conference on Industrial Technology (ICIT), Athens, Greece, March 18-22, 2012.
- September 2012 *Integration of Triple Graph Grammars and Constraints*. 7th International Workshop on Graph Based Tools (GraBaTs 2012), Bremen, Germany, September 24, 2012.
- October 2012 *The MDELab Tool Framework for the Development of Correct Model Transformations with Triple Graph Grammars*. MoDELS Workshop on Analysis of Model Transformations, Innsbruck, Austria, October 02, 2012.

Christian Krause

- October 2012 *Probabilistic Graph Transformation Systems*. International Conference on Graph Transformations (ICGT), Bremen, Germany, October 2012.

Leen Lambers

- March 2012 *Attribute Handling for Bidirectional Model Transformations: The Triple Graph Grammar Case*. First International Workshop on Bidirectional Transformations (BX 2012), Tallin, Estonia, March 2012.

September 2012 *Toward Bridging the Gap between Formal Foundations and Current Practice for Triple Graph Grammars*. International Conference on Graph Transformation (ICGT'12), Bremen, Germany, September 2012.

September 2012 *Towards Automatic Verification of Behavior Preservation for Model Transformation via Invariant Checking*. International Conference on Graph Transformation (ICGT'12), Bremen, Germany, September 2012.

Stefan Neumann

September 2012 *Automatic Transformation of Abstract AUTOSAR Architectures to Timed Automata*. International Workshop on Model Based Architecting and Construction of Embedded Systems (ACES-MB), Innsbruck, Austria, September 31, 2012.

Thomas Vogel

April 2012 *Executable Megamodels for Feedback Loops in Self-Adaptive Systems*. Spring Workshop and Ph.D. Retreat of the HPI Research School on Service-Oriented Systems Engineering, Potsdam, Germany, April 20, 2012.

June 2012 *A Language for Feedback Loops in Self-Adaptive Systems: Executable Runtime Megamodels*. 7th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS 2012), Zurich, Switzerland, June 4-5, 2012.

June 2012 *Engineering Self-Adaptive Software Systems with Runtime Models*. 7th Annual HPI Symposium on Future Trends in Service-Oriented Computing (FutureSOC 2012), Potsdam, Germany, June 14-15, 2012.

October 2012 *Models@run.time for Engineering Self-Adaptive Software*. Fall Workshop and Ph.D. Retreat of the HPI Research School on Service-Oriented Systems Engineering, Rheinsberg, Germany, October 18-19, 2012.

Sebastian Wätzoldt

April 2012 *Adaptation of Cyber-Physical Systems*. Spring Workshop and Ph.D. Retreat of the HPI Research School on Service-Oriented Systems Engineering, Potsdam, Germany, April 20, 2012.

October 2012 *Software Development for Embedded Robotic Systems*. Fall Workshop and Ph.D. Retreat of the HPI Research School on Service-Oriented Systems Engineering, Potsdam, Germany, October 19-20, 2012.

November 2012 *Integrated Software Development for Embedded Robotic Systems*. Simulation, Modeling, and Programming for Autonomous Robots Conference (SIMPAN), Tsukuba, Japan, November 2012.

December 2012 *Reconfiguration in Cyber-Physical Systems*. HPI Symposium, Walldorf, Germany, December 17, 2012.

9 Herausgeberschaft

9.1 Bücher und Tagungsbände

- [B1] Andrew Fish and Leen Lambers, editors. *Proceedings of the 11th International Workshop on Graph Transformation and Visual Modeling Techniques (GTVMT 2012)*, volume 47 of *EC-EASST*, 2012.
- [B2] Holger Giese and Grigore Rosu, editors. *Formal Techniques for Distributed Systems: Joint 14th IFIP WG 6.1 International Conference, FMOODS 2012 and 32nd IFIP WG 6.1 International Conference, FORTE 2012, Stockholm, Sweden, June 13-16, 2012, Proceedings*, volume 7273 of *Lecture Notes in Computer Science (LNCS)*. Springer-Verlag, 2012.

10 Web-Portale und -Services

10.1 Self-adaptive.org

Das Online-Angebot <http://www.self-adaptive.org> dient als Übersichtsseite für das jährliche Symposium *Software Engineering for Adaptive and Self-Managing Systems (SEAMS)* im Rahmen der *International Conference on Software Engineering (ICSE)*. Auf der Webseite sind alle Call for Papers für aktuelle und vergangene SEAMS Symposien, eine umfassende themenspezifische Bibliographie, Informationen zu weiterführenden Veranstaltungen wie den Dagstuhl Seminaren 08031 und 10431 sowie eine Liste von Wissenschaftlern, die auf dem Gebiet forschen, zu finden.

10.2 MDELab.org

Mit dem Online-Angebot <http://www.mdelab.org> informieren wir über Forschungsarbeiten unseres Fachgebiets im Bereich des *Model-Driven Engineering (MDE)*. Dabei liegt der Schwerpunkt auf Werkzeugen unter anderem für die modellgetriebene Softwareentwicklung, die an unserem Fachgebiet entwickelt werden und die zum Download bereitstehen.

