

Abschlussbericht zum Verbundprojekt InPULSE

(Integrative Pattern- und UML- orientierte Lern- und System-Entwicklungsumgebung)

Zuwendungsempfänger: Kuratorium OFFIS e.V.	Förderungskennzeichen 01ISC19D
--	-----------------------------------

Vorhabensbezeichnung	InPULSE – Integrative Pattern- und UML-orientierte Lern- und System-Entwicklungsumgebung
-----------------------------	--

Laufzeit des Vorhabens	01.01.2004 bis 30.09.2005
Berichtszeitraum	01.01.2004 bis 30.09.2005
Datum	17.11.2005

Mitarbeiter:

Prof. Dr. Claus Möbus
Dipl.-Inform. Hilke Garbe
Dipl.-Inform. Claudia Janßen
Dipl.-Inform. Heiko Seebold
Dipl.-Inform. Holger de Vries
Dipl.-Inform. Jochen Zurborg

Kontakt:

claus.moebus@uni-oldenburg.de
hilke.garbe@uni-oldenburg.de
claudia.janssen@offis.de

I.	KURZDARSTELLUNG DES VORHABENS	4
1.	AUFGABENSTELLUNG.....	4
2.	VORAUSSETZUNG, UNTER DENEN DAS VORHABEN DURCHGEFÜHRT WURDE.....	4
3.	PLANUNG UND ABLAUF DES VORHABENS.....	5
4.	WISSENSCHAFTLICHER UND TECHNISCHER STAND, AN DEN ANGEKNÜPFT WURDE ...	6
5.	ZUSAMMENARBEIT MIT ANDEREN STELLEN.....	11
II.	EINGEHENDE DARSTELLUNG DER ERGEBNISSE	12
1.	ERZIELTE ERGEBNISSE	12
1.1.	<i>HAP 1: Entwurfsmuster-Ontologie</i>	<i>13</i>
1.1.1.1.	UAP 1.2: Wissensakquisition	13
1.1.1.2.	UAP 1.3: Aufbau der Musterontologie	15
1.2.	<i>HAP 2: Verfahren zur Design-Überprüfung auf Mustermerkmale</i>	<i>19</i>
1.2.1.	UAP 2.2: Entwicklung des Case Based Reasoning Systems	19
1.2.1.1.	CBR-Verfahren.....	19
1.2.1.2.	Korrespondenzregeln	21
1.2.1.3.	Arbeitsweise des Dialogsystem.....	22
1.2.1.3.1.	Dialogführung mit Begriffsverbänden	22
1.2.1.3.2.	Wissensrepräsentation mit Bayes-Netzen.....	24
1.2.1.3.2.1.	Bayes-Netze.....	25
1.2.1.3.2.2.	Beschreibung des auf Bayes-Netzen basierenden Dialogspiels	26
1.2.2.	UAP 2.3: Entwicklung des Hypothesentesten.....	29
1.2.2.1.	Architektur	30
1.2.2.2.	Ablauf des Hypothesentestens.....	31
1.3.	<i>HAP 6: Entwurf/Realisierung E-Learning und Assistenzsystem</i>	<i>34</i>
1.3.1.	UAP 6.4: Adaption der e-stat-Plattform.....	34
1.3.2.	UAP 6.5: Content-Engineering.....	35
1.3.3.	UAP 6.6: Integration der Algorithmen aus HAP 2 in die E-Learning-Plattform	36
1.4.	<i>Evaluation.....</i>	<i>37</i>
2.	VORAUSSICHTLICHER NUTZEN	41
3.	FORTSCHRITT BEI ANDEREN STELLEN	43
4.	VERÖFFENTLICHUNGEN	43
III.	ERFOLGSKONTROLLBERICHT.....	45
1.	BEITRAG DES ERGEBNISSES ZU DEN FÖRDERPOLITISCHEN ZIELEN	45
2.	WISSENSCHAFTLICH-TECHNISCHE ERGEBNISSE.....	45
3.	FORTSCHREIBUNG DES VERWERTUNGSPLANS	45
4.	ARBEITEN, DIE ZU KEINER LÖSUNG GEFÜHRT HABEN	46
5.	PRÄSENTATIONSMÖGLICHKEITEN FÜR ANDERE NUTZER.....	46
6.	EINHALTUNG DER AUSGABEN- UND ZEITPLANUNG	46

Abbildung 1: Ausschnitt aus dem Projekt-Meilensteinplan	5
Abbildung 2: Curricular-Planung mit Hasse-Diagramm	14
Abbildung 3: Screenshot des Paarvergleich-Tools	15
Abbildung 4. Grafische Benutzungsoberfläche von KARaCAs	17
Abbildung 5. Aus dem Repertory Grid erzeugter Kontext	17
Abbildung 6. Aus dem Kontext erzeugtes Liniendiagramm	18
Abbildung 7: Liniendiagramm als Ergebnis der Formalen Begriffsanalyse	20
Abbildung 8: Beispielhafte Nutzung des CBR-Systems.....	21
Abbildung 9: Aktivitätsdiagramm für den Dialogablauf	23
Abbildung 10: Qualitative Struktur des Bayes-Netzes	26
Abbildung 11: Bayes-Netz zu Strukturmustern	28
Abbildung 12: Aktivitätsdiagramm für den Dialogablauf mit einf. Fragestrategie	29
Abbildung 13. UML-Komponenten-Diagramm für das Hypothesentesten	30
Abbildung 14: Lösungsentwurf des Benutzers.....	31
Abbildung 15: Dialog zur Semantik der Diagrammelemente	31
Abbildung 16: Vervollständigung des Benutzerentwurfes durch das System	32
Abbildung 17: Dozenten-Tool zur Erstellung neuer Aufgaben	33
Abbildung 18: Assoziations-Beziehung und Aggregations-Beziehung	34
Abbildung 19: InPULSE-CMS, http://inpulse.uni-oldenburg.de/inpulse	35
Abbildung 20: Content im InPULSE-LCMS.....	36
Abbildung 21: InPULSE-Dialogspiel zur Musteridentifikation.....	37
Abbildung 22: Fragebogenauswertung - Entwurfsmusterkenntnisse	38
Abbildung 23: Fragebogenauswertung - Nutzung von LCMS.....	39
Abbildung 24: Fragebogenauswertung - Detaillierungsgrad der Informationen	39
Abbildung 25: Fragebogenauswertung – punktuelle Nutzung der Informationen	39
Abbildung 26: Fragebogenauswertung - Angemessenheit der Textlänge	40
Abbildung 27: Fragebogenauswertung - Anzahl der Beispiele	40
Abbildung 28: Fragebogenauswertung - Workflow-Unterstützung.....	40
Abbildung 29: Fragebogenauswertung - zukünftige Systemnutzung.....	41
Abbildung 30: Fragebogenauswertung - Verbesserungsvorschläge	41

I. Kurzdarstellung des Vorhabens

1. Aufgabenstellung

OFFIS e.V. entwickelte im Rahmen des Projektes InPULSE eine Ontologie-basierte Methode zur Unterstützung der Anwendung von Mustern im Forward-Engineering.

Es wurde ein werkzeugunterstützter Workflow gestaltet, der die Wissensakquisition, die Wissensrepräsentation und schließlich auch die Wissensanwendung abdeckt.

Die Methode wurde innerhalb eines E-Learningsystems implementiert. Das E-Learningsystem soll die Verwendung von Mustern schulen und damit einen wesentlichen Beitrag zur Forcierung der standardisierten Entwicklung von Software leisten.

2. Voraussetzung, unter denen das Vorhaben durchgeführt wurde

Das Kuratorium OFFIS e.V. ist ein an die Carl-von-Ossietzky Universität Oldenburg angeschlossenes Forschungsinstitut. Prof. Dr. Claus Möbus ist Mitglied in beiden Organisationen und seine Abteilung Lehr- und Lernsysteme setzt sich aus Universitäts- und OFFIS-Mitarbeitern zusammen.

In der Abteilung von Prof. Dr. C. Möbus (Universität Oldenburg, Fachbereich Informatik) werden seit 1985 wissensbasierte CBT-Programme, sogenannte Intelligent Problem Solving Environments (IPSE) [Möb95] entwickelt. Das Design der IPSE basiert auf der in der Arbeitsgruppe entwickelten, kognitionswissenschaftlich orientierten ISPD – Theorie (impasse - success - problem solving driven learning) [MTS93a,b, MST94]. Diese Theorie beschreibt den Wissenserwerb beim Lösen von Aufgaben und Problemen. Sie erklärt die Modifikation von Wissen in Stocksituationen und die Optimierung von Wissen nach erfolgreicher Aufgabenbearbeitung. Beim Problemlösen werden die Handlungsphasen Abwägen, Planen Ausführen und Bewerten differenziert unterstützt.

IPSE sind Lehr- und Problemlöswerkzeuge, die aktives Lernen fördern und die Eigenaktivität der Lernenden unterstützen. Die Lernenden haben die Freiheit, entdeckend Probleme zu lösen und selbständig Hypothesen zu formulieren und zu überprüfen. Eine große Bandbreite von Lösungsentwürfen - auch unvollständige Entwürfe – werden untersucht, kommentiert und mit adaptiven Hilfen unterstützt.

Bisher wurden IPSE für unterschiedliche Wissensgebiete entwickelt [z.B. Möb95, MST96, Möb+02a,c, Wil02, YGMT02]: Softwareentwurf (Funktionale Programmierung), Modellierung und Beschreibung verteilter Systeme (Petri-Netze), Raumausstattung, CAD (Hydraulik / Pneumatik und Elektrotechnik), Polymerisation von Kunststoffen, betriebswirtschaftliche Planspiele, Modellierung und Diagnosen bei unsicheren komplexen Sachverhalten (Umweltmedizin), Konstruktion elektrischer Schaltungen und Qualitätssicherung mit Statistischer Prozesskontrolle. Die Förderung dieser Projekte erfolgte durch die DFG, BMBF, BMWI, EU, ESF, BIBB u.a.

Im Unterschied zu Intelligenten Tutor Systemen beinhalten IPSE kein aufwendiges und „teures“ Lehr- und Lernermodul. Das Curriculum wird durch eine Sequenz obligatorischer domänenrelevanter Aufgaben realisiert. Informationen werden nicht

durch Instruktionen sondern problemorientiert in Form von Hilfen nach dem „Pull-Prinzip“ geliefert. Während beim „Pull-Prinzip“ Lernen „on-demand“ ermöglicht wird, verlangt das „Push-Prinzip“ Lernen auf Vorrat. Lerneradaptivität wird durch die Methode des Hypothesentestens [Möb95, TMS97, WM00] erreicht. Dabei selektiert der Lerner einen Teil seiner Lösung, den er für korrekt hält und das System prüft mit Hilfe eines generativen Expertensystems, ob sich die Teillösung zu einer Gesamtlösung erweitern lässt. Auf diese Weise wird individuell auf den Lernerentwurf Bezug genommen.

In dem Projekt I-CAN-EIB wurde ein Beratungs- und Informationssystem erstellt, das in Form eines innovativen avatarbasierten Wissens-Chat über die Möglichkeiten des Elektronischen Installationsbusses (EIB) informiert. In e-stat [Möb+02b] wurde eine E-Learning-Plattform realisiert, in die Lerninhalte mit deklarativem Wissen der Statistikdomäne aus unterschiedlichen Quellen eingestellt werden können und die es ermöglicht, diese view-orientiert nach unterschiedlichen didaktischen Methoden zu vermitteln. Diese Plattform wurde in InPULSE für das E-Learningsystem adaptiert.

3. Planung und Ablauf des Vorhabens

Die zeitliche Planung unserer UAP ist dem Ausschnitt aus dem Meilensteinplan (vgl. Abbildung 1) zu entnehmen. Der vollständige Meilensteinplan ist im Rahmenantrag enthalten. Dem Plan sind die für die jeweiligen Aufgaben erforderlichen Voraussetzungen und die Ergebnisse zu entnehmen. Ebenso ist daraus die Vernetzung zwischen den Projektpartnern ersichtlich.

Arbeitspaket-Nr.	Arbeitspaket-Bezeichnung	Bearbeitungszeitraum im Projekt																										
		2003			2004						2005																	
		10	11	12	01	02	03	04	05	06	07	08	09	10	11	12	01	02	03	04	05	06	07	08	09			
Meilensteine														M 1											M 2			
HAP 0	Projektmanagement																											
	DaimlerChrysler	Summe:	4																									
JAP 0.0	Konsortialführerschaft																											
	DaimlerChrysler	Summe:	0,5						2												1,5							
HAP 1	Entwurfsmuster-Ontologie	Summe:	18,5																									
	DaimlerChrysler	Summe:	2,5																									
JAP 1.1	Erstellung eines Musterkatalogs mit allen Gamma-Mustern				E 1.1.1																							
	OFFIS	Summe:	2,5																									
JAP 1.2	Wissensakquisition				(E1.1.1)	(E2.4.1)																						
	OFFIS	Summe:	4																									
JAP 1.3	Aufbau der Musterontologie					(E3.4.2)				E 1.3.1															E 1.3.2			
	OFFIS	Summe:	6,5																									
HAP 2	Verfahren zur Design-Überprüfung auf Mustermerkmale	Summe:	37,25																									
	DaimlerChrysler	Summe:	6																									
JAP 2.1	Analyse musterbasierter Entwurf bei Embedded Software									(E1.3.1)					E 2.1.1													
	OFFIS	Summe:	17																									
JAP 2.2	Entwicklung des Case Based Reasoning Systems									E 2.2.1	(E1.3.1)				E 2.2.2	(E1.3.2)				(E3.8.1)		(E3.8.2)		E 2.2.3				
	OFFIS	Summe:	7																									
JAP 2.3	Entwicklung des Hypothesentestens									(E3.5.1)					E 2.3.1				(E3.5.2)				(E3.7.1)	E 2.3.2				
	OFFIS	Summe:	3,5																									
...																												
	Embedded SW-Systemen	Summe:	32																						7,25			
	OFFIS	Summe:	32																						7,25			
JAP 6.4	Adaption der e-stat Plattform									E 6.4.1																		
	OFFIS	Summe:	1,5																						2,25			
JAP 6.5	Content-Engineering									(E6.4.1)																		
	OFFIS	Summe:	4,5																						4,5			
JAP 6.6	Integration der Algorithmen aus HAP2 in die E-Learning Plattform									(E2.2.1)					E 6.6.1	(E2.2.2)									E 6.6.2	(E2.2.3)	(E2.3.1)	E 6.6.3
	OFFIS	Summe:	1,5																									

Abbildung 1: Ausschnitt aus dem Projekt-Meilensteinplan

Demnach waren folgende Hauptarbeits- und Unterarbeitspakete geplant, die im Abschnitt „Erzielte Ergebnisse“ eingehend vorgestellt werden:

HAP 1: Entwurfsmuster-Ontologie

UAP 1.2: Wissensakquisition

UAP 1.3: Aufbau der Musterontologie (Deliverables E1.3.1, E1.3.2)

HAP 2: Verfahren zur Design-Überprüfung auf Mustermerkmale

UAP 2.2: Entwicklung eines Case Based Reasoning Systems
(Deliverables E2.2.1, E2.2.2, E2.2.3)

UAP 2.3: Entwicklung des Hypothesentesten
(Deliverables E2.3.1, E.2.3.2)

HAP 6: Entwurf/Realisierung E-Learning und Assistenzsystem

UAP 6.4: Adaption der e-stat-Plattform (Deliverable E6.4.1)

UAP 6.5: Content Engineering

UAP 6.6: Integration der Algorithmen aus HAP 2 in die E-Learning-Plattform
(Deliverable 6.6.1, E6.6.2, E6.6.3)

Evaluation

Alle Projektarbeiten wurden plangemäß durchgeführt. Änderungen im Zeitplan haben sich nicht ergeben.

Die Wissensakquisition mit dem Werkzeug KARaCAs wurde während der gesamten Projektlaufzeit weitergeführt. So konnten neue Informationen zu den Mustern kontinuierlich in die Ontologie integriert werden. Auch die Content-Erstellung für das E-Learning-System, die Entwicklung des Hypothesentestens und des Dialogspiels wurde kontinuierlich betrieben.

Die Ergebnisse wurden zu den vorher definierten Meilensteinen angefertigt und werden insbesondere im Teil „Erzielte Ergebnisse“ (II.1.1) erläutert.

4. Wissenschaftlicher und technischer Stand, an den angeknüpft wurde

Die Arbeiten hinsichtlich der E-Learning-Plattform stützten sich auf dem e-stat-Projekt. Die dort entwickelte Plattform wurde adaptiert und weiterentwickelt. Das Hypothesentesten wurde u. a. im Projekt mileET bereits erfolgreich eingesetzt und war Basis für die Entwicklung des Hypothesentestens mit den spezifischen Anforderungen im InPULSE-Projekt.

Literatur

[Bortz89] Bortz, J.: Statistik für Sozialwissenschaftler. Springer-Verlag, 1989.

[Fahrmeir+03] Fahrmeir, L.; Künstler, R.; Pigeot, I. & Tutz, G.: Statistik. Ein Weg zur Datenanalyse. Springer-Verlag, 2003.

