

Gesellschaft für Informatik e.V. (GI)

publishes this series in order to make available to a broad public recent findings in informatics (i.e. computer science and information systems), to document conferences that are organized in co-operation with GI and to publish the annual GI Award dissertation.

Broken down into the fields of

- Seminar,
- Proceedings
- Dissertations
- Thematics

current topics are dealt with from the fields of research and development, teaching and further training in theory and practice. The Editorial Committee uses an intensive review process in order to ensure the high level of the contributions.

The volumes are published in German or English

Information: <http://www.gi-ev.de/service/publikationen/lni/>

ISSN 1614-3213

ISBN 978-3-88579-439-4



GI-Edition

Lecture Notes in Informatics

Gesellschaft für Informatik (Hrsg.)

Informatiktage 2007

Fachwissenschaftlicher Informatik-Kongress

30. und 31. März 2007

B-IT Bonn-Aachen International Center
for Information Technology, Bonn

Gesellschaft für Informatik (Hrsg.): Informatiktage 2007





Gesellschaft für Informatik (Hrsg.)

Informatiktage 2007

Fachwissenschaftlicher Informatik-Kongress

30. und 31. März 2007

**B-IT Bonn-Aachen International Center for Information
Technology in Bonn**

Gesellschaft für Informatik e.V. (GI)

Lecture Notes in Informatics (LNI) - Seminars

Series of the Gesellschaft für Informatik (GI)

Volume S-5

ISBN 978-3-88579-439-4

ISSN 1614-3213

Volume Editor

Gesellschaft für Informatik e.V.

Ahrstraße 45

53175 Bonn

E-Mail: gs@gi-ev.de

Redaktion: Ludger Porada

E-Mail: ludger.porada@gi-ev.de

Series Editorial Board

Heinrich C. Mayr, Universität Klagenfurt, Austria (Chairman, mayr@ifit.uni-klu.ac.at)

Jörg Becker, Universität Münster, Germany

Ulrich Furbach, Universität Koblenz, Germany

Axel Lehmann, Universität der Bundeswehr München, Germany

Peter Liggesmeyer, TU Kaiserslautern und Fraunhofer IESE, Germany

Ernst W. Mayr, Technische Universität München, Germany

Heinrich Müller, Universität Dortmund, Germany

Heinrich Reinermann, Hochschule für Verwaltungswissenschaften Speyer, Germany

Karl-Heinz Rödiger, Universität Bremen, Germany

Sigrid Schubert, Universität Siegen, Germany

Dissertations

Dorothea Wagner, Universität Karlsruhe, Germany

Seminars

Reinhard Wilhelm, Universität des Saarlandes, Germany

© Gesellschaft für Informatik, Bonn 2007

printed by Köllen Druck+Verlag GmbH, Bonn

Wissenschaftliche Tagungsleitung:

Prof. Dr. Matthias Jarke, GI-Präsident, RWTH Aachen

Prof. Dr. Karl Neunast, Fachhochschule Bonn-Rhein-Sieg

Programmkomitee

Rainer Beedgen, Berufsakademie Mannheim

Jochen Berger, Comma Soft AG

Anja Grothausmann, BearingPoint

Tim Gugel, sd&m AG

Jens Lüssem, Fachhochschule Wolfenbüttel

Dirk Primbs, Microsoft Deutschland GmbH

Klaus Schmid, Universität Hildesheim

Eberhard Stickel, Hochschule der Sparkassen

Bernd Liesner, integranova GmbH

Johannes Windeln, IBM Mainz

Alfred Zimmermann, Hochschule Reutlingen

Detlev Zimmermann, Private FernFachhochschule Darmstadt

Übersicht über das Rahmenprogramm und Partner der Informatiktage 2007

Neben den Workshops wurde den Teilnehmerinnen und Teilnehmern ein attraktives Rahmenprogramm geboten. Wir bedanken uns außerdem bei unseren Partnern, ohne deren Unterstützung die Veranstaltung nicht möglich wäre.

Rahmenprogramm:

30.03. 17:00 Uhr:

Stephan Reichart, Hauptgeschäftsführer G.A.M.E. Bundesverband der Entwickler von Computerspielen e.V.

„Die Computerspielebranche in Deutschland von A wie Ausbildung bis Z wie Zukunft“
Ein Überblick über wirtschaftliche Eckdaten, Berufsbilder und Potentiale der deutschen Spielebranche

31.03. 9:00 Uhr

Dr. Johannes Windeln, Manager des IBM Technology Center, ISC EMEA, Mainz
Nanoscience - Nanotechnologie

Moderation: Prof. Dr. Karl Neunast, Fachhochschule Bonn-Rhein-Sieg

Postersession:

30.03. 15:15 Uhr und 31.03. 13:00 Uhr

Partner und Sponsoren der Informatiktage:

Partner:

B-IT Bonn-Aachen Center
for Information Technology
IBM Deutschland GmbH

www.b-it-center.de
www.ibm.de

Sponsoren:

BearingPoint GmbH
Comma soft AG
integranova GmbH
Microsoft Deutschland
sd&m AG
SIZ – Informatikzentrum der
Sparkassenorganisation GmbH

www.bearingpoint.de
www.comma-soft.com
www.programmiermaschine.de
www.microsoft.com
www.sdm.de
www.siz.de

Medienpartner:

Computer Zeitung

www.computerzeitung.de

Informatiktage 2007 – Vorwort zum Tagungsband

Sehr geehrte Damen und Herren, liebe Teilnehmerinnen und Teilnehmer,

das Informatikjahr 2006 liegt hinter uns. Fast eine halbe Million Menschen haben an den über 1.500 Veranstaltungen teilgenommen, mit denen versucht wurde, Bewusstsein für die Bedeutung der Informatik in breiten Teilen der Bevölkerung zu schaffen und Wege zur Überwindung der digitalen Spaltung aufzuzeigen, aber auch die Innovationskraft der Informatik in allen Bereichen unserer Gesellschaft über das Label „dank Informatik“ sichtbar zu machen. Die Politik hat diese Impulse mit dem ersten IT-Kanzlergipfel in der Geschichte der Bundesrepublik aufgegriffen, der am 18. Dezember – dem Abschlusstag des Informatikjahrs – in Potsdam stattfand. Konkret bedeutet das für die Informatik ein neues Förderprogramm IKT-2020, das Bundesforschungsministerin Dr. Annette Schavan anlässlich der CeBIT-Eröffnung im März 2007 veröffentlicht hat.

Der pünktlich zum Informatikjahr beginnende Aufschwung rückt aber auch den strategischen IT-Fachkräftemangel in Deutschland wieder in den Blickpunkt, der einige Jahre nach dem Platzen der Internetblase verdeckt war. Die stark gestiegene Beteiligung der Wirtschaft an den Informatiktagen 2007 demonstriert die exzellenten Berufschancen, die sich Informatikabsolventen in den nächsten Jahren bieten, vor allem dann, wenn sie mit wachem Blick die aktuellen Trends der ubiquitären Einbettung der Informatik in Technik, Medizin, Unternehmen und Haushalte verfolgen, durch gute Sprachkenntnisse und interkulturelle Kommunikationsfähigkeit in der Lage sind, in globalen Teams mitzuarbeiten und exzellente technische Kompetenzen mit dem Verständnis verbinden, dass auch Kosten und Marktattraktivität wichtig sind.

Die Informatiktage der Gesellschaft für Informatik bieten alljährlich einigen der besten deutschen Studierenden Gelegenheit, ihre fachlichen Ideen und ihre Zukunftspläne mit Vertretern der Wirtschaft und erfahrenen Wissenschaftlern zu diskutieren. Aus der Ausschreibung um die Beteiligung an den Informatiktagen 2007 sind 41 studentische Beiträge aus 27 Hochschulen hervorgegangen, ferner findet auf den Informatiktagen die Preisverleihung des informatiCup-Studierendenwettbewerbs statt. Die Informatiktage 2007 finden wieder am Bonn-Aachen International Center for Information Technology (B-IT) statt, einer stiftungsgeförderten gemeinsamen Einrichtung der RWTH Aachen und der Universität Bonn, die ich gemeinsam mit dem Bonner Kollegen Armin Cremers 2003 gegründet habe, um internationale Masterstudiengänge in der Angewandten Informatik auf Eliteniveau anzubieten. Wir begrüßen Sie herzlich und wünschen spannende Diskussionen und zukunftsweisende Kontakte.



Matthias Jarke
Präsident der Gesellschaft für Informatik e.V. (GI)

Inhaltsverzeichnis

Grundlagen der Informatik

Bär B.

Benchmark–Vergleich verschiedener Datenstrukturen
für das IP–Lookup–Problem 13

Heinemann C., Nicolai J., Zetsche G.

GI-Gaia - Ein Beitrag zum ersten GI-Studierendenwettbewerb 17

Lülfesmann M.

Partielle Berechnung von Jacobi-Matrizen mittels Graphfärbung 21

Künstliche Intelligenz

Franke M., Gent D., Grieme F., Hühn S., Lewin H.

Ein KI-Verfahren für das Kartenspiel Magic: The Gathering 25

Friede U.

Betrugserkennung bei Versicherungen mit Case Based Reasoning 29

Hentschel C.

Automatische Annotation von Bilddaten auf Grundlage eines
verlässlichen visuellen Wörterbuchs 36

Holz D.

Continuous 3D Environment Sensing for Autonomous Robot
Navigation and Mapping 39

Lauder M.

Erzeugung von vorlesbarem Text aus LATEX-Dokumenten 43

Müller B.

EPOS: Experimental prototype for ontology based search
in a multi agent system 47

Strauch J.

Ontologiebasierte Sprachvernetzung zur intelligenten
Suchunterstützung von Bildinhalten 51

Softwaretechnik

Bartel S.

Werkzeugunterstützung zur Einführung von Adaptionstrukturen
basierend auf einer Service-orientierten Architektur 55

Feja S.

Service-orientierte Integration im E-Commerce 59

Hummel M., Lessner T.

PRREP – Projekt Repository 63

Matzat M., Schmiedel D.

