

# **Modellgetriebene, agile Entwicklung und Evolution mehrbenutzerfähiger Enterprise Applikationen mit MontiEE**

Von der Fakultät für Mathematik, Informatik und Naturwissenschaften der  
RWTH Aachen University zur Erlangung des akademischen Grades  
eines Doktors der Naturwissenschaften genehmigte Dissertation

vorgelegt von

**Diplom-Informatiker Diplom-Wirtschaftsinformatiker  
Markus Look**

aus Düren-Lendersdorf

Berichter: Universitätsprofessor Dr. rer. nat. Bernhard Rumpe  
Universitätsprofessor Dr. rer. nat. Wilhelm Hasselbring

Tag der mündlichen Prüfung: 15. Dezember 2016

Diese Dissertation ist auf den Internetseiten der Universitätsbibliothek online verfügbar.



# **Aachener Informatik-Berichte, Software Engineering**

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

Band 27

**Markus Look**  
RWTH Aachen University

**Modellgetriebene, agile Entwicklung  
und Evolution mehrbenutzerfähiger  
Enterprise Applikationen mit MontiEE**

Shaker Verlag  
Aachen 2017

**Bibliografische Information der Deutschen Nationalbibliothek**

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

Zugl.: D 82 (Diss. RWTH Aachen University, 2016)

Copyright Shaker Verlag 2017

Alle Rechte, auch das des auszugsweisen Nachdruckes, der auszugsweisen oder vollständigen Wiedergabe, der Speicherung in Datenverarbeitungsanlagen und der Übersetzung, vorbehalten.

Printed in Germany.

ISBN 978-3-8440-5131-5

ISSN 1869-9170

Shaker Verlag GmbH • Postfach 101818 • 52018 Aachen  
Telefon: 02407 / 95 96 - 0 • Telefax: 02407 / 95 96 - 9  
Internet: [www.shaker.de](http://www.shaker.de) • E-Mail: [info@shaker.de](mailto:info@shaker.de)

# Kurzfassung

Um Geschäftsprozesse eines Unternehmens in der IT abzubilden, werden Enterprise Applikationen eingesetzt. Wesentliche Funktionalitäten dieser sind Speicherung und Bereitstellung von Daten für Benutzer mit unterschiedlichen Rollen und heterogenen Clients. Enterprise Applikationen können aus mehreren Servern mit unterschiedlichen Aufgaben bestehen und von unterschiedlichen Clients verwendet werden. Kommunikation findet zwischen Applikationsservern, zwischen diesen und Datenbanken, aber auch zwischen mehreren Clients und einzelnen Applikationsservern statt. Sowohl die Kommunikationstechnologie als auch der Inhalt muss den Beteiligten bekannt sein. Die Kommunikation ist meistens durch den Client initiiert, so dass die konkreten Clients dem Applikationsserver unbekannt sind. Der Applikationsserver kann unter Verwendung verschiedener Kommunikationstechnologien und -formate clientspezifische Schnittstellen bereitstellen.

Im Software Engineering werden Modelle zu unterschiedlichen Zwecken, wie zur Spezifikation und zur Codegenerierung, eingesetzt. Durch häufig wiederkehrende Aufgaben bei der Entwicklung und der Verwendung einer Schichtenarchitektur eignet sich der Einsatz von Generatoren gut zur Umsetzung von Enterprise Applikationen. Im Rahmen dieser Dissertation wird mit MontiEE die modellgetriebene, agile Entwicklung und Evolution mehrbenutzerfähiger Enterprise Applikationen durch Generatoren unterstützt. MontiEE enthält eine Sprachfamilie zur Modellierung sowie verschiedene Generatoren zur Umsetzung von Enterprise Applikationen. MontiEE unterstützt die Modellierung der Datenpersistenz, der Mehrbenutzerfähigkeit und heterogener Clients. Durch die Generatoren kann von benötigten technologiespezifischen Informationen abstrahiert werden, so dass sich vollständig auf die Entwicklung der Geschäftslogik fokussiert werden kann. Darüber hinaus werden Sprachen zur Modellierung der Evolution und Generatoren zur Datenmigration zur Verfügung gestellt.