- [Castillo+97] Castillo, E., Gutiérrez, J.M., Hadi, A.S.: Expert Systems and Probabilistic Network Models, Berlin: Springer, 1997
- [C+04] Cimiano, P.; Hotho, A.; Stumme, G.; Tane, J.: Conceptual Knowledge Processing with Formal Concept Analysis and Ontologies, ICFA'04, 2004.
- [Cowell+99] Cowell, R.G., Dawid, A.P., Lauritzen, St.L., Spiegelhalter, D.J.: Probabilistic Networks and Expert Systems, Berlin: Springer, 1999
- [DaimlerChrysler03] DaimlerChrysler: Merkmale vom Entwurfsmustern nach Gamma et al. Version 1.0
- [Eden02] Eden, A.H., 'LePUS: A Visual Formalism for Object-Oriented Architectures', 6th World Conference on Integrated Design and Process Technology, Pasadena, California, (2002)
- [Eden01] Eden, A.H., Formal Specification of Object Oriented Design. International Conference on Multidisciplinary Design in Engineering CSME-MDE 2001 (Nov. 21-22, 2001), Montreal, Canada.
- [Eden+01] Eden, A.H., Hirshfeld, Y., Principles in formal specification of object oriented design and architecture, CASCON '01: Proceedings of the 2001 conference of the Centre for Advanced Studies on Collaborative research, Toronto, Ontario, Canada, IBM Press, 2001
- [Eden +98] Eden, A.H., Hirshfeld, Y., Yehudai, A., LePUS - A Declarative Pattern Specification Language, Technical report 326/98, department of computer science, Tel Aviv University, citeseer.ist.psu.edu/112816.html, letzter Zugriff: 17.12.2004
- [Eden04] Eden, A.H., LePUS 2: Sample Specifications, <http://www.eden-study.org/lepus/specifications/specifications.html>, letzter Zugriff : 17.12.2004
- [FMS96] Folckers, J., Möbus, C., Schröder, O. & Thole, H.J.: An Intelligent Problem Solving Environment for Designing Explanation Models and for Diagnostic Reasoning in Probabilistic Domains, in: C. FRASSON, G. GAUTHIER, A. LESGOLD (eds), Intelligent Tutoring Systems, ITS 96, Montreal, Canada, Proceedings, p. 353-362, Berlin: Springer (LNCS 1086), 1996
- [Gamma+95] Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, "Design Patterns - Elements of Reusable Object-Oriented Software", Addison Wesley, 1995.
- [Ganter+96] B. Ganter, R. Wille. Formale Begriffsanalyse, Springer, 1996.
- [Gar+03] GARBE, H., YAKIMCHUK, V., MÖBUS, C., OSTERLOH, J.-P., THOLE, H.-J., WEBER, L., WAGNER, E. mileET - Knowledge Based Assistance for Electrical Engineering Education in 7. Workshop Multimedia für Bildung und Wirtschaft, TU Ilmenau, 2003, ISSN 1436-4492, S. 109-114

- [GW96] GANTER, B.; WILLE, R.: Formale Begriffsanalyse, Springer, 1996.
- [Grand+02] GRAND, Mark: Patterns in Java. Volume 1. 2. Auflage. John Wiley & Sons, November 2002. ISBN 0-471-22729-3
- [Jensen01] Jensen, F.V.: Bayesian Networks and Decision Graphs, Statistics for Engineering and Information Science, Berlin: Springer, 2001
- [Kirchhoff+01] Kirchhoff, S.; Kuhnt, S.; Lipp, P. & Schlawin, S.: Der Fragebogen. Datenbasis. Konstruktion. Auswertung. Leske + Budrich, 2001.
- [Maplesden+02] MAPLESDEN, D.; HOSKING, J.G.; GRUNDY, J.C.: Design Pattern Modelling and Instantiation using DPML, In Proceedings of Tools Pacific 2002, Sydney, 18-21 February, 2002, CRPIT Press.
- [Mak+04] MAK, Jeffrey K.H.; CHOY, Clifford S.T.; LUN, Daniel P.K.: Precise Modeling of Design Patterns in UML, In Proceedings of the 26th International Conference on Software Engineering (ICSE '04).
- [Maudet+98] N. Maudet, F. Evrard. A generic framework for dialogue game implementation; in Second Workshop on Formal Semantics and Pragmatics of Dialogue (TWLT13) May 13-15, 1998, University of Twente, Enschede, The Netherlands, Hulstijn J. and Nijholt A. eds.
- [Mattuch+02] W. Mattuch, J. Caumanns. Medien in der betrieblichen Weiterbildung – Eigenschaften von Lernmaterialien für die arbeitsprozessorientierte Weiterbildung. Fraunhofer Institut Software- und Systemtechnik 2002, <http://www.apo-it.de/Bekanntmachung/Medien.pdf>.
- [Möb+02a] C. Möbus, B. Albers, S. Hartmann, H.J. Thole, J. Zurborg. Towards a Specification of Distributed and Intelligent Web Based Training Systems, in: St.A. Cerri, G. Gouarderes & F. Paraguacu (eds), Intelligent Tutoring Systems, Proceedings of the 6th International Conference, ITS2002, Biarritz, France and San Sebastian, Spain, June, 2002, 291 - 300, Berlin: Springer, Lecture Notes in Computer Science.
- [Möb+02b] C. Möbus, B. Albers, S. Hartmann, J. Zurborg. Intelligent WBT: Specification and Architecture of the Distributed, Multimedia e-Learning System e-STAT, in: W. HÄRDLE & B.RÖNZ (Hrsgb), COMPSTAT 2002, Proceedings in Computational Statistics, 401 - 406, Heidelberg: Physica Verlag, 2002.
- [Möb+02c] C. Möbus, B. Albers, H. Garbe, S. Hartmann, H.J. Thole, V. Yakimchuk, J. Zurborg. Towards an AI-Specification of Intelligent Distributed Learning Environments. KI-Heft 1/03 "Wissensmodellierung und Wissenskommunikation in Lernszenarien" – im Druck.

- [Möb95] C. Möbus. Towards an Epistemology on Intelligent Problem Solving Environments: The Hypothesis Testing Approach, in J. GREER (ed), Proceedings of AI-ED 95, World Conference on Artificial Intelligence and Education, Washington, DC, Association for the Advancement of Computing in Education (AACE), 1995, S. 138 – 145
- [Möbus+04] MÖBUS, Claus; GARBE, Hilke; JANßEN, Claudia; DE VRIES, Holger; ZURBORG, Jochen: Erste Version der Musterontologie, Whitepaper 2004, HAP 1, E 1.3.1, Projekt InPULSE.
- [Möbus+04a] MÖBUS, Claus; GARBE, Hilke; JANßEN, Claudia; SEEBOLD, Heiko; DE VRIES, Holger: Zweite Version der Musterontologie, Whitepaper 2004, HAP 1, E 1.3.2, Projekt InPULSE.
- [Möbus+04b] MÖBUS, Claus; GARBE, Hilke; JANßEN, Claudia; DE VRIES, Holger; ZURBORG, Jochen: Erste Version des CBR-Verfahrens, Whitepaper 2004, HAP 2, E 2.2.1, Projekt InPULSE.
- [Möbus+04c] MÖBUS, Claus; GARBE, Hilke; JANßEN, Claudia; SEEBOLD, Heiko; DE VRIES, Holger: Zweite Version des CBR-Verfahrens, Whitepaper 2004, HAP 2, E 2.2.2, Projekt InPULSE.
- [Möbus+04d] MÖBUS, Claus; GARBE, Hilke; JANßEN, Claudia; SEEBOLD, Heiko; DE VRIES, Holger: DTD für die Modulstruktur, Whitepaper 2004, HAP 6, E 6.4.1, Projekt InPULSE.
- [Möbus+04e] MÖBUS, Claus; GARBE, Hilke; JANßEN, Claudia; OSTERLOH, Jan-Patrick; DE VRIES, Holger: Designvorgaben im Forward Engineering, Whitepaper 2004, HAP 2, E 2.3.1, Projekt InPULSE.
- [Möbus+05] MÖBUS, Claus; GARBE, Hilke; JANßEN, Claudia; OSTERLOH, Jan-Patrick; DE VRIES, Holger: InPULSE Hypothesentesten. Installations- und Bedienungsanleitung, Whitepaper 2005, HAP 2, E 2.3.2, Projekt InPULSE.
- [MSG05] MÖBUS, C., SEEBOLD, H., GARBE, H. A greedy knowledge acquisition method for the rapid prototyping of knowledge structures, in K-CAP '05: Proceedings of the 3rd international conference on Knowledge capture, Banff, Alberta, Canada, New York, NY, USA: ACM Press, 2005, ISBN 1-59593-163-5, S. 211-212
- [MST94] C. Möbus, O. Schröder, H.J. Thole. Diagnosing and Evaluating the Acquisition Process of Programming Schemata, in J.E. GREER, G. McCALLA (eds), Student Modelling: The Key to Individualized Instruction (Proceedings of the NATO Advanced Research Workshop on Student Modelling, in St.Adele, Quebec, Canada), Berlin: Springer (NATO ASI Series F: Computer and Systems Sciences, Vol. 125), 1994, S. 211-264

- [MST96] C. Möbus, O. Schröder, H.J. Thole. WULPUS - An Intelligent Problem Solving Environment Delivering Knowledge Based Help and Explanations in Business Management Simulation, in C. Frasson, G. Gauthier, A. Lesgold (eds), Intelligent Tutoring Systems, Proceedings of the Third International Conference ITS 96, Montreal, Canada, June 1996, Berlin: Springer (LNCS 1086), ISBN 3-540-61327-7, 1996
- [MTS93a] C. Möbus, H.J. Thole, O. Schröder. Interactive Support of Planning in a Functional, Visual Programming Language, in P. BRNA, S. OHLSSON, H. PAIN (eds), Proceedings AI-ED 93, World Conference on Artificial Intelligence and Education, Edinburgh, 1993, S. 362 – 369
- [MTS93b] C. Möbus, H.J. Thole, O. Schröder. Diagnosis of Intentions and Interactive Support of Planning in a Functional, Visual Programming Language, in D.M. TOWNE, T. de JONG, H. SPADA (eds), Simulation-Based Experiential Learning (Proceedings of the NATO Advanced Research Workshop "The Use of Computer Models for Explication, Analysis, and Experiential Learning", Bonas, France, Oct. 12-14, 1992), Berlin: Springer (NATO ASI Series F: Computer and Systems Sciences, Vol. 122), 1993, S. 61-76
- [Pe97] Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference (Revised Second Printing), Morgan Kaufman Publishers, San Mateo, CA., 1998
- [Pearl01] Pearl, J.: Causality: Models, Reasoning, and Inference, Cambridge, UK: Cambridge University Press, 2001
- [Shalloway+03] SHALLOWAY, Alan; TROTT, James R.: Entwurfsmuster verstehen: eine neue Perspektive auf objektorientierte Software-Entwicklung. 1. Auflage. mitp, 2003. ISBN 3-8266-1345-7
- [Sch89] SCHACHTER, R. D.: Bayes-Ball: The rational pastime (for determining irrelevance and requisite information in belief networks and influence diagrams), in G. F. Cooper and S. Moral (Hrsg.), Uncertainty in Artificial Intelligence: Proceedings of the Fourteenth Conference, Morgan Kaufmann, San Francisco, CA, 1998, S. 480-487
- [Sc92] Scheer, J. W.: Psychologie der persönlichen Konstrukte und Repertory Grid-Technik. Ein idiographischer Ansatz in klinischer und medizinischer Psychologie. Erschienen in: Schmidt, L. R. (Hrsg.): Psychologische Aspekte medizinischer Maßnahmen. Jahrbuch der medizinischen Psychologie, Band 7. Berlin, Heidelberg, Springer, S. 273-290.
- [Scheer+93] SCHEER, Jörn W. (Hrsg.) ; CATINA, Ana (Hrsg.): Einführung in die Repertory Grid-Technik. Band 1: Grundlagen und Methoden. 1. Auflage. Hans Huber, 1993. ISBN 3-456-82074-7
- [Scheer+93a] SCHEER, Jörn W. (Hrsg.) ; CATINA, Ana (Hrsg.): Einführung in die Repertory Grid-Technik. Band 2: Klinische Forschung und Praxis. 1. Auflage. Hans Huber, 1993. ISBN 3-456-82075-5

- [Streitferdt04] STREITFERDT, Detlev: Anforderungen an einen Sourcecodeanalysator, Whitepaper 2004, HAP3, E 3.4.1, Projekt InPULSE.
- [TMS97] H.J. Thole, C. Möbus, O. Schröder. Domain Knowledge Structure, Knowledge Representation and Hypotheses Testing, in: B. Boulay, R. Mizoguchi (eds.): Artificial Intelligence in Education Proceedings of AI-ED 97 World Conference on Artificial Intelligence in Education, Kobe, Japan, Amsterdam: IOS Press, 1997
- [Warsitz02] B. Warsitz. Zur Strategischen Ausrichtung und Implementierung von E-Learning Umgebungen, CML Services, 2002, <http://www.cml-services.de/>
- [Wil02] WILLMS, J., Konzeption einer intelligenten Problemlöseumgebung für die Patentanmeldung und -prüfung, Dissertation, Universität Oldenburg, Fachbereich Informatik, 2002, <http://www.dissertation.de>
- [WM00] WILLMS, J. & MÖBUS, C., Evolution of the Hypotheses Testing Approach in Intelligent Problem Solving Environments, in G. Gauthier, C. Frasson, K. VanLehn (eds), Intelligent Tutoring Systems, Proceedings of the 5th International Conference, ITS 2000, Montreal, Canada, June 2000, Berlin: Springer (LNCS 1839), ISBN 3-540-67655-4, 2000
- [YGMT02] V. Yakimchuk, H. Garbe, C. Möbus, H.J. Thole. Eine intelligente Problemlöseumgebung für die Grundlagen der Elektrotechnik, in: Tagungsband zum 6. Workshop Multimedia für Bildung und Wirtschaft, TU Ilmenau, 26. & 27. 9. 2002.
- [Yak+03]YAKIMCHUK, V., GARBE, H., MÖBUS, C., THOLE, H.-J., WAGNER, E., An Intelligent Problem Solving Environment in the Domain of Electrical Engineering.. In: Hoppe, U. (Hrsg.) ; Verdejo, F. (Hrsg.) ; Kay, J. (Hrsg.): Artificial Intelligence in Education: Shaping the Future of Learning through Intelligent Technologies Bd. 97. Amsterdam, Berlin, Oxford, Tokyo, Washington DC : IOS Press Ohmsa, 2003. - ISSN 0922-6389, S. 518-520.
- [Yak+04]YAKIMCHUK, V., GARBE, H., THOLE, H.J., MÖBUS, C., WAGNER, E., mileET: Problemorientiertes Lernen in einer wissensbasierten und adaptiven Lernumgebung für die Grundlagen der Elektrotechnik. In: REBENSBURG, K. (Hrsg.): Grundfragen Multimedialen Lehrens und Lernens, 2. Workshop GML2 2004, Alcatel SEL Stiftung für Kommunikationsforschung, Books on Demand, 2004, ISBN 3-8334-1573-8, S. 73-84.

5. Zusammenarbeit mit anderen Stellen

Die Vernetzung und damit die Zusammenarbeit zwischen den Konsortialpartnern ist dem Rahmenantrag und auch dem Ausschnitt des Meilensteinplans im Abschnitt „Planung und Ablauf des Vorhabens“ zu entnehmen. Für das OFFIS gab es daneben auch Kontakte außerhalb des Konsortiums, z. B. im Zuge der Wissensakquisition. Im Einzelnen ergaben sich folgende Zusammenarbeiten:

Konsortialpartner gesamt:

- regelmäßig durchgeführte Projekttreffen
- Austausch der Ergebnisse in Form von Deliverables

OFFIS Oldenburg und Projektpartner (Software-Entwickler):

- Evaluation der InPULSE-Plattform und Feststellung der konkreten Bedürfnisse von Software-Entwicklern bzgl. der Unterstützung im Forward Engineering

DaimlerChrysler AG:

- Integration von Analysemustern aus dem Bereich Embedded Systems in die InPULSE-Plattform

eXXelent solutions:

- Konzeption einer Schnittstelle zum Austausch von UML-Modellinformationen zwischen Together und dem Tool zum Hypothesentesten

TU Ilmenau:

- Wissensakquisition: Diskussion von Anwendungsmerkmalen im Mustereinsatz und Integration in eine gemeinsame Ontologie
- Vorbereitung einer entsprechenden gemeinsamen Publikation

Universität Oldenburg:

- Gespräche zwecks Integration des InPULSE-LCMS in andere OFFIS-Projekte
- Vorbereitungen zur Verwendung des InPULSE-LCMS in Lehrveranstaltung der Universität Oldenburg

Projektgruppe (Studierende):

- Evaluation der InPULSE-Plattform

Professoren und wissenschaftliche Mitarbeiter:

- Wissensakquisition zu Anwendungsmerkmalen von Entwurfsmustern
- Durchführung des Paarvergleichs zur Erhebung von Lernreihenfolgen

II. Eingehende Darstellung der Ergebnisse

1. Erzielte Ergebnisse

Im Rahmen des Projektes InPULSE wurde am OFFIS e.V. unter der Leitung von Prof. Möbus eine Ontologie-basierte Methode zur Unterstützung der Anwendung von Mustern im Forward-Engineering entwickelt.

Die Grundlage für das zu entwickelnde **Assistenzsystem** bildete eine **Expertenbefragung** für die die Repertory Grid Methode [SC93] adaptiert wurde, um Merkmale zu erfassen, die Entwurfsmuster beschreiben. Auf Basis der identifizierten Merkmale wurde unter Verwendung der Methode der Formalen Begriffsanalyse [GW96] eine abstraktionsbasierte Hierarchisierung der Muster vorgenommen. Darüber hinaus wurde aus den erhobenen Daten ein Bayes-Netz [Pe97] generiert, das als Grundlage für einen Dialog dient, der dem Benutzer ein adäquates Muster vorschlägt. Mit Hilfe des Bayes-Netzes wird ein Softwareentwurf im Hinblick auf einen möglichen Entwurfsmuster-Einsatz geprüft, indem sukzessiv Designmerkmale

von dem Benutzer abgefragt werden. Das **Dialogsystem** ist in der Lage, nach Erfragen der Modellierungssituation, dem Benutzer ein geeignetes Muster zum Einsatz zu empfehlen.

Über das im Rahmen des Projektes entwickelte **InPULSE-LCMS¹** sind diese Systeme erreichbar.

Die folgenden Abschnitte beschreiben die Projektarbeiten entsprechend der im Meilensteinplan definierten Hauptarbeitspakete (HAP) und Unterarbeitspakete (UPA).

1.1. HAP 1: Entwurfsmuster-Ontologie

Das HAP 1 befasst sich mit der Erstellung der Entwurfsmuster-Ontologie. Im Folgenden werden die Methoden der Wissensakquisition und die entsprechende Wissensrepräsentation beschrieben. Den Anfang machte im Arbeitspaket der Wissensakquisition eine Befragung zur Feststellung des Bildungsbedarfs. Im Anschluss daran wurden Expertenbefragungen zur Curricular-Planung und zum Aufbau einer Entwurfsmuster-Ontologie durchgeführt. Ihren Einsatz findet die Entwurfsmuster-Ontologie im Cased-Based-Reasoning-System/Dialogspiel. Eine Erweiterung der Ontologie um Wahrscheinlichkeiten (bzgl. einzelner Muster, d. h. Zustände und Zustandübergänge) erlaubt die Generierung eines Bayes-Netzes. Das Hypothesentesten basiert auf diesem Bayes-Netz.

1.1.1.1. UAP 1.2: Wissensakquisition

Ein Bestandteil der Wissensakquisition war (besonders zu Projektbeginn) eine Fragebogenaktion zur Feststellung des Bildungsbedarfs. Der Fragebogen befasste sich mit dem Mustereinsatz in der Praxis und hatte das Ziel, den Informationsbedarf (der Softwareentwickler) festzustellen. Gefragt waren u. a. praxisrelevante Entwurfsmuster und sinnvolle Lernreihenfolgen für Kurse. An der Befragung nahmen insgesamt 52 Personen mit unterschiedlichen Programmiererfahrungen und Musterkenntnissen teil.