10.3 CPSLab.org

Mit dem Online-Angebot <http://www.cpslab.org> informieren wir über Aktivitäten im Kontext unseres Labors im Bereich *Cyber-Physical-Systems*. Inhalte beziehen sich auf vergangene, aktuelle als auch geplante Forschungsarbeiten. Weiterhin werden ausgewählte Projekte, welche im Kontext der Lehre umgesetzt wurden, repräsentiert.

11 Mitgliedschaften, Programmkomitees und Gutachtertätigkeiten

11.1 Mitgliedschaften

Prof. Dr. Holger Giese

- Mitglied der Association for Computing Machinery (ACM)
- Mitglied der folgenden Special Interest Groups: SIGSOFT, SIGBED, SIGPLAN
- Mitglied der IEEE (Valued IEEE Member, Member since 1994)
- Mitglied der IEEE Computer Society
- Mitglied der folgenden Technical Councils: TCSE, TCDP, TCRTS, TFAAS
- Mitglied der IEEE Systems, Man, and Cybernetics Society
- Mitglied der Gesellschaft für Informatik e.V. (GI)
- Mitglied der folgenden Fachgebiete und Fachgruppen: ST, TAV, OOSE, ASE, PN, SPECS, FOMSESS
- Mitglied des Deutschen Hochschulverbandes (DHV)

Basil Becker

- Mitglied der Association for Computing Machinery (ACM)
- Mitglied der folgenden Special Interest Groups: SIGSOFT
- Mitglied der Gesellschaft für Informatik e.V. (GI)

Gregor Gabrysiak

- Mitglied der IEEE
- Mitglied der IEEE Computer Society

Regina Hebig

- Mitglied der Association for Computing Machinery (ACM)
- Mitglied der folgenden Special Interest Groups: SIGSOFT

Thomas Vogel

- Mitglied der Association for Computing Machinery (ACM)
- Mitglied der folgenden Special Interest Groups: SIGSOFT
- Mitglied der Gesellschaft für Informatik e.V. (GI)

Sebastian Wätzoldt

- Mitglied der Association for Computing Machinery (ACM)
- Mitglied der folgenden Special Interest Groups: SIGSOFT, SIGBED
- Mitglied der Gesellschaft für Informatik e.V. (GI)

11.2 Mitarbeit in Programmkomitees

Prof. Dr. Holger Giese

- 34th International Conference on Software Engineering (ICSE)
Zurich, Switzerland, June 2-9, 2012, [↗ website](#)
- 15th International Conference on Model Driven Engineering Languages and Systems (MODELS) – Foundations Track
Innsbruck, Austria, September 30 - October 5, 2012, [↗ website](#)
- 15th International Conference on Fundamental Approaches to Software Engineering (FASE)
Tallin, Estonia, March 25 - April 1, 2012, [↗ website](#)
- 10th International Conference on Business Process Management (BPM)
Tallin, Estland, September 3-6, 2012, [↗ website](#)
- 6th International Conference on Graph Transformation (ICGT)
Bremen, Germany, September 24-29, 2012, [↗ website](#)
- 31st International Conference on Computer Safety, Reliability and Security (SAFECOMP)
Magdeburg, Germany, September 25-28, [↗ website](#)
- 7th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS)
Zurich, Switzerland, June 4-5, 2012, [↗ website](#)
- 38th Euromicro Conference on Software Engineering and Advanced Applications (SEAA) – Track on Model-based development, Components and Services (MOCS)
Cesme, Izmir, Turkey, September 5-8, 2012, [↗ website](#)
- 3rd International Conference on Simulation, Modeling, and Programming for Autonomous Robots (SIMPAN)
Tsukuba, Japan, November 5-8, 2012, [↗ website](#)
- 13th International Conference on Industrial Technology (ICIT) – Track on Embedded and Cyberphysical Systems in Industrial Applications
Athens, Greece, March 19-21, 2012, [↗ website](#)
- 27th ACM Symposium on Applied Computing (SAC 2012) – 7th Track on Dependable and Adaptive Distributed Systems (DADS)
Riva del Garda (Trento), Italy, March 25-29, 2012, [↗ website](#)
- 7th International Workshop on Models@run.time (MRT)
Innsbruck, Austria, October 2, 2012, [↗ website](#)
- 6th International Workshop on Multi-Paradigm Modeling (MPM)
October 1, 2012, Innsbruck, Austria, [↗ website](#)
- Doctoral Symposium of the 6th International Conference on Graph Transformation (ICGT)
Bremen, Germany, September 26, 2012, [↗ website](#)
- 4th International Workshop on Modelling in Software Engineering (MiSE)
Zurich, Switzerland, June 2-3, 2012, [↗ website](#)
- 11th International Workshop on Graph Transformation and Visual Modeling Techniques (GT-VMT)
Tallin, Estland, March 24-25, 2012, [↗ website](#)

- 1st Workshop on the Analysis of Model Transformations (AMT)
Innsbruck, Austria, October 2, 2012, [↗ website](#)
- 11th International Workshop on Foundations of Coordination Languages and Self Adaptation (FOCLASA)
Newcastle, U.K., September 8, 2012, [↗ website](#)
- Model-based Methodologies for Pervasive and Embedded Software (MOMPES)
Essen, Germany, September 04, 2012, [↗ website](#)
- 13th International Workshop on Agent-Oriented Software Engineering (AOSE)
Valencia, Spain, June 4, 2012, [↗ website](#)
- Fachtagung Modellierung 2012
Bamberg, Germany, March 14-16, 2012, [↗ website](#)
- Workshop Modellierung in der Automatisierungstechnik (MAT)
March 14-16, 2012, Bamberg, Germany, [↗ website](#)

Regina Hebig

- Doctoral Symposium at MODELS Conference
Innsbruck, Austria, October 2, 2012

Dr. Leen Lambers

- 6th International Conference on Graph Transformation (ICGT) – Foundations Track
Bremen, Germany, September 24-29, 2012, [↗ website](#)
- 5th International Conference on Model Transformation (ICMT), co-located with TOOLS Europe
Prague, Czech Republic, May 28-29, 2012, [↗ website](#)
- 7th Workshop on Applied and Computational Category Theory (ACCAT), co-located with ETAPS
Tallin, Estonia

Dr. Christian Krause

- 14th International Conference on Coordination Models and Languages (COORDINATION)
Stockholm, Sweden

11.3 Organisation von Tagungen und Workshops

Prof. Dr. Holger Giese

- 8. Dagstuhl-Workshop: Modellbasierte Entwicklung eingebetteter Systeme (MBEES)
Schloss Dagstuhl, Wadern, Deutschland, February 2-8, 2012, [↗ website](#)
- Formal Techniques for Distributed Systems: Joint 14th IFIP WG 6.1 International Conference, FMOODS 2012 and 32nd IFIP WG 6.1 International Conference, FORTE 2012
Stockholm, Sweden, June 13-16, 2012, [↗ website](#)

Dr. Leen Lambers

- 11th International Workshop on Graph Transformation and Visual Modeling Techniques (GT-VMT)
Tallin, Estland, March 24-25, 2012, [↗ website](#)

Dr. Christian Krause

- 7th International Workshop on Graph Based Tools (GraBaTs)
Bremen, Germany, September 24, 2012, [↗ website](#)

11.4 Gutachtertätigkeiten**11.4.1 Forschungsprojekte****Prof. Dr. Holger Giese**

- European Research Council (ERC)
- European Union Seventh Framework Programme (EU FP7)
- Deutsche Forschungsgemeinschaft (DFG)
- Die niederländische Organisation für wissenschaftliche Forschung (NWO)
- Swedisch Knowledge Foundation (KK-stiftelsen)
- Austrian Science Fund (FWF)
- Natural Sciences and Engineering Research Council (NSERC) Canada

11.4.2 Zeitschriften und Magazine**Prof. Dr. Holger Giese**

- Science of Computer Programming (Zeitschrift)
- Transactions on Software Engineering and Methodology (Zeitschrift)
- Formal Aspects of Computing (Zeitschrift)
- IEEE Computer (Magazin)
- IEEE Robotics and Automation (Magazine)
- IEEE Software
- IEEE Transactions on Control Systems Technology
- IEEE Transactions on Industrial Informatics
- IEEE Transactions on Software Engineering
- Information and Software Technology
- Journal of Systems and Software (Zeitschrift)

- Journal of Visual Languages and Computing (Zeitschrift)
- Requirements Engineering (Zeitschrift)
- Simulation: Transactions of the Society for Modeling and Simulation International (Zeitschrift)
- Software Quality Journal (Zeitschrift)
- Software and Systems Modeling (Zeitschrift)
- Journal of Software Engineering for Robotics (JOSER)
- International Journal on Software Tools for Technology Transfer (STTT)
- International Journal of Aerospace Engineering (IJAE)
- Information Systems (IS) (Zeitschrift)
- Concurrency and Computation: Practice and Experience (Zeitschrift)
- Mechatronics (Zeitschrift)
- Algorithms (Zeitschrift)

Dr. Leen Lambers

- Journal of Visual Languages and Computing, Elsevier (Zeitschrift)
- Journal of Software and Systems Modeling, Springer, (Zeitschrift)
- Progress in Informatics, National Institute of Informatics, Japan, (Zeitschrift)
- Information and Software Technology, Elsevier, (Zeitschrift)

Dr. Christian Krause

- Algorithms (Zeitschrift)

Thomas Vogel

- Journal of Systems and Software (JSS), Special Issue on State of the Art in Self-Adaptive Software Systems (Zeitschrift)
- Software and Systems Modeling (SOSYM) (Zeitschrift)
- Computing (Zeitschrift)