Deutsche Forschungsgruppen in der Softwaretechnik

Inhaltsverzeichnis

RWTH Aachen **44**

RWTH Aachen **45**

Universit“ **47**

Universit“ **49**

Universit“ **51**

RWTH Aachen

Rheinisch Westfälische Technische Hochschule
Lehrstuhl für Informatik III (Softwaretechnik)

Prof. Dr. Horst Lichter

1 Arbeitsschwerpunkte

Die Lehr- und Forschungsgruppe Informatik III hat ihre Arbeit Anfang 1998 aufgenommen und befindet sich dementsprechend inhaltlich und personell im Aufbau.

Ein Schwerpunkt der Forschungsarbeiten ist der Bereich der systematischen Software-Konstruktion. Ziel der Arbeiten ist, Konstruktionstechniken für spezielle Anwendungsdomänen zu definieren. Diese Konstruktionstechniken sollen geeignet sein, um adäquate softwaretechnische Lösungen in der Anwendungsdomäne zu erstellen. Von zentraler Bedeutung sind dabei Konstruktionstechniken im Bereich der Meta-, Makro- und der Mikroarchitektur. Ein weiterer wichtiger Aspekt ist die Wiederverwendung von software-technischem Konstruktions-Know-how in einer Anwendungsdomäne. Dieses ist heute lediglich in wenigen Bereichen der Stand der Praxis (z.B. Compiler-Entwicklung, Editor-Entwicklung). Als Grundlage für domänen-spezifische Konstruktionstechniken dienen im wesentlichen Techniken der Objektorientierung. Diese haben sich in den letzten Jahren als tragfähig erwiesen, um stabile Software-Architekturen zu entwerfen.

Ein weiteres Arbeitsgebiet liegt im Bereich der konstruktiven und analytischen Qualitätssicherung von objektorientierten Systemen. Die bisher mit dem objektorientierten Ansatz gemachten Erfahrungen sind in vielen Bereichen positiv. Allerdings gibt es zum Teil schwerwiegende Probleme bei der Prüfung solcher Systeme. Die objektorientierten Techniken wie Vererbung und der Einsatz von Rahmenwerken stellen besondere Anforderungen an adäquate Maßnahmen zur Qualitätssicherung. Der Einsatz von kommerziell verfügbaren Werkzeugen bringt in vielen Fällen nicht den gewünschten Erfolg. Ein wesentlicher Grund für diese Probleme ist darin zu sehen, daß die Anforderung der Testbarkeit nicht im Entwurf und in der Implementierung berücksichtigt wird.

2 Projekte

Software-Prüfstand

Ziel dieses Projektes ist zu untersuchen, welche Auswirkungen die Anforderung der Testbarkeit an die Architekturen und an die Implementierung objektorientierter Systeme stellt. Dabei soll gezeigt werden, daß es insbesondere bei Anwendungsfamilien, die auf Rahmenwerken basieren, lohnend sein kann, angepaßte *Software-Prüfstände* zu entwickeln.

SPI (Software Process Improvement)

Um qualitativ hochwertige Software-Systeme entwickeln zu können, wird ein entsprechend hochwertiger Entwicklungsprozeß benötigt. Ziel dieses Projektes ist, zusammen mit dem Kooperationspartner geeignete Verbesserungsmaßnahmen für die vorhandenen Software-Entwickungsprozesse zu definieren und umzusetzen. Wesentlich ist dabei auch, daß der Erfolg der Verbesserungsmaßnahmen durch geeignete Metriken gemessen wird. Förderung: ABB

NetComp

Dieses Projekt hat zum Ziel am Beispiel der Anwendungsdomäne *Simulationssysteme für Engineeringaufgaben* Konzepte und Architekturen zu entwickeln, um Simulationsaufgaben verteilt in heterogenen Firmennetzen berechnen zu lassen. Dabei sollen auch die vorhandenen Anwendungen in diesem Bereich untersucht und integriert werden. Als zentrale Technolgie wird Java und dazugehörende Middleware eingesetzt. Förderung: ABB

3 Veröffentlichungen

1. Lichter H., P. Mandl-Striegnitz (1998): *A Case Study on Software Project Management in Industry - Experiences and Conclusions*, Proc. of FESMA 98 Business Improvement through Software Measurement, Antwerp, May 6-8.
2. Lichter H. (1997): *Improving software quality by static program analysis*, Proc. of SPI 97 Software Process Improvement, 1-4 Dec, Barcelona.
3. Lichter, H, C. Welsch (1997): *Software Process Improvement at ABB - Common Issues and Lessons Learnt*, Proc. of Software Quality Management SQM 97, Bath UK, March 1997.
4. Bäumer, D., W. Bischofberger, H. Lichter, H. Züllighoven (1996): *User Interface Prototyping - Concepts, Tools, and Experience*, Proceedings of the ICSE-18, Berlin, IEEE Computer Society Press, pp. 532-541.