Die Auswertung erfolgte mit statistischen Methoden. Eine multiple lineare Regression z. B. belegt den Tool-Bedarf bei komplexeren Mustern. Aus berechneten Korrelationen ließ sich schließen, dass Muster nicht ohne Hilfsmittel angewendet werden, wenn diese subjektiv als komplex eingestuft werden.

Als weiterer Aspekt wurde die Frage nach der Voraussetzung für das Verstehen eines Musters gestellt („Welches Muster erleichtert das Verständnis anderer Muster?“). Ziel hierbei war, eine partielle Ordnung zwischen den Mustern zwecks Curricular-Planung aufzubauen.

¹ <http://inpulse.uni-oldenburg.de/inpulse/>

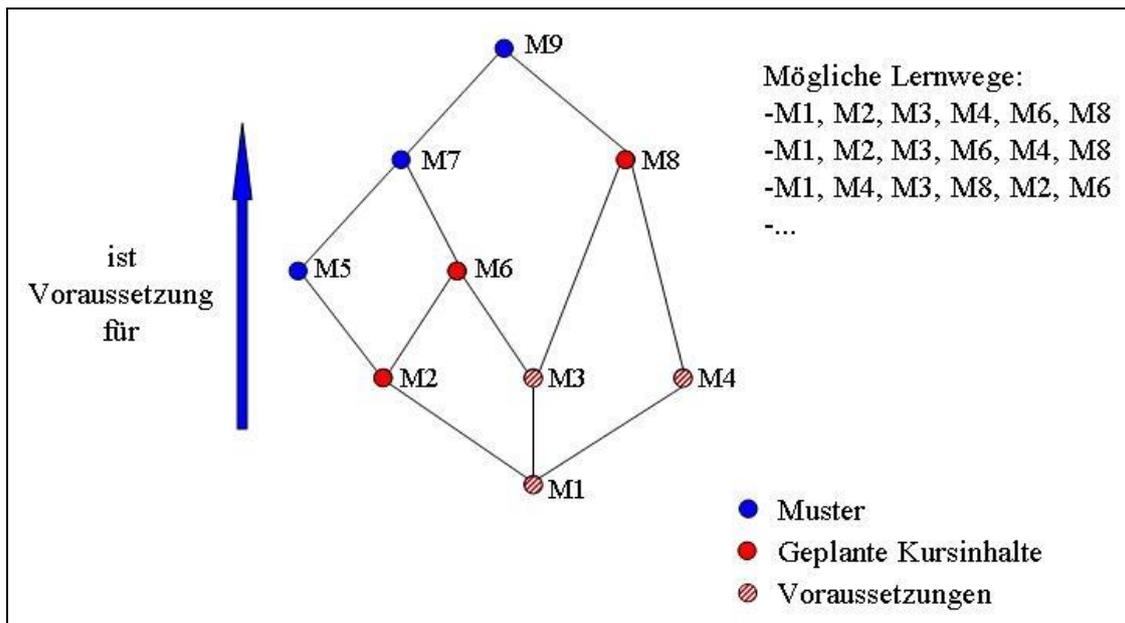


Abbildung 2: Curricular-Planung mit Hasse-Diagramm

In der Abbildung 2 ist ein fiktives Hasse-Diagramm mit den Konzepten M1, M2, ... M9 zu sehen. Diese Konzepte stehen in diesem Fall jeweils für ein Entwurfsmuster. Das Diagramm enthält die Muster mit aufsteigendem Schwierigkeitsgrad, angeordnet von unten nach oben. Der Schwierigkeitsgrad wird hier insbesondere dadurch gekennzeichnet, dass ein Muster alle direkten und indirekten Vorgänger als Vorwissen voraussetzt. Demnach ist z. B. das Muster M1 Voraussetzung für die Muster M2, M3 und M4. Gemäß der Transitivität aber auch für alle anderen Muster dieses Hasse-Diagramms. Soll ein Kurs mit den Inhalten M2, M6 und M8 angeboten werden, sind notwendige Voraussetzung für das Verständnis die Muster M1, M3 und M4. Über die Reihenfolge von Mustern auf gleicher Ebene wird in dieser Ordnung noch keine Aussage gemacht, so dass verschiedene Lernpfade möglich sind.

Die Fragebogenauswertung hat allerdings aufgrund einiger Antworten nicht zu einer partiellen Ordnung in Form eines Hasse-Diagramms geführt. Dies waren Antworten, die bzgl. der Ordnung von Mustern Zyklen und Schleifen zur Folge haben. Da die Voraussetzungsanalyse (iterative Vertiefung der Lerninhalte) aber für die Curricularplanung sehr relevant ist, wurde eigens zu diesem Zweck das KARaCAs-Tool um den Paarvergleichsalgorithmus GRAP (**G**reedy **K**nowledge **A**cquisition **P**rocedure) erweitert. Der Paarvergleichsalgorithmus ist in [MSG05] (vgl. auch Abschnitt 4) ausführlicher erläutert. Die Veröffentlichung ist diesem Abschlussbericht beigelegt. Die Abbildung 3 zeigt einen Ausschnitt des Tools, in dem die Muster Dekorierer und Fassade geordnet werden.

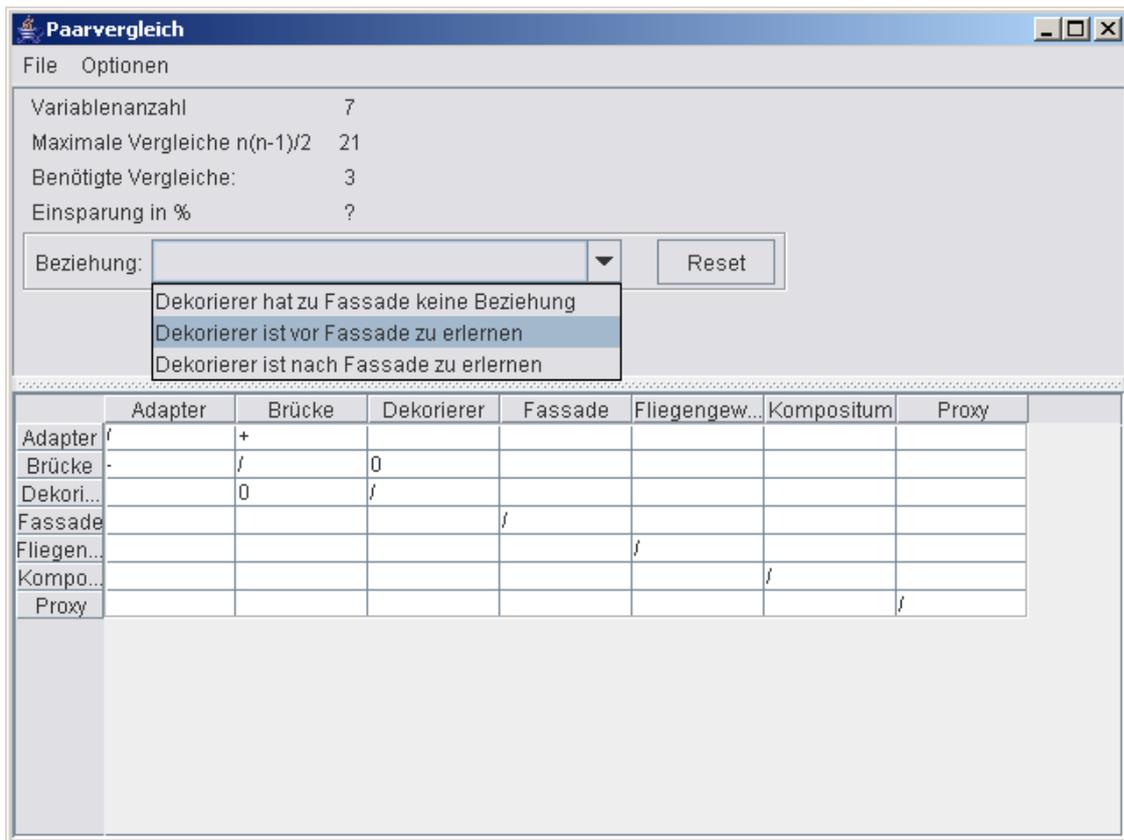


Abbildung 3: Screenshot des Paarvergleich-Tools

Ein trivialer Paarvergleich aller 23 Muster würde zu 253 ($=n*(n-1)/2$) zu bewertenden Paaren führen. Durch die Anwendung von GRAP konnte diese Anzahl auf 136, 104 bzw. 73 Paarvergleichen bei 3 befragten Experten reduziert werden. Dies entspricht einer Einsparung von 47 bis 71 Prozent.

1.1.2. UAP 1.3: Aufbau der Musterontologie

Den Kernbereich des Arbeitspaketes der Wissensakquisition bildete die Expertenbefragung zum Aufbau der Musterontologie.

Ein besonders relevanter Aspekt beim Ontologieaufbau ist das Herausarbeiten von Gemeinsamkeiten und Unterschieden zwischen den einzelnen Entwurfsmustern. Werden z. B. ausschließlich unterscheidende Merkmale von den Experten benannt, führt dies zu einer sehr breiten Ontologie mit vielen Konzepten auf gleicher Ebene. Werden dagegen viele Konzepte mit gemeinsamen Merkmalen annotiert und die Unterschiede vernachlässigt, sind in dieser Ontologie „Muster-Cluster“ enthalten, die es nur schwer erlauben, ein konkretes Muster zu identifizieren. Aus diesem Grund wurden die Experten-Interviews derart geleitet durchgeführt, dass die Experten explizit zur Angabe von diskriminierenden und gemeinsamen Merkmalen für konkrete Musterpaare/Mustermengen aufgefordert wurden.

Zu diesem Zweck wurde für die Befragung die Methode der Repertory Grid-Technik aus der Psychologie der Persönlichen Konstrukte von Kelly [SC93] adaptiert. Dabei handelt es sich um eine Methode zur Erfassung von Elementen, Konstrukten und den Relationen zwischen den Konstrukten. Konstrukte stehen für Merkmale, mit denen begriffliche Unterscheidungen vorgenommen werden können. Es wird die Gewohnheit ausgenutzt, in Kontrasten (Dichotomiesatz) zu denken: „Der Begriff

‚groß‘ existiert nicht ohne eine Vorstellung von ‚klein‘. Die Elemente sind Gegenstände, auf die sich die Konstrukte beziehen. [Sc92]

Die Formale Begriffsanalyse (FBA) ist eine mathematische Theorie zur qualitativen Analyse von Daten. Gegenstand der formalen Begriffsanalyse ist ein formaler Kontext. Ein formaler Kontext (G, M, I) ist wie folgt definiert, G ist eine Menge von (formalen) Gegenständen (Objekten), M ist eine Menge von (formalen) Merkmalen (Attributen) und der Inzidenzrelation $(I \subseteq G \times M)$. Der Ausdruck g / m bedeutet: "Der Gegenstand g hat das Merkmal m ". Ein formaler Kontext wird als Kontexttabelle veranschaulicht, in der Kreuze die Elemente der Inzidenzrelation kennzeichnen [GW96] [Li99].

Ein Begriffsverband kann gesehen werden als [GW96]:

- Hierarchische Klassifikation von Gegenständen.
- System von Merkmalsimplikationen.
- Struktur zur Darstellung und Abfrage von Wissen.

Ein Anwendungsgebiet der FBA ist die Unterstützung des Ontologie Engineering. Sie dient der Extraktion von Begriffshierarchien aus gegebenen Datensätzen. Zur Identifizierung der Konzepte einer Ontologie können sowohl die formalen Konzepte (Begriffe) als auch Attribute der FBA herangezogen werden, da es im Bereich der Ontologien keine scharfe Unterscheidung zwischen Konzepten und Attributen wie in der FBA-Theorie gibt [C+04]. Im Rahmen des Projekts InPULSE werden die durch Expertenbefragung entstanden Ontologien mit Hilfe der FBA zusammengeführt, eine mögliche Methode (FCA-MERGE) wird in [SM01] beschrieben. Die FBA ist nicht nur Hilfsmittel des Ontologie Engineering, sondern wird auch in (anderen) Bereichen des Software Engineering verwendet, Requirements Analyse, Formal Specification, Komponenten Retrieval, Reengineering, Reverse Engineering [Li99] [T+04].

Die Kombination von Repertory Grids zur Befragung und der FBA zur Auswertung der Ergebnisse wurde bereits in verschiedenen Domänen eingesetzt, z. B. [SW88].

In InPULSE wurde **KARaCAs** entwickelt, um die Befragung von Experten für Entwurfsmuster zu unterstützen.

Um innerhalb des Repertory Grids auch Merkmale zuzulassen, die keine „echten“ Gegensatzpaare bilden, wurde die Methode adaptiert:

Im ersten Schritt (1) werden die Elemente erhoben bzw. vorgegeben: die relevanten Entwurfsmuster. Der zweite Schritt (2) erfasst die Merkmale, indem der Proband für jeweils drei Elemente (Triade) angibt, „in welcher Weise zwei der drei Elemente einander ähnlich sind und sich darin von dem dritten unterscheiden“. Im dritten Schritt (3) werden die Experten gebeten, aus der Gesamtmenge der Objekte (hier Mustern) so viele wie möglich einem der beiden Merkmale zuzuordnen. Im vierten Schritt (4) kann der Experte quantifizieren, mit welcher Sicherheit das Merkmal (die Anwendungssituation) auf das Muster zutrifft.

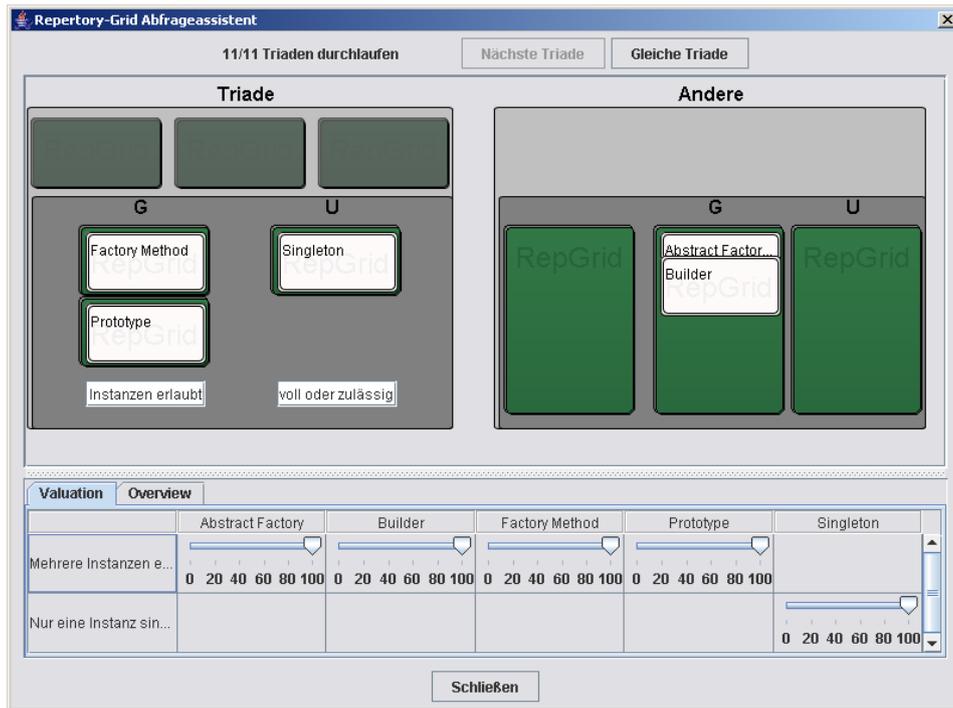


Abbildung 4. Grafische Benutzeroberfläche von KARaCAs

Da es sich bei der Repertory-Grid Methode nach Kelly um eine Kartenlegetechnik handelt, wurde die grafische Benutzeroberfläche von KARaCAs an diese Kartenmetapher angelehnt (vgl. Abbildung 4)

Die erhobenen Daten können mithilfe der FBA analysiert (vgl. Abbildung 5) und als Liniendiagramm (vgl. Abbildung 6) dargestellt werden.

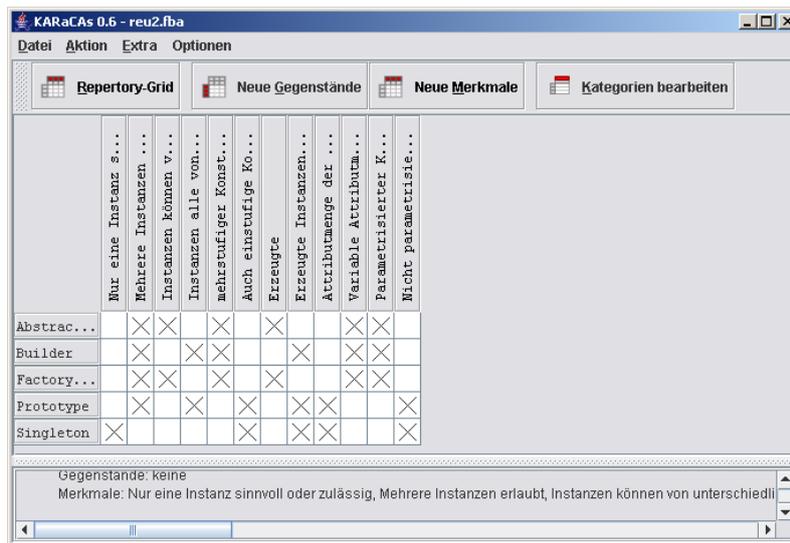


Abbildung 5. Aus dem Repertory Grid erzeugter Kontext

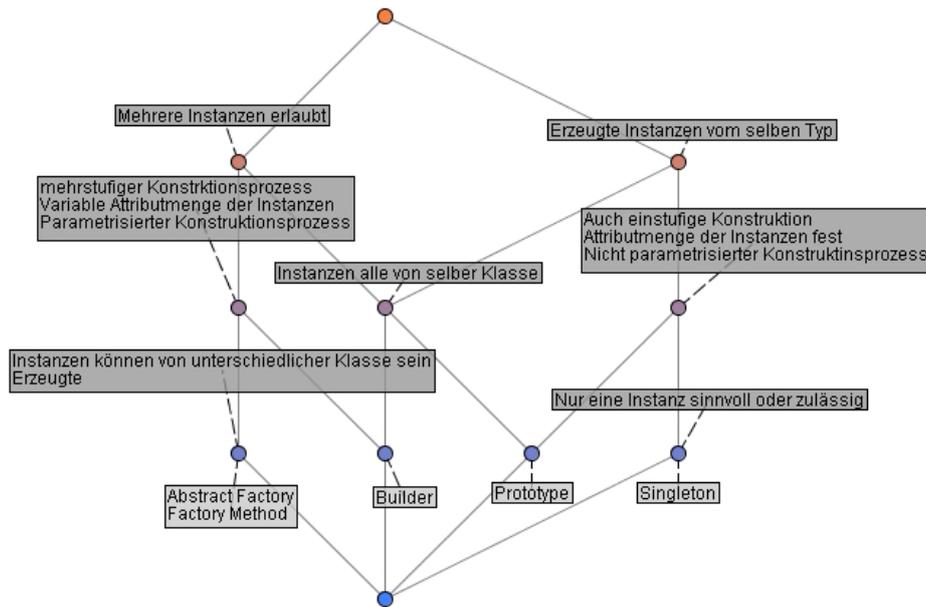


Abbildung 6. Aus dem Kontext erzeugtes Liniendiagramm

Um die mit KARaCAs entwickelte Ontologie und die zusätzlich angegebenen Sicherheitsmaße für das Dialogsystem zu operationalisieren, wird aus den erhobenen Daten ein Bayes-Netz generiert (z. B. [Pe97], [FMS96], [SMF96]).