Test von Serviceorientierten Architekturen 67

Nordholz J.

PED - Python External Decompiler 73

Pehmöller A.

Software-Industrialisierung: Auswirkungen der automatisierten
Software-Erstellung auf die Arbeit der Entwickler 77

Reinhardt W.

Pair Programming: Implikationen der industrialisierten Softwareentwicklung
auf die universitäre Informatikausbildung 83

Datenbanken und Informationssysteme

Bednarz M., Bienemann J., Christ A., Kather S., Kelleter N., Pfeiffer D., Röttgerding M., Tiebe D., Tomasoni J., Vehring N.

Entwicklung einer Verkaufsförderungsplattform für integrierte Promotions 87

Holder S.

Towards a Metrics Suite for Object-Relational Mappings 91

Lohe T.

Identifying Protein Domain Insertions in Large Databases

Mlynarski M., Nelkner T., Reinhardt W.

Vom Informations- zum Wissensmanagement:
Der Knowledge Bus als flexibler Lösungsansatz 95

Parchmann J.

How does a project-related Personal Information
Collection reflect the View of a Project? 103

Scheidewig S.

Supporting Presentation Reuse with a Presentation Repository System 107

Wall C.	
Wissensmanagement : Herausforderung Prozesswissen in Konzernen	111

Technische Informatik

Comanns M.	
Wiederverwendbarkeit von Prozessspuren in prozessintegrierten Werkzeugen	115

Frenzel M.	
Entwicklung eines Spiels für Mobile Endgeräte mit Unterstützung wie GPS und UMTS auf Basis der Java2 Micro-Edition	119

Goldau M.	
Kommunikationssensitiver dynamischer Lastausgleich in Multi-Clustern mit Raumfüllenden Kurven	123

Kneissl M. C.	
Challenges across two worlds: The concerns of IT Change Managers to deal with SAP WebAS Java and and WebAS ABAP	127

Pinhammer A.	
Konzeption und Realisierung eines Frameworks für Gruppenkommunikationsanwendungen auf mobilen Endgeräten unter Nutzung von IMS-Diensten	131

Schäfer J.	
Lebenszyklusorientierte Performance-Instrumentierung verteilter Anwendungen	135

Stein T.	
Ein Kommunikations-Framework für die Selbstorganisation von Software-Systemen	139

Graphische Datenverarbeitung

Forster J.	
Reducing Time and RAM Requirements in Content-Based Image Retrieval using Retrieval Filtering	143

Glander T.	
Semantikgetriebene automatische Verfahren für die kartografische Generalisierung von 3D Gebäudemodellen und ihre Visualisierung	147

Kläser A.	
Human Detection and Character Recognition in TV-Style Movies	151

Schneider D.C. Curvature Correction for Camera-based Document Processing	155
--	-----

Vieth M. Reliable Real-time Detection and Tracking of Human Body Parts in Work Safety Relevant Environments	159
--	-----

Zilken O. Reliable Detection and Tracking of Hands in Work Safety Relevant Environments	163
--	-----

Sicherheit – Schutz und Zuverlässigkeit

Holm C., Schwardt C. Verwundbarkeiten von Web-Anwendungen	167
---	-----

Kasper M. Virtualisation of a SIM-Card using Trusted Computing	171
--	-----

Koltes A. Efficient implementation of rainbow table based cryptanalysis tools on general purpose microprocessors	179
---	-----

Trobitius J. Anwendung der "Common Criteria for Information Technology Security Evaluation" (CC) / ISO 15408 auf ein SOA Registry-Repository	183
---	-----

Benchmark–Vergleich verschiedener Datenstrukturen für das IP–Lookup–Problem

Bettina Bär

bbaer@informatik.uni-freiburg.de

Abstract: In den letzten Jahren sind viele Algorithmen für hochperformantes *IP-Address-Lookup* vorgestellt worden. Als mögliche Lösung für das IP-Lookup-Problem wurde eine Datenstruktur auf Basis von Prioritätssuchbäumen vorgeschlagen. Ein Alternativvorschlag sind die Min-augmented Range Trees und die Priority Search Pennants. Diese drei Datenstrukturen sind mit Hilfe eines Benchmarks vergleichend untersucht worden. Alle Operationen auf den drei Strukturen sind in $O(\log n)$ ausführbar. Anhand der verschiedenen Testdaten sind die tatsächlichen Laufzeiten der Operationen durch Messungen verglichen worden. Dabei wurden verschiedene Test-Szenarien durchgeführt.

1 Einleitung

Infolge des rapiden Wachstums des Internetverkehrs sind Backbone¹-Verbindungen von mehreren Gigabits pro Sekunde entwickelt worden. Um diese hohen Übertragungsraten zu bewerkstelligen, müssen die Backbone-Router in der Lage sein, Millionen von Paketen pro Sekunde weiterzuleiten. Diese Weiterleitung ist oft ein Engpass in hochperformanten Routern. Ein schnelles *IP-Address-Lookup* in den Routern, das die Zieladresse des Pakets benutzt, um den Weg für jedes Paket zum nächsten Switch oder Router zu bestimmen, ist wichtig, um die benötigten Paketweiterleitungsraten zu erreichen. Experimentelle Ergebnisse haben ergeben, dass das *IP-Address-Lookup* 56% der Zeit für die gesamte Paketverarbeitung benötigt [LBC03].

In den letzten Jahren sind viele Algorithmen für hochperformantes *IP-Address-Lookup* vorgestellt worden. Lu und Sahni schlagen in ihrem Paper [LS04] Prioritätssuchbäume als eine mögliche Lösung für das IP-Lookup-Problem vor. In [LOD07] präsentieren Lauer, Ottmann und Datta zwei alternative Datenstrukturen, die das Problem lösen. In diesem Artikel werden diese drei Datenstrukturen Prioritätssuchbäume, Min-augmented Range Trees und Priority Search Pennants anhand eines Benchmarks vergleichend untersucht.

¹Backbone kann mit Kernnetz übersetzt werden. Das Backbone-Netz bildet eine Infrastruktur um einen Informationsaustausch bei größeren Vernetzungen mit mehreren gleichen oder unterschiedlichen Netzwerkstrukturen zu schaffen.

2 Das IP-Lookup-Problem

Das IP-Lookup-Problem ist ein spezielles eindimensionales Paketklassifizierungsproblem. Informell bestimmt das Paketklassifizierungsproblem anhand eines oder mehrerer Felder im Header den *Flow*, zu dem das Paket gehört. Die *Flows* werden anhand von Regeln, die auf ankommende Pakete angewendet werden, festgelegt [GM01]. Das Paketklassifikationsproblem besteht nun darin, für jedes Paket die passendste Regel in der Weiterleitungstabelle zu finden. Dabei soll die spezifischste Regel, auf die die gegebene Adresse passt, ausgewählt werden. Die Aufgabe des IP-Lookup-Problems besteht nun darin, die passendste Regel, den sogenannten *most specific range*, $msr(d)$ zu finden.

In [LS04] wurde beobachtet, dass die dynamische Version des eindimensionalen IP-Lookup-Problems in ein geometrisches Problem überführt werden kann. Die Intervalle werden dabei durch Punkte in der Ebene repräsentiert und umgekehrt. Im folgenden wird jedes Intervall $[l, r]$ mit $l \leq r$ auf den Punkt (r, l) abgebildet. Für eine Menge R von Intervallen ist diese Abbildung eine Menge von Punkten auf oder unterhalb der Hauptdiagonalen. Dies wird in Abb. 1 veranschaulicht. Die ursprünglichen Intervalle sind sichtbar als horizontale Linien von der Hauptdiagonalen zu den abgebildeten Punkten.

Anfragen von Paketen können als Aufspießanfragen umgesetzt werden. Dabei spießt ein Punkt d ein Intervall A auf, wenn das Intervall den Punkt enthält (siehe Abb. 1). Aufspießanfragen für eine Menge von Intervallen entsprechen Bereichsanfragen für sogenannte südlich gegründete, semi-unbegrenzte Bereiche von Punkten in der Ebene.

Das IP-Lookup-Problem kann nun gelöst werden, indem der am weitesten links oben liegende Punkt rechts unterhalb des Punktes (d, d) (*most specific range*, $msr(d)$) gefunden wird. Dafür müssen nicht alle Punkte in dem südlich begrenzten Rechteck betrachtet werden. Da es sich ausschließlich um geschachtelte oder disjunkte Intervalle handelt, ist es ausreichend, die $MinXinRectangle(d, \infty, d)$ -Anfrage, zu beantworten (siehe Abb. 1). Diese liefert den linkensten Punkt in dem Rechteck, das durch d, ∞, d und der x -Achse begrenzt ist, wenn ein solcher Punkt existiert [LOD07]. Die Komplexität der Anfragen hängt von der unterliegenden Struktur ab.

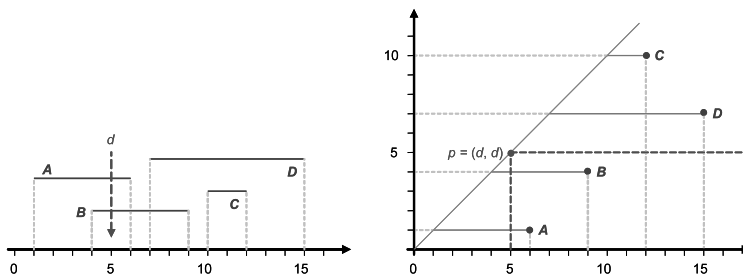


Abbildung 1: Eine Menge von Intervallen wird auf eine Menge von Punkten in der Ebene abgebildet, aus [LOD07]

3 Verwendete Datenstrukturen

Die verwendeten Datenstrukturen sind Strukturen, die es erlauben, Intervalle zu verwalten, so dass direkt Überlappings- und IP-Lookup-Anfragen beantwortet werden können. Dies sind Prioritätssuchbäume (PST) [McC85], Min-augmented Range Trees (MART) [LOD07] und Priority Search Pennants (PSP) [Hin01]. Das Entfernen, Einfügen und die *msr*-Suchen können auf allen drei Strukturen in $O(\log n)$ ausgeführt werden, vorausgesetzt, die Bäume sind balanciert [LOD07]. Lu und Sahni schlagen PSTs als Basis für Router-Tabellen-Designs vor. Da beim MART und PSP mehr Freiheit in Bezug auf das Balancierungsschemata möglich ist, werden diese von Lauer, Ottmann und Datta als Alternativen vorgeschlagen.