Die wichtigsten Ergebnisse dieser Arbeit sind:

- Eine Methodik zur Ableitung domänenspezifischer Sprachen zur schematischen Anreicherung von Modellen mit technologiespezifischen Informationen.
- Eine Sprachfamilie zur Modellierung von Mehrbenutzerfähigkeit, zur Berücksichtigung heterogener Clients und zur Modellierung von Evolution.
- Generatoren zur Generierung weiter Teile, wie Persistenz, Kommunikations- und Migrationsinfrastruktur, von Enterprise Applikationen.
- Eine Methodik zur Entwicklung von Enterprise Applikationen mit MontiEE.

MontiEE wurde in verschiedenen Fallstudien eingesetzt, in denen gezeigt werden konnte, dass MontiEE insgesamt ein umfassendes Framework zur agilen Modellierung, Generierung und Evolution mehrbenutzerfähiger Enterprise Applikationen ist. MontiEE unterstützt sowohl die Entwicklung als auch die Weiterentwicklung von Enterprise Applikationen.



# Abstract

To reflect a company's business processes in IT Enterprise Applications are put in place. The major functionality of such applications is persisting and offering data to different users with different roles using heterogeneous clients. Enterprise Applications may consist of several servers. Clients may use these servers which are fulfilling different tasks. Communication takes place between application servers, between application servers and databases as well as between multiple clients and a single application server. The communication technology as well as the content has to be known to all participants. Usually, communication is initiated by clients unknown to the application server. The application server can offer specific interfaces to various clients, enabling different communication technologies and formats.

Software Engineering uses models for different purposes, such as specification and code generation. Due to repeatedly reoccurring tasks during the development of enterprise applications and the utilisation of a tiered architecture the use of code generators does fit the implementation of enterprise applications well. In this dissertation MontiEE supports the model-driven, agile development of multi-user Enterprise Applications through code generators. MontiEE contains a language family for modelling as well as various generators for implementing Enterprise Applications. MontiEE supports modelling data persistency, multi-user capability, and heterogeneous clients. Because of the code generators an abstraction is made possible so that the developer can completely focus on implementing the business logic. Furthermore, languages for modelling evolution and generators for data migration are provided.

The main contributions of this thesis are:

- A methodology for deriving domain specific languages for a systematic enrichment of models with technology specific information.
- A language family for modelling multi-user capability, for considering heterogeneous clients, and for modelling evolution.
- Generators for generating major parts of Enterprise Applications, such as persistency, communication and migration infrastructure.
- A methodology for developing Enterprise Applications with MontiEE.

MontiEE has been used in different case studies showing that overall MontiEE offers a comprehensive framework for the agile modelling and generating of multi-user Enterprise Applications. MontiEE supports development as well as advancement of Enterprise Applications.



# Danksagung

An dieser Stelle möchte ich mich bei all den Menschen bedanken, die mich während meiner Promotion begleitet und unterstützt haben. Sie alle haben wesentlich zum Gelingen beigetragen.

Mein erster Dank gilt Prof. Dr. Bernhard Rumpe für die Möglichkeit der Promotion am Lehrstuhl Software Engineering der RWTH Aachen. Seine Betreuung, seine konstruktiven Diskussionen und seine wertvollen Ratschläge formten die Arbeit maßgeblich. Neben der abwechslungsreichen und spannenden akademischen Arbeit wurden mir auch tiefe Einblicke und vielschichtige Erfahrungen in der Praxis, bei der Erstellung von Projektanträgen, der Durchführung dieser und der Leitung der großartigen Arbeitsgruppe *Energie* ermöglicht.

Darüber hinaus möchte ich Prof. Dr. Wilhelm Hasselbring für die Bereitschaft zur Übernahme der Zweitkorrektur danken. Auch Prof. Dr. Ir. Joost-Pieter Katoen möchte ich für die Leitung sowie Prof. Dr. Urlik Schroeder für die Mitarbeit in meinem Prüfungskomitee danken.

Auch meinen Kollegen am Lehrstuhl möchte ich großen Dank aussprechen. Die vielen Diskussionen, spannenden Forschungsvorhaben und gemeinsamen Errungenschaften, aber auch gemeinsame Freizeitaktivitäten haben mir immer viel Freude bereitet. Wir haben viel erlebt, viel gearbeitet und viel gelacht. Es war mir eine Freude, die letzten Jahre mit euch zusammen arbeiten und auch feiern zu dürfen. Daher möchte ich Kai Adam, Vincent Bertram, Marita Breuer, Lennart Bucher, Arvid Butting, Gereon Bürvenich, Robert Eikermann, Timo Greifenberg, Dr. Arne Haber, Lars Hermerschmidt, Dr. Christoph Herrmann, Andreas Horst, Katrin Hölldobler, Oliver Kautz, Thomas Kurpick, Evgeny Kusmenko, Achim Lindt, Ben Mainz, Klaus Müller, Antonio Navarro Pérez, Dr. Pedram Mir Seyed Nazari, Sebastian Oberhoff, Jerome Pfeiffer, Dr. Claas Pinkernell, Dimitri Plotnikov, Manuel Pützer, Dr. Dirk Reiss, Dr. Holger Rendel, Dr. Jan Oliver Ringert, Alexander Roth, Dr. Martin Schindler, Christoph Schulze, Brian Sinkovec, Minh Tran, Galina Volkova, Max Voß, Michael von Wenckstern und Dr. Andreas Wortmann herzlich für die letzten Jahre danken.

Ganz besonders erwähnen möchte ich die Mitglieder der Arbeitsgruppe *Energie*. Es hat stets sehr viel Spaß gemacht, gemeinsam mit euch Projekte zu bearbeiten und neue zu akquirieren. Ebenfalls hervorheben möchte ich die vielen hilfsbereiten Freunde und Kollegen, die sich bereit erklärt haben, die Arbeit Korrektur zu lesen und Verbesserungsvorschläge einzubringen. Danke Andreas, Claas, Dimitri, Ingrid, Katrin, Robert, Ruth, Timo.

Darüber hinaus möchte ich mich bei den Kollegen bedanken, die immer ein offenes Ohr für mich hatten, stets meinen unterschiedlichsten Anliegen Aufmerksamkeit geschenkt haben und mir mit Rat zur Seite standen. Auch allen beteiligten Hiwis, Studenten, Abschlussarbeitern, die ich betreuend unterstützen durfte, möchte ich danken. Ihr alle,

Kollegen, Freunde und Studenten, habt mir sehr geholfen.

Großen Dank möchte ich meinen Eltern, Ingrid und Norbert, die mich fortwährend unterstützt haben, aussprechen. Sie haben mir diese Promotion und den Weg dorthin erst ermöglicht und waren jederzeit für mich da. Bei meinen Schwiegereltern Petra und Alfred möchte ich mich ebenfalls bedanken. Auch auf ihre Unterstützung konnte ich immer bauen.

Besonderer Dank gilt meiner Frau Ruth Maria, die mich stets unterstützt und motiviert hat. Ihre Unterstützung, ihr Zuspruch und ihre Geduld haben maßgeblich zum Erfolg dieser Arbeit beigetragen.