Kontaktadresse:

Prof. Dr. Horst Lichter
RWTH Aachen
Lehr- und Forschungsgebiet Informatik III
Ahornstr. 55
52074 Aachen
Tel.: 0241 / 8021330, Fax: 0241 / 8888 352
lichter@informatik.rwth-aachen.de
http://www-lufgi3.informatik.rwth-aachen.de

RWTH Aachen

Rheinisch Westfälische Technische Hochschule
Lehrstuhl für Informatik III
(Softwaretechnik)

Prof. Dr.-Ing. Manfred Nagl

1 Arbeitsschwerpunkte

Generalthema der Forschungsaktivitäten des Lehrstuhls sind Sprachen, Methoden und Werkzeuge der Softwaretechnik für die Softwareentwicklung in der Informatik oder in anderen Anwendungsbereichen. In den letzten 10 Jahren hat sich der Schwerpunkt von der *allgemeinen* Softwaretechnik (Werkzeuge, Unterstützung für beliebige Software-Entwicklung) zu *speziellen* Softwaretechniken hin entwickelt (spezielle Anwendungsbereiche, bestimmte Klassen von Systemen, spezifische Projektarten).

Als *Anwendungsbereiche* treten die Softwareentwicklung in der Informatik auf als auch in betriebswirtschaftlichen Anwendungen, technischen Anwendungen (Fertigungstechnik, Verfahrenstechnik) sowie für Multimediasysteme. Die Klasse von Systemen, mit denen sich der Lehrstuhl intensiv auseinandergesetzt hat, sind interaktive und intelligente Systeme (Entwicklungssysteme) und deren Integration (A priori als auch A posteriori).

Der Lehrstuhl ist an einer Reihe von *Projekten* innerhalb der RWTH als auch mit Partnern außerhalb beteiligt. Er koordiniert einige dieser Projekte. Zur genaueren Übersicht vgl. `http://www-i3.informatik.rwth-aachen.de/research/index.html`.

Neben der Grundausbildung im Studiengang Informatik bzw. Elektrotechnik bietet der Lehrstuhl ein breites Lehrangebot im Vertiefungsgebiet Softwaretechnik im Hauptstudium des Diplomstudienganges Informatik an (vgl. `http://www-i3.informatik.rwth-aachen.de/department/teaching.html`).

Insbesondere hat der Lehrstuhl dazu beigetragen, daß die Informatik an der Technischen Hochschule Aachen ihren spezifischen Platz gefunden hat.

2 Projekte

IPSEN (Incremental and Integrated Software Project Support Environment)

IPSEN-Werkzeuge haben (i) enge Integration verschiedener Werkzeuge auf einem Dokument, (ii) enge feingranulare Integration zwischen verschiedenen Dokumenten sowie (iii) grobgranulare Integration über ein verallgemeinertes Workflowsystem (vgl. SU-

KITS) nachgewiesen. IPSEN verfolgt den a-priori-Ansatz. Teile dieser Ideen sind in Industrieprojekte eingeflossen. Insbesondere hat das Projekt gezeigt, daß der Bau integrierter Werkzeuge strukturiert und mechanisiert werden kann: Produktwiederverwendung über ein Rahmenwerk für allgemein verwendbare Teile bis hin zu dem Nichtstandard-Datenbanksystem GRAS [6], Prozeßwiederverwendung mit Spezifikation und Generierung. Das Projekt hat ca. 110 PJ Kapazität gebunden und wurde von diversen Fördergeldgebern (DFG, VW, EU) unterstützt. Die Ergebnisse sind wissenschaftlich ausführlich dokumentiert [10].

adt (Analysis and Development Tool)

adt ist eine schlanke Version von IPSEN. Fokus ist die Unterstützung der Architekturmodellierung [12, 13] und der Programmierung sowie ihre Verzahnung. Abweichend von IPSEN werden bestehende Werkzeuge mit einbezogen. Kernpunkt ist die Annotation der Architektur nach ihrer Erstellung, um Bindungsmechanismen, Verteilungsaspekte etc. einzufügen [7].

DFG-Forschergruppe SUKITS (Software- und Kommunikationsstrukturen in technischen Systemen, Sprecher: Prof. Nagl)

Die Forschergruppe (WZL, IKV, Lehrstühle Informatik III und IV) widmete sich der Steigerung der Effizienz von der Produktentwicklung in der Fertigungstechnik [11]. Unser Beitrag lag in der Bereitstellung eines verallgemeinerten Workflowsystems (Aufgaben-, Versions-, Varianten-, Konfigurationsverwaltung [2]) mit dessen Hilfe die Entwicklungssysteme grobgranular integriert wurden. Ein Rahmenwerk für die a-posteriori-Integration wurde entwickelt. Kernpunkt der Forschung war das Management dynamischer Entwicklungsprozesse (Concurrent and Simultaneous Engineering, Rückgriffshandhabung [4, 5]). Die Forschergruppe hat 3 Begehungen mit Erfolg absolviert. Förderung: DFG von 10/91 bis 7/97.