4 Ergebnisse

Für die Ausführung der Tests wird unterschieden zwischen *prefix ranges* und *nonintersecting ranges*. Der Grund für diese Trennung ist das Überprüfen auf Überschneidungen vor dem Einfügen eines Intervalls mit den im Baum bereits vorhandenen Intervallen, welches bei den *nonintersecting Ranges* notwendig ist [LOD07]. Bei den Präfix-Intervallen ist die Überprüfung auf Überschneidungen nicht erforderlich, da diese sich nicht überlappen. Getestet wurden unterschiedliche Größen von 1000 bis 1 Million einzufügenden Intervallen. Dabei wurden für jede Größe 20 verschiedene Datensätze generiert und getestet. Aus den Ergebnissen für die 20 Datensätze wurde anschließend der Durchschnitt gebildet. Zur Ausführung von *msr*-Suchen wurden Anfragepunkte, die gleichverteilt über den ganzen IPv4-Adressraum sind, erzeugt. Für die Entferne-Operation wurden zufällig 33% der Intervalle aus den generierten Datensätzen ausgewählt. Die Tests sind auf einem 1.7 GHz AMD Athlon Prozessor mit einem 256 KB L2-Cache und 1 GB RAM ausgeführt worden. Für die Tests wurden sowohl zufällige Daten als auch Simulationsdaten, die versuchen, echte Router-Tabellen abzubilden, generiert.

Aufgrund der Flexibilität bezüglich des Balancierungsschemas bei dem PSP und dem MART sind zwei unterschiedliche Balancierungsmöglichkeiten näher untersucht worden. Außerdem wurden zwei unterschiedliche Varianten des PST getestet. Für die *msr*-Suchen wurde sowohl die iterative als auch die rekursive Variante getestet. Es zeigt sich, dass die iterative Variante der Implementierung deutlich weniger Zeit benötigt. Für das Überprüfen auf Überschneidungen wurde der von Lu und Sahni [LS04] vorgestellte Ansatz und die Lösung, die von Lauer, Ottmann und Datta [LOD07] präsentiert wurde, verglichen.

Ein exemplarisches Hauptergebnis ist in Abb. 2 zu sehen. Dabei sind 100.000 Präfix-Intervalle eingefügt worden. Es stellt sich heraus, dass der PSP für das Einfügen und Entfernen die effizienteste Struktur ist. Die Suchzeiten sind jedoch beim MART geringer. Bei den *msr*-Suchen zeigt sich, dass sich Vergleichsoperationen wesentlich mehr auf die Laufzeit auswirken als das Verfolgen von Zeigern.

Für das Einfügen von *nonintersecting ranges* ist der MART die schnellste Struktur. In diesem Ergebnis macht sich bemerkbar, dass der MART bei der MinXinRectangle-Anfrage

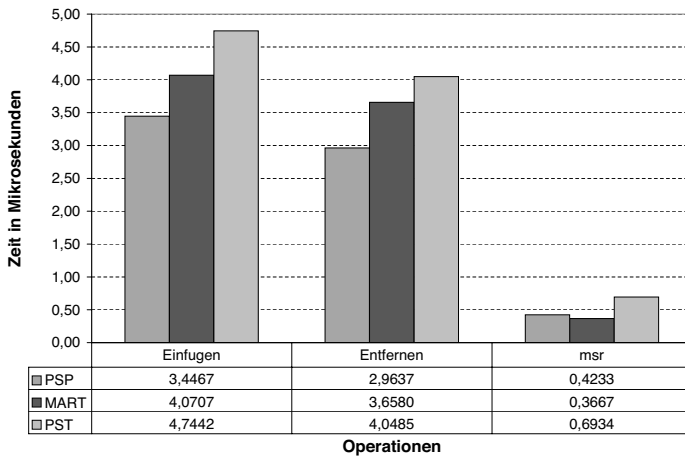


Abbildung 2: Zeitvergleich pro Operation für das Einfügen, Entfernen und Suchen in Präfix-Intervallen.

schneller ist als der PSP. Da vor dem Einfügen eines *nonintersecting ranges* zwei Min-XinRectangle-Anfragen (Überprüfen auf Überschneidungen) ausgeführt werden müssen, ist dies mit großer Wahrscheinlichkeit ausschlaggebend dafür, dass der MART weniger Zeit benötigt. Auch für die *nonintersecting ranges* erfordern die getesteten Operationen auf dem MART und dem PSP weniger Zeit als auf dem PST.

Zusammenfassend lässt sich festhalten, dass Einfügen und Entfernen auf dem MART und PSP aufgrund ihres einfacheren Konzepts wie erwartet schneller sind als beim PST. Als überraschend und positiv stellt sich heraus, dass die *msr*-Suchen auf diesen beiden Strukturen ebenfalls weniger Zeit benötigen.

Literatur

- [GM01] P. Gupta und N. McKeown. Algorithms for packet classification. *Network*, 15(2):24–32, 2001.
- [Hin01] Ralf Hinze. A Simple Implementation Technique for Priority Search Queues. In *International Conference on Functional Programming*, Seiten 110–121, 2001.
- [LBC03] Yan Luo, Laxmi Narayan Bhuyan und Xi Chen. Shared Memory Multiprocessor Architectures for Software IP Routers. *IEEE Transactions on Parallel and Distributed Systems*, 14(12):1240–1249, 2003.
- [LOD07] T. Lauer, T. Ottmann und A. Datta. Update-Efficient Data Structures for Dynamic IP Router Tables. *Int'l Journal of Foundations of Computer Science*, 18(1):139–161, 2007.
- [LS04] Haibin Lu und Sartaj Sahni. $O(\log n)$ Dynamic router-tables for prefixes and ranges. *IEEE Transactions on Computers*, 53(10):1217–1230, 2004.
- [McC85] Edward M. McCreight. Priority Search Trees. *SIAM Journal on Computing*, 14(2):257–276, 1985.

GI-Gaia - Ein Beitrag zum ersten GI-Studierendenwettbewerb

Christian Heinemann*

Johannes Nicolai†

Georg Zetzsche‡

Abstract: Der InformatiCup (früher unter dem Namen GI-Studierendenwettbewerb) richtet sich bundesweit an Studierende in der ersten Hälfte des Studiums. Teilnehmen können Gruppen mit zwei oder drei Teilnehmern. In diesem Beitrag wird die Ausarbeitung der Siegergruppe des ersten Wettbewerbs, bestehend aus Christian Heinemann (Humboldt-Universität zu Berlin), Johannes Nicolai (Hasso-Plattner-Institut Potsdam) und Georg Zetzsche (Universität Hamburg), kurz vorgestellt. Die bearbeitete Aufgabe bestand darin, die Berechnung des größten gemeinsamen Teilers einer endlichen Zahlenmenge mithilfe eines verteilten Algorithmus zu implementieren, wobei die an der Berechnung beteiligten Prozesse in einem logischen Ring angeordnet sind. Stellvertretend für die gesamte Ausarbeitung sollen die Algorithmen für Start und Berechnung sowie ihre Modellierung als gefärbtes Petrinetz vorgestellt werden.

1 Einführung

Der InformatiCup ist ein Wettbewerb der Gesellschaft für Informatik, der jährlich stattfindet und sich an Studierende in der ersten Hälfte des Studiums richtet. Er fand zum ersten Mal im Jahr 2005 statt und hieß damals GI-Studierendenwettbewerb (die zugehörige Homepage ist unter [GISW] zu finden). Die Teilnehmer bearbeiten eine von drei vorgegebenen Aufgaben in Heimarbeit und senden ihren Beitrag ein. Die Teams mit den besten Beiträgen der ersten Runde werden dann zur zweiten Runde eingeladen, um dort eine Präsentation zu ihrer Ausarbeitung und eventuellen Verbesserungen zu halten.

Unsere Gruppe entschied sich für die zweite Aufgabe, deren Gegenstand die Implementierung von verteilten Algorithmen zur Berechnung des größten gemeinsamen Teilers war. In der Aufgabenstellung (ebenfalls unter [GISW] verfügbar) war vorgegeben, wie die Berechnung geschehen sollte, d.h. wie auf ankommende Nachrichten im verteilten Algorithmus reagiert werden soll. Außerdem sollte die Kommunikation der Prozesse in einem Ring erfolgen. Es wurden jedoch viele Aspekte offengelassen, die dann durch die Teilnehmer zu konkretisieren waren. Dazu gehörten unter vielen anderen die folgenden Fragen.

1. Wie wird die Berechnung gestartet?
2. Wie wird festgestellt, ob die Berechnung fertiggestellt ist?

*Humboldt-Universität zu Berlin, heinemann.christian@googlemail.com

†Hasso-Plattner-Institut für Softwaresystemtechnik, johannes.nicolai@student.hpi.uni-potsdam.de

‡Universität Hamburg, 3zetzsche@informatik.uni-hamburg.de

3. Arbeitet die gewählte Kombination von Algorithmen korrekt?

Zur ersten Frage muss geklärt werden, welche Prozesse am Anfang der Berechnung Nachrichten aussenden und an wen. Der *Startalgorithmus*, der dazu verwendet wird, muss insbesondere sicherstellen, dass beim vorgegebenen Berechnungsverfahren die Nachrichten nicht vor dem Erreichen des Ergebnisses „versiegen“, sondern die Berechnung nach endlicher Zeit zu einem Ergebnis kommt.