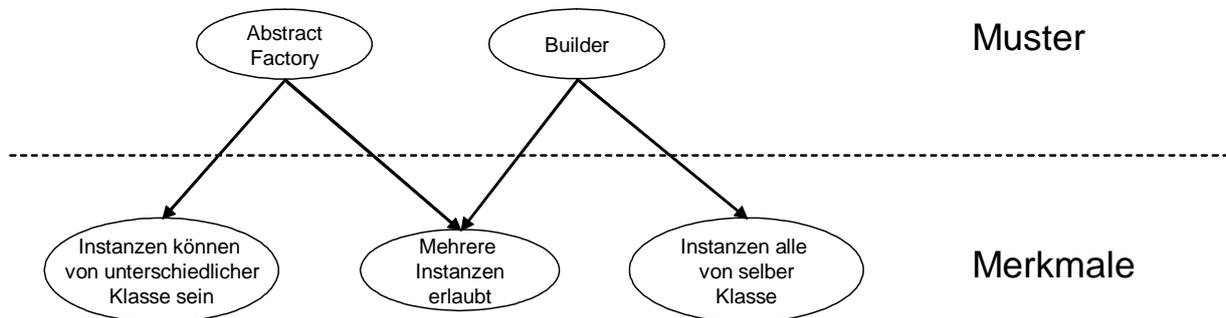


Abbildung 4. Qualitative Struktur des Bayesnetzes

Dazu wird die qualitative Struktur des Bayes-Netzes in zwei Schichten (s. Abbildung 4) aufgeteilt: Auf der obersten Schicht befinden sich die Muster selbst, diese besitzen Merkmale, die in der zweiten Schicht repräsentiert werden. Die verschiedenen Muster und ihre Merkmale, die durch das Repertory-Grid-Verfahren gewonnen wurden, werden in diese Schichten eingeordnet und zueinander in Beziehung gesetzt. Dabei kann der Benutzer bereits während des Repertory-Grid-Verfahrens bewerten, wie sicher ein Muster ein bestimmtes Merkmal erfüllt. Diese Sicherheiten bilden die Grundlage zur Berechnung der Wahrscheinlichkeiten im modellierten Bayes-Netz.

Das Bayes-Netz kann eingesetzt werden, um die Einsatzmöglichkeit eines Musters unter bestimmten vorherrschenden Fakten zu prüfen. Dazu wird der Anwender in einem **Dialogspiel** zum Vorhandensein verschiedener Merkmale befragt. Die daraus resultierenden Fakten werden als so genannte Evidenzen in das Bayes-Netz eingetragen. Dieses berechnet daraufhin die Wahrscheinlichkeiten für die

Einsatzmöglichkeit der modellierten Muster und liefert dem Anwender einen entsprechenden Designvorschlag.

1.2. HAP 2: Verfahren zur Design-Überprüfung auf Mustermerkmale

Im Hauptarbeitspaket 2 wird ein Case-Based-Reasoning-Verfahren zur Unterstützung des Forward Engineerings entwickelt. Grundlage des Verfahrens ist die Entwurfsmuster-Ontologie aus HAP 1, die hier zu einem Bayes-Netz weiterentwickelt wurde. Weiterhin wurde das Forward Engineering durch eine Design-Überprüfung unterstützt. Dazu wurde ein Verfahren zum Hypothesen-Testen konzipiert und implementiert. Durch HAP 1 und HAP 2 haben wir so einen werkzeugunterstützten Workflow gestaltet, der die Wissensakquisition, die Wissensrepräsentation und schließlich auch die Wissensanwendung abdeckt.

1.2.1. UAP 2.2: Entwicklung des Case Based Reasoning Systems

Im Unterarbeitspaket 2.2 wurde ein Case-Based-Reasoning (CBR)-Verfahren zur Unterstützung des Forward Engineerings in Form eines kritischen Dialogs (Dialogspiel) entwickelt. Im Verlaufe des Dialogs werden anwendungsspezifische Merkmale erfragt und kritisch hinterfragt. Sukzessiv werden auf Basis der Musterontologie aus HAP 1 zu allen Mustermerkmalen Ja/Nein-Fragen gestellt.

1.2.1.1. CBR-Verfahren

Das Wissen wird in Form von so genannten Fällen in einer Fallbasis gespeichert. Besteht eine Ähnlichkeit zwischen der aktuellen Problemsituation und einem Fall in der Fallbasis, wird die entsprechende Lösungsmethode für die aktuelle Situation vorgeschlagen. Grundlage der Ähnlichkeitsberechnung sind die Anwendungsmerkmale, die in der Ontologie aus HAP 1 definiert werden. Die Fälle (Problembeschreibungen) bestehen aus einer Menge von Anwendungsmerkmalen, die zugehörigen Lösungen entsprechen den Entwurfsmustern. Das CBR-System erhält als Input die vom Benutzer erfragten Anwendungsmerkmale. Für diese Problembeschreibung wird in der Fallbasis der Fall mit der höchsten Ähnlichkeit gesucht. Die Lösung, die für den Fall in der Fallbasis in der Vergangenheit bereits erfolgreich war, wird jetzt für das aktuelle Problem vorgeschlagen. Die Ähnlichkeit der Merkmale wird durch das Liniendiagramm (Ergebnis der Formalen Begriffsanalyse) dargestellt (Abbildung 7 und Abbildung 8).

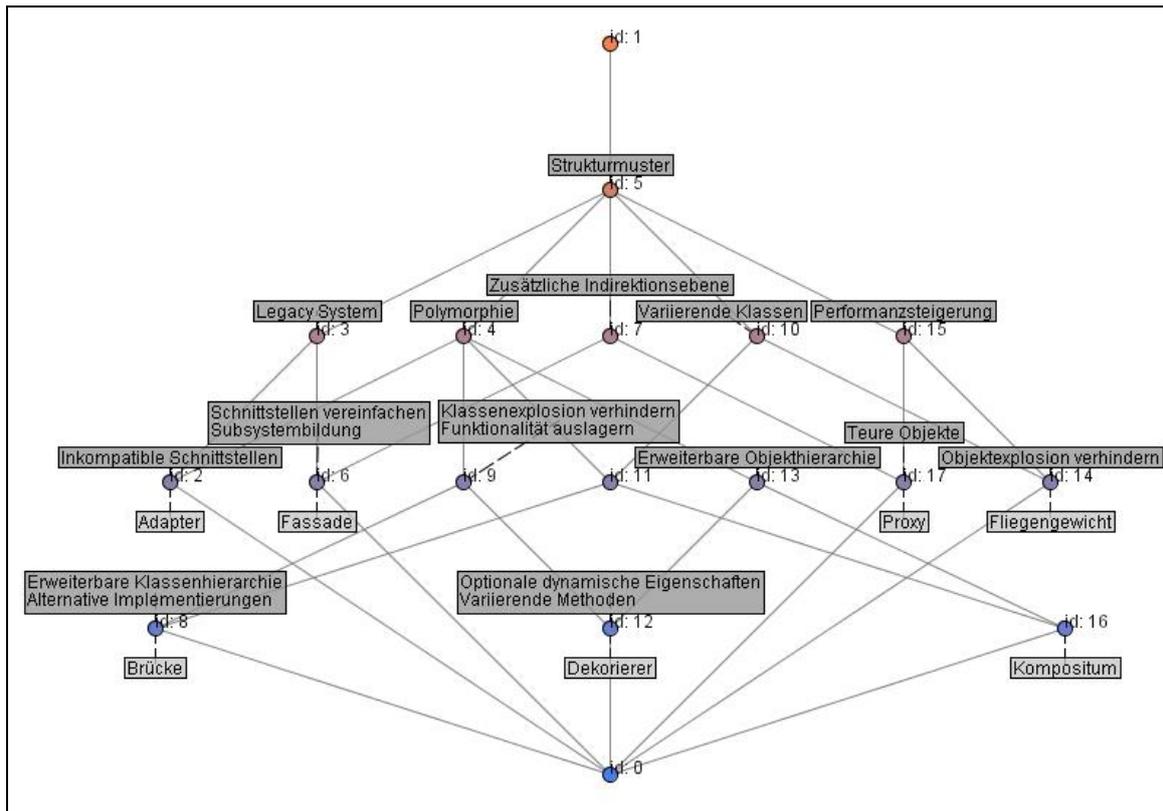


Abbildung 7: Liniendiagramm als Ergebnis der Formalen Begriffsanalyse

Die Abbildung 8 illustriert folgendes Beispiel: Im Rahmen des kritischen Dialoges werden nacheinander die Anwendungsmerkmale „Strukturmuster“, „Perfomanzsteigerung“ und „Teure Objekte“ ermittelt. Aufgrund dieser Problembeschreibung wird in der Fallbasis eine ähnliche Problembeschreibung gesucht. Die größte Ähnlichkeit besteht zu einem Fall mit den Anwendungsmerkmalen „Strukturmuster“, „Perfomanzsteigerung“, „Teure Objekte“ und „Zusätzliche Indirektionsebene“. Die zugehörige Lösung ist das Proxy-Entwurfsmuster, das nun auch für das aktuelle Problem empfohlen wird. Das Liniendiagramm wird sowohl zur Durchführung des kritischen Dialoges (Auswahl von Dialogzügen) wie auch zur Ähnlichkeitsbestimmung der Fälle herangezogen.

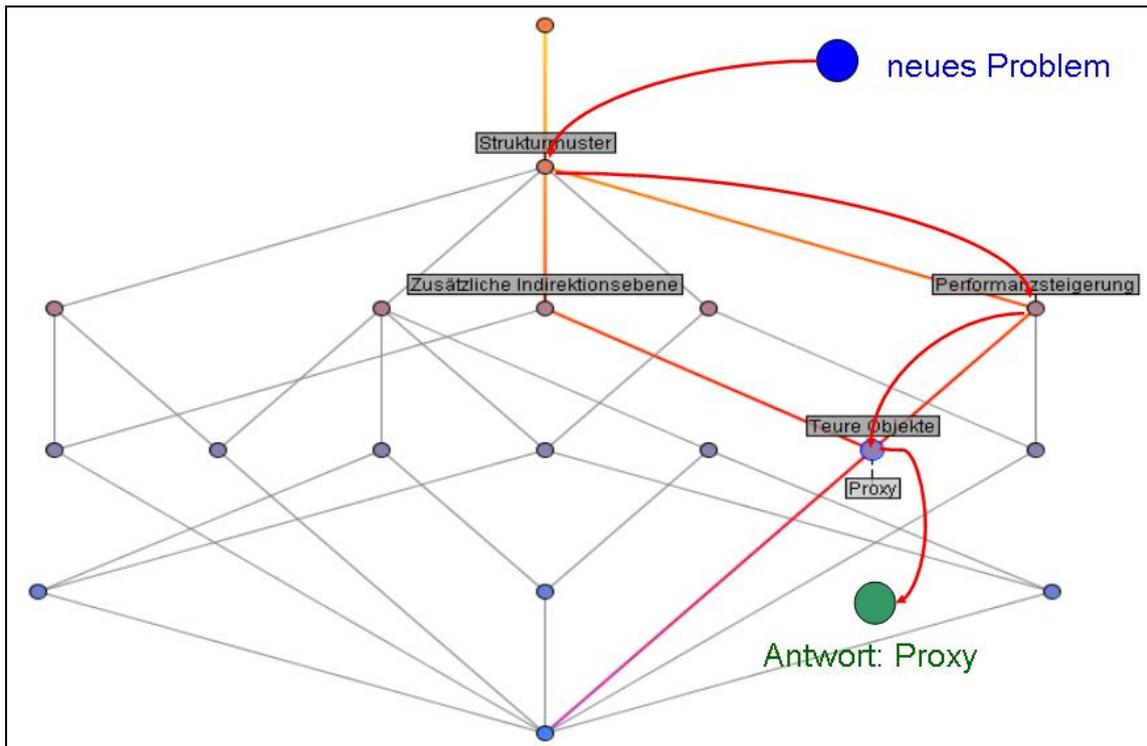


Abbildung 8: Beispielhafte Nutzung des CBR-Systems

1.2.1.2. Korrespondenzregeln

Um einen Dialog mit dem Benutzer über die Merkmale führen zu können, müssen die Merkmale durch Fragen ergänzt werden, mit deren Hilfe sie vom Benutzer erfragt werden. Um den Dialog möglichst flexibel und an den Benutzer anpassbar zu machen, werden diese Fragen in verschiedenen Abstraktionsebenen oder Sichten angelegt. Momentan realisiert sind Fragen aus der Sicht der Objektorientierten Programmierung (OOP) und Fragen aus der Anwendungssicht. Die Fragen aus der OOP-Sicht wird ein Benutzer z. B. beantworten können, wenn er sich bereits im Entwurf befindet und mit dem objektorientierten Vokabular vertraut ist. Falls er sich noch relativ am Anfang seines Entwurfs befindet, oder eine Frage aus der OOP-Sicht nicht versteht, so können die Fragen aus der Anwendungssicht hilfreich sein. Die Fragen werden den Merkmalen des Verbandes durch so genannte Korrespondenzregeln zugeordnet, z. B.:

Merkmal: „Inkompatible Schnittstellen“

- o OOP: Haben Sie ein Problem mit inkompatiblen Schnittstellen?
- o AS: Soll Code von außerhalb verwendet werden, aber die Namen und Parameterlisten der Methoden sind nicht gleich oder es kann nicht passend abgeleitet werden?
- o AS: Soll eine bestehende Klasse oder Methode genutzt werden, deren Schnittstelle aber nicht Ihren Bedürfnissen entspricht?
- o AS: Haben die bereits vorhandenen Klassen unveränderliche Schnittstellen oder ist der Sourcecode nicht zugänglich?

Die Intention aller drei Fragen ist, zu erfahren, ob der Anwender ein Problem mit

inkompatiblen Schnittstellen hat. Während im ersten Fall direkt danach gefragt wird, wird im zweiten und dritten Fall ein Anwendungsproblem erfragt, das als Indikator für inkompatible Schnittstellen dient.

1.2.1.3. Arbeitsweise des Dialogsystem

Mithilfe des Dialogs wird, wie oben beschrieben, die Problemsituation des Benutzers bestimmt. Wenn sich dabei eine Anwendungssituation für ein Muster erkennen lässt, wird dieses Muster vorgeschlagen. Zur Umsetzung des Dialogsystems wurden zwei alternative Vorgehensweisen evaluiert.

1.2.1.3.1. Dialogführung mit Begriffsverbänden

Grundlage für die erste Version des Dialogs bilden zum einen der Begriffsverband und zum anderen die Korrespondenzregeln.

Während des Dialogs wird der Begriffsverband von dem allgemeinsten Begriff (dem obersten Knoten) aus bis zu einem Begriff durchlaufen, der ein Muster enthält. Dabei werden dem Benutzer Fragen aus den, den Attributen der Begriffe zugeordneten, Korrespondenzregeln gestellt, um die Problemsituation des Benutzers einzuschätzen. Der Benutzer kann diese Fragen mit „ja“, „nein“ und „weiß ich nicht“ beantworten. Abbildung 9 stellt den Dialogablauf als Aktivitätsdiagramm dar. Beim Durchlaufen des Verbandes werden die Knoten, Attribute und Fragen markiert. Knoten werden mit „besucht“ markiert, sobald ein zugehöriges Attribut vom Benutzer erfragt wird oder der Knoten kein Attribut enthält. Wird die Frage nach einem Attribut vom Benutzer positiv beantwortet, so wird der Knoten mit „ja“ markiert. Wird die Frage dagegen negativ beantwortet, so wird der Knoten und alle Unterbegriffe beinhaltenden Knoten mit „nein“ markiert. Ein „nein“ muss durch den Graphen propagiert werden, da ein Attribut, das für einen Begriff gilt auch für alle seine Unterbegriffe gültig ist. Attribute werden mit „besucht“ markiert, sobald der Benutzer nach ihnen gefragt wird. Wird die Frage positiv beantwortet werden sie mit „ja“ markiert, bei einer negativen Antwort mit „nein“. Gleiches gilt für die zu den Attributen gehörigen Fragen. Da es zu einem Attribut mehrere Fragen gibt, müssen diese einzeln markiert werden. Der Dialog beginnt bei dem Wurzelknoten des Verbandes (dem allgemeinsten Begriff). Enthält er bereits ein Attribut, so wird der Benutzer nach dem Vorhandensein dieses Attributs in seiner Problemsituation gefragt. Anfangs wird eine Frage aus der OOP-Sicht gestellt, kann er diese Frage nicht beantworten, so wird die Sicht gewechselt und eine Frage aus der „Anwendungssicht“ zu dem gleichen Attribut gestellt. Kann auch diese Frage nicht beantwortet werden, so wird die nächste Frage aus der „Anwendungssicht“ gestellt, usw. Wird die Frage mit „nein“ beantwortet, so kann in diesem Fall kein Muster vorgeschlagen werden, da das Attribut, welches in dem obersten Begriff enthalten ist, auch für alle Unterbegriffe gilt. Beantwortet er eine Frage mit „ja“, so wird der Dialog mit einem der Unterbegriffe fortgesetzt. Für die Antworten „ja“ und „weiß ich nicht“ gilt hier das gleiche wie bereits beschrieben. Wird hier jedoch mit „nein“ geantwortet, so wird der Knoten und alle Knoten von Unterbegriffen mit „nein“ markiert und der Dialog mit einem nicht markierten Kindknoten eines mit „ja“ markierten Elternknotens fortgesetzt. Wird bei diesem Durchlaufen des Verbandes ein Knoten erreicht, der ein Muster enthält, so wird dem Benutzer dieses Muster für den Einsatz vorgeschlagen. Kann durch die Antworten des Benutzers kein solcher Knoten erreicht werden, so kann kein Mustervorschlag gemacht werden.