SFB IMPROVE (Informatische Unterstützung übergreifender Entwicklungsprozesse in der Verfahrenstechnik, Sprecher: Prof. Nagl)

6 Institute/Lehrstühle der RWTH widmen sich der verbesserten Unterstützung von Entwicklungsprozessen in der Verfahrenstechnik (Qualitäts- und Effizienzverbesserung) [11] . Dabei wird ein a-posteriori-Ansatz verfolgt, der nicht nur das Zusammenschalten der bestehenden Systeme zum Ziel hat. Statt dessen wird die Integration über neue Funktionalität erreicht. Der Lehrstuhl ist mit 3 Teilprojekten vertreten, die sich der Unterstützung der managementseitigen Koordination, dem Bau feingranularer Entwicklungswerkzeuge [8] und der Softwarekoordination des SFB widmen. Die Formalisierung von Entwicklungsprozessen ist der Schlüssel für neue Werkzeuge und auch für essentielle Wiederverwendung. Auch hier entsteht ein Rahmenwerk für die a-posteriori-Integration von Gesamtentwicklungsumgebungen. Förderung: DFG ab 7/97.

Reengineering bzw. Verteilung betriebswirtschaftlicher Anwendungen

In Zusammenarbeit mit der AMI und der GEZ werden bis dato zentralistische Systeme umstrukturiert und verteilt. Kernpunkt ist die Entwicklung neuer Werkzeuge (Strukturerkennung, Architekturveränderung, Verteilungsdurchführung). Dabei wird die Entwicklungsmaschinerie des Lehrstuhls eingesetzt [3, 14]. Förderung: MWF/BMBW.

REGINA Softwarebibliothek

Mit 11 Mitgliedern des Regionalen Industrie-Club Informatik Aachen e.V. widmet sich dieses Projekt der Wiederverwendung von Softwarebausteinen. Ein Informationssystem im WWW mit Benutzeroberfläche mit generierten html-Seiten bzw. in JAVA wurde erzeugt. In der strukturierten Suche wird auch die mit Suchmaschinen unterstützt [1] . Förderung: REGINA/MWF.

PROGRES System

Ein am Lehrstuhl entwickeltes Spezifikationssystem zur Veränderung graphartiger Datenstrukturen, wird für die Lehrstuhlprojekte aber auch an anderen Stellen in anderen Forschungsgruppen eingesetzt. Mit seiner Hilfe ist ein verschränktes Arbeiten von Erstellen, Verändern, Analysieren und Ausführen solcher Spezifikationen möglich [15, 16]. Förderung in der Vergangenheit: DFG, VW.

Prototypengenerierung

Aus PROGRES-Spezifikationen wird Code generiert und zusammen mit einer TCL/TK-Oberfläche interaktiv genutzt. Es sind visuelle Programmierumgebungen für diverse Anwendungsbereiche entstanden [9].Förderung: DFG.

Multimedia-Projekt

In Zusammenarbeit mit dem Springer-Verlag ist der Lehrstuhl am SPP "Digitale Bibliotheken" beteiligt. Es entsteht eine Strukturierungs-, Autoren- und Leserumgebung. Im Rahmen des Projekts "Virtuelle Wissensfabrik" untersucht der Lehrstuhl feingranulare Konsistenthaltung zwischen verschiedenen Wissensartefakten. Förderung: DFG/MWF.

Industrieprojekte

In Zusammenarbeit mit der örtlichen Industrie wurde eine Reihe von kleinen Industrieprojekten durchgeführt zu Fragen von Verteilung, Verteilungsplattformen, WWW, JAVA. Förderung: Industriefirmen.

3 Veröffentlichungen

1. A. Behle: *An Internet-based Information System for Cooperative Software Reuse*, erscheint in: Proc. ICSR '98

2. R. Conradi, B. Westfechtel: *Version Models for Software Configuration Management*, ACM Computing Surveys, 30(2), Juni 1998

3. K. Cremer: *A Tool Supporting the Re-Design of Legacy Applications*, in P. Nesi et al. (eds.): Proc. CSMR '98, 142-148, 1998

4. P. Heimann, G. Joeris, C.-A. Krapp, B. Westfechtel: *DYNAMITE: Dynamic Task Nets for Software Process Management*, Proc. 18th ICSE, 331-341

5. P. Heimann, C.-A. Krapp, B. Westfechtel, G. Joeris: *Graph-based software process management, Intern. Journ. of Softw. Eng. and Knowl. Eng.*, 7(4), Dec. 1997, 431-455

6. N. Kiesel, A. Schürr, B. Westfechtel: *GRAS, a Graph-Oriented Database System for (Software) Engineering Applications*, Information Systems, 20, 1, 1995, 21-51

7. P. Klein, A. Schürr: *Constructing SDEs with the IPSEN Meta Environment*, Proc. 8th Conf. on Software Engineering Environments SEE '97, IEEE Computer Society Press 1997, 2-10

8. Ch. Kohring, M. Lefering, M. Nagl: *Requirements Engineering Environment within a Tightly Integrated SDE*, Requirements Engineering 1, 3, 137-156, 1996

9. C.-A. Krapp: *An Adaptable Environment for the Management of Development Processes*, Diss. RWTH Aachen, 1998

10. M. Nagl (Ed.): *Building Tightly Integrated Software Development Environments: The IPSEN Approach*, LNCS 1170, Springer, 1996