Die zweite Frage zielt auf einen *Terminationsalgorithmus*. Seine Aufgabe ist es, zuverlässig festzustellen, wann die Berechnung zu einem Ergebnis gekommen und beendet ist. Außerdem soll er dafür sorgen, dass das Ergebnis an einen ausgezeichneten Prozess gelangt, der es dann dem Benutzer darstellen kann. Im Beitrag haben wir hierfür zwei Terminationsalgorithmen vorgestellt, die sich jeweils für verschiedene Einsatzzwecke eignen.

Die in der dritten Frage geforderte Verifikation der Algorithmen haben wir in der eingereichten Ausarbeitung in Form von Korrektheitsbeweisen für alle Algorithmen geliefert. Die Korrektheit von Start- und Berechnungsalgorithmus wurde dabei anhand einer Petrinetz-Modellierung der Algorithmen und mithilfe von Argumenten aus der Transaktionstheorie von Datenbanken bewiesen.

Die vollständige Ausarbeitung kann im Web unter [Gaia] gefunden werden.

2 Berechnung

Der in der Aufgabenstellung vorgegebene Algorithmus zur Berechnung des ggT legt fest, welche Daten die Prozesse verwalten und wie sie auf ankommende Nachrichten reagieren.

Wir nehmen an, zu den natürlichen Zahlen z_0, \dots, z_{n-1} der größte gemeinsame Teiler berechnet werden. Im Algorithmus soll es genau so viele Prozesse geben wie es Zahlen gibt, deren größter gemeinsamer Teiler berechnet werden soll, d.h. n Stück. Die Prozesse werden mit P_0, \dots, P_{n-1} bezeichnet. Im Verlauf der Berechnung besitzt dabei jeder Prozess eine Zahl, die zu Anfang die Zahlen z_0, \dots, z_{n-1} sind.

Die gesendeten Nachrichten enthalten ebenfalls jeweils eine Zahl. Empfängt nun Prozess P_i eine Nachricht mit der Zahl y von seinem Nachbarn, so prüft er, ob die empfangene Zahl kleiner ist als seine eigene Zahl x . Ist dies der Fall, so berechnet er $((x - 1) \bmod y) + 1$, wählt dies als seine neue eigene Zahl und sendet sie an seine beiden Nachbarn. Andernfalls ignoriert er die Nachricht.

3 Start

Der Startalgorithmus wird auf Initiative des Benutzers vom Koordinator gestartet. Dabei werden alle Prozesse vom Koordinator dazu veranlasst, mit der Berechnung zu beginnen. Dazu sendet jeder Prozess seine eigene Zahl an seinen rechten Nachbarn, der gemäß des vorgegebenen Berechnungsalgorithmus darauf reagiert.

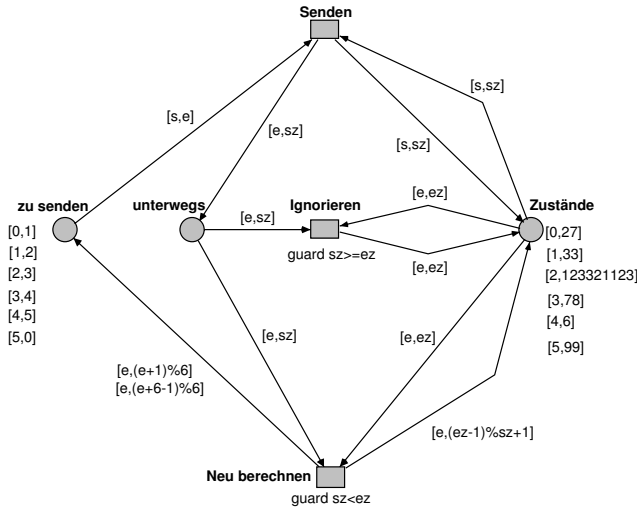


Abbildung 1: Gefärbtes Petrinetz zum Startverfahren und zum vorgegebenen Berechnungsverfahren

Das Startverfahren wird zusammen mit dem vorgegebenen Berechnungsverfahren in Abbildung 1 als gefärbtes Petrinetz¹ dargestellt. Die Kantenbeschriftung ist dabei nach der Syntax des Petrinetz-Tools Renew [Renew] gewählt: Eine Marke der Form $[x, y]$ ist eine Liste, die die beiden natürlichen Zahlen x und y enthält. Die Beschriftung $\text{guard } sz \leq ez$ bzw. $\text{guard } sz < ez$ an den Transitionen *Ignorieren* bzw. *Neu berechnen* bedeuten, dass diese Transitionen nur für solche Belegungen der Variablen schalten dürfen, die die Bedingung $sz \leq ez$ bzw. $sz < ez$ erfüllen. Die Ausdrücke $(e+1) \% 6$, $(e+6-1) \% 6$, bzw. $(ez-1) \% sz + 1$ sind schließlich, wie in Renew üblich, als Java-Ausdrücke zu verstehen, d.h. hier wird $(e+1) \bmod 6$, $(e+6-1) \bmod 6$ bzw. $((ez-1) \bmod sz) + 1$ berechnet.

In der Abbildung ist eine beispielhafte Anfangsmarkierung für sechs beteiligte Prozesse angegeben. Wir erklären kurz die Bedeutung der einzelnen Stellen und Transitionen.

- Die Stelle *Zustände* enthält stets zu jedem Prozess seine momentane Zahl, d.h. eine Marke $[x, xz]$ in dieser Stelle bedeutet, dass Prozess xz in diesem Moment die Zahl von Prozess x ist. Sind n Prozesse an der Berechnung beteiligt, so werden die Prozesse im Netz durch die Nummern $0, \dots, n-1$ repräsentiert; dabei soll im Ring jeweils der Prozess i die Nachbarn $(i+1) \bmod n$ bzw. $(i+n-1) \bmod n$ haben. Man beachte, dass jede Transition durch zwei Kanten mit dieser Stelle verbunden ist, und zwar so, dass stets (d.h. in jeder erreichbaren Markierung) zu jedem Prozess genau eine Marke in dieser Stelle liegt.
- Die Marken in der Stelle *zu senden* beschreiben Nachrichten, die noch gesendet werden sollen, d.h. eine Marke $[x, y]$ in dieser Stelle bedeutet, dass Prozess x eine

¹ Gefärbte Petrinetze werden in [Jen92] beschrieben.

Nachricht an Prozess y senden wird.

- Die Marken in der Stelle *unterwegs* repräsentieren Nachrichten, die gerade zu einem Prozess unterwegs sind. Eine Marke $[x, yz]$ bedeutet, dass eine Nachricht mit der Zahl yz an den Prozess x unterwegs ist.
- Die Transition *Senden* führt den Wunsch eines Prozesses aus, eine Nachricht an einen anderen Prozess zu senden. Dafür nimmt sie eine entsprechende Marke aus der Stelle *zu senden*, liest den Zustand des Senders aus der Stelle *Zustände* und legt eine Marke in *unterwegs* ab. Somit ist dann s die Nummer des sendenden Prozesses, e die Nummer des Empfängers und sz der momentane Zustand des Senders (die Zahl, die versandt wird).
- Die Transitionen *Ignorieren* bzw. *Neu berechnen* beschreiben den Empfang einer Nachricht aus der Stelle *unterwegs*. Beide Transitionen vergleichen die Zahl in der Nachricht sz mit dem aktuellen Zustand ez des Empfängers, indem die zum Empfänger gehörige Zustandsmarke $[e, ez]$ gelesen wird. Die Transition *Ignorieren* schaltet nur dann, wenn die Zahl in der Nachricht mindestens so groß ist wie die Zahl des Empfängers und ändert den Zustand des Empfängers nicht. Ist jedoch die Zahl in der Nachricht kleiner als die des Empfängers, so schaltet *Neu berechnen* und die Zustandsmarke des Empfängers ändert sich wie im vorgegebenen Algorithmus gefordert ($[e, (ez-1) \% sz + 1]$ wird als neue Zustandsmarke abgelegt) und es werden zwei neue Nachrichten-Marken an die Nachbarn des Empfängers gesendet.

Das von uns verwendete Startverfahren wird im Netz durch die Anfangsmarkierung dargestellt, in der Abbildung beispielhaft für sechs Prozesse: für jeden Prozess i enthält die Stelle *zu senden* eine Marke, die das Senden einer Nachricht von i an den rechten Nachbarn von i zur Folge hat. Die Stelle *Zustände* enthält für jeden Prozess i eine Marke, die dessen Zahl zu Beginn der Berechnung beschreibt.

Literatur

- [Gaia] Das Projekt GI-Gaia bei SourceForge: <http://gi-gaia.sf.net/>.
- [GISW] Die Homepage des 1. GI-Studierendenwettbewerbs: <http://www.gi-ev.de/wir-ueber-uns/wettbewerbe/informaticup/1-studierendenwettbewerb-der-gi-20052006/>
- [Jen92] K. Jensen. *Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use. Volume 1: Basic Concepts*. EATCS Monographs on Theoretical Computer Science. Springer-Verlag, 1992.
- [Renew] Renew: The Reference Net Workshop, <http://www.renew.de/>.

Partielle Berechnung von Jacobi-Matrizen mittels Graphfärbung

Michael Lulfesmann

Lehrstuhl für Informatik 12 (Hochleistungsrechnen), RWTH Aachen

D-52056 Aachen

lulfesmann@sc.rwth-aachen.de

Abstract: Zur Vorkonditionierung eines linearen Gleichungssystems wird oftmals nur eine Teilmenge der Nichtnullelemente der Koeffizientenmatrix benötigt. Deshalb wird ein Verfahren betrachtet, das es ermöglicht, diese Teilmenge von Elementen mit geringerem Aufwand im Vergleich zu allen Nichtnullelementen zu berechnen. Wegen der Dünnbesetztheit der Koeffizientenmatrix wird eine Graphmodellierung verwendet.