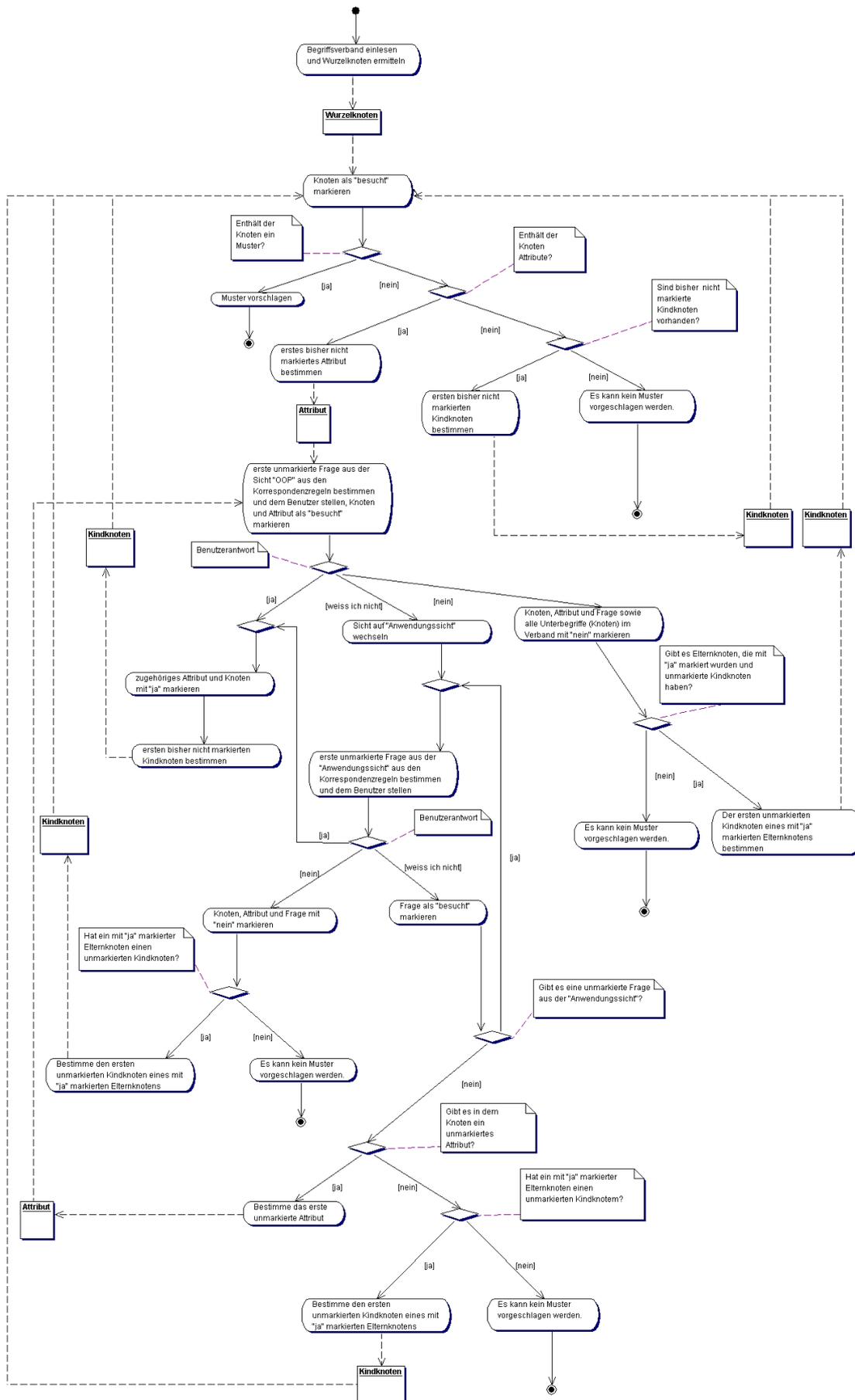


Abbildung 9: Aktivitätsdiagramm für den Dialogablauf

1.2.1.3.2. Wissensrepräsentation mit Bayes-Netzen

In dieser ersten Version des CBR-Verfahrens (E 2.2.1) wurde ein Vorgehen für den Dialog vorgestellt, das direkt auf der Struktur des Begriffsverbandes arbeitet. Um die Anwendungssicht zu integrieren, wurden Korrespondenzregeln angegeben, die die Fragen aus dieser Sicht den Merkmalen aus der OOP-Sicht zuordnen. Bei diesem Vorgehen ergeben sich jedoch Probleme, die zum Teil durch einen erhöhten Aufwand innerhalb des Algorithmus, der auf den Verbänden arbeitet, gelöst werden können. Im Einzelnen ergeben sich folgende Probleme:

- „Oder-Beziehungen“ zwischen Merkmalen:

Zum Teil können zwischen Merkmalen, die einem Muster zugeordnet werden „Oder-Beziehungen“ bestehen. Ein Beispiel ist: Das Proxy-Muster kann eingesetzt werden, um die Zugriffszeiten auf ein Objekt oder die Sicherheit beim Zugriff zu erhöhen. Da solche „Oder-Beziehungen“ nicht im Begriffsverband modelliert werden können, ist eine Möglichkeit abstraktere Beschreibungen zu finden, die beide Situationen abdecken. Eine andere wäre, den Begriffsverband mit einer geeigneten Datenstruktur zu ergänzen.

- Ergebnisse aus Voranalysen:

Es ist ein möglichst früher Einsatz des Dialogsystems, also möglichst bereits während der Analysephase, intendiert. Daher ist es möglich, dass aus dieser Phase bereits Merkmale vorliegen, der Dialog aber noch nicht abgeschlossen werden konnte, da noch Informationen zur eindeutigen Identifizierung der Situation fehlten. Ein so begonnener Dialog kann dann während der späteren Entwurfsphase fortgesetzt werden. Die Informationen, die aus einer solchen Voranalyse stammen, können in dem Begriffsverband „verstreut“ vorliegen. Um diese Informationen in die Dialogsteuerung aufnehmen zu können, müsste der oben (E 2.2.1) vorgestellte Algorithmus erweitert werden.

- Gemeinsam verwendete Muster:

Einige Muster werden häufig gemeinsam verwendet, wie z. B. die Abstrakte Fabrik und das Singleton-Muster, denn bei der Implementierung der abstrakten Fabrik wird häufig das Singleton-Muster verwendet. [Gamma+01] gibt eine Grafik dieser Abhängigkeiten an. Eine solche Beziehung kann ebenfalls in dem Begriffsverband nicht dargestellt werden, wenn die Merkmale aus den Anwendungssituationen gewählt werden. Denkbar wäre nun, einen zweiten Begriffsverband zu erstellen, der diese Abhängigkeit abbildet und beide Begriffsverbände für den Dialog zu nutzen. Die genannten drei Probleme lassen sich durch eine Ergänzung des Algorithmus durchaus lösen, jedoch mit einem gewissen Aufwand.

Ein Problem, welches sich zum Teil bei den Interviews mit den Experten ergeben hat, lässt sich allerdings mit den Begriffsverbänden nicht auflösen:

- Entscheidungen unter Unsicherheit:

Zum einen können die Experten für die Merkmale nur einen gewissen Grad an Sicherheit angeben, mit dem diese auf die Muster zutreffen. So ist zum Beispiel das Vorhandensein von altem einzubindendem Programm-Code ein Hinweis darauf, dass ein Adapter verwendet werden sollte, es ist aber nicht zwingend notwendig. Es ist ebenso möglich, dass eigene Klassen nicht direkt geändert werden sollen und deshalb ein Adapter verwendet werden kann. Zum anderen kann bei den Benutzern

des Dialogs Unsicherheit über die eigene Modellierungssituation bestehen. So kann es vorkommen, dass sie eine Frage anders beantworten, als es in ihrer Situation erwartet würde. Um die geschilderten Probleme zu lösen, wurde der Begriffsverband für die zweite Version des Dialogs um ein Bayes-Netz ergänzt.

1.2.1.3.2.1. Bayes-Netze

Die wahrscheinlichkeitstheoretische Methode der Bayes-Netze ist eine Modellierungssprache durch die unsicheres und unvollständiges Expertenwissen über eine Domäne zum Ausdruck gebracht werden kann. Gerade in Domänen unsicheren Wissens sind Bayes-Netze ideal für die Entwicklung von Systemen zur Entscheidungsfindung ([Pearl97]; [Castillo97]; [Cowell99]; [Jensen01]).

Die Modellierung des unsicheren Wissens erfolgt in Bayes-Netzen auf zwei Ebenen: Auf der qualitativen Ebene ist ein Bayes-Netz ein azyklischer gerichteter Graph, in dem die unterschiedlichen Konzepte der Domäne einheitlich durch Variablenknoten und Beziehungen zwischen diesen Konzepten durch Pfeile repräsentiert werden. Zudem besitzt jeder Variablenknoten verschiedene Ausprägungen, die er annehmen kann. Auf der quantitativen Ebene wird die Unsicherheit der Zusammenhänge im Netz spezifiziert. Diese Angaben erfolgen als Wahrscheinlichkeiten, bei denen die Ausprägungen der direkten Vorgängerknoten berücksichtigt werden müssen.

In InPULSE werden Bayes-Netze eingesetzt, um das Wissen über die Anwendbarkeit eines Musters zu modellieren. Dazu wird die qualitative Struktur des Bayes-Netzes in drei Schichten (s. Abbildung 10) aufgeteilt: Auf der obersten Schicht befinden sich die Muster selbst, diese besitzen Merkmale, die in der zweiten Schicht der OOP-Sicht repräsentiert werden. Diese Merkmale lassen sich in der Anwendungssicht, der dritten Schicht, durch Frühindikatoren identifizieren. Diese Frühindikatoren entsprechen den Fragen aus der Anwendungssicht im Begriffsverband, so dass durch diese Art der Modellierung keine zusätzlichen Korrespondenzregeln notwendig sind. Die verschiedenen Muster und ihre Merkmale, die durch das Repertory-Grid-Verfahren (E 1.3.1) gewonnen wurden, werden in diese Schichten eingeordnet und zueinander in Beziehung gesetzt. Dabei kann der Benutzer bereits während des Repertory-Grid-Verfahrens bewerten, wie sicher ein Muster ein bestimmtes Merkmal erfüllt. Diese Sicherheiten bilden dann die Grundlage zur Berechnung der Wahrscheinlichkeiten im modellierten Bayes-Netz. Durch die angegebenen Wahrscheinlichkeiten und die Netzstruktur lassen sich problemlos die oben erwähnten „Oder-Beziehungen“ modellieren. Ein weiterer Vorteil des Einsatzes von Bayes-Netzen ist, dass „verstreute“ Informationen aus Voranalysen durch die Inferenzalgorithmen im Netz propagiert und ohne zusätzlichen Aufwand für den Dialog genutzt werden können.

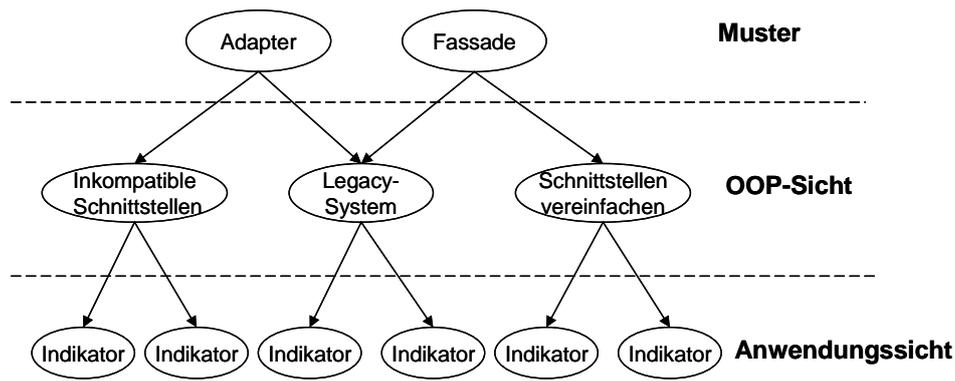


Abbildung 10: Qualitative Struktur des Bayes-Netzes

Das Bayes-Netz wird eingesetzt, um die Einsatzmöglichkeit eines Musters unter bestimmten vorherrschenden Fakten zu prüfen. Dazu wird der Anwender, wie bereits oben beschrieben, in einem Dialogspiel zum Vorhandensein verschiedener Merkmale befragt. Zur Ermittlung der Fragereihenfolge sind verschiedene Strategien unterschiedlicher Komplexität möglich (Zufällige Frageauswahl bis hin zu komplexen Sensitivitätsanalysen ([Jensen01]). Abhängig von der gewählten Strategie ändern sich sowohl die Fragereihenfolge als auch die Anzahl der benötigten Fragen, um ein Muster zu identifizieren. Die Fragen können sich dabei je nach Fragestrategie und Wissensstand des Anwenders direkt auf Merkmale aus der OOP-Sicht oder auf Frühindikatoren beziehen. Die aus den Antworten resultierenden Fakten werden als so genannte Evidenzen in das Bayes-Netz eingetragen. Dieses berechnet daraufhin die Wahrscheinlichkeiten für die Einsatzmöglichkeit der modellierten Muster und liefert dem Anwender einen entsprechenden Designvorschlag.

1.2.1.3.2.2. Beschreibung des auf Bayes-Netzen basierenden Dialogspiels

Mithilfe des Dialogs wird die Problemsituation des Benutzers bestimmt. Wenn sich dabei eine Anwendungssituation für ein Muster erkennen lässt, wird dieses Muster vorgeschlagen. Die Grundlage für die erste Version des Dialogs bildet das Bayesnetz (Abbildung 11), das auf Basis der Ontologie aus E 2.2.1 zu den Strukturmustern von [Gamma+01], modelliert wurde.

Während des Dialogs werden dem Benutzer Fragen zu verschiedenen Merkmalen auf den unterschiedlichen Ebenen gestellt, um die Problemsituation des Benutzers einzuschätzen. Da derzeit nur die Ebenen Muster und OOP-Sicht modelliert sind, erhält der Benutzer ausschließlich Fragen auf OOP-Ebene. Der Benutzer kann diese Fragen mit „ja“, „nein“ und „ich bin mir nicht sicher“ beantworten. Abbildung 12 stellt den Dialogablauf als Aktivitätsdiagramm dar.

Jedes Merkmal, zu welchem dem Benutzer eine Frage gestellt wurde, wird als bereits erfragt markiert, um zu verhindern, dass Fragen zu Merkmalen doppelt gestellt werden. Allerdings hat der Benutzer jederzeit die Möglichkeit bereits gegebene Antworten zu erfragten Merkmalen nachträglich zu ändern. Wird die Frage zu einem Merkmal von dem Benutzer mit „ja“ bzw. „nein“ beantwortet, wird die Antwort als Evidenz in das Bayes-Netz eingetragen und propagiert. Kann der Benutzer eine Frage nicht eindeutig beantworten („ich bin mir nicht sicher“), wird lediglich das Merkmal als erfragt markiert. Es wird keine Evidenz in das Bayes-Netz eingetragen.

Diese Version des Dialogspiels verfolgt eine einfache Fragestrategie: Dem Benutzer wird immer die Frage zu dem aktuell wahrscheinlichsten Merkmal gestellt. Durch Eintragen der Antworten des Benutzers in das Bayes-Netz verändern sich die vorherrschenden Wahrscheinlichkeiten, so dass abhängig von den Benutzerangaben ein unterschiedlicher Dialogverlauf entsteht. Zur dynamischen Reduzierung der betrachteten Knoten des Bayes-Netzes wird der Bayes-Ball-Algorithmus [Sch98] verwendet. Im optimalen Fall (alle Fragen werden durch den Benutzer mit „Ja“ beantwortet) ergibt sich durch die Anwendung des Algorithmus eine Reduzierung von 75 Prozent. Werden zufällig Fragen mit „Ja“, „Nein“ und „ich bin mir nicht sicher“ beantwortet, so beträgt die Reduzierung noch 30 Prozent. Der Dialog wird beendet und das identifizierte Muster dem Benutzer empfohlen, sobald durch die gegebenen Antworten die Wahrscheinlichkeit für die Anwendbarkeit eines Musters auf mehr als 90% steigt.

