



Fakultät II – Informatik, Wirtschafts- und Rechtswissenschaften
Department für Informatik

A Model for Adaptive Applications on the Semantic Web

Dissertation for the degree
Doctor of Engineering
(Doktor der Ingenieurwissenschaften)

of

Dipl.-Inf. Ammar Memari

Oldenburger Schriften zur Wirtschaftsinformatik

Band 16

Ammar Memari

**A Model for Adaptive Applications
on the Semantic Web**

Shaker Verlag
Aachen 2015

Bibliographic information published by the Deutsche Nationalbibliothek

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

Zugl.: Oldenburg, Univ., Diss., 2014

Copyright Shaker Verlag 2015

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

Printed in Germany.

ISBN 978-3-8440-3384-7

ISSN 1863-8627

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

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

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

Acknowledgement

I would like to express my special appreciation and thanks to my advisor Prof. Dr. Jorge Marx Gómez for his invaluable guidance, patience and his continuous faith in my abilities. I would also like to thank Prof. Dr. Andreas Winter for his support during much of this research as my second supervisor.

This work would not have been achievable if not for the support of the German Academic Exchange Service (DAAD) who had provided me with a full scholarship for three and a half years.

While working towards my degree, I was also an employee in the department of Embedded Hardware/-Software Systems (EHS) and worked with the Smart Resource Integration (SRI) group at OFFIS Institute. I thank them, and in particular Prof. Dr. Wolfgang Nebel, for providing me with an opportunity to explore a new domain and develop my scientific horizon.

Additionally, I would like to thank Prof. Dr. Jürgen Sauer and Dr. André Bolles for serving as my committee members, providing important feedback and perspective, and letting my defense be an enjoyable event.

I acknowledge with the deepest gratitude the role played by my colleagues both at the VLBA department and at OFFIS SRI group. They have provided valuable feedback, guidance, inspiration, and the cheerful unforgettable moments. Thank you Dr. Nico Brehm, Dr. Daniel Söpke, Prof. Dr. Liane Haak, Dr. Jan Aalmink, Benjamin Wagner vom Berg, and Dr. Gunnar Schomaker especially for your mentoring, guidance, and thought-provoking discussions, Neil Ross Paterson, Alaa Memari, Alexander Zeißig, and Alexander Borgerding for taking time out of your busy lives to proof-read this document.

Besides colleagues at the university and OFFIS, it has been a pleasure to work with the students doing their bachelors' and masters' theses as well as those participating in the project group that I had supervised.

My mother and father have always believed in me, which has helped me get to where I am now. Thank you for all the sacrifices you have made, and for buying my first computer back in 1999; it was expensive by then and I hope this dissertation goes some way to repaying your investment!

My brother Alaa has been a great friend and supporter, he was always there for me even during what has sometimes been a difficult journey. My little brother Issa, and through his enthusiasm for studying computer science, has inspired me to go further.

Most importantly, I want to thank my beautiful wife Leen for her love and understanding. She never complained when I had to spend weekends and vacation days working on this dissertation, always offered words of encouragement when I was feeling down, spent sleepless nights backing me up, and somehow always found a simple solution to any problem I proposed.

Finally, but not least, my sincere thanks to my many friends for their emotional support, patience, and understanding. You were all fantastic.

Abstract

Modern applications need to process and be aware of much more than direct input. Indirect input includes information about the application user, the surrounding context, and the available resources as examples.

User-sensitive applications deserve special attention in today's information industry due to massive amounts of information made available hourly for individuals. Filtering and recommendation of pieces of information are therefore "common practices" rather than "additional features".

Context-aware applications are growing rapidly in the environment of ubiquitous and mobile computing. More and more applications are able to sense and adapt to screen size, connection speed, battery status, and possible input methods of the device on the one hand, and to time of day, current weather, and traffic conditions on the other.

Resource-critical conditions are affecting a growing number of applications not only on mobile devices but also within data centers. Energy, bandwidth, memory, and storage restrictions apply to applications that provide services. Sensitivity of the application itself to availability of these resources is a welcomed and encouraged feature.

Adaptive Hypermedia, Adaptive Computing, Context-aware applications, User-oriented applications, Recommendation Systems, Collaborative Filtering applications, Personalized Information Retrieval, and many other similar techniques have been used to tackle the above-mentioned problems. Even though they have a lot in common, they still live on separate islands. Each of them could have had benefits from advances made in the others, had a common ground existed.

This common ground, in order for it to guarantee a healthy collaboration, needs to clearly set common terms and concepts, and then use them to define the different categories and classify applications onto them. A formal version of this process would require a common metamodel representing the terms, a reference model representing their possible interactions, and transformations of the two which allow the models to mutate to represent existing and emerging systems.

Moreover, the common ground should provide interoperability on the *data* level. This calls for a metadata technology that can guarantee a data representation understandable by the different applications. Ontologies promise to deliver this feature, and they, together with the rest of the Semantic Web stack of technologies have proven useful for such tasks.

In order to fulfill these requirements, this work sheds light on adaptive applications on the Semantic Web following a design science research methodology. It provides a rough-sets-theory-based definition for adaptive applications and other neighboring categories, in addition to a framework composed of a UML metamodel, a reference model, and their transformations.

The models are designed in accordance with best practices extracted from existing models, and guided by relevance to problems of existing systems running on the ground. Design patterns for approaching some problems are defined as parts of the reference model.

Quality of the resulting artifacts is assured by prototyping, where the models are implemented and evaluated in two distinct prototypes. The first prototype serves as a proof-of-concept for parts of the

models, and the second poses as a life-size implementation and case study.

The result is a framework for developing adaptive applications that utilize Semantic Web technologies. The framework accounts for the business environment and produces viable applications which account for its risks and opportunities. It provides solutions for aligning the adaptive functionality with the corporate information system and organizational infrastructure.

Zusammenfassung

Moderne Anwendungen müssen heute sehr viel mehr verarbeiten und berücksichtigen als direkte Eingaben. Weitere, indirekte Eingaben enthalten beispielsweise Informationen über den Nutzer der Anwendung, den umgebenden Kontext und verfügbare Ressourcen.

Auf Grund der gewaltigen Menge an Daten, die dem Einzelnen verfügbar gemacht werden, verdienen benutzersensitive Anwendungen besondere Beachtung. Das Filtern und Vorschlagen spezifischer Informationen ist heute übliche Praxis und nicht mehr als Zusatzfunktion zu bezeichnen.

Im Umfeld des ubiquitären und mobilen Computings wächst die Zahl kontextsensitiver Anwendungen rapide. Immer mehr Anwendungen sind in der Lage sich an verschiedene Displaygrößen, Bandbreiten, Akkuladungen und Eingabemethoden der Endgeräte sowie auch an die Tageszeit, das aktuelle Wetter und Verkehrsbedingungen anzupassen.

Ressourcen-kritische Bedingungen tangieren eine wachsende Zahl von Anwendungen, nicht nur auf mobilen Geräten, auch in Rechenzentren. Energie-, Bandbreiten-, Speicher- und Kapazitätsbeschränkungen betreffen Anwendungen, die Dienste anbieten. Die Fähigkeit solcher Anwendungen auf die Verfügbarkeit dieser Ressourcen zu reagieren ist eine willkommene und auch forcierte Funktionalität.

Adaptive Hypermedia, Adaptive Computing, kontextsensitive Anwendungen, benutzerorientierte Anwendungen, Empfehlungssysteme, personalisiertes Information Retrieval und viele ähnliche Techniken wurden genutzt, um die genannten Probleme zu lösen. Obwohl sie viel gemeinsam haben, stammen sie doch aus verschiedenen Bereichen. Jede könnte Vorteile aus den Fortschritten der anderen ziehen, hätte es eine gemeinsame Plattform gegeben.

Offensichtlich muss diese Plattform, um einen gesunden Schulterschluss zu garantieren, gemeinsame Begriffe und Konzepte festlegen, und diese dann verwenden, um Kategorien zu definieren und Anwendungen diesen zuzuordnen.

Eine formale Variante dieser Prozesse würde ein gemeinsames Metamodell, das die Begriffe abbildet, ein Referenzmodell, das die Interaktionen abbildet, sowie Transformationen, die es den Modellen ermöglicht existierende und neu entstehende Systeme abzubilden, erfordern.

Ferner sollte die gemeinsame Plattform Interoperabilität auf der Datenebene ermöglichen. Dies erfordert eine Metadaten-Technologie, die garantiert, dass die Datenrepräsentation von den verschiedenen Anwendungen verstanden wird. Ontologien genügen diesen Anforderungen und haben sich, zusammen mit dem Bündel an Semantic Web Technologien, als tauglich für diese Aufgabe erwiesen.