Im wissenschaftlichen Rechnen sind häufig kombinatorische Probleme zu lösen, die durch Graphen modelliert werden können. Insbesondere in der numerischen linearen Algebra sind Graphen allgegenwärtig, wenn die zugrundeliegenden Matrizen dünnbesetzt sind. Der Ursprung der in dieser Arbeit betrachteten Problemstellungen liegt in der Lösung eines linearen Gleichungssystems, dessen Koeffizientenmatrix eine dünnbesetzte, große und nicht-singuläre Jacobi-Matrix J ist. Zum Lösen dieser Systeme werden iterative Verfahren genutzt, die in der Praxis durch eine Vorkonditionierung beschleunigt werden [SK04]. Zur Vorkonditionierung können Methoden verwendet werden, die nur eine Teilmenge der Nichtnullelemente von J benötigen. Bisher wurden zur Bestimmung des Vorkonditionierers alle Nichtnullelemente von J berechnet. Darum wird dieses Vorgehen als vollständige Berechnung bezeichnet. Bei der partiellen Berechnung muss hingegen nur noch diejenige Teilmenge der Nichtnullelemente, die zur Vorkonditionierung benötigt werden, mit möglichst geringem Aufwand berechnet werden.

Bisher wurde eine Jacobi-Matrix J vollständig berechnet, z.B. mit der Methode der Finiten Differenzen [SK04] oder dem Automatischen Differenzieren [Gri00, Ra181]. Im einfachsten Fall werden dabei alle Spalten oder alle Zeilen einzeln berechnet. Mit entsprechender Initialisierung können Berechnungen gespart werden, da mehrere Zeilen/Spalten von J durch eine einzelne Linearkombination berechnet werden können. Die jeweiligen Spalten bzw. Zeilen dürfen gleichzeitig berechnet werden, wenn sie paarweise kein Element in derselben Komponente enthalten [CPR74]. Linearkombinationen nur von Spalten oder nur von Zeilen können durch eine sogenannte unidirektionale Partitionierung von J berechnet werden, wobei eine Linearkombination dann einer Gruppe der Partition entspricht [CPR74]. Die Nichtnullelemente aus J können dabei aus den berechneten Linearkombinationen abgelesen werden. In Abb. 1, in der Nichtnullelemente durch das Symbol \times gekennzeichnet sind, ist an der Farbe der Spalten zu erkennen, dass die Matrix J_1 in fünf Gruppen $\{\{1\}, \{2\}, \{3\}, \{4\}, \{5\}\}$ partitioniert worden ist, so dass alle Spalten einzeln berechnet werden. Die Matrix J_2 , die dieselbe Struktur wie J_1 hat, ist hingegen nur in drei Gruppen $\{\{1, 2\}, \{3\}, \{4, 5\}\}$ partitioniert worden, obwohl trotzdem alle Nichtnull-

elemente berechnet werden. Die Spalten 1 und 2 von J_2 dürfen z.B. zusammengefasst werden, da in keiner Zeile beide Spalten ein Nichtnullelement enthalten. Für die Spalten 2 und 3 gilt dieses hingegen nicht, weil sie beide in der 3. Komponente ein Element haben.

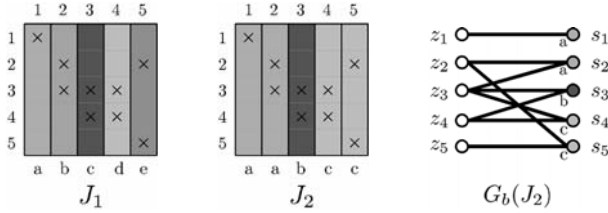


Abbildung 1: J_1, J_2 : partitionierte Matrizen (entsprechend mit Farben aus der Menge $\{a, \dots, e\}$ gefärbt), Dünnbesetztheitsmuster einer Jacobi-Matrix J ; $G_b(J_2)$: bipartiter Graph der J_2 repräsentiert und entsprechend der Partitionierung gefärbt ist

Dieses Partitionieren von J kann auch als einseitige Graphfärbung eines bipartiten Graphen modelliert werden [CM83, HS98]. Die bidirektionale Partitionierung, bei der Linearkombinationen von Spalten und Zeilen berechnet werden, kann entsprechend als zweiseitige Färbung modelliert werden [CV98, HS98, HS01, HS02]. Es wird in beiden Fällen von vollständiger Färbung gesprochen, da alle Nichtnullelemente von J berechnet werden sollen. Hierzu wird die Dünnbesetztheitsstruktur einer Jacobi-Matrix J durch einen bipartiten Graphen $G_b(J) = (V_z, V_s, E)$, dessen Knotenmenge V_z für jede Zeile i einen Knoten z_i und dessen Knotenmenge V_s für jede Spalte k einen Knoten s_k enthält, eindeutig dargestellt. Die Kantenmenge E enthält für jedes Nichtnullelement $j_{i,k}$ eine Kante (z_i, s_k) . Bei der einseitigen Färbung werden alle Knoten, die entweder den Spalten oder den Zeilen entsprechen, so gefärbt, dass zwei Knoten, die über einen Pfad der Länge zwei verbunden sind, unterschiedlich gefärbt sind. Die Spalten bzw. Zeilen von J , deren entsprechende Knoten im Graphen identisch gefärbt sind, können durch eine einzelne Linearkombination berechnet werden. Die Minimierung der Gruppen von Spalten oder Zeilen entspricht also der Minimierung der Farbanzahl im Graphen. Die kompliziertere, zweiseitige Färbung wird der Kürze wegen ausgelassen.

Zur partiellen Berechnung wird die Initialisierung so angepasst, dass eine Teilmenge der Elemente von J oftmals mit noch weniger Linearkombinationen berechnet werden kann [Geb03, GMP05]. Hierzu können nun auch Spalten oder Zeilen zusammengefasst werden, wenn in derselben Komponente nur zwei nicht benötigte Elemente enthalten sind. Als Beispiel für die partielle Berechnung betrachten wir in Abb. 2 die Matrizen J_2 und J_3 . Die benötigten Elemente werden durch das Symbol \otimes gekennzeichnet und die nicht benötigten Elemente durch das Symbol \times . Die Spalten 2 und 3 der Matrix J_3 können zusammengefasst werden, da hier beide 3. Komponenten im Gegensatz zur Matrix J_2 jeweils ein nicht benötigtes Element enthalten. Dadurch werden nur zwei Gruppen anstelle von drei Gruppen benötigt. Die Anpassung des Graphmodells für die partielle Berechnung besteht darin, die Kanten danach zu unterscheiden, ob sie einem benötigten oder einem nicht benötigten Element entsprechen. Im Graphen $G_b(J_2)$ in Abb. 2 wird das dadurch erreicht, dass die benötigten Kanten dicker gezeichnet werden als die nicht benötigten Kanten. Bei der einseitigen, beschränkten Färbung müssen somit nur noch die Knoten unterschiedlich gefärbt werden, die durch einen Pfad verbunden sind, der die Länge zwei hat und eine Kante

enthält, die einem benötigten Element entspricht. Somit müssen in $G_b(J_3)$ die Knoten s_2 und s_3 im Gegensatz zu den Knoten s_2 und s_4 nicht unterschiedlich gefärbt werden, da sie nur über den Pfad (s_2, z_3, s_3) miteinander verbunden sind und dieser Pfad keine Kante enthält, die einem benötigten Element entspricht. Die Färbungsbedingungen für die zweiseitige, beschränkte Färbung werden der Kürze wegen wiederum ausgelassen.

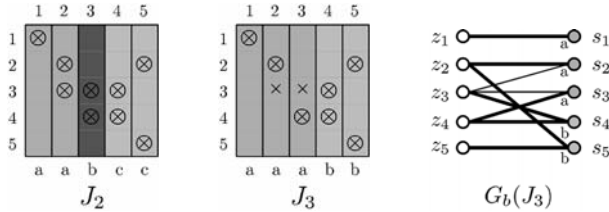


Abbildung 2: J_2 , J_3 : partitionierte Matrizen, Dünnbesetzmuster einer Jacobi-Matrix J ; $G_b(J_3)$: bipartiter Graph der J_3 repräsentiert und entsprechend der Partitionierung gefärbt ist (benötigtes Element: \otimes , nicht benötigtes Element: \times , Farben: a, b und c)

In der zugrundeliegenden Diplomarbeit [Lü06] wird für diese vier Minimierungsprobleme (einseitige und vollständige, zweiseitige und vollständige, einseitige und beschränkte sowie zweiseitige und beschränkte Färbung) erstmals die NP-Schwierigkeit gezeigt. Deshalb werden auch Heuristiken zum Färben der bipartiten Graphen vorgestellt. Dazu wird eine Heuristik zur einseitigen und vollständigen Färbung und eine Heuristik zur zweiseitigen und vollständigen Färbung betrachtet. In [Lü06] werden diese erstmals auf die jeweilige partielle Färbung angepasst und anschließend implementiert. Zudem werden Algorithmen zur Vorsortierung von Knotenmengen betrachtet, um die Anzahl der benötigten Farben beim Färben zu reduzieren.

Abschließend wird hier ein kleiner Ausschnitt der Ergebnisse aus [Lü06] betrachtet. Dazu werden die bipartiten Graphen von vier Matrizen (Quelle: [Dav]) zuerst vollständig und dann beschränkt gefärbt, wobei bei der beschränkten Färbung die Hauptdiagonalelemente als benötigten Elemente betrachtet werden. In Tabelle 1 sind die von den Heuristiken zum Färben benötigten Farbanzahlen aufgeführt. Beide beschränkte Färbungen benötigen für die gesuchte Teilmenge dieselbe Farbanzahl. Die Färbungen benötigten im Vergleich zur einseitigen, vollständigen Färbung nur zwischen 0,003% und 30% der Farben (Linearkombinationen) und im Vergleich zur zweiseitigen, vollständigen Färbung, die im Allgemeinen weniger Farben als die beschränkte, einseitige Färbung benötigt, nur 5% bis 30% der Farben. Also wird der Aufwand zur partiellen Berechnung zumindest bei den benötigten Elementen, die hier gewählt wurden, signifikant reduziert.