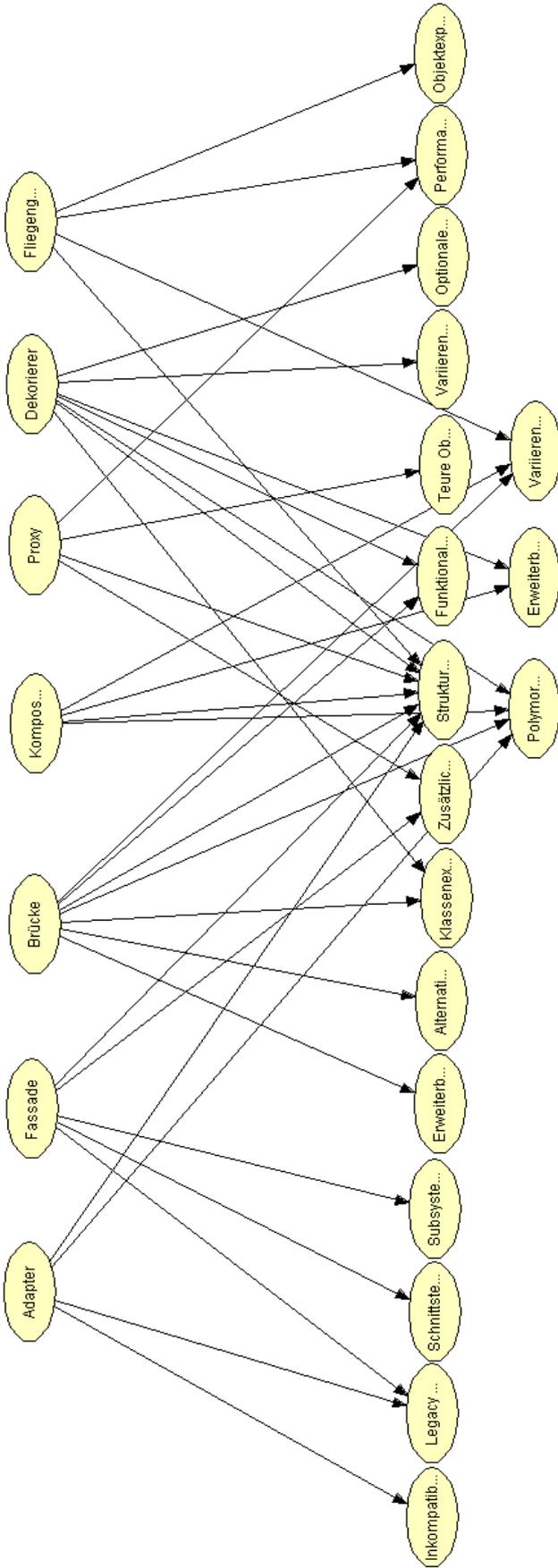


Abbildung 11: Bayes-Netz zu Strukturmustern

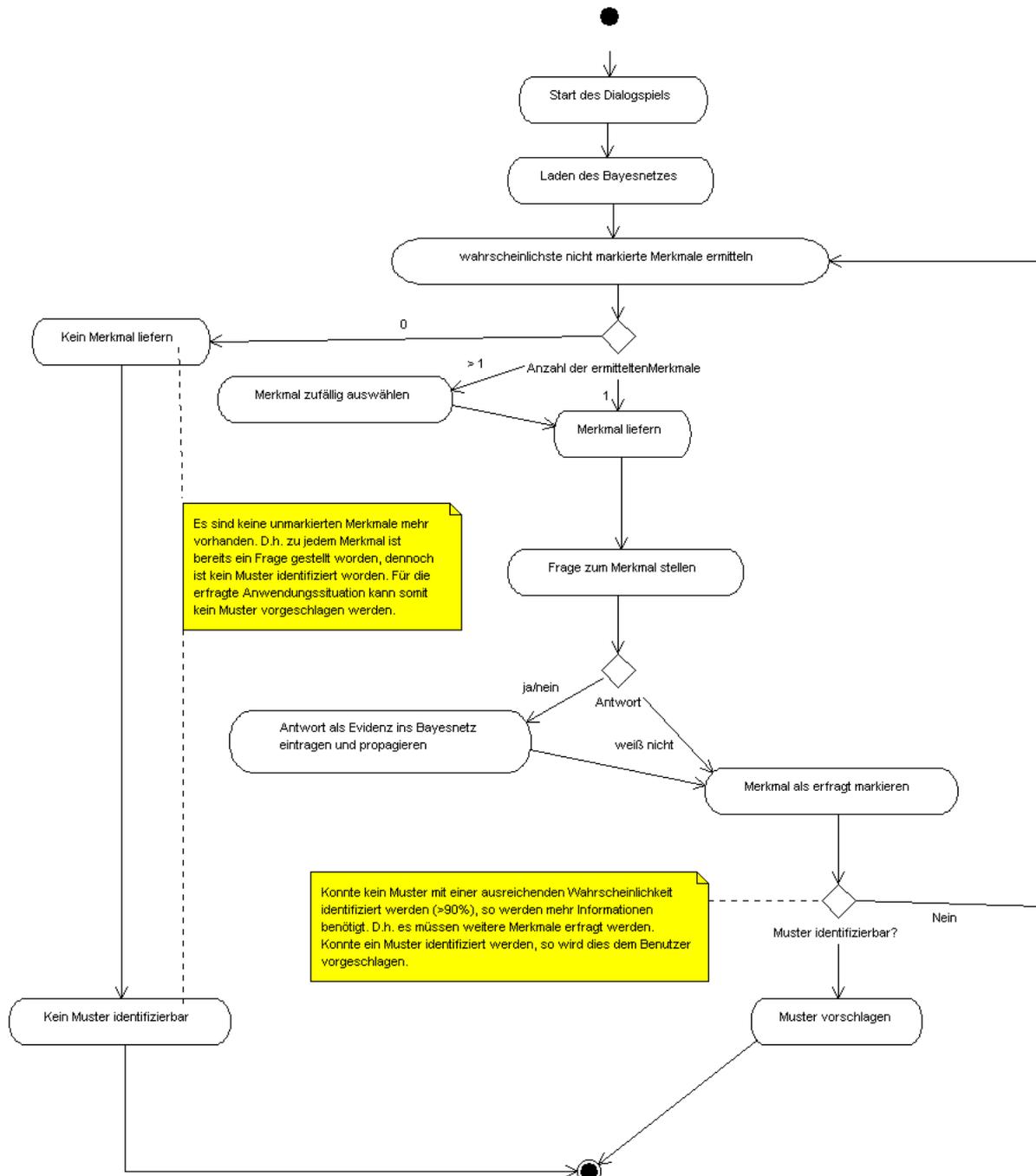


Abbildung 12: Aktivitätsdiagramm für den Dialogablauf mit einf. Fragestrategie

1.2.2. UAP 2.3: Entwicklung des Hypothesentesten

Dieser Abschnitt beschreibt die Konzeption und Realisierung des Hypothesentestens [Möbus95, Thole+97, Willms+00].

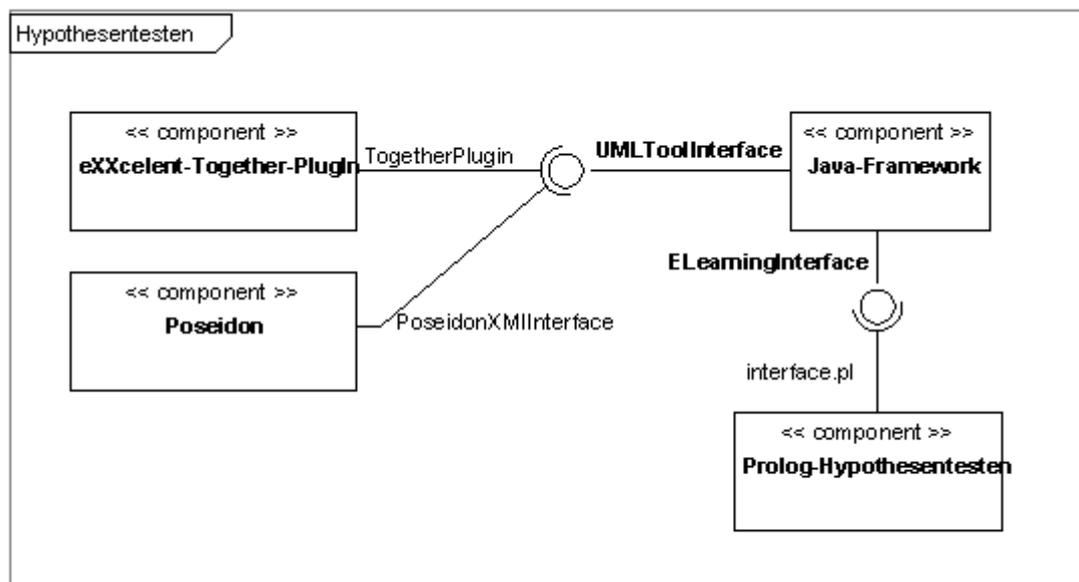
Beim Hypothesentesten bearbeiten Lernende Aufgaben einer Domäne. Die Aufgaben in InPULSE beziehen sich auf die Modellierung von UML-Klassendiagrammen unter Verwendung von Entwurfsmustern. Für InPULSE war die Integration des Verfahrens in das Tool Together der Firma Borland geplant, so dass Lernende dieses Tool zur Bearbeitung der Aufgaben verwenden können. Im Verlauf

der Bearbeitung einer Aufgabe kann der Lerner einen Teil seiner Lösung selektieren, den er für korrekt hält, und das System prüft mit Hilfe eines generativen Expertensystems, ob sich die Teillösung zu einer Gesamtlösung erweitern lässt. Auf diese Weise wird individuell auf den Lernerentwurf Bezug genommen. Im Rahmen des Forward-Engineering wird im ersten Schritt geprüft, ob die strukturellen Merkmale der in der Aufgabenbeschreibung geforderten Muster im Lösungsentwurf des Lerners vorhanden sind. Im negativen Fall hat der Lerner die Möglichkeit, bestimmte Teile seines Entwurfs zu markieren und somit als Hypothese zu kennzeichnen: „Dieser Teil meines Entwurfs ist Bestandteil des Musters M.“ Der markierte Teil des Entwurfs wird erneut auf das Vorhandensein struktureller Mustermerkmale geprüft. Im positiven Fall handelt es sich bei dem markierten Bereich um ein Musterfragment. Fällt die Prüfung wieder negativ aus, sind noch Elemente vorhanden, die nicht zu dem Muster gehören. Im weiteren Verlauf kann der Lerner seine Hypothese weiter einschränken und jeweils erneut prüfen lassen.

1.2.2.1. Architektur

Um das Hypothesentesten möglichst flexibel und toolunabhängig einsetzen zu können und die Anbindung an ein oder mehrere UML-Tools (z. B. Together, Poseidon) zu vereinfachen, wurde ein Java-Framework als Verbindungsglied entwickelt (Abbildung 13). Das Framework beinhaltet eine vereinfachte Modellierung der UML.

Das Modell des Benutzers im UML-Tool wird für das Hypothesentesten zunächst in diese Struktur übersetzt. Eine solche Übersetzung ist für XMI Dateien (Poseidon) und über ein Plug-In der Firma eXXcellent solutions (Together) bereits vorhanden. Aus dieser unabhängigen UML-Repräsentation wird die für das Hypothesentesten benötigte Struktur (Prolog) erzeugt. Die Analyse der Benutzer-Entwürfe ist in Prolog implementiert worden. Die vom Hypothesentesten erzeugte Lösung wird wiederum in das Java-Framework überführt und anschließend in die Originalmodelle des UML-Tools.



Erstellt mit Poseidon for UML Community Edition. Nicht zur kommerziellen Nutzung.

Abbildung 13. UML-Komponenten-Diagramm für das Hypothesentesten

1.2.2.2. Ablauf des Hypothesentestens

Das Hypothesentesten unterstützt Lernende beim Lösen von Modellierungsaufgaben. Die Aufgabentexte der durch das Hypothesentesten unterstützten Aufgaben befinden sich in einer Aufgabensammlung zu Entwurfsmustern im InPULSE-CMS (siehe Abschnitt 1.1.3, HAP6). Dort findet der Lernende neben den Aufgaben auch Lernmodule zu den einzelnen Entwurfsmustern.

Im Folgenden wird der Ablauf des Hypothesentestens mit Together beschrieben: Hat ein Benutzer in Together zu einer der unterstützten Aufgaben einen Lösungsvorschlag als UML-Klassenmodell erstellt (Abbildung 14), so kann er das entsprechende Plug-In über einen Menüpunkt auswählen und seinen Entwurf von dem wissensbasierten Assistenten analysieren lassen.

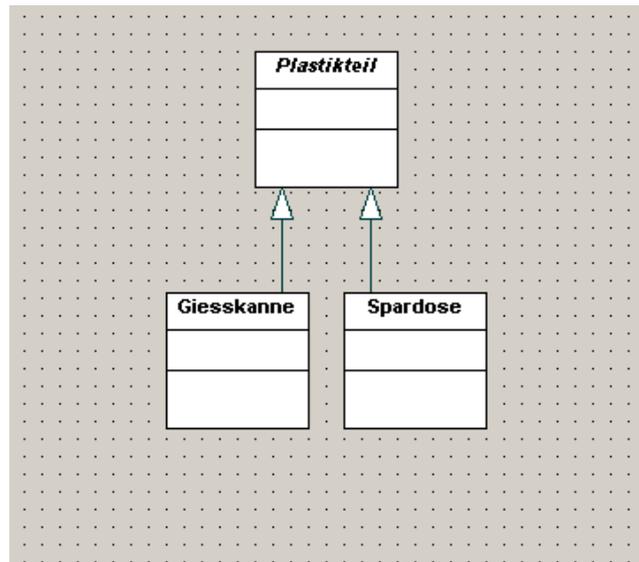


Abbildung 14: Lösungsentwurf des Benutzers

Das System führt auf syntaktischer Ebene eine Analyse durch, ob es sich bei dem Lösungsentwurf um einen korrekten Teilentwurf handelt, der zu einer Lösung vervollständigt werden kann. Ist dies der Fall, so wird der Benutzer nach der Semantik seiner Diagrammelemente gefragt (Abbildung 15). Diese semantische Prüfung der Elemente ist notwendig, um den Benutzern die freie Benennung ihrer Diagramm-Elemente zu ermöglichen und gleichzeitig sicherzustellen, dass die Interpretation der syntaktischen Elemente durch das System korrekt war.

Matches from Design to Pattern			
Your solution proposal may possibly be embedded in a correct solution, but we need further information about your diagram elements.			
Please answer the following questions concerning your diagram elements with "yes" or "no".			
UML element	Question	Yes	No
Plastikteil	Entspricht diese Klasse dem allgemeinen Plastikteil?	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Spardose	Entspricht diese Klasse der Spardose?	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Giesskanne	Entspricht diese Klasse der Gießkanne?	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Abbildung 15: Dialog zur Semantik der Diagrammelemente

Werden alle Fragen durch den Benutzer mit „ja“ beantwortet, so kann das System eine Vervollständigung des Benutzerentwurfes zu einer Lösung vorschlagen (Abbildung 16). Das vervollständigte UML-Klassendiagramm wird dem Benutzer in

einem separaten Fenster angezeigt. Es wurde ein eigener speziell angepasster UML-Editor entwickelt, in dem das veränderte Modell angezeigt wird, um Änderungen nicht direkt in dem Together-Modell des Benutzers durchzuführen. Ergänzungen im Together-Modell seitens des Plugins finden erst mit Zustimmung des Benutzers statt. Erst dann werden die Elemente in seinen Entwurf integriert und von Together z. B. Quelltext generiert.

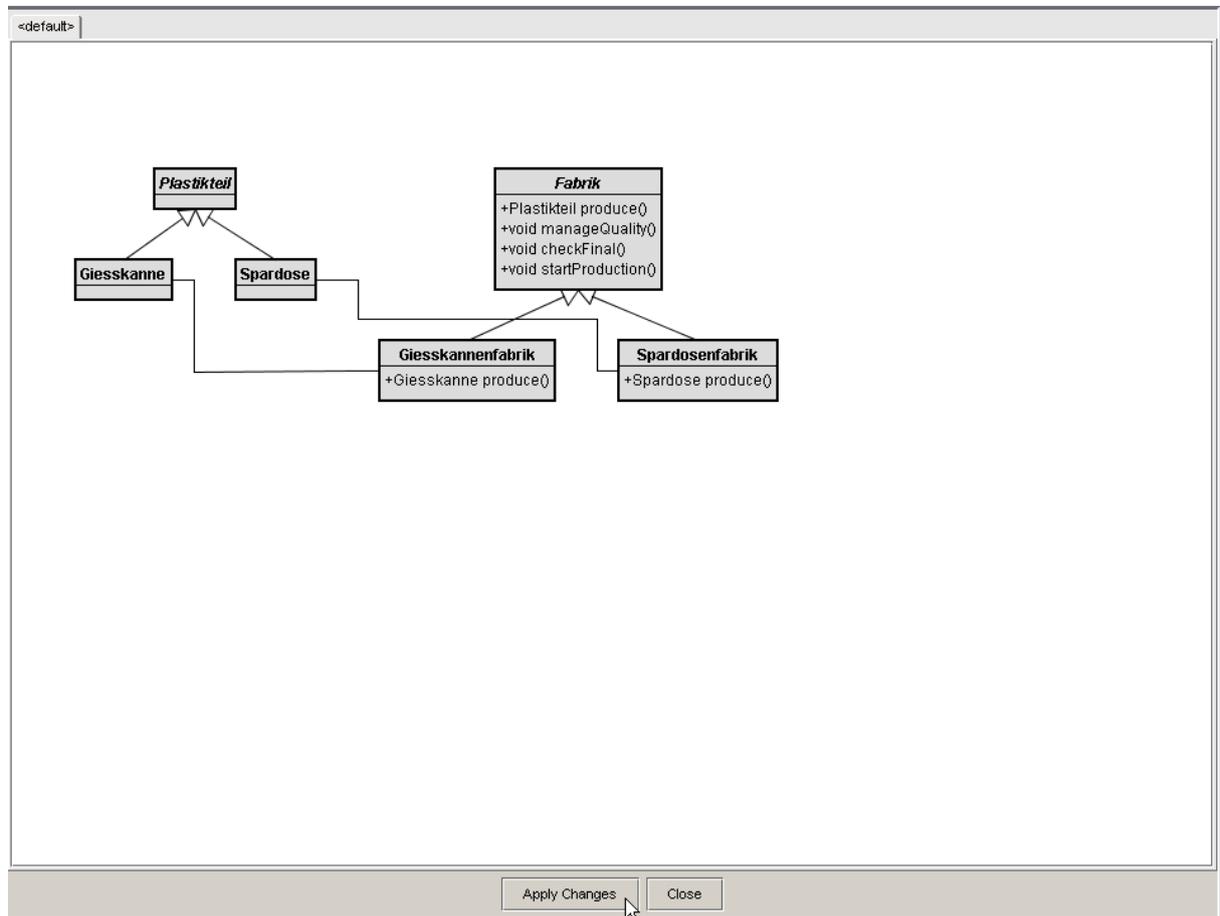


Abbildung 16: Vervollständigung des Benutzerentwurfes durch das System

Werden in dem Dialog zur Semantik der Diagrammelemente allerdings Fragen mit „nein“ beantwortet, so versucht das System, eine alternative Lösung auf syntaktischer Ebene zu finden. Gelingt dies, so werden dem Benutzer neue Fragen präsentiert. Kann keine alternative Lösung gefunden werden, so wird der Benutzer darüber informiert, dass sich sein Entwurf nicht zu einer Lösung vervollständigen lässt. Er hat dann die Möglichkeit, Teile seines Diagramms zu markieren und diese dem System erneut zur Überprüfung zu übergeben. Auf diese Weise kann er seinen Entwurf sukzessive auf einen einbettbaren Kern reduzieren.

Um das Hypothesentesten möglichst flexibel an neue Aufgaben und Beispiele anpassen und so erweitern zu können, wurde ein Autorentool (Abbildung 17) entwickelt. Mit diesem Autorentool wird es Dozenten auf komfortable Weise ermöglicht, neue Aufgaben zu erstellen. Zu einer Aufgabenstellung können verschiedene Lösungsmöglichkeiten angegeben werden. Der Dozent kann diese Lösungen z.B. in Poseidon modellieren, als XMI Dateien abspeichern und dann für die Aufgabe als Lösungen angeben. Zu den Diagramm-Elementen müssen Fragen

angegeben werden, die sich auf die Semantik der Elemente beziehen und dem Benutzer später zu diesen präsentiert werden (Abbildung 17).

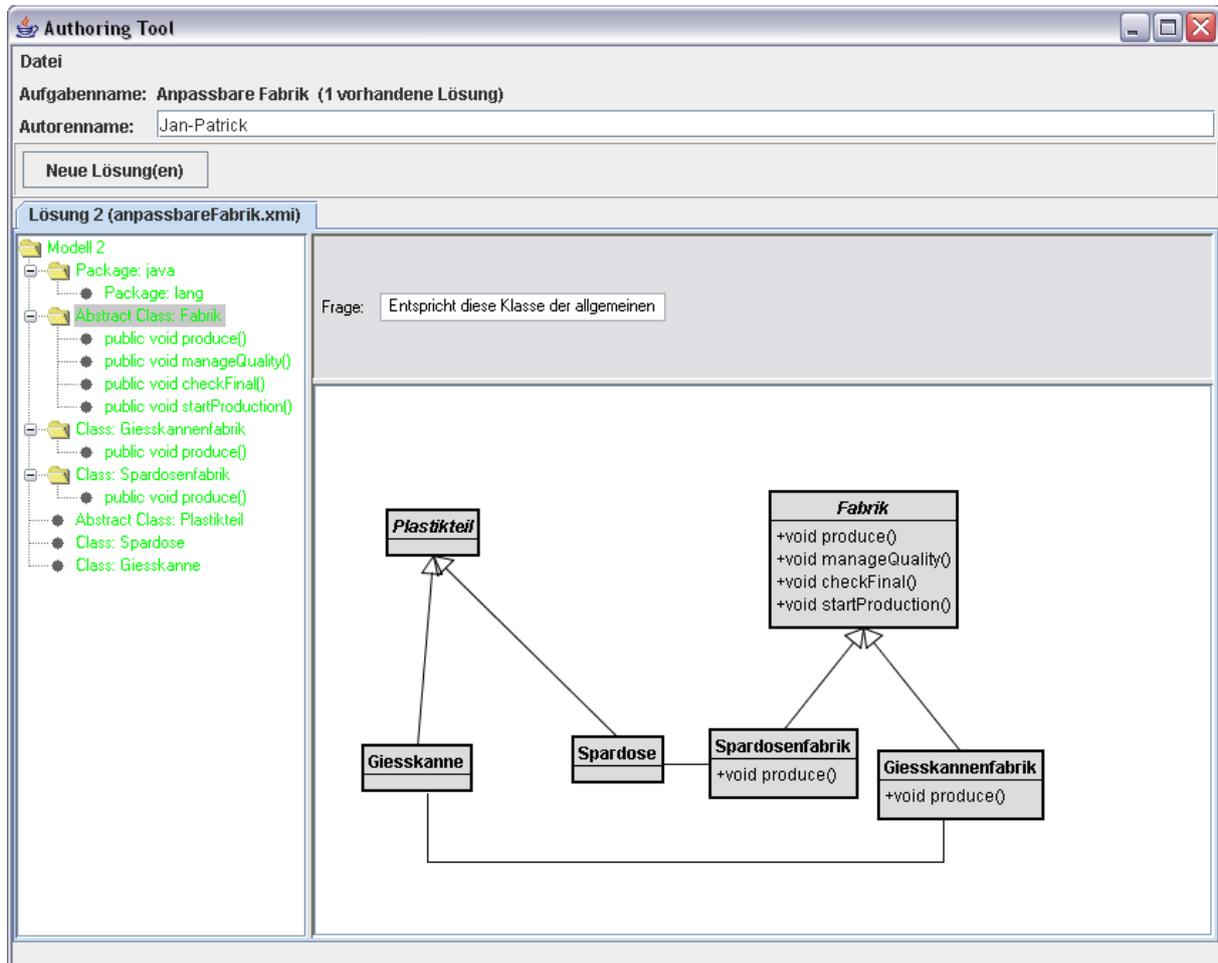


Abbildung 17: Dozenten-Tool zur Erstellung neuer Aufgaben

Die angegebenen Lösungen werden in der Wissensbasis des Systems gespeichert. Um auch Variationen der Muster erkennen zu können, enthält der Analyse-Prozess zusätzlich Regeln zu möglichen Alternativen für einzelne Diagrammelemente. Auf diese Weise wird zum Beispiel eine Assoziations-Beziehung, die der Benutzer zwischen zwei Klassen angibt, nicht als Fehler erkannt, wenn die durch den Dozenten angegebene Lösung statt dessen eine Aggregations-Beziehung enthält (Abbildung 18).

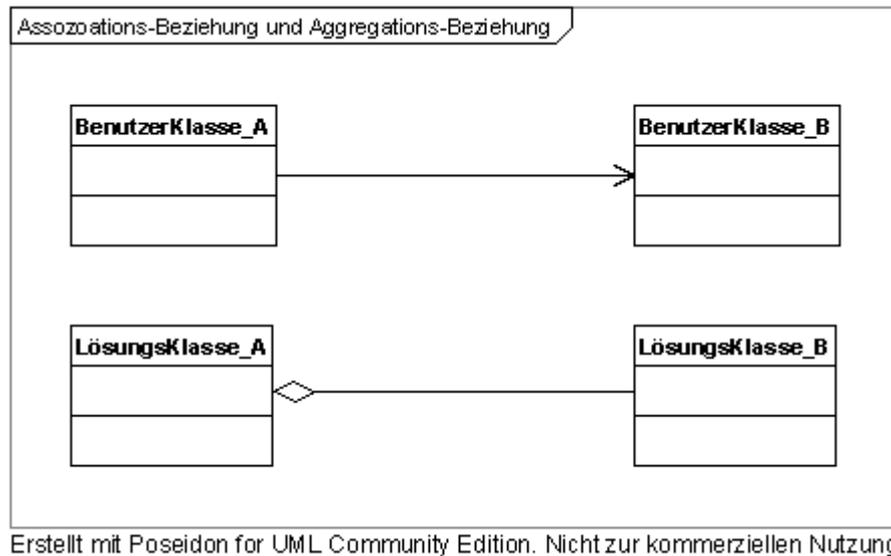


Abbildung 18: Assoziations-Beziehung und Aggregations-Beziehung

1.3. HAP 6: Entwurf/Realisierung E-Learning und Assistenzsystem

Das E-Learning- und Assistenzsystem dient zum einen als LCMS, in dem Informationen zu Entwurfsmustern abrufbar sind. Für Dozenten oder Content-Entwickler ist durch ein Schema die Struktur des Contents definiert, so dass die Kompatibilität mit dem System gesichert ist. Die Integration des neuen Contents in das LCMS ist durch einen Upload-Client unterstützt.

Zum anderen sind in dem E-Learning- und Assistenzsystem die entwickelten Methoden des Dialogspiels und des Hypothesentestens integriert.

In diesem Abschnitt wird zunächst kurz die Adaption der bereits vorhandenen e-stat-Plattform beschrieben und kurz auf die Content-Entwicklung eingegangen. Screenshots geben eine Vorstellung über das System.

1.3.1. UAP 6.4: Adaption der e-stat-Plattform

In InPULSE wurde die in e-stat konzipierte und implementierte Plattform wieder verwendet und an die projektspezifischen Bedürfnisse angepasst. Diese Adaption beinhaltete anfangs die Definition eines XML-Schemas für die Lernmodulstruktur, die als Grundlage zur Erstellung von Content dient.

Um das E-Learning-System im industriellen Umfeld einsetzen zu können, wurde eine Rollen- und Rechteverwaltung entwickelt. So können Zugriffsrechte nutzergruppenspezifisch angepasst werden.

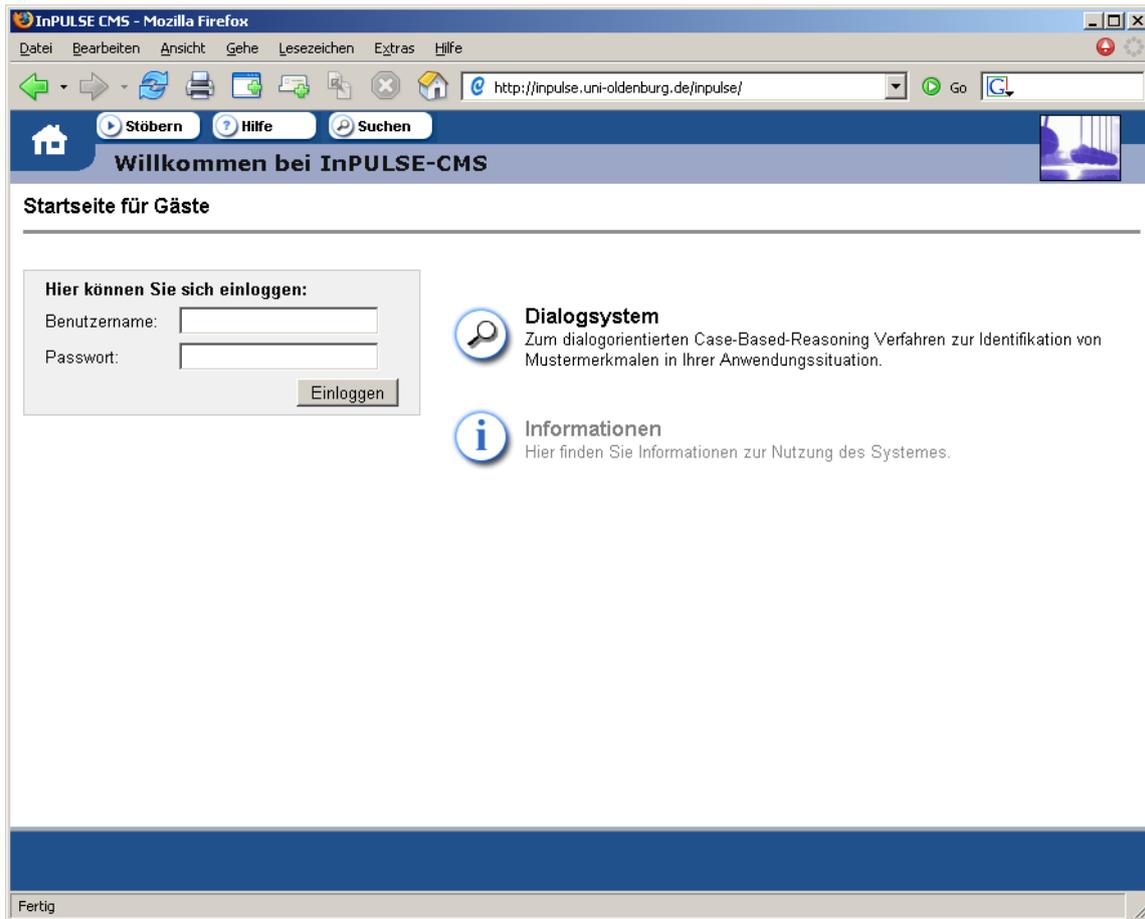


Abbildung 19: InPULSE-CMS, <http://inpulse.uni-oldenburg.de/inpulse>

Auf der InPULSE-Startseite (s. Abbildung 19) können sich Benutzer einloggen und anschließend entsprechend ihrer Zugriffsrechte durch den Entwurfsmuster-Content stöbern sowie das Dialogsystem nutzen.

1.3.2. UAP 6.5: Content-Engineering

Content-Entwicklern steht das XML-Schema inkl. einer Beschreibung der dort definierten Elemente zur Verfügung. Zusätzlich existiert eine XSLT, mit der in einem XML-Tool eine Preview des Contents angezeigt werden kann. So kann sich der Autor schon während der Content-Erstellung einen Eindruck der späteren Darstellung verschaffen. Der Upload wird durch ein Tool (Upload-Client) unterstützt. Im LCMS sind die notwendigen administrativen Aufgaben, wie etwa Content-Verwaltung und – Upload verankert.

Abbildung 20: Content im InPULSE-LCMS

Die Abbildung 20 zeigt exemplarisch eine Seite aus dem System. Zu erkennen ist im rechten Bereich die so genannte Karteikarten-Reiter-Darstellung. Diese erlaubt einen schnellen Zugriff auf die einzelnen Textabschnitte, die in allen Mustern vorhanden sind. Im linken Bildschirmbereich sind die vorhandenen Muster über eine Baumdarstellung zugriffsbereit.

1.3.3. UAP 6.6: Integration der Algorithmen aus HAP 2 in die E-Learning-Plattform
Das bereits im Abschnitt 1.2 beschriebene Dialogspiel ist ebenfalls über das InPULSE-LCMS erreichbar. Beispielhaft zeigt die Abbildung 21 einen beliebigen Punkt während des Dialogs. Der linke Fensterbereich zeigt die aktuell zu beantwortende Frage. Im rechten Bereich werden alle Fragen protokolliert. Schließlich liefert das Dialogsystem am Ende einen Vorschlag für ein situationsadäquates Entwurfsmuster.

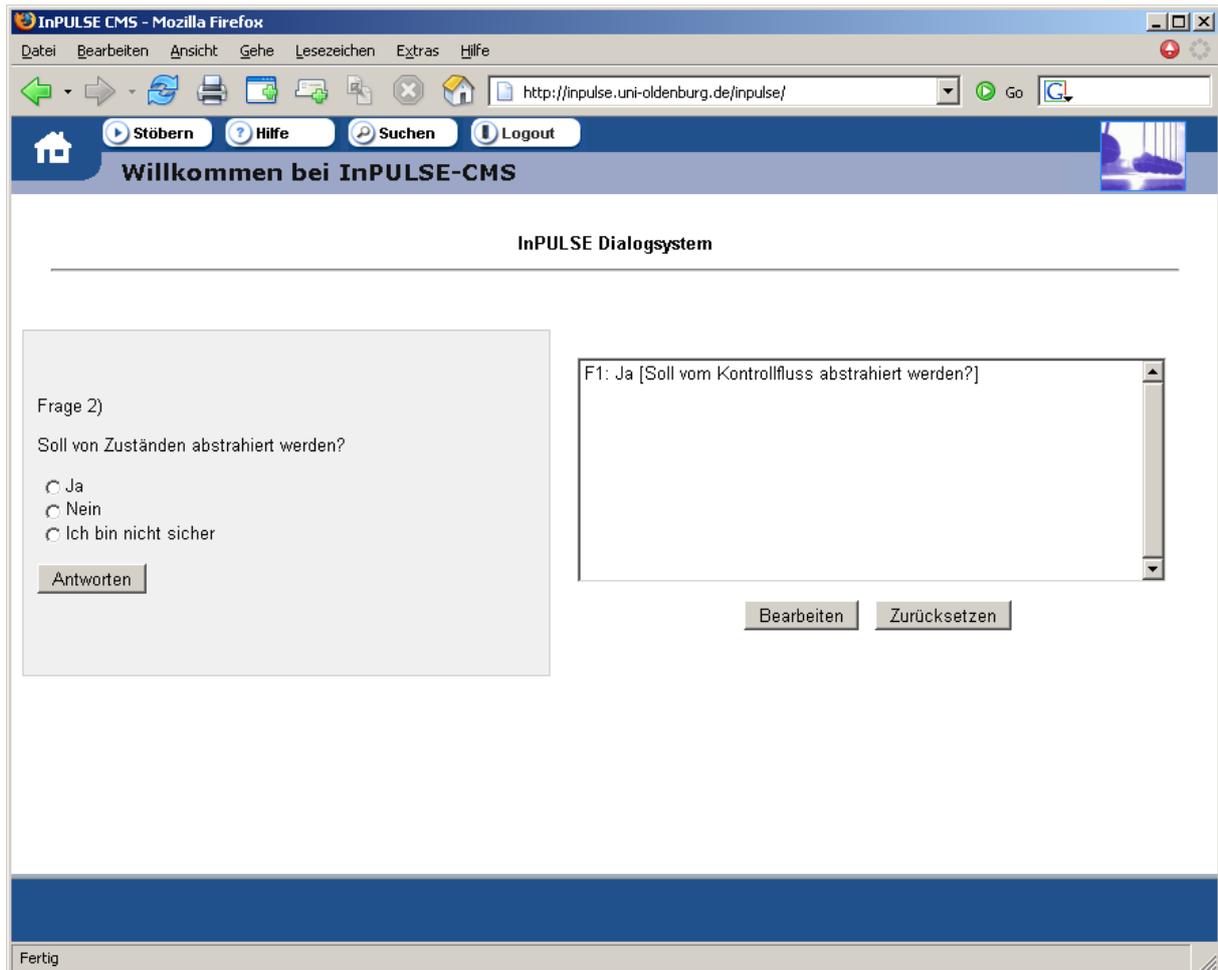


Abbildung 21: InPULSE-Dialogspiel zur Musteridentifikation

1.4. Evaluation

Zur Feststellung der Praxistauglichkeit der Plattform *InPULSE-(L)CMS*² wurde eine Evaluation durchgeführt. Da das LCMS in der Lehre eingesetzt werden soll und darüber hinaus der Einsatz und die Weiterentwicklung im Rahmen von Folgeprojekten geplant sind, können Anpassungen an die speziellen Anforderungen erforderlich werden. Diese Evaluation zum Ende des Projektes hatte aus diesem Grund sowohl formativen wie auch summativen Charakter. In dieser Evaluation wurde der Fokus ausschließlich auf das LCMS gesetzt und deshalb die Methoden zum Hypothesen-Test und das Dialogspiel nicht einbezogen.

Testdesign

Teilnehmern einer Projektgruppe der Universität Oldenburg wurde eine Auswahl von Modellierungsaufgaben angeboten. Die Teilnehmer erstellten mit Unterstützung eines UML-Tools eine Lösung für eine ausgewählte Aufgabe. Als Hilfsmittel stand ihnen das *InPULSE-(L)CMS* zur Verfügung.

Nach der Bearbeitung der Aufgabe wurden die Teilnehmer gebeten, einen Fragebogen auszufüllen, um das *InPULSE-(L)CMS* zu bewerten.

Aufgaben

² <http://inpulse.uni-oldenburg.de/inpulse/>

Die zur Verfügung stehenden Modellierungsaufgaben sind im InPULSE-(L)CMS enthalten³.

Es handelt sich um Modellierungsaufgaben in Form von Textaufgaben, die keinen expliziten Hinweis auf das zu verwendende Entwurfsmuster geben. Das LCMS lässt sich für andere Kurse auch um Aufgaben mit anderem Schwierigkeitsgrad erweitern.

Fragebogen

Der Fragebogen erfasste im 1. Block die persönlichen Daten der Teilnehmer, wie z. B. Programmierkenntnisse, Kenntnisse im Umgang mit Entwurfsmustern und die grundsätzliche Einstellung zu LCMS. Im 2. Block wurde das LCMS hinsichtlich Aufgabenangemessenheit, Selbstbeschreibungsfähigkeit, Erwartungskonformität, Fehlerrobustheit, Individualisierbarkeit, Erlernbarkeit, Dokumentation, Hilfe und Suchfunktion bewertet. Abschließend wurde den Teilnehmern im 3. Block die Möglichkeit gegeben, auf allgemein gehaltene offene Fragen ihren Eindruck vom System wiederzugeben.

Neben möglichen Freitextantworten gab es bei einigen Fragen als Antwortvorgaben eine Skala von -3 bis +3, die zwischen den Extremen „inakzeptabel“ und „ganz große Klasse“ von den Teilnehmern selbst belegt werden sollte.

Auswertung und Interpretation

Im Folgenden werden ausgewählte Bereiche der Evaluationsauswertung vorgestellt.

Block 1:

Teilnehmer waren Studierende einer Projektgruppe der Universität Oldenburg, die während des Studiums bereits Programmier- und UML-Kenntnisse erworben haben. Der Großteil der Teilnehmenden hatte bereits Erfahrungen mit Entwurfsmustern nach Gamma [Gamma+95], so dass sichergestellt war, dass die Aufgaben mit entsprechender Hilfestellung bearbeitet werden konnten (vgl. Abbildung 22).

F7: Sind Ihnen Entwurfsmuster (nach Gamma) vertraut?

		Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	ja	83,3	83,3	83,3
	nein	16,7	16,7	100,0
	Gesamt	100,0	100,0	

Abbildung 22: Fragebogenauswertung - Entwurfsmusterkenntnisse

Bezüglich des Nutzungsverhaltens von LCMS-Systemen wurden folgende Angaben gemacht (vgl. Abbildung 23):

³ <http://inpulse.uni-oldenburg.de/inpulse/>

F5: Welche Einstellung haben Sie grundsätzlich zu LCMS?

	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig nutze ich gelegentlich	50,0	50,0	50,0
nutze ich häufig	16,7	16,7	66,7
nutze ich grundsätzlich nicht	33,3	33,3	100,0
Gesamt	100,0	100,0	

Abbildung 23: Fragebogenauswertung - Nutzung von LCMS

Auch hier sind den Studierenden LCMS grundsätzlich bekannt, so dass auch dies keine Hürde in der Durchführung der Evaluation darstellte.

Block 2 und 3:

Die Teilnehmer gaben in diesem Block durch die Beantwortung der Fragen zu verstehen, dass sie durch das LCMS bei der Bearbeitung ihrer Aufgaben unterstützt wurden und die notwendigen Informationen schnell finden konnten. Die zu Rate gezogenen Informationen wurden als verständlich bewertet. Die Gestaltung gemäß der Karteikartenmetapher wurde begrüßt.

Für uns war es interessant zu erfahren, ob die benötigten Informationen in einer für den Lernenden als angemessen empfundenen Zeit eingeholt werden können. Deshalb wurden Fragen zur Detailliertheit hinzugenommen. Ebenso ist es interessant zu erfahren, ob die Anzahl der Beispiele als ausreichend empfunden wird. Entsprechende Hinweise darauf lieferten die Antworten auf folgende Fragen (Abbildung 24, Abbildung 25, Abbildung 26, Abbildung 27).

F15: Sind die Informationen detailliert genug?

	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig Absolut	16,7	16,7	16,7
ich finde schon	16,7	16,7	33,3
ja	33,3	33,3	66,7
teilweise zu detailliert	16,7	16,7	83,3
Vielleicht wäre noch ein technisch sehr detaillierter Abschnitt für Interessierte gut?!	16,7	16,7	100,0
Gesamt	100,0	100,0	

Abbildung 24: Fragebogenauswertung - Detaillierungsgrad der Informationen**F16: War es möglich, einzelne Informationen herauszugreifen, ohne dass zu viele Textpassagen gelesen werden mussten?**

	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig 0	16,7	16,7	16,7
1	33,3	33,3	50,0
2	33,3	33,3	83,3
3	16,7	16,7	100,0
Gesamt	100,0	100,0	

Abbildung 25: Fragebogenauswertung – punktuelle Nutzung der Informationen

F17: Sind die Informationen zu ausführlich, so dass die Lesezeit als zu lang empfunden wurde?

		Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	-1	16,7	16,7	16,7
	-2	33,3	33,3	50,0
	1	16,7	16,7	66,7
	2	16,7	16,7	83,3
	3	16,7	16,7	100,0
	Gesamt	100,0	100,0	

Abbildung 26: Fragebogenauswertung - Angemessenheit der Textlänge**F18: Halten Sie die Anzahl der Beispiele für ausreichend?**

		Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	-1	33,3	33,3	33,3
		16,7	16,7	50,0
	0	16,7	16,7	66,7
	3	33,3	33,3	100,0
	Gesamt	100,0	100,0	

Abbildung 27: Fragebogenauswertung - Anzahl der Beispiele

Exemplarisch können die Antworten auf die Fragen 19, 32 und 35 Indizien dafür sein, dass das LCMS alleine für Softwareentwickler noch nicht zufrieden stellend ist, da der Workflow noch nicht vollständig unterstützt wird. In den Antworten wurde z. B. der Wunsch nach Feedback und einem Tutoren-System deutlich. Das Tool zum Hypothesentesten könnte diese Lücke schließen.

F19: Wird der Workflow/Lernprozess optimal unterstützt?

		Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	-2	16,7	16,7	16,7
	0	16,7	16,7	33,3
	2	50,0	50,0	83,3
	3	16,7	16,7	100,0
	Gesamt	100,0	100,0	

Abbildung 28: Fragebogenauswertung - Workflow-Unterstützung**F32: Würden Sie das System regelmäßig nutzen? Bitte begründen Sie Ihre Antwort.**

		Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Evtl. bei länger andauernden Programmierjobs	16,7	16,7	16,7
	Ich denke schon, zumindest, wenn es sich dahingehend bewährt hat, dass die zu findenden Informationen wirklich ausreichend sind.	16,7	16,7	33,3
	Ja. Gute Referenz. Gelungene Aufbereitung der Informationen, so dass neue Pattern schnell erfasst werden können.	16,7	16,7	50,0
	nein, ich bevorzuge andere Quellen zu Entwurfsmustern.	16,7	16,7	66,7
	siehe Frage 31	16,7	16,7	83,3

Wahrscheinlich nicht, man muss alle Entwurfsmuster kennen, um eins anzuwenden. Bei der Implementierung ein Entwurfsmuster anzuwenden ist eigentlich schon zu spät. Daher ist eine Bibliothek für Entwurfsmuster nur nützlich, wenn man nicht genau weiß, ob ein	16,7	16,7	100,0
Gesamt	100,0	100,0	

Abbildung 29: Fragebogenauswertung - zukünftige Systemnutzung

F35: Was hat Ihnen gefehlt?

	Prozent	Gültige Prozepte	Kumulierte Prozepte
Gültig "Schnellsuch" Eingabefeld, das von überall zugreifbar ist.	16,7	16,7	16,7
Bilder	16,7	16,7	33,3
Hilfe	16,7	16,7	50,0
keine Angaben	16,7	16,7	66,7
Lösungen zu den Aufgaben. Oder Lösungshinweise zumindest. Irgendwie möchte man da eine Rückmeldung haben.	16,7	16,7	83,3
Tutoren-System	16,7	16,7	100,0
Gesamt	100,0	100,0	

Abbildung 30: Fragebogenauswertung - Verbesserungsvorschläge

Zusammenfassung und Resümee

Zusammenfassung haben wir durch die Evaluation die Einsatzmöglichkeiten des LCMS für Softwareentwickler überprüft. Die Systemgestaltung, insbesondere die Karteikarten-Metapher, und die Content-Strukturierung wurden von den Studierenden akzeptiert und für gut befunden, nicht zuletzt auf Grund der Möglichkeit schnell und gezielt die jeweils relevanten Methoden herauszugreifen. Aber auch das Stöbern aus „reinem Interesse“ gefiel.

Verbesserungsvorschläge gab es z. B. hinsichtlich der vollständigen Abdeckung des Workflows, woraus geschlossen werden könnte, dass das LCMS grundsätzlich durchaus zur Informationsgewinnung herangezogen würde. Die Studierenden zeigten in diesem Zusammenhang auch Interesse an den Lösungen zu den Aufgaben und Feedback während der Bearbeitung. Das Hypothesentest-Plugin ist für diese Form der Interaktivität geeignet. Ergänzungen wurden z. B. auch hinsichtlich besonders technischer Muster-Informationen gewünscht.

Insgesamt lässt sich schließen, dass das LCMS inkl. der integrierten Methoden eine akzeptierte Workbench darstellen würde und sich durchaus als praxistauglich erwies.

2. Voraussichtlicher Nutzen

Das InPULSE-LCMS inkl. der integrierten Methoden kann sowohl im Direktstudium, als auch zum Selbstlernen eingesetzt werden. Es besteht die Möglichkeit, das System zur Vorbereitung auf Seminare und zur Vertiefung des Vorlesungsstoffes zu nutzen.

Weiterentwicklungen erfährt das Projekt punktuell im Rahmen von Diplomarbeiten und Individuellen Projekten von Studierenden der Universität Oldenburg. Darüber hinaus sind der Einsatz und die Weiterentwicklung der Methoden in Folgeprojekten geplant.

Das System wird den Projektpartnern als Web-Anwendung zur Verfügung gestellt. Weitere Abteilungen der Universität Oldenburg planen den Einsatz ebenfalls.

Die folgende Auflistung stellt die wesentlichen Einsatzgebiete zusammen:

Einsatz und Weiterentwicklung der Methoden im Rahme der Lehre

- Zurzeit laufen die Vorbereitungen, das InPULSE-System in Vorlesungen im Bereich der Angewandten Informatik (Schwerpunkt E-Learning und Software Engineering) einzusetzen. In diesem Rahmen sind die Integration weiterer Inhalte und die Evaluation des Systems geplant.
- Das System wird Studierenden zum Selbststudium zur Verfügung gestellt

Einsatz bei den InPULSE-Projektpartnern

Das InPULSE-CMS enthält neben den vom OFFIS erstellten Modulen auch Module von DaimlerChrysler zu Analyse- und Entwurfsmustern im Embedded-Systems-Bereich. Das InPULSE-CMS kann daher als Hilfe- und Nachschlagssystem für diese Themen genutzt werden. Aus den im Rahmen von InPULSE bei DaimlerChrysler entwickelten Tools kann direkt auf die entsprechenden Themen verlinkt werden. So kann ein Anwender schnell und unkompliziert Zugriff auf Informationen zu den verwendeten Mustern erhalten.

Einsatz und Weiterentwicklung der Methoden im Rahmen von Diplomarbeiten und Individuellen Projekten

abgeschlossene Arbeiten:

1. Individuelles Projekt: „Java Klassenmodell für UML.“
2. Individuelles Projekt: „Algorithmen zum automatischen Zeichnen von Softwarepatterns.“

laufenden Arbeiten:

1. Individuelles Projekt: „Erstellung eines Lernmoduls zur Modellierung von Klassendiagrammen.“
2. Diplomarbeit: „Entwurf und Implementation eines Ontologie-basierten Navigationskonzeptes für Lerninhalte des InPULSE-E-Learningsystems.“
3. Diplomarbeit: „Entwurf und Implementation eines Content-Editors für das InPULSE-CMS“

Einsatz und Weiterentwicklung der Methoden z. B. im Rahmen von Folgeprojekten

Im Rahmen der BMBF-Förderlinie „Forschungsoffensive Software Engineering 2006“ befindet sich ein Projektantrag nach positiv beurteilter Skizze in der

Beurteilungsphase, an der die Abteilung von Herrn Prof. Möbus als Projektpartner beteiligt ist. Das beantragte Verbundprojekt „MINT – Modellgetriebene Integration von Informationssystemen“ sieht als ein Unterarbeitspaket die Entwicklung eines wissensbasierten Assistenzsystem zur Integration der im Projekt entwickelten Methoden und Verfahren vor. In MINT ist u. a. geplant, das InPULSE-Dialogsystem durch domänenspezifische Anwendungsmerkmale von Mustern (nach Gamma und J2EE-Pattern) sowie Musterkombinationen zu. Somit soll eine Anwendung des Dialogsystems bereits in der Analysephase möglich werden, um früh Hinweise auf die spätere Architektur zu erhalten. Diese Erweiterung erlaubt den Einsatz des Systems in der Model Driven Architecture (MDA) bzw. Model Driven Engineering (MDE). Zur zeitnahen Schulung von Mitarbeitern und zur Gewährung der Nachhaltigkeit der Projektergebnisse über das Projektende hinaus, werden die Verfahren in das E-Learning-System integriert. Als Hilfs- und Informationssystem wird es dem „Training on the Job“ dienen.

Es ist geplant, das Hypothesentesten um eine Komponente zur umfassenderen Fehlererkennung zu erweitern. Diese Arbeiten beinhalten die Integration die Methode der Constraint-Basierten-Tutoren, die ebenfalls in einem beim BMBF in der Förderlinie B „Neue Medien in der Bildung“ beantragten Projekt Get.Netz eingesetzt werden soll. Die Domäne im Projekt Get.Netz wäre Elektrotechnik. Durch die parallele Entwicklung für zwei unterschiedliche Domänen sind synergetische Effekte zu erwarten.

3. Fortschritt bei anderen Stellen

4. Veröffentlichungen

erfolgte Publikationen:

Möbus, Claus & Seebold, Heiko; A Greedy Knowledge Acquisition Method for the Rapid Prototyping of Bayesian Belief Networks, in: Ch-K. Looi, G. McCalla, B. Bredeweg & J. Breuker (eds), Artificial Intelligence in Education, Amsterdam: IOS Press, 2005, p. 875 - 877, ISBN 1-58603-530-4

Möbus, Claus; Seebold, Heiko; Garbe, Hilke; A Greedy Knowledge Acquisition Method for the Rapid Prototyping of Knowledge Structures, pages: 211 - 212; in: Clark, Peter; Schreiber, Guus; Proceedings of the 3rd International Conference on Knowledge Capture, 2005, New York, NY: ACM Press, ISBN: 1-59593-163-5; Order Number:607051; DOI-Bookmark: <http://doi.acm.org/10.1145/1088622.1088673>

erfolgte Vorträge:

- regelmäßige Projekttreffen
- Review-Treffen mit Projektträger im November 2004

Möbus, C., A Greedy Knowledge Acquisition Method for the Rapid Prototyping of Bayesian Belief Networks, AIED 2005, Artificial Intelligence in Education, Uni Amsterdam, 21.07.2005

geplante Publikationen:

- bzgl. Wissensakquisition und –repräsentation mit KARaCAs und Erstellung einer Musterontologie und Anwendungsmerkmalen in Zusammenarbeit mit der TU Ilmenau im Frühjahr 2006
- bzgl. der automatisierten Erzeugung eines Bayes-Netzes für Dialogsysteme aus dem Ergebnis der Wissensakquisition im Frühjahr 2006

geplante Vorträge:

- Abschlusspräsentation im Dezember 2005

III. Erfolgskontrollbericht

1. *Beitrag des Ergebnisses zu den förderpolitischen Zielen*

Durch die Verwendung von Mustern vereinfacht und unterstützt man das Verstehen existierender Softwaresysteme für die Systemwartung. Der Aufwand kann somit gesenkt werden. Darüber hinaus wird die Langlebigkeit der Softwareprodukte durch Verringerung der Gefahr eines Qualitätsverlustes der Softwarearchitektur erhöht. Voraussetzung dafür ist das musterbasierte Entwerfen und Implementieren von Software einerseits und das methodische Auffinden von Mustern in existierenden Systemen andererseits. Obwohl der in [Gamma+95] beschriebene Satz von Standard-Mustern breite Akzeptanz in Wissenschaft und Industrie erfahren hat, wird das musterorientierte Vorgehen bisher kaum methodisch und durch Werkzeuge unterstützt.

Im Rahmen von InPULSE hat unsere Arbeitsgruppe Methoden entwickelt, die den Software-Entwickler während des Forward Engineerings unterstützen.

Das Vorhaben entspricht den förderpolitischen Zielen im Programm „IT-Forschung 2006“, Bereich Softwaresysteme, Themenfeld Komponentenorientierung/Wiederverwendung. Behandelt wurde die Problematik der Wiederverwendung durch Nutzung und Erkennung von immateriellen Entwürfen zur Lösung von standardisierten bzw. domänenorientierten Lösungsstrukturen für stereotype Entwurfsprobleme.

2. *Wissenschaftlich-technische Ergebnisse*

Vgl. hierzu die eingehende Beschreibung der Projektergebnisse im Abschnitt 1 („Erzielte Ergebnisse“).

3. *Fortschreibung des Verwertungsplans*

Die wissenschaftlichen und technischen Erfolgsaussichten nach Projektende sowie die wirtschaftliche Anschlussfähigkeit für eine mögliche nächste Phase sind im Abschnitt I.2 „Voraussichtlicher Nutzen“ erläutert. Es ist geplant, in Folgeprojekten die Methoden weiterzuentwickeln, auch die Wiederverwendung des bestehenden LCMS ist vorgesehen. Darüber hinaus finden die entwickelten Methoden Einzug in die Lehre und stehen Studierenden auch im Rahmen ihrer individuellen Projekte zur Verfügung.

Die Methoden und das LCMS lassen sich für unterschiedliche Anwendergruppen adaptieren bzw. erweitern. Ein bedeutender funktionaler Vorteil der Content-Strukturierung und -Erfassung ist die Nutzung von XML-Schemata, so dass Inhalt und Optik auf spezielle Bedürfnisse anpassbar sind. Auch der Transfer auf andere Domänen ist somit möglich.

Erfindung und Schutzrechtanmeldungen sind nicht erfolgt.

4. Arbeiten, die zu keiner Lösung geführt haben

Alle Arbeiten wurden planmäßig durchgeführt und haben zum angestrebten Ziel geführt.

5. Präsentationsmöglichkeiten für andere Nutzer

Das InPULSE-LCMS ist unter der URL <http://inpulse.uni-oldenburg.de/inpulse/> erreichbar.

6. Einhaltung der Ausgaben- und Zeitplanung

In der Ausgaben- und Zeitplanung haben sich keine Änderungen ergeben.

Anlagen:

- **Publikation: MÖBUS, CLAUS & SEEBOLD, HEIKO & GARBE, HILKE; A**
Greedy Knowledge Acquisition Method for the Rapid Prototyping of Knowledge Structures, (accepted paper), K-CAP 2005, Third International Conference on Knowledge Capture, October 2-5, 2005, Banff, Canada.