Um diese Anforderungen zu erfüllen richtet diese Arbeit ihren Fokus auf adaptive Anwendungen im Semantic Web. Sie folgt dabei dem Entwurfs- und Entwicklungsparadigma des Design Science. Sie liefert eine Rough-Sets-Theorie gestützte Definition für adaptive Anwendungen und benachbarte Kategorien in Ergänzung zu einem Framework bestehend aus einem UML-Metamodell, einem Referenzmodell und ihren Transformationen.

Die Modelle wurden in Übereinstimmung mit der guten fachlichen Praxis aus existierenden Modellen extrahiert und durch die Relevanz für reale laufende Systeme vor Ort geleitet. Als Teil des Referenzmodells werden Entwurfsmuster definiert, um sich einigen Problemen anzunähern. Die Qualität der

resultierenden Artefakte wird durch das Erstellen von Prototypen gesichert, wobei die Modelle in zwei unterschiedlichen Prototypen implementiert werden. Der erste Prototyp dient als Machbarkeitsnachweis für Teile des Modells, der Zweite stellt eine lebensgroße Implementierung und eine Fallstudie dar.

Das Ergebnis ist ein Framework zur Entwicklung von adaptiven Anwendungen, die Semantic Web Technologien verwenden. Das Framework trägt den Rahmenbedingungen in der Wirtschaft Rechnung und erzeugt funktionsfähige Anwendungen, die deren Risiken und Möglichkeiten nachweisen. Es stellt Lösungen bereit, um adaptive Funktionalität mit dem Informationssystem der Unternehmung und der organisatorischen Infrastruktur zu verbinden.

Table of Contents

Acknowledgement	iii
Abstract	v
Zusammenfassung	vii
1 Introduction	1
1.1 Motivation	1
1.2 Research Question	2
1.3 Contribution of this Thesis	2
1.4 Research Methodology of Choice	3
1.5 Structure of the Thesis and Typographical Conventions	6
2 Background	9
2.1 Adaptation	9
2.2 Hypertext, Hypermedia, and Adaptive Hypermedia	9
2.2.1 Adaptation Methods and Techniques	11
2.2.2 Adaptation Knowledge Submodels	12
2.3 Semantic Web and Software Agents	13
2.3.1 Ontologies and Reasoning	14
2.3.2 Software Agents	16
2.4 Personalization in Business Environment	21
3 Research Domain: Adaptive Applications on the Semantic Web	23
3.1 Definitions in Literature	23
3.1.1 Definition of Lenz	23
3.1.2 Definition in Adaptive Hypermedia	24
3.1.3 Other related classifications	24
3.2 An Informal Rough-Sets-based Definition	24
3.3 Set of Example Every-Day Applications	25
3.4 Set of Attributes Relevant to Adaptivity	25
3.4.1 Models Adaptation Domain Metamodel	25
3.4.2 Models Adaptation Subject	25
3.4.3 Models Adaptation Object	28
3.4.4 Models Where and How to Adapt	29
3.4.5 Extends Models by Reasoning	33
3.4.6 Has Navigable Conceptual Space on Models	34
3.4.7 Has Order of Adaptation	35
3.4.8 Has Adaptable Architecture	37
3.4.9 Respects Business Constraints	37
3.5 Adaptive Applications Definition	38
3.5.1 Lower and Upper Approximation	38

3.5.2	The Reduct	39
3.5.3	Other Definitions	39
3.6	Summary	40
4	Research Rigor and Related Works	43
4.1	Related Works on Classification and Interpretation	43
4.1.1	Adaptive Systems in the Era of the Semantic and Social Web: a Survey	43
4.1.2	A Comprehensive Survey of Adaptive Hypermedia Methods and Techniques	46
4.1.3	Personalized Information Retrieval: Survey and Classification	49
4.2	Related Reference Models	51
4.2.1	Dexter Hypertext Reference Model	51
4.2.2	The Tower Model	53
4.2.3	Amsterdam Model of Hypermedia	53
4.2.4	Adaptive Hypermedia Application Model - AHAM	54
4.2.5	Munich Reference Model	54
4.2.6	Goldsmiths Adaptive Hypermedia Model - GAHM	56
4.2.7	Layered WWW AHS Authoring Model - LAOS	56
4.2.8	Fundamental Open Hypermedia Model - FOHM	57
4.2.9	Generic Adaptivity Model - GAM	58
4.2.10	General User Model Ontology - GUMO	59
4.2.11	Enhanced Adaptive Hypermedia Application Model	59
4.2.12	General Ontological Model for Adaptive Web Environments - GOMAWE	59
4.2.13	Generic Adaptation Framework - GAF	60
4.3	Summary	60
5	Requirements and Problem Relevance	63
5.1	Reaching at the Problem Area	63
5.2	Unambiguous Domain	63
5.3	Open Corpus Adaptation	64
5.3.1	Open Corpus Adaptive Hypermedia	64
5.3.2	Freshness versus Completeness	65
5.3.3	Adaptive Crawling	66
5.4	Second-Order Adaptation	66
5.5	Knowledge Re-usability and Conceptual Space	67
5.6	User Clustering and Data Mining	68
5.7	Context and Neighborhood Awareness	69
5.8	Business Ethical Constraints	70
5.8.1	Personal Information Constraints	70
5.8.2	Potential Objects of Personalization	72
5.8.3	Depth of Personalization	74
5.9	Business Risks and Opportunities	76
5.10	Adaptation to a Variable Environment	77
5.11	Summary	78
6	The Approach: Design as a Search Process	79
6.1	Reference Modeling and Design Patterns	79
6.2	Modeling Language and Notation	80
6.3	Process of Reaching at Solutions for Found Problems	82
6.3.1	Extension of Modeling and Adaptation Process	82
6.3.2	Ontology-Based Models	83
6.3.3	Handling the Open Corpus	84

6.3.4	Second-Order Adaptation Definition and Solution Process	89
6.3.5	Fitting into Business Environment	93
6.3.6	User Model Hosted by the CRM System	94
6.3.7	Diversity to Manage Variability	101
6.4	Summary	102
7	The Artifact: A Framework for Adaptive Applications	105
7.1	The Metamodel	105
7.1.1	Functional Dimension	105
7.1.2	Knowledge Dimension	107
7.1.3	Concrete Layer	108
7.2	The Reference Model	110
7.2.1	Aggregations on Dimensions	112
7.2.2	Adaptation Parameter Pattern	113
7.2.3	Open and Closed Corpuses	113
7.2.4	Second-Order Adaptation	118
7.2.5	Ontologies on the Knowledge and Functional Dimensions	118
7.2.6	Diversified Actor Pattern	121
7.2.7	Actor DNA Pattern	123
7.3	Transformations and Mappings to Other Models	126
7.3.1	Model Landscape	127
7.3.2	Mapping to Dexter, AHAM, and the Enhanced AHAM	128
7.3.3	Mapping to LAOS	129
7.3.4	Alignment with Organizational Infrastructure	129
8	Design Evaluation: Proof of Concept and Case Studies	133
8.1	PurpleBee Framework	133
8.1.1	Network: libpurple	134
8.1.2	PurpleBee for Building Adaptive Content Modelers	134
8.1.3	Incarnation of the Patterns	135
8.1.4	Implementation of the framework	137
8.1.5	Example Application IntHarvesters	140
8.1.6	Experiments and Results	145
8.1.7	Summary	148
8.2	Use Case: Jinengo	150
8.2.1	Jinengo Architecture	150
8.2.2	Domain Metamodel	151
8.2.3	Goal Model	151
8.2.4	User Model	153
8.2.5	Content and Context Models	154
8.2.6	Neighborhood Model	157
8.2.7	Adaptation in Jinengo	158
8.2.8	Presentation Composer	164
8.2.9	Frontend and Graphical User Interface	165
8.2.10	Evolution in Jinengo	167
8.2.11	Jinengo Evaluation	167
8.3	Compliance Study: Facebook Ads Engine	168
8.3.1	Adaptivity Attributes and Classification	169
8.3.2	Mapping to the Artifacts	172
8.3.3	Conclusion	173

9 Conclusion and Outlook	175
9.1 Revisiting the Research Question	175
9.2 Outlook and Future Work	176
List of Figures	179
List of Tables	182
List of Acronyms and Abbreviations	183
Bibliography	186