Matrix	Vollständige Färbung		Beschränkte Färbung	
	Einseitig	Zweiseitig	Einseitig	Zweiseitig
barrier2-10	8.437	118	26	26
hcircuit	1.399	23	5	5
para-5	6.931	532	27	27
turon_m	23	23	7	7

Tabelle 1: Mit den Färbungsheuristiken aus [Lü06] berechnete Farbanzahlen

Literatur

- [CM83] T. F. Coleman und J. J. Moré. Estimation of Sparse Jacobian Matrices and Graph Coloring Problems. *SIAM J. Numer. Anal.*, 20(1):187–209, 1983.
- [CPR74] A. R. Curtis, M. J. D. Powell und J. K. Reid. On the Estimation of Sparse Jacobian Matrices. *Journal of the Institute of Mathematics and Applications*, 13:117–119, 1974.
- [CV98] T. F. Coleman und A. Verma. The Efficient Computation of Sparse Jacobian Matrices Using Automatic Differentiation. *SIAM J. Sci. Comput.*, 19(4):1210–1233, 1998.
- [Dav] T. Davis. University of Florida Sparse Matrix Collection. <http://www.cise.ufl.edu/research/sparse/matrices>. NA Digest, vol. 92, no. 42, October 16, 1994, NA Digest, vol. 96, no. 28, July 23, 1996, and NA Digest, vol. 97, no. 23, June 7, 1997.
- [Geb03] A. H. Gebremedhin. *Practical Parallel Algorithms for Graph Coloring Problems in Numerical Optimization*. Dissertation, University of Bergen, February 2003.
- [GMP05] A. H. Gebremedhin, F. Manne und A. Pothén. What Color Is Your Jacobian? Graph Coloring for Computing Derivatives. *SIAM Review*, 47(4):629–705, 2005.
- [Gri00] A. Griewank. *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation*. Number 19 in Frontiers in Appl. Math. SIAM, Philadelphia, PA, 2000.
- [HS98] S. Hossain und T. Steihaug. Computing a Sparse Jacobian Matrix by Rows and Columns. *Optimization Methods and Software*, 10:33–48, 1998.
- [HS01] S. Hossain und T. Steihaug. Reducing the Number of AD Passes for Computing a Sparse Jacobian Matrix. In G. Corliss, C. Faure, A. Griewank, L. Hascoët und U. Naumann, Hrsg., *Automatic Differentiation of Algorithms: From Simulation to Optimization*, Computer and Information Science, Kapitel 31, Seiten 263–270. Springer, New York, NY, 2001.
- [HS02] S. Hossain und T. Steihaug. Sparsity Issues in the Computation of Jacobian Matrices. In *ISSAC '02: Proceedings of the 2002 International Symposium on Symbolic and Algebraic Computation*, Seiten 123–130, New York, NY, USA, 2002. ACM Press.
- [Lü06] M. Lülfsmann. Graphfärbung zur partiellen Berechnung von Jacobi-Matrizen. Diplomarbeit, Fachgruppe Informatik, RWTH Aachen, 2006.
- [Ral81] L. B. Rall. *Automatic Differentiation: Techniques and Applications*, Bd. 120 of *Lecture Notes in Computer Science*. Springer, Berlin, 1981.
- [SK04] H.R. Schwarz und N. Köckler. *Numerische Mathematik*. Teubner, 2004.

Ein KI-Verfahren für das Kartenspiel Magic: The Gathering

Marco Franke, Daniel Gent, Florian Grieme,
Kaan Hatunoglu, Sebastian Hühn, Heiko Lewin,

Universität Bremen
Informatik

{franke, deg, griemo, targa, shuehn, hlewin}@informatik.uni-bremen.de

Art der Arbeit: Projekt über vier Semester im Hauptstudium Informatik (Diplom).

Betreuer der Arbeit: Prof. Dr. Karl-Heinz Rödiger, Dr. Jan Plath

Abstract: Ziel des Projekts ist, im Kartenspiel Magic: The Gathering menschliche Spieler durch eine algorithmische Lösung zu ersetzen. Die Lösung setzt auf dem Programm UCA¹ auf; sie ist in der Lage, einen Anfänger² durch eine integrierte KI zu besiegen.

1 Einleitung

Das Projekt LOSER beschäftigt sich mit der Entwicklung einer algorithmischen Lösung (AL) für das Spiel Magic: The Gathering (MTG). MTG ist ein strategiebasiertes Sammelkartenspiel. Rundenbasiert wechseln sich zwei Spieler mit dem Ausspielen von Karten ab. Der Reiz bei MTG liegt darin, dass es Karten gibt, die während eines gegnerischen Zuges ausgespielt werden können. Zusätzlich ist es möglich durch geschicktes Kombinieren unterschiedlicher Karten Teile des Regelwerks vorübergehend oder anhaltend zu verändern. Ein vollständiges Regelwerk (Comprehensive Rules) findet sich auf der folgenden Seite:

<http://www.wizards.com/default.asp?x=magic/rules/tourneyplayer>

2 Lösungsansätze für MTG

Bislang entwickelte Lösungen arbeiten mit festen Mustern und unveränderbaren Decks; sie sind leicht innerhalb kurzer Zeit zu besiegen. Nur wenige arbeiten mit einer KI. Beispielsweise besitzt das Programm Magic The Gathering: Duels of the Planeswalkers³ eine solche KI. Allerdings sind nur Karten mit vereinfachten Regelsatz gewählt wurden.

¹ UCA wird von Jan Plath entwickelt.

² Anfänger ist ein Spieler, der jede Runde eine Kreatur ins Spiel bringt und damit angreift.

³ Am 31.01.1998 von Atari veröffentlicht.

Zusätzlich erscheinen einige von der KI ausgewählte Aktionen unnötig zu sein; sie führen unter Umständen zu einem schlechteren Spielzustand.

MagicMachine (MagMa)⁴ bietet ebenfalls die Möglichkeit gegen einen Computergegner zu spielen. Zur Einschätzung der Spielstärke wurden einige Dutzend Runden gegen die stärkste Stufe des Programms gespielt. Nach jeder Runde wurden die Decks untereinander ausgetauscht. Die KI konnte mit einer Gewinnquote von weniger als 10% nicht überzeugen. Anfänger können die KI innerhalb weniger Spiele besiegen.

Die hohe Komplexität des beliebten und auf den ersten Blick einfachen Spiels erlaubt offensichtlich bislang keine halbwegs überzeugende algorithmische Lösung. Die Berechnung der Angreifer- und Blockerdeklarationen beispielsweise führt selbst unter vereinfachten Bedingungen zu exponentiellem Aufwand.

Die beim Schachspiel mit einem Computer verwendeten Methoden sind für MTG praktisch unbrauchbar. So wird heutzutage trotz vieler Ansätze, Spiele mit wissensbasierten Methoden zu lösen, die aus dem menschlichen Denken abgeleitet sind, in den meisten Fällen noch mit Suchbaum-basierten Brute-Force-Methoden gearbeitet. Bei komplexen Spielen wie MTG nimmt der Suchbaum Dimensionen an, die mit heutigen Rechnern – und auch mit den nächsten Rechnergenerationen - nicht abzarbeiten sind.

3 Zielsetzung und Inhalt des Projekts

Ziel des Projekts ist, die Schwächen der bisherigen KI-Ansätze zu beseitigen und eine Lösung zu entwickeln, die in der Lage ist, einen Anfänger zu besiegen. Wir wollen keine festen Muster und Abläufe verwenden; vielmehr werden Entscheidungen mittels Plänen und Zustandsbewertungen getroffen. Zusätzlich werden Suchbäume benutzt, die mit vereinfachten Annahmen und Pruning arbeiten. Die KI soll dabei so flexibel sein, dass sie dem menschlichen Agieren und Reagieren möglichst nahe kommt. Die Zustandsbewertungen werden deshalb so angelegt, dass Pläne ggf. korrigiert werden können. Dies wird durch eine dynamische Anpassung der Gewichte unterstützt.

3.1 Zustandsbewertungen

Zustandsbewertung [RN03] ist ein zentrales Thema für dieses Projekt. Soll die Lösung für MTG auch nur annähernd intelligent spielen, ist die Auswahl an relevanten Spielinformationen von größter Bedeutung. Das Spiel ist zu komplex, um alle Informationen zu jedem Zeitpunkt erfassen und bewerten zu können. Deshalb haben wir im Vorfeld relevante Informationen spezifiziert. Beispielsweise werden die Historie und einzelne Effekte von Karten, die sich im Spiel befinden, bewertet. Zu den einzelnen Informationen werden entsprechende Bewertungskriterien erstellt und mit einem Gewicht versehen. Diese Gewichte werden beim Spielstart gesetzt; ihre Werte hängen

⁴ Das Programm wurde ursprünglich von Constanzo Zingrillo entwickelt und später von Alessandro Bernadi weitergeführt.[Z107]

von der Kategorie des Spieler-Decks ab. Im Verlauf des Spiels werden einzelne Gewichte dynamisch angepasst. Die Gewichte werden immer dann angepasst, wenn neue Karten ins Spiel gebracht oder Karten aus dem Spiel entfernt werden. Das Ausspielen einer Effekt-verändernden Karte erfordert ebenfalls eine neue Gewichtung. Dabei ist es möglich, dass mehrere Karten ein Gewicht beeinflussen.

3.2 Pläne

In MTG ist eine gute Aktion zumeist das Resultat einer vorteilhaften Kombination von konkreten Spielaktionen, d.h. das Ausführen von Aktionen in einer bestimmten Reihenfolge zu bestimmten Zeitpunkten. Dies ergibt sich zum Teil daraus, dass Aktionen Voraussetzungen haben, die sich auf ihre Kosten oder die Ziele ihrer Effekte beziehen, und zum Teil erst durch vorheriges Ausführen von anderen Aktionen erfüllt werden können. Wegen der hohen Komplexität von MTG wäre es zu aufwändig, solche Aktionsfolgen durch eine klassische Spielbaumsuche zu finden. Deshalb versuchen wir, durch Verwendung von Planungsalgorithmen mit geringerem Aufwand zu diesen Aktionssequenzen zu gelangen.

Voraussetzungen von Aktionen sind oftmals Anforderungen an bestimmte Merkmale, die an der Aktion beteiligte Objekte aufweisen müssen. Wir definieren für diese Merkmale jeweils die Menge der Aktionen, die das betreffende Merkmal potenziell verändern können. Zusätzlich werden für weitere abstrakte Ziele (z.B. Zonenwechsel von Spielobjekten) Mengen von Aktionen bestimmt, die diese erreichen können. Mit diesem Ansatz sollen anhand von Zielen Aktionssequenzen erzeugt werden können. Dafür müssen Ziele entweder von vornherein vorhanden sein oder dynamisch gefunden werden können.

Dazu wurde eine Komponente entwickelt, deren Aufgabe die Beobachtung von Spielobjekten und deren Auswirkungen auf den Spielzustand ist. Wenn dabei Spielobjekte auffallen, die die Bewertung des Spielzustands maßgeblich verschlechtern, wird ein Ziel definiert, mittels dessen das Spielobjekt entfernt wird. Eine weitere Komponente, die Ziele generieren kann, ist die Deckerkennung; durch sie werden Decks kategorisiert. In MTG existieren einige Archetypen von Decks, die gewisse Grundstrategien verfolgen. Diese sollen durch einige grundlegende Ziele modelliert werden, die dann während des Spiels von der algorithmischen Umsetzung verfolgt werden.

3.3 Kampf

Während der Kampfphase hat der aktive Spieler die Möglichkeit, mit seinen Kreaturen anzugreifen. Hierfür deklariert er eine Teilmenge seiner Kreaturen als Angreifer. Der Gegner darf dann mit seinen Kreaturen beliebige angreifende blocken. Nach dem Deklarieren der Blocker wird der Kampfschaden verrechnet. Stärkere Kreaturen zerstören dabei schwächere; ungeblockte Kreaturen ziehen dem Spieler Lebenspunkte ab. Die Angreifer-Menge wird durch einfache Reduktion ermittelt: Zunächst wird ein Angriff mit allen möglichen Angreifern durchgeführt und bewertet. Anschließend

werden die Blockzuordnungen einzeln bewertet und der Angreifer mit der schlechtesten Bewertung aus der Menge entfernt. Dieser Vorgang wird fortgesetzt, bis die Angreifermenge leer ist. Danach wird die beste der bewerteten Angreifermengen ausgewählt.

Für die Zuweisung einer Blockerkombination zu einer gegebenen Angreifermenge wird eine Alpha-Beta-Suche [RN03] über einen Strategieraum gestartet. Jedem Angreifer wird eine Block-Strategie zugewiesen, die eine sinnvolle Blockzuordnung generiert. Dieses Verfahren reduziert die Anzahl der zu berechnenden Zuweisungen im Vergleich zu einer Brute-Force-Suche [RN03] über alle möglichen Blockerzuordnungen erheblich. Die Blockzuordnungen werden einer vereinfachten Bewertung unterzogen; sie werden in der Reihenfolge ihrer Wertungen abgearbeitet, wodurch die Wahrscheinlichkeit eines frühen Alpha-Cutoff [RN03] erhöht wird. Die Schadensverteilung lässt sich als Variante des Rucksackproblems auffassen. Um dieses in annehmbarer Zeit zu lösen, wird der Algorithmus von Nemhauser und Ullmann verwendet [WE04].

4 Ausblick

Wir erhoffen uns, dass diese Lösung nach Beendigung des Projekts von vielen Spielern genutzt wird. Zudem ist einer Weiterentwicklung und Verbesserung der verwendeten Algorithmen über die reguläre Projektzeit hinaus, beispielsweise in Diplomarbeiten, denkbar.

Literaturverzeichnis

- [RN03] Russel, S.J.; Norvig, P: Artificial Intelligence - A Modern Approach; 2nd ed., Upper Saddle River, NJ 2003.
- [WE04] Wetzel, E: Random Knapsack in Expected Polynomial Time; Universität Paderborn, Seminar "Perlen der Theoretischen Informatik", Paderborn WS 2003/2004, <http://www.cs.uni-paderborn.de/cs/ag-madh/vorl/Perlen/EVGENIJA.pdf>; Abruf: 15.03.2007.
- [ZI07] Zingrillo, C.: Webseite von Constanzo Zingrillo; <http://magicmachine.tripod.com/>; Abruf: 15.03.2007.

Betrugserkennung bei Versicherungen mit Case Based Reasoning

Ursula Friede

Private FernFachhochschule Darmstadt

ursula.friede@gmail.com

Art der Arbeit: Diplomarbeit
Betreuer/in der Arbeit: Dipl.-Inform. (FH) Michael Best

Abstract: Herkömmliche Verfahren der maschinellen Erkennung von Versicherungsbetrug haben den Nachteil, dass sie nicht selbstlernend sind. Daher müssen die Regeln einer permanenten Wartung unterzogen werden. Auch weiterentwickelte Verfahren weisen noch gravierende Mängel auf. Sie entwickeln eine ungewollte Eigendynamik und benötigen eine lange Vorlaufzeit bis sie richtig einsatzfähig sind. In dieser Arbeit wird ein Ansatz erarbeitet, der Versicherungsbetrug mit Hilfe des Fallbasierten Schließens erkennt. Das Verfahren ist bisher einzigartig, da die Lösung sich am Erinnerungsprozess der menschlichen Denkweise eines Experten orientieren soll.

1 Einleitung

„Der deutschen Versicherungswirtschaft entsteht durch Versicherungsbetrug jährlich ein Schaden von rund vier Milliarden Euro“ [1]. Dies ist einer der Hauptgründe, weswegen jede Versicherungsgesellschaft ein großes Augenmerk auf die Abwehr von Versicherungsbetrug legt.

Neben der Zusammenarbeit mit der Kriminalpolizei und beauftragten Ermittlern kommt es auf die Kompetenz der speziell geschulten Betrugssachbearbeiter an, ob ein Versicherungsbetrug aufgedeckt wird oder nicht. Jährlich werden Millionen von Schäden von Service-Mitarbeitern reguliert. Diese verfügen i.d.R. nicht über besondere Kenntnisse im Betrugssektor. Dem speziell geschulten Betrugssachbearbeiter werden daher quantitativ und qualitativ nicht die richtigen Schadenfälle zur Bearbeitung vorgelegt. Dies wiederum mindert die Anzahl der echten Versicherungsbetrugsfälle, die entdeckt werden können.

Diese wissenschaftliche Arbeit stellt nicht den ersten Ansatz dar dieses Problem technisch zu lösen. Im nachfolgenden Kapitel werden Systeme, die es bereits auf dem deutschen Markt gibt, näher erläutert.

2 Prüfung von Alternativen

Im produktiven Einsatz bei deutschen Versicherern befinden sich verschiedene Systeme. U.a. ein Regelbasiertes System mit Fuzzy-Logik. Dieses System hat den Nachteil, dass es nicht selbstlernend ist. Durch die ständig sich verändernde Umwelt muss es einer ständigen Wartung und Anpassung unterzogen werden. Mit einer wachsenden Anzahl von Regeln wird das System zu komplex, ist dadurch sehr entwicklungs- und wartungsintensiv. Die Fehlersuche gestaltet sich immer diffiziler.

Weiterhin sind Neuronale Netze im Einsatz. Diese sind selbstlernend. Durch das Lernen und Vergessen entwickeln sie jedoch eine gewisse Eigendynamik, die ggf. nicht mehr kontrolliert werden kann. Sie sind erst nach einer längeren „Einarbeitungszeit“ effektiv einsatzfähig. Es bleibt weiterhin eine Blackbox im System, die nicht händelbar ist.

3 Grundlagen

Case Based Reasoning (CBR) bedeutet in der deutschen Sprache Fallbasiertes Schließen. Konkret auf das Thema Versicherungsbetrug angewendet bedeutet ein Fall nichts weiter als einen Versicherungsschadenfall. Dieser Schadenfall hat bestimmte Kriterien, nach denen er als Betrugsfall oder als regulierwürdiger Schadenfall eingeschätzt wird. Die Lösung des Falles ist dann also das Ergebnis: Betrugsverdacht ja/nein. Sofern man eine Mehrzahl repräsentativer Fälle hat, kann man bei einem neuen Fall, aufgrund der bisherigen Erfahrungen, ein Ergebnis daraus geschlossen werden.

Die Kriterien werden in Kategorien eingeteilt. Es wird die Person des Anspruchstellers und des Versicherungsnehmers geprüft. Hier spielen u.a. Alter, Geschlecht, Nationalität und Beruf eine Rolle. Auch ist die Relation zwischen den Beteiligten von Bedeutung. In welcher Beziehung stehen Anspruchsteller und Fahrer und wie groß ist die Altersdifferenz zwischen den Fahrern. Ein weiteres Prüfkriterium sind die beteiligten Fahrzeuge. Neben dem Hersteller und Fahrzeugtyp, sind u.a. Schadenhöhe, Alter des Fahrzeuges, Vorschäden und TÜV von Bedeutung. Abschließend spielen Angaben zum Unfallhergang, -ort und -zeit eine Rolle. Zur Veranschaulichung, wie diese verschiedenen Kriterien zusammenspielen, dazu folgendes Beispiel aus der Realität:

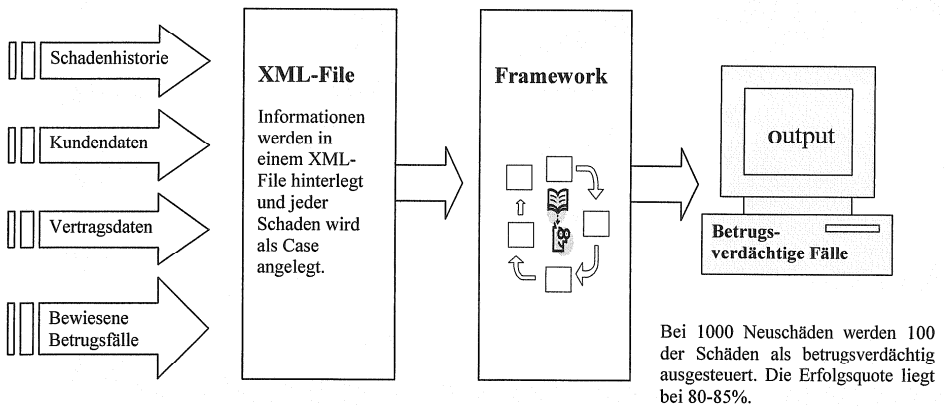
„Die herbeigerufenen Polizisten hegten keinen Verdacht, alles sah nach einem gewöhnlichen Autounfall aus: Der Fahrer eines Mietwagens war in Wuppertal gegen ein Uhr morgens mit einem entgegenkommenden PKW kollidiert, dessen Scheinwerferlicht ihn geblendet hatte (...) auf den zweiten Blick zeigten sich Ungereimtheiten: Der Unfall ereignete sich zur Nachtzeit an einem abgelegenen Ort. Zudem wohnte der Fahrer des Mietwagens zwar in der Stadt Mettmann, aufgewachsen war er jedoch in Wuppertal – im gleichen Haus, in dem das Unfallopfer wohnte. Bei einer Wohnungsdurchsuchung fand sich ein Foto einer Fußballmannschaft, auf dem die beiden Männer, die ausgesagt hatten, sich nicht zu kennen, nebeneinander standen. Unter der erdrückenden Beweislast räumten sie die Manipulation des Schadenfalles ein.“ [2]

4 Lösungsansatz und Implementierung

Der Lösungsansatz für die genannte Problematik ist durch eine repräsentative Anzahl von tatsächlichen Betrugsfällen eine Falldatenbasis zu schaffen. Diese Fälle wurden bereits von Spezialisten bearbeitet und abschließend hat sich der Betrug bestätigt. Jeder Betrugsfall ist ein „case“ und wird mit seinen entsprechenden Bedingungen in ein XML-File („PTcasebase“) eingetragen. Dieses XML-File ist der Input für ein CBR-Framework (Abbildung 1).

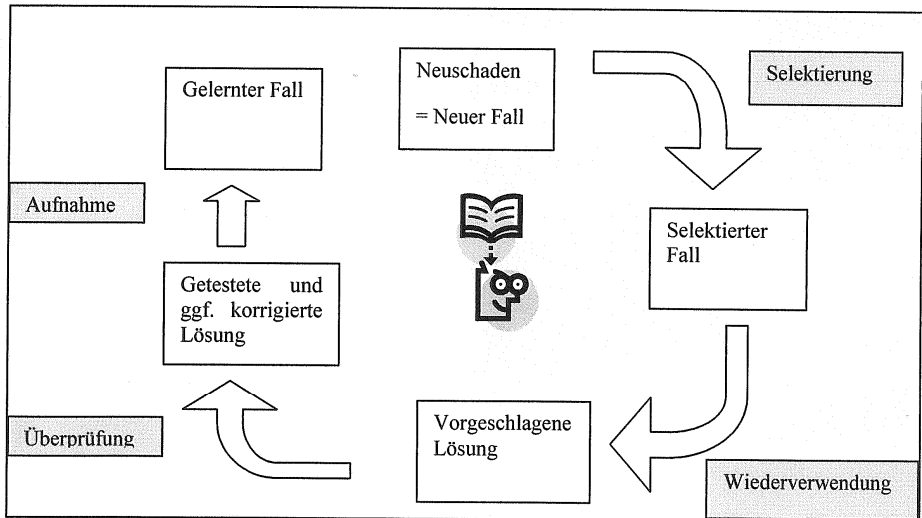
Mitten unter vielen kommerziellen Tools für Fallbasiertes Schließen existieren einige wenige Open Source Frameworks, wie z.B. jColibri. Zur Realisierung des Prototyps wurde jColibri verwendet. JColibri ist ein objektorientiertes Framework in Java. Es erfüllt alle Bedingungen, die zur Lösung der Aufgabenstellung erforderlich sind.

Abbildung 1



Innerhalb des Frameworks gestaltet sich der CBR-Zyklus (Abbildung 2) wie folgt: Der Input für den Zyklus ist der Neuschaden, der mit den vorhandenen Daten vom Sachbearbeiter im System erfasst wird. Für das Framework ist dies der neue Fall. Als Erstes erfolgt die *Selektierung* des ähnlichen Falls oder der ähnlichen Fälle. Es wird also geprüft, ob es in der Konstellation schon einen Fall in der Datenbasis gibt oder zumindest einen ähnlichen Fall. Von hier startet die *Phase der Wiederverwendung*. Der ähnliche oder die ähnlichen Fälle werden mit dem Neuschaden abgeglichen, und es wird eine Lösung für den aktuellen Schaden generiert. Diese vorgeschlagene Lösung muss allerdings noch ausgetestet werden. Dies ist die *Phase der Überprüfung*. Ein Betrugsexperte nimmt die Evaluation vor. Sofern sich der Betrugsverdacht als korrekt erwiesen hat besteht anschließend die Möglichkeit mit der *Aufnahme* den neuen Fall in die Fallbasis aufzunehmen und zu integrieren.

Abbildung 2



5 Anwendung und Nutzungsmöglichkeit

Eine Anwendung in der Praxis ist nur sinnvoll, wenn das Versicherungsunternehmen über ein Schadensystem verfügt, das auch die erforderlichen Kriterien erfasst und speichert. Weiterhin muss eine Vielzahl repräsentativer Fälle vorliegen, dieses ist jedoch in jedem großen Versicherungsunternehmen gegeben. Das System ist dann in der Lage nur diese Fälle zu selektieren, die den Betrugsverdacht haben. Dies hat den Vorteil, dass Servicemitarbeiter, die nicht eine spezielle Ausbildung in der Fachrichtung Betrug haben, die Schadenfälle, die nicht betrugsverdächtig sind, kundenfreundlich und zügig abarbeiten können und die betrugsverdächtigen Fälle direkt dem speziellen Betrugssachbearbeiter vorlegen können. Dieser wiederum bekommt nur eine stark eingegrenzte Anzahl an Schäden vorgelegt, die er intensiv bearbeiten kann.

6 Zusammenfassung

Im Rahmen dieser Arbeit wurde eine Lösung auf Basis des Case Based Reasoning zur automatischen Erkennung von Versicherungsbetrug entwickelt. Es wurde gezeigt, wie der Einsatz eines Tools dazu beitragen kann, Betrugsfälle gezielter zu erkennen und dadurch die Erfolgsquote der Betrüger zu verringern.

Literaturverzeichnis

- [1] Gesamtverband der Deutschen Versicherungswirtschaft e.V.: *Versicherungsbetrug: Banden auf dem Vormarsch*. Dresden: Presseforum, 2003, S.1.
- [2] Baumann, Marc: *Der Krieg gegen die Autobumser*. In: Ausgabe Nr. 47 - März/April.
http://www.gdv.de/Publikationen/Zeitschrift_Positionen_Linkliste/Positionen_Nr._47_inhaltsseite665.html. (Zugriff: 26.01.2007, 11.30MEZ)

Automatische Annotation von Bilddaten auf Grundlage eines verlässlichen visuellen Wörterbuchs

Christian Hentschel
Otto-von-Guericke Universität Magdeburg
chentsch@student.uni-magdeburg.de

Art der Arbeit: Masterarbeit
Betreuer der Arbeit: Jun.-Prof. Dr. Andreas Nürnberger,
Dr. habil. Marcin Detyniecki

Der vermehrte Einsatz digitaler Technologien zur Produktion, Verarbeitung und Verbreitung bildlicher Information hat zu einem enormen Anstieg der weltweit verfügbaren Digitalbilder geführt. Im September 2006 hat der Bundesverband Informationswirtschaft, Telekommunikation und neue Medien e.V. (BITKOM) die Anzahl der monatlich erzeugten Digitalfotos in Deutschland auf rund zwei Milliarden geschätzt¹. Neben diesen meist privaten Nutzern hat sich eine Medienindustrie etabliert, die mehr denn je zuvor auf Bilder angewiesen ist, um die täglichen Nachrichten in TV, Internet und Zeitungen zu illustrieren.

Die bloße Anzahl bildlicher Daten erfordert computergestützte Methoden, die ein gezieltes Durchsuchen großer Bildsammlungen nach bestimmten visuellen Informationen erlauben (*Image Retrieval*). In den letzten Jahren haben sich zwei Verfahren etabliert, deren primärer Unterschied in der Formulierung der Suchanfrage besteht. *Content-based Image Retrieval* versucht durch den syntaktischen Vergleich eines Anfragebildes mit allen Bildern der Datenbank semantisch ähnliche Bilder zu finden. Die Notwendigkeit eines Anfragebildes welches möglichst ähnlich zum gesuchten Bild ist, disqualifiziert dieses Verfahren jedoch typischerweise in den oben genannten Anwendungsfälle. *Annotation-based Image Retrieval* durchsucht eine Bilddatenbank nach Bildern anhand einer textuellen Beschreibung (Annotation) ihres Inhalts. Hierfür ist es notwendig, dass der dargestellte Bildinhalt aller Bilder in der Sammlung in eine textuelle Repräsentation überführt wird. Eine manuelle Annotation ist ein zeitaufwändiger und ermüdender Prozess. *Automatic Image Annotation* versucht daher, auf Grundlage eines kleinen Trainingsdatensatzes manuell annotierter Bilder automatisch auf die Beschreibung aller Bilder in einer Kollektion zu schließen.

Automatic Image Annotation beschreibt die überwachte Klassifikation von Bilddaten. Jede Bildklasse enthält Bilder, die wenigstens eine textuelle Annotation gemein haben. Typischerweise sind Annotationen einfache Schlüsselwörter wie "Baum" oder "Himmel". Da ein Bild häufig mit mehr als einer Annotation versehen werden kann, gehören die meisten Bilder mehreren Klassen an. Ein Klassifizierer versucht, ein Bild einer oder mehreren

¹BITKOM, Nachlässiger Umgang mit digitalen Fotos, http://bