Softwareunterstützung für adaptive eHome-Systeme

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

Daniel Retkowitz

aus Münster in Westfalen

Berichter: Universitätsprofessor Dr.-Ing. Manfred Nagl
Universitätsprofessor Dr. rer. nat. Kurt Geihs
Universitätsprofessor Dr. rer. nat. Bernhard Rumpe

Tag der mündlichen Prüfung: 13. Januar 2010

Aachener Informatik-Berichte, Software Engineering

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

Band 3

Daniel Retkowitz

Softwareunterstützung für adaptive eHome-Systeme

Shaker Verlag
Aachen 2010

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, 2010)

Copyright Shaker Verlag 2010

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

Printed in Germany.

ISBN 978-3-8322-9246-1

ISSN 1869-9170

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

Zusammenfassung

eHomes sind Umgebungen, die sich durch eine Vernetzung der vorhandenen Geräte auszeichnen, und in denen komplexe geräteübergreifende Funktionalitäten in Form von Diensten angeboten werden. Übergreifende Funktionalitäten gehören im Automobil bereits heute zum Standard, in Wohnumgebungen sind sie jedoch bislang nicht verbreitet. Aber auch hier gibt es verschiedenste Anwendungsfelder für solche Funktionalitäten, z. B. in den Bereichen Sicherheit, Komfort, Multimedia, Kommunikation oder auch bei der medizinischen Überwachung und der Unterstützung von Senioren. Daher werden eHomes in Zukunft an Bedeutung gewinnen.

Für die Entwicklung von eHomes müssen eine Reihe von Herausforderungen überwunden werden. Ein wesentlicher Aspekt ist die Berücksichtigung von *Mobilität* und *Dynamik* in eHomes, die sich sowohl auf die Bewohner als auch auf die Geräteumgebung beziehen. Da in eHomes eine Vielzahl von Standards Anwendung finden, muss außerdem der daraus resultierenden *Heterogenität* Rechnung getragen werden, um die Interoperabilität der Dienste zu gewährleisten. Damit eHomes kostengünstig realisiert werden können, muss insbesondere eine geeignete *Infrastruktur* bereitgestellt werden, die die Entwicklung von Diensten und die Interaktion mit dem eHome-System vereinfacht. Sowohl für die Dienstentwicklung als auch für den Betrieb von eHomes sind daher geeignete *Softwarewerkzeuge* erforderlich.

Bisher wurde am Lehrstuhl für Informatik 3 (Softwaretechnik) der RWTH Aachen bereits ein Ansatz erarbeitet, der die Entwicklung von eHome-Diensten durch den Einsatz wiederverwendbarer Standardkomponenten für die Dienstimplementierung vereinfacht. Im sogenannten SCD-Prozess, der sich aus den Phasen *Spezifizierung*, *Konfigurierung* und *Deployment* zusammensetzt, werden die Benutzerwünsche erfasst, und es wird darauf basierend eine eHome-spezifische Dienstkomposition erstellt und dann zur Ausführung gebracht. Ein entsprechendes Werkzeug unterstützt diesen Prozess.

Die vorliegende Arbeit erweitert die bisherigen Konzepte durch Mechanismen zur dynamischen Adaption. Dabei wird der bisherige SCD-Prozess, der die Installationsphase eines eHomes abdeckt, zu einem *kontinuierlichen SCD-Prozess* erweitert, der eine Unterstützung im laufenden Betrieb des eHomes ermöglicht.

Die *strukturelle Adaption* dient der dynamischen Anpassung der Dienstkomposition. Änderungen, die sich aus Mobilität und Dynamik ergeben, können so zur Laufzeit berücksichtigt werden. Damit dies soweit möglich automatisch durchgeführt werden kann, werden Erweiterungen der Dienstspezifikation eingeführt, die es dem Entwickler erlauben, das spätere Bindungsverhalten sowie mögliche Einschränkungen der Bindungen festzulegen. Durch die neu eingeführten Mechanismen kann die Mehrfachentwicklung von Querschnittsfunktionalitäten zur personen- und kontextbezogenen Konfigurierung vermieden werden, was den Aufwand der Dienstentwicklung reduziert.

Die *semantische Adaption* ermöglicht die Überwindung syntaktischer Inkompatibilitäten, die sich aus der Heterogenität in eHomes ergeben. Dazu wird die Dienstspezifikation um eine semantische Beschreibung ergänzt, die die syntaktischen Elemente der Dienstschnittstellen auf semantische Konzepte einer domänenspezifischen Ontologie abbildet. Dies schafft die Grundlage für eine Dienstkomposition auf Basis semantisch übereinstimmender Funktionalitäten. Ein im Rahmen der vorliegenden Arbeit entwickelter Mechanismus zur automatischen Adaptergenerierung ermöglicht die praktische Umsetzung des Ansatzes.

Die Ergebnisse dieser Arbeit liefern einen wissenschaftlichen Beitrag in verschiedenen Bereichen. Der verfolgte Ansatz ist spezifisch auf die *Anwendungsdomäne eHomes* ausgerichtet. Darin unterscheidet er sich von anderen Ansätzen, die sich häufig mit allgemeinen Fragen der dynamischen Dienstkomposition befassen. So ist eine gezieltere Unterstützung der Besonderheiten im eHome möglich, z. B. bei der personen- und kontextbezogenen Konfigurierung. Der kontinuierliche SCD-Prozess wird auf Basis eines globalen Graphmodells in einem modellgetriebenen Entwicklungsprozess realisiert. Die *graphbasierten Techniken* ermöglichen dabei den einfachen Zugriff auf alle relevanten Daten. Das Graphmodell wird außerdem für die Werkzeugentwicklung genutzt. Ein weiterer wichtiger Beitrag dieser Arbeit ist der Laufzeitmechanismus zur *dynamischen Adaptergenerierung* auf Basis der semantischen Dienstbeschreibung. Dieser zeigt, wie mittels struktureller Reflection in Java eine Adaption zwischen syntaktisch inkompatiblen Schnittstellen erreicht werden kann. Weitere Ergebnisse sind ein Ansatz zur *Laufzeitüberwachung der Dienstkommunikation*, für dessen Realisierung auf die aspektorientierte Programmierung zurückgegriffen wird, und die Entwicklung von *Werkzeugen*, die sowohl die Dienstentwicklung als auch die Laufzeit des eHome-Systems abdecken.

Danksagung

Zum Entstehen der vorliegenden Arbeit haben zahlreiche Menschen beigetragen, bei denen ich mich an dieser Stelle für ihre Unterstützung bedanken möchte.

Zunächst gilt mein besonderer Dank Professor Manfred Nagl, der meine Arbeit als Doktorvater betreut und mir die Promotion an seinem Lehrstuhl ermöglicht hat. Seine Unterstützung meines Promotionsvorhabens, die anregenden Diskussionen aber auch der mir gewährte Freiraum haben meine Promotion zu einer lehrreichen Erfahrung gemacht. Ebenso danke ich Professor Kurt Geihs für sein Interesse an meiner Arbeit und die Übernahme des Zweitgutachtens. Professor Bernhard Rumpe gilt mein Dank dafür, dass er sich als weiterer Gutachter zur Verfügung gestellt hat.

Außerdem danke ich allen Mitgliedern des eHome-Projekts, die im Laufe der Zeit an diesem spannenden Thema mitgearbeitet und so zum Gelingen der Arbeit beigetragen haben. Besonderer Dank gilt meinem Kollegen Ibrahim Armaç für die vielen fruchtbaren Diskussionen, die richtungsweisenden Anregungen beim Korrekturlesen dieser Arbeit und die sehr gute Zusammenarbeit in all den Jahren. Meinen ehemaligen Kollegen im eHome-Projekt Michael Kirchhof und Ulrich Norbistrath danke ich für die Unterstützung in der Anfangsphase und ihre Vorarbeiten, die die Basis der vorliegenden Arbeit geliefert haben. Allen Studentinnen und Studenten, die im eHome-Projekt mitgearbeitet haben, danke ich für ihren Beitrag zum Erfolg des Projekts und damit auch dieser Arbeit. Mein besonderer Dank gilt dabei Ralf Frotscher, Marvin Hoffmann, Sven Kulle, Monika Pienkos, Mark Stegelmann und Chengzhi Xue. Die vielen intensiven Gespräche mit ihnen haben bedeutenden Einfluss auf die verschiedenen Themenbereiche dieser Arbeit genommen.

Während meiner Zeit am Lehrstuhl haben mich zahlreiche Kollegen begleitet. Ihnen allen möchte ich an dieser Stelle für das hervorragende Arbeitsklima und die vielen gemeinsamen Erlebnisse danken. Es war eine spannende und schöne Zeit, die mir dauerhaft in Erinnerung bleiben wird. Besonderer Dank gilt meinem ehemaligen Kollegen, Kommilitonen und Nachbarn Christof Mosler für die unterhaltsamen gemeinsamen Unternehmungen, seine tatkräftige Unterstützung sowie die kritische Durchsicht dieser Arbeit. Bodo Kraft danke ich für die vielen lehrreichen Ratschläge, die mir die Anfangszeit am Lehrstuhl erleichtert haben,

und die gemeinsamen Konferenzreisen, die immer eine bereichernde Erfahrung während meiner Promotion waren. Auch den weiteren „alten“ Kollegen Simon Becker, Boris Böhlen, Christian Fuß, Thomas Haase, Markus Heller und Ulrike Ranger möchte ich für die vielen Tipps zur Promotion danken, die mir eine wertvolle Hilfe waren. Thomas Heer, Cem Mengi, Theresa Körtgen, Erhard Weinell und René Würzberger danke ich für die gemeinsame Zeit, die fachlichen Gespräche und nicht zuletzt auch die vielen Erlebnisse außerhalb des akademischen Alltags. Allen „neuen“ Mitarbeitern, die zusammen mit Professor Bernhard Rumpel an den Lehrstuhl für Informatik 3 nach Aachen gekommen sind, wünsche ich viel Erfolg bei ihren Promotionsvorhaben und eine gute und lehrreiche Zeit.

Nicht zuletzt danke ich meiner Familie für die stete Unterstützung. Auf meine Eltern Klaus und Maria Retkowitz konnte ich immer zählen. Ihr Rückhalt hat mir geholfen, meinen Weg zu gehen. Darüber hinaus gilt meiner Mutter, Maria Retkowitz, besonderer Dank für die Korrekturhinweise zur vorliegenden Arbeit.

Meiner Freundin Nadine danke ich ganz herzlich für ihre Unterstützung und Geduld auch in den schwierigeren Phasen der Promotion. Sie hat jederzeit an mich geglaubt und mir den Ausgleich und das Glück während der vergangenen Jahre gegeben. Ohne ihre Unterstützung wäre diese Arbeit nicht möglich gewesen.

Aachen, Januar 2010

Daniel Retkowitz

Inhaltsverzeichnis

1	Einführung	1
1.1	Motivation	4
1.2	Zielsetzung	6
1.3	Projektkontext	10
1.4	Lösungsansatz	12
1.5	Gliederung der Arbeit	18
2	eHome-Systeme	21
2.1	Hintergrund und Terminologie	21
2.2	Anforderungen aus Benutzersicht	29
2.3	Technische Herausforderungen	34
2.4	Anwendungsbeispiele	38
2.4.1	Komfortszenario	38
2.4.2	Sicherheitsszenario	43
2.4.3	Unterstützung von Senioren	45
2.4.4	Allgemeine Dienste	45
2.5	eHome-Systeme in der Praxis	46
2.6	Zusammenfassung	50
3	Grundlagen	53
3.1	Komponentenbasierte Software	53
3.1.1	Begriffsklärung	54
3.1.2	Anwendung von Komponenten	56
3.2	Serviceorientierte Architektur	58
3.2.1	Von Komponenten zu Services	59
3.2.2	Aufbau serviceorientierter Systeme	60

Inhaltsverzeichnis

3.2.3	Webservices	62
3.2.4	Die OSGi Service Plattform	64
3.3	Modellgetriebene Softwareentwicklung	70
3.3.1	Graphersetzungssysteme	71
3.3.2	UML-basierte Ansätze	74
3.4	Vorarbeiten	76
3.4.1	Klassischer Entwicklungsprozess	77
3.4.2	Konfigurierungsansatz	78
3.4.3	Dienstkonzept	83
3.4.4	Modellierungsansatz	88
3.4.5	Werkzeugunterstützung	90
3.4.6	Bewertung	91
3.5	Zusammenfassung	96
4	Strukturelle Adaption	97
4.1	Mobilität und Dynamik in eHome-Systemen	97
4.1.1	Einflussfaktoren	98
4.1.2	Adaptionsansatz	102
4.2	Erweiterter Entwicklungsprozess	104
4.3	Dienstentwicklung	107
4.3.1	Diensttypen	108
4.3.2	Bindungsbeschränkungen	112
4.3.3	Bindungsstrategien	129
4.3.4	Bindungstypen	132
4.4	Kontinuierlicher SCD-Prozess	140
4.4.1	Spezifizierung	140
4.4.2	Konfigurierung	149
4.4.3	Deployment	159
4.4.4	Ausführung	176
4.5	Verwandte Arbeiten	183
4.6	Zusammenfassung	196
5	Semantische Adaption	199
5.1	Heterogenität in eHome-Systemen	199
5.1.1	Problemfelder	201
5.1.2	Zielsetzung des semantischen Ansatzes	202

5.2	Semantik und Ontologien	204
5.2.1	Begriffsklärung	204
5.2.2	Semantikbasierte Komposition	210
5.2.3	Adaption	214
5.3	Teilprozess der semantischen Adaptierung	218
5.4	Semantikdefinition	222
5.4.1	Ontologiesprachen	224
5.4.2	Struktur der Darstellungsontologie	226
5.4.3	Anwendung zur Ontologiemodellierung	230
5.5	Semantische Abbildung	234
5.5.1	Dienstschnittstellen	235
5.5.2	Inkompatibilitäten	238
5.5.3	Transformationen	240
5.6	Semantisches Matching	245
5.7	Adaptierung	251
5.7.1	Adaptierbarkeit von Komponenten	251
5.7.2	Funktionalitätsüberdeckung	253
5.7.3	Aktive und passive Komponenten	260
5.7.4	Adaptergenerierung	263
5.8	Verwandte Arbeiten	270
5.9	Zusammenfassung	284
6	Realisierung	287
6.1	Systemübersicht	287
6.2	Technische Grundlagen	289
6.3	Werkzeugunterstützung	292
6.3.1	Service-Editor	293
6.3.2	Umgebungseditor	297
6.3.3	Laufzeit-Manager	298
6.4	Anwendungsumgebungen	301
6.5	Umfang der Implementierung	306
6.6	Zusammenfassung	309
7	Schlussbemerkungen	311
7.1	Zusammenfassung	311
7.2	Bewertung	315

Inhaltsverzeichnis

7.3 Ausblick	320
Abbildungsverzeichnis	325
Listingverzeichnis	329
Literaturverzeichnis	331
Index	355