11. M. Nagl, B. Westfechtel (Hrsg.): *Integration von Entwicklungssystemen und Ingenieuranwendungen - Substantielle Verbesserung der Entwicklungsprozesse unter Nutzung existierender Systeme*, Springer, 1998

12. M. Nagl: *Die Softwaretechnik-Programmiersprache Ada '95*, 5. Aufl., Vieweg, 1998

13. M. Nagl: *Softwaretechnik: Methodisches Programmieren im Großen*, Springer, 1990, Neuauflage 1999

14. A. Radermacher: *Integration of Existing Programs into CORBA*, in H. König et al. (eds.): Proc. DAIS '97, 157-168, 1997

15. A. Schürr: *Programmed Graph Replacement Systems*, G. Rozenberg (ed.): Handbook on Graph Grammars: Foundations, Vol. 1, World Scientific, Singapore 1997, 479-546

16. A. Schürr, A. Winter, A. Zündorf: *Graph Grammar Engineering with PROGRES*, ESEC '95, W. Schäfer, P. Botella (Eds.): LNCS 989, Springer, 219-234

4 Produkte und Werkzeuge

Zu den meisten der obengenannten Projekte sind Prototypen bzw. Demonstratoren verfügbar (Nachfrage bei den entsprechenden Teilprojekten).

Der Lehrstuhl bietet eine Reihe von Dienstleistungen an, die von Sprachen (Ada '95, JAVA), über Methoden, konkrete Werkzeuge bis hin zu Architekturfragen spezieller Systeme in diversen Anwendungsbereichen reichen.

Kontaktadresse:

Prof. Dr.-Ing. Manfred Nagl
RWTH Aachen
Lehrstuhl für Informatik III
Ahornstr. 55
52074 Aachen
Tel.: 0241 / 8021300, Fax: 0241 / 8888 218
nagl@i3.informatik.rwth-aachen.de
http://www-i3.informatik.rwth-aachen.de

Universität Koblenz-Landau

Universität Koblenz-Landau,
Abteilung Koblenz
Institut für Informatik

Prof. Dr. Jürgen Ebert

1 Arbeitsschwerpunkte

Die Forschungsschwerpunkte der Arbeitsgruppe liegen auf den folgenden Gebieten:

- Softwaretechnik, insbesondere formale Methoden und Werkzeuge für die Softwareentwicklung, -wartung und -evaluation,

- Programmiersprachen, insbesondere funktionale und visuelle Sprachen und

- Graphentechnologie, insbesondere Modellierung mittels Graphen und Entwicklung graphbasierter Werkzeuge.

Im Rahmen einer inhaltliche Unterscheidung können die folgenden Themen identifiziert werden:

- Formale Ansätze in der Softwaretechnik
- Graphentechnologie
- MetaCASE
- Programmverstehen
- Software Reengineering
- Organisationsmodellierung und Software-Evaluation

Im Bereich der formalen Ansätze in der Softwaretechnik geht es zentral um die Formalisierung von Entwurfsbeschreibungssprachen. In diesem Zusammenhang werden die Semantik von Statecharts, Vererbung von Verhalten, Formalisierung und Vergleich objekt-orientierter Methoden untersucht. Weitere Themen im Bereich der formalen Ansätze sind die Formalisierung von Verhaltensbeschreibungen im Netzmanagement, die Prototypisierung mit Z und die Konfigurationskontrolle.

Die Graphentechnologie befaßt sich mit Realisierung von Anwendungssystemen durch Graphen und mit Hilfe graphentheoretischer Hilfsmittel und Algorithmen. Es werden typisierte, attributierte und angeordnete, gerichtete Graphen (TGraphen) verwendet. Klassen von Tgraphen werden über erweiterte Entity-Relationship-Diagramme (EER-Diagramme) beschrieben. EER-Diagramme können durch weitere Bedingungen in der Z-ähnlichen Sprache GRAL (Graph Specification Language) ergänzt werden. TGraphen sind als Datenstruktur mit Operationen für die Manipulation und Traversierung mit Hilfe des Graphenlabors (Gralab) effizient implementiert. Für Anfragen an TGraphen existieren mit G^2QL und GReQL entsprechende Anfragesprachen. G^2QL ist eine auf dem Entity-Relationship-Paradigma beruhende visuelle Sprache. GReQL dagegen ist eine textuelle Sprache, die im Rahmen des GUPRO-Projekts für die Extraktion von tabellenartigen Informationen aus Graphen entwickelt wurde. Für GReQL existiert auch ein Auswerter, der GReQL-Anfragen auf mit dem GraLab gespeicherten TGraphen auswertet.

Der MetaCASE-Bereich wird im wesentlichen durch das Projekt KOGGE (KOblenzer Generator für Graphische Entwurfsumgebungen) abgedeckt. Der KOGGE-Ansatz basiert auf einem Interpreter, der eine formale Beschreibung eines CASE-Werkzeugs interpretiert und es damit konkret realisiert. Dabei läßt sich die Beschreibung eines KOGGE-CASE-Werkzeugs selbst mit Hilfe eines KOGGE-Systems, der UrKOGGE, definieren. Es existieren KOGGE-Werkzeuginstanzen für die BON-Methode, für Datenflußdiagramme sowie für eine Anforderungserfassung in der Software-Evaluation. Eine Entwurfsunterstützung für Feature Invocation Graphs wird gerade erstellt.

Der Bereich des Programmverstehens wird durch das Projekt GUPRO (Generische Umgebung zum PROgrammverstehen) repräsentiert. In GUPRO geht es um die Entwicklung eines Generators für die Erzeugung sprachübergreifender Programmverstehenswerkzeuge. Die vom Anwender individuell zu bestimmenden Konzepte der Altsoftware werden in Form von EER-Diagrammen (Konzeptdiagrammen) erfaßt. Mit Hilfe entsprechender Parser werden Quelltexte in den Diagrammen entsprechende TGraphen überführt. Die Analyse der Quelltexte kann damit auf der Basis der TGraphen erfolgen. In diesem Zusammenhang stehen geeignete Anfrage- und Browsingwerkzeuge zu Verfügung.

Im Bereich des Software Reengineering können drei weitere Teilaktivitäten identifiziert werden. Dabei geht es zunächst um eine Aufgabenunterstützung im Software Reengineering im Rahmen eines MetaCARE-Ansatz. Über einen Zyklus aus extract, transform und rewrite (ETR) sollen Aufgabenstellungen im Zusammenhang mit beliebigen Programmiersprachen unterstützt werden. Eine weitere Aktivität innerhalb des Software Reengineerings ist die Beteiligung am Reverse Engineering Demonstration Project, das eine internationale Studie zwischen kommerziellen und nicht-kommerzielle Forschungsgruppen im Bereich des Reengineering darstellt. Darüber hinaus wird in einem Kooperationsprojekt mit dem Lehrstuhl für Wirtschaftsinformatik, Universität Regensburg (Prof. Dr. Franz Lehner) versucht, die unternehmensweiten Software-Reengineering-Aktivitäten zu koordinieren, um so zu einem unternehmensweiten Ansatz für das Reengineering (Enterprise-wide Reengineering Approach (ERA)) zu kommen.

Der Bereich der Organisationsmodellierung und Software-Evaluation befaßt sich mit der Software Evaluation im allgemeinen sowie mit der Evaluation von Krankenhaus-Informationssystemen im speziellen. Desweiteren wird ein Referenzschema entwickelt, das strukturierte Beschreibungsmittel aus der Softwaretechnik und aus der Organisationstheorie konzeptionell zusammenfaßt. Bei den Aktivitäten im Bereich von Krankenhaus-Informationssysteme geht es außerdem um die Herleitung und Beschreibung von Referenzmodellen für Krankenhaus-Informationssysteme.

2 Projekte

KOGGE (Koblenzer Generator für Graphische Entwurfsumgebungen)

KOGGE (Koblenzer Generator für Graphische Entwurfsumgebungen) ist ein gemeinsam mit der rbeitsgruppe Prof. Dr. Manfred Rosendahl entwickeltes MetaCASE-System, d. h. ein Werkzeug, das der Erzeugung von CASE-Werkzeugen dient. (Weitere Info unter: `http://www.uni-koblenz.de/~ist/kogge.html`)

GUPRO (Generische Umgebung zum PROgrammverstehen (BMBF)

GUPRO ist ein gemeinsames Projekt mit der Aachener und Münchener Informatik-Service GmbH, Hamburg, und dem IBM Wissenschaftlichen Zentrum, Heidelberg. Es wird gefördert durch das Bundesministeriums für Bildung, Wissenschaft, Forschung und Technologie (BMBF), Förderprogramm Softwaretechnologien, Förderkennzahl 01 IS 504. (Weitere Info unter: `http://www.uni-koblenz.de/~ist/gupro.html`)

3 Veröffentlichungen

1. J. Ebert, B. Kullbach, A. Panse: *The Extract-Transform-Rewrite Cycle - A Step towards MetaCARE*, in P. Nesi, F. Lehner: Proceedings of the 2nd Euromicro Conference on Software Maintenance & Reengineering IEEE Computer Society, pg. 165-170, Los Alamitos, 1998

2. J. Ebert, R. Süttenbach, I. Uhe: *Meta-CASE in Practice: a Case for KOGGE*, in A. Olive, J. A. Pastor: Advanced Information Systems Engineering, Proceedings of the 9th International Conference, CAiSE'97, Barcelona, Catalonia, Spain, June 16-20, 1997 Springer, LNCS 1250, S. 203-216, Berlin, 1997

3. J. Ebert, R. Gimnich, A. Winter: *Wartungsunterstützung in heterogenen Sprachumgebungen, Ein Überblick zum Projekt GUPRO*, in F. Lehner: Softwarewartung und Reengineering - Erfahrungen und Entwicklungen Gabler, S. 263-275, Wiesbaden, 1996

4. J. Ebert, A. Winter, P. Dahm, A. Franzke, R. Süttenbach: *Graph Based Modeling and Implementation with EER/GRAL*, in B. Thalheim: 15th International Conference on Conceptual Modeling (ER'96), Proceedings Springer, LNCS 1157, pg. 163-178, Berlin, 1996

5. A. Winter, J. Ebert: *Ein Referenz-Schema zur Organisationsbeschreibung*, in J. Becker, G. Vossen: Geschäftsprozeßmodellierung und Workflows Thomson, S. 101-123, Bonn, 1996

6. A. Franzke: *GRAL: A Reference Manual*, Fachbericht Informatik 3/97, Universität Koblenz-Landau, Fachbereich Informatik, Koblenz, 1997

7. P. Dahm, J. Ebert, C. Litauer: *Das EMS-Graphenlabor 3.0,* Interner Projektbericht 3/94, Universität Koblenz-Landau, Institut für Softwaretechnik, Koblenz, 1994

4 Produkte und Werkzeuge

Graphenlabor: Das Graphenlabor GraLab ist eine C++-Klassenbibliothek für die Handhabung von Graphen als interne Datenstruktur.

KOGGE: KOGGE ist ein MetaCASE-System, das in C++ implementiert und unter UNIX lauffähig ist.

GUPRO: GUPRO ist ein Meta-Programmverstehenswerkzeug, das unter OS/2 und Solaris verfügbar ist.

Kontaktadresse:

Prof. Dr. Jürgen Ebert
Universität Koblenz-Landau
Institut für Informatik
Rheinau 1
56075 Koblenz
Tel. (0261) 9119-412 Fax -499
ebert@informatik.uni-koblenz.de

Universität - GH Paderborn

Universität - GH Paderborn
Fachbereich Elektrotechnik
Fachgebiet Angewandte Datentechnik
(Softwaretechnik)

Prof. Dr. Fevzi Belli

1 Arbeitsschwerpunkte

Den zentralen Gegenstand der Arbeiten des Fachgebiets bilden formale und semi formale Methoden und Hilfsmittel für die Konstruktion verläßlicher Systeme für sicherheitskritische Anwendungen. Die Arbeitsschwerpunkte sind (s. auch http:/adt.uni-paderborn.de):

- Entwurf und Spezifikation: Berücksichtigung von voraussehbaren Irregularitäten im Nutzungsprofil, d.h. unerwünschten Ereignissen, die zu Ausfällen führen können. Behandelt werden u.a.

  - Methoden, die neben der Modellierung der Systemeigenschaften auch eine Modellierung fehlerhafter Zustände und Ereignisse ermöglichen,
  - Integration der Methoden der Software-Fehlertoleranz in den Konstruktionsprozeß,
  - Multiparadigmatische Betrachtung, z.B. logisch-objektorientiert, logisch-prozedurale.

- Analyse und Prüfung: Nachweiserbringung der Erfüllung der Nutzungsanforderungen.
  - Analyse durch formalisierte Review-Methoden,
  - Systematisches Testen, Test-Theorien,
  - Modellierung der Software-Zuverlässigkeit.

Neben tradiierten Methoden der Softwaretechnik werden auch Techniken des Knowledge-Engineering, insbes. Wissensrepräsentation und -akquisition eingesetzt.

Die Ergebnisse der Forschungs- und Entwicklungsarbeiten fließen in die Lehrveranstaltungen über Programmierung und Softwaretechnik, auch in Form von Praktika (z.Z.: Software-Testwerkzeuge), ein.

Dissertationen:

A. Azem, "Reliability Determination of Logic and Conventional Programs", Reihe "Programmierung komplexer Systeme", deGruyter, Berlin, New York etc.(1995)

R. Crisan, "Towards Automation of Code-Reviews based on Checklists", Dissertationsschrift eingereicht an die Universität - GH Paderborn, FB 14 (1998)

J. Dreyer, "Program Segmentation for Controlling Software Testing and Analysis", Dissertationsschrift eingereicht an die Universität - GH Paderborn, FB 14 (1998)

O. Jack, "Software Testing for Conventional and Logic Programming", Reihe "Programmierung komplexer Systeme", deGruyter, Berlin, New York etc. (1996)

A. Pollmann, "An Approach to Logic/Object-Oriented Concurrent Robot Programming and Performance Analysis", Reihe "Programmierung komplexer Systeme", deGruyter, Berlin, New York etc. (1996)

S. Schurig, "Regelbasierte Planung kollisionsfreier Bewegungsbahnen für Industrieroboter mittels objekt orientierter Logikprogrammierung", Dissertationsschrift angenommen von der Universität - GH Paderborn, FB 14 1996)

2 Projekte

Einige bisher geförderte Projekte sind

DANGOOD ("A Knowledge-Based System for the Transport of Dangerous Goods and Fire Department Consulting") gefördert von der Stadt Bremerhaven,

ASH ("A Knowledge-Based Workbench for User-Programmable Control of Public Administration Tasks") in Zusammenarbeit mit der Kommunalverwaltung einiger Städte in Süddeutschland und dem Softwarehaus GfS, Markt Oberdorf

PROMIFF ("Modelling and Implementation of a Flexible Multi-Robot Cell"), gefördert durch Mitsubishi, Europe

PROLoop ("An Extension of PROLOG for Object-Oriented Programming in Logic") internes Projekt

PROBot ("Distributed and Transputer-Based Robot Controlling through Object-Orient ed Programming in PROLOG"), teilgefördert durch Mitsubishi, Europe,

PROMPCD ("Motion Planning and Collision Detection in Multi-Robot Systems"), teilgefördert durch AEG/MODCOMP

PROTest ("Development of a Test Environment for Conventional and LogicProgramming") zur Teilautomatisierung von Code-Reviews mit Reverse-Engineering-Komponente, teilgefördert von einem Unternehmen aus der Auto-Industrie.

3 Veröffentlichungen

1. F. Belli, *"Einführung in die logische Programmierung mit PROLOG"*, Bibliographisches Institut - Wissenschaftsverlag, Mannheim etc., B.I.-Hochschultaschenbuch, HTB 630, 2. erweiterte Auflage (1988)

2. F. Belli, *"Pascal - Anleitung zur systematischen Programmierung und Konstruktion zuverlässiger Programme mit Anwendungsbeispielen in Standard- und Turbo-Pascal", Band I: "Einführung"*,

3. B.I.-Hochschultaschenbuch (HTB) 635 (1989), Band II: *"Weitere Methoden"*, HTB 636 (1989), Band III: *"Programmierpraxis"* HTB 640 (1989), Bibliographisches Institut - Wissenschaftsverlag, Mannheim etc.

4. A. Azem, F. Belli, P. Jedrzejowicz, *"Reliability Prediction & Estimation of Prolog Programs"*, IEEE Trans. on Reliability (1994), pp. 542-549

5. F. Belli, O. Jack, *"Declarative Paradigm of Test Coverage"*, J. Software Testing, Verification & Reliability (1998)

6. F. Belli, K.-E. Großpietsch, *"Specification of Fault-Tolerant System Issues by Predicate/Transition Nets and Regular Expressions - Approach and Case Study"*, IEEE Trans. on Software Eng. (1991), pp. 513-526

7. F. Belli, P. Jedrzejowicz, *"An Approach to the Reliability Optimization of Software with Redundancy"*, IEEE Trans. on Software Eng. (1991), pp. 310-312

8. F. Belli, P. Jedrzejowicz, *"Fault-Tolerant Programs and Their Reliability"*, IEEE Trans. on Reliability, Vol. 39, No. 2 (1990), pp. 184-192

4 Produkte und Werkzeuge

Die Ergebnisse der meisten Projekte (PROLoop, PRORel, PROTest, PROLyse, ...) mündeten in Form von Implementierungen entwickelter Algorithmen in die Programmier- und Produktsicherungsumgebung des Fachgebiets ein.

Kontaktadresse:

Prof. Dr. Fevzi Belli
Dr.-Ing. Dipl.-Math. Oliver Jack
Universität-GH Paderborn
Fachbereich 14 (Elektrotechnik)
Fachgebiet Angewandte Datentechnik (Softwaretechnik)
33098 Paderborn
Tel.: +49-5251-60 3445-3449
Fax: +49-5251-60 3246
bellioliver@adt.uni-paderborn.de
http://adt.uni-paderborn.de

Universität - GH Paderborn

Universität - GH Paderborn
Fachbereich Mathematik/Informatik
AG Informationssysteme

Prof. Dr. Gregor Engels

1 Arbeitsschwerpunkte

Objektorientierte Modellierung

Die von der OMG als Standard akzeptierte objektorientierte Modellierungssprache UML (Unified Modeling Language) umfaßt zwar eine präzis definierte Syntax, jedoch nur eine informelle und z.T. auch unvollständige Festlegung der Semantik. Als Vorarbeiten für eine derartige präzise Semantikdefinition untersuchen wir unter anderem die Vererbung und Aggregation von Zustandsdiagrammen [EE 94] und die Übersetzung von UML Kollaborations- und Ablaufdiagrammen nach Java. Weiterhin untersuchen wir weitergehende objektorientierte Konzepte zur Modellierung von kooperierendem Verhalten von Objekten [EGK96]. Dies baut auf eigenen Arbeiten im Rahmen der Entwicklung der objektorientierten Spezifikationssprache SOCCA auf [EG 94].

Modellierung von Multimedia-Anwendungen

Multimedia-Anwendungen werden heutzutage in der Regel mit Hilfe sogenannter Autorensysteme wie Toolbook oder Macromedia Director entwickelt. Die bei der Verwendung dieser Autorensysteme unterstützte prototypische, ad-hoc Vorgehensweise führt dazu, daß weder zunächst ein Modell der Anwendung entwickelt wird, noch daß eine ausreichende Dokumentation erstellt wird. Um dies zu verbessern, entwickeln wir ausgehend von der objektorientierten Modellierungssprache UML eine Modellierungssprache für Multimedia-Anwendungen. Zur Entwurfsunterstützung entwickeln wir darüberhinaus ein Framework für Multimedia-Anwendungen, welches für eine konkrete Anwendung spezialisiert werden kann und dann auch als Dokumentation verwendet werden kann.

Visuelle Spezifikationssprachen

Ausgehend von der Idee, daß Systemzustände durch Graphen und Veränderungen durch Graphtransformationen beschrieben werden können, untersuchen wir visuelle Spezifikationssprachen auf der Grundlage von Graphersetzungssystemen. Insbesondere untersuchen wir Strukturierungsmöglichkeiten für derartige Spezifikationen, wie z.B. Modularisierungskonzepte und Sichtendefinitionen [EHTE 97a/b]. Anwendungen dieser Sprache finden sich dann z.B. in der Syntax- und Semantikdefinition visueller Sprachen [AE 96] oder in der Modellierung komplexer Systemen [AER 97].

System- und Datenbankintegration

In diesem Arbeitsschwerpunkt untersuchen wir verschiedene Aspekte der Integration. Zum einen untersuchen wir, wie durch ein mehrdimensionales Rahmenwerk ein Software-Entwickler bei der Integration verschiedener Aspekte im Rahmen der Software-Entwicklung unterstützt werden kann. Im Themen-bereich föderierter Datenbanksysteme untersuchen wir, welche Auswirkungen lokale und globale Schemaänderungen auf das integrierte, föderierte System haben.

2 Projekte

OMMMA (Objektorientierte Modellierung von Multimedia-Anwendungen)

Im Rahmen dieses Projektes wird eine Entwurfssprache und -umgebung für den Entwurf und die Realisierung von Multimedia-Anwendungen entwickelt.

PROMOTER II

Im Rahmen dieser ESPRIT Basic Research Working Group wird zusammen mit anderen Partnern an der Entwicklung von prozeßgesteuerten Softwareentwicklungsumgebungen gearbeitet. In enger Zusammenarbeit mit der Reichuniversität Leiden (Niederlande) wird an der Entwicklung der objektorientierten Modellierungssprache SOCCA für Softwareprozesse gearbeitet [EG 94].

APPLIGRAPH / GETGRATS

Im Rahmen dieser beiden von der EU geförderten Projekte wird an Spezifikationssprachen auf der Basis von Graphtransformationssystemen gearbeitet. In enger Zusammenarbeit mit der TU Berlin wird an Strukturierungskonzepten für Graphtransformationssysteme gearbeitet [EHTE 97 a/b].

Virtuelle Universität

Im Rahmen des Projekts "Multimedia in der wirtschaftswissenschaftlichen Lehre" wird an einem Prozeßmodell für die Entwicklung multimedialer Lehr- und Lerneinheiten gearbeitet.

3 Veröffentlichungen

[AE 96] M. Andries, G. Engels: *A Hybrid Query Language for the Extended Entity Relationship Model*, Journal of Visual Languages and Computing, Vol. 7, No. 3, September 1996, 321-352

[AER 97] M.A.A. Andries, G. Engels, J. Rekers: *How to represent a visual specification*, In K. Marriot, B. Meyer (eds.): Visual Language Theory, Springer, Berlin 1997, 241-255

[EE 94] J. Ebert, G. Engels: *Structural and Behavioural Views on OMT-Classes*, In E. Bertino, S. Urban (eds.): Proceedings International Symposium on Object-Oriented Methodologies and Systems (ISOOMS), Palermo, Italy, September 21-22, 1994, LNCS 858, Springer, Berlin 1994, 142-157

[EHTE 97a] G. Engels, R .Heckel, G. Taentzer, H. Ehrig: *A Combined Reference Model- and View-Based Approach to System Specification*, International Journal on Software Engineering and Knowledge Engineering, Vol.7, No. 4, December 1997, 457-477

[EHTE 97b] G. Engels, R. Heckel, G. Taentzer, H. Ehrig: *A View-Oriented Approach to System Modelling Using Graph Transformations*, In M. Jazayeri, H. Schauer (eds.): Proceedings European Software Engineering Conference (ESEC) 1997, Zürich, LNCS 1301, Springer 1997, 327-343

[EG 94] G. Engels, L.P.J. Groenewegen: *SOCCA: Specifications of Coordinated and Cooperative Activities*, In A. Finkelstein, J. Kramer, B.A. Nuseibeh (eds.): Software Process Modelling and Technology, Research Studies Press, Taunton 1994, 71-102

[EGK 96] G. Engels, L.P.J. Groenewegen, G. Kappel : *Object-Oriented Specification of Coordinated Collaboration*, In N. Terashima, Ed. Altman: Proc. IFIP World Conference on IT Tools, 2-6 September 1996, Canberra, Australia. Chapman & Hall, London 1996, 437-449

Kontaktadresse:

Prof. Dr. Gregor Engels Universität-GH Paderborn
Fachbereich Mathematik/Informatik
AG Informationssysteme
Warburger Straßse, 33098 Paderborn
Tel.: +49-5251-60 3336/3337
Fax: +49-5251-60 3431
engels@uni-paderborn.de
http://www.uni-paderborn.de/cs/engels.html