Aachen, Februar 2017  
Markus Look

# Inhaltsverzeichnis

<b>I</b>	<b>Grundlagen</b>	<b>1</b>
<b>1</b>	<b>Einführung</b>	<b>3</b>
1.1	Wichtigste Ziele der Arbeit . . . . .	6
1.2	Wichtigste Ergebnisse der Arbeit . . . . .	7
1.3	Aufbau der Arbeit . . . . .	8
<b>2</b>	<b>Grundlagen der modellgetriebenen Entwicklung</b>	<b>11</b>
2.1	Domänenspezifische Sprachen . . . . .	16
2.2	Die Language Workbench MontiCore . . . . .	18
2.2.1	MontiCore Grammatiken . . . . .	19
2.2.2	Generierung der AST-Klassen . . . . .	22
2.2.3	Sprachintegrationsmechanismen in MontiCore . . . . .	23
2.2.4	Kontextbedingungen in MontiCore . . . . .	24
2.3	Die Sprachfamilie UML/P . . . . .	29
2.3.1	Klassendiagramme . . . . .	30
2.3.2	Objektdiagramme . . . . .	33
2.4	Zusammenfassung . . . . .	35
<b>3</b>	<b>Entwicklung von Enterprise Applikationen</b>	<b>37</b>
3.1	Enterprise Applikationen . . . . .	37
3.2	Architektur von Enterprise Applikationen . . . . .	40
3.2.1	Clientkommunikation . . . . .	42
3.2.2	Datenbankkommunikation . . . . .	48
3.3	Implementierung von Enterprise Applikationen . . . . .	50
3.4	Verwandte Frameworks . . . . .	53
3.5	Szenario . . . . .	62
3.5.1	Benutzung des Systems . . . . .	64
3.5.2	Entwicklung des Systems . . . . .	67
3.6	Zusammenfassung . . . . .	72
<b>II</b>	<b>Die Sprachfamilie MontiEE</b>	<b>75</b>
<b>4</b>	<b>Modellierung der Persistenz</b>	<b>77</b>
4.1	Überblick . . . . .	78
4.2	Das Domänenmodell . . . . .	80

4.3	Taggingssprachen . . . . .	82
4.3.1	Die Tagdefinitionssprache . . . . .	85
4.3.2	Die Tagschemasprache . . . . .	91
4.3.3	Kontextbedingungen . . . . .	97
4.4	Zusammenfassung . . . . .	100
<b>5</b>	<b>Modellierung der Kommunikation</b>	<b>101</b>
5.1	Überblick . . . . .	102
5.2	Sichten auf Klassendiagramme . . . . .	103
5.2.1	Die Sichtensprache . . . . .	105
5.2.2	Kontextbedingungen . . . . .	109
5.2.3	Transformation von Sichten in Klassendiagramme . . . . .	113
5.3	Rollenbasierte Zugriffskontrolle . . . . .	116
5.3.1	Die Rollensprache . . . . .	119
5.3.2	Kontextbedingungen der Rollensprache . . . . .	120
5.3.3	Die Rechtesprache . . . . .	122
5.3.4	Automatisierte Ableitung eines Rechtediagramms . . . . .	125
5.3.5	Kontextbedingungen der Rechtesprache . . . . .	127
5.3.6	Die Mappingsprache . . . . .	128
5.3.7	Kontextbedingungen der Mappingsprache . . . . .	131
5.4	Zusammenfassung . . . . .	133
<b>6</b>	<b>Modellierung der Evolution</b>	<b>135</b>
6.1	Überblick . . . . .	136
6.2	Modellevolution . . . . .	136
6.2.1	Die Deltasprache . . . . .	140
6.2.2	Kontextbedingungen . . . . .	149
6.3	Zusammenfassung . . . . .	150
<b>III</b>	<b>MontiEE-Generatoren</b>	<b>151</b>
<b>7</b>	<b>Generierung der Persistenz</b>	<b>153</b>
7.1	Überblick . . . . .	154
7.2	Die MontiEE Tagtypen . . . . .	157
7.3	Generierung und Befüllung der Infrastruktur für Tags . . . . .	165
7.3.1	Generierung der Infrastruktur . . . . .	165
7.3.2	Befüllung der Infrastruktur . . . . .	168
7.4	Der Entity-Generator . . . . .	170
7.4.1	Abbildung der Modellierungskonzepte auf das Generat . . . . .	170
7.4.2	Generaterweiterungen . . . . .	184
7.4.3	Kontextbedingungen . . . . .	186
7.5	Der DAO-Generator . . . . .	190
7.5.1	Abbildung der Modellierungskonzepte auf das Generat . . . . .	191

7.6	Der SQL-Generator . . . . .	196
7.6.1	Abbildung der Modellierungskonzepte auf das Generat . . . . .	197
7.6.2	Kontextbedingungen . . . . .	204
7.7	Zusammenfassung . . . . .	204
<b>8</b>	<b>Generierung der Kommunikationsinfrastruktur</b>	<b>207</b>
8.1	Überblick . . . . .	207
8.2	Der DTO-Generator . . . . .	211
8.2.1	Abbildung der Modellierungskonzepte auf das Generat . . . . .	212
8.2.2	Generierung der Requests . . . . .	214
8.2.3	Generierung des DTO-Assemblers . . . . .	218
8.3	Der BusinessAPI-Generator . . . . .	220
8.3.1	Abbildung der Modellierungskonzepte auf das Generat . . . . .	220
8.4	Der Facade-Generator . . . . .	224
8.4.1	Abbildung der Modellierungskonzepte auf das Generat . . . . .	225
8.5	Zusammenfassung . . . . .	232
<b>9</b>	<b>Generierung der Evolutionsinfrastruktur</b>	<b>235</b>
9.1	Überblick . . . . .	235
9.2	Evolution des Klassendiagramms . . . . .	238
9.3	Migration der Objektdiagramme . . . . .	242
9.4	Serialisierung und Deserialisierung von Objektdiagrammen . . . . .	246
9.5	Zusammenfassung . . . . .	250
<b>10</b>	<b>Eine MontiEE-basierte Methodik</b>	<b>251</b>
10.1	Methodik . . . . .	252
10.2	Konfiguration und Ausführung von MontiEE . . . . .	256
10.2.1	Konfiguration . . . . .	257
10.2.2	Ausführung . . . . .	260
10.3	Anwendung der MontiEE-basierten Methodik auf das Szenario . . . . .	264
10.3.1	Konfiguration von MontiEE . . . . .	265
10.3.2	Generierung der Entitäten . . . . .	266
10.3.3	Generierung der DAOs . . . . .	270
10.3.4	Generierung der SQL-Skripte . . . . .	271
10.3.5	Modellierung der Sichten . . . . .	271
10.3.6	Generierung der DTOs . . . . .	272
10.3.7	Modellierung der Autorisierung . . . . .	273
10.3.8	Generierung der Fassaden . . . . .	274
10.4	Deployment des Generats . . . . .	275
10.4.1	Erstellung der Konfigurationsdateien . . . . .	277
10.5	Werkzeugunterstützung . . . . .	281
10.6	Fallstudien . . . . .	283
10.6.1	Integration in bestehende Systeme . . . . .	283
10.6.2	Verwendung in Neuentwicklungen . . . . .	288

10.7 Zusammenfassung . . . . .	293
<b>IV Epilog</b>	<b>295</b>
11 Zusammenfassung und Ausblick	297
Literaturverzeichnis	303
<b>V Anhänge</b>	<b>327</b>
<b>A Markierungen in Abbildungen und Listings</b>	<b>329</b>
<b>B Abkürzungen</b>	<b>331</b>
<b>C Modelle und Grammatiken</b>	<b>335</b>
C.1 Vollständiges Klassendiagramm des sozialen Netzwerks . . . . .	335
C.2 Grammatik des sprachunabhängigen Teils der Tagdefinitionssprache . . .	336
C.3 Grammatik des klassendiagrammspezifischen Teils der Tagdefinitionssprache . . . . .	339
C.4 Grammatik des sprachunabhängigen Teils der Tagschemasprache . . . . .	340
C.5 Grammatik des klassendiagrammspezifischen Teils der Tagschemasprache	342
C.6 Grammatik der Sichtensprache . . . . .	342
C.7 Grammatik der Rollensprache . . . . .	349
C.8 Grammatik der Rechtesprache . . . . .	350
C.9 Grammatik der Mappingsprache . . . . .	352
C.10 Grammatik der Deltasprache . . . . .	353
C.11 Grammatik der erweiterten Deltasprache . . . . .	359
<b>D Lebenslauf</b>	<b>367</b>
<b>Abbildungsverzeichnis</b>	<b>369</b>
<b>Listingsverzeichnis</b>	<b>375</b>
<b>Tabellenverzeichnis</b>	<b>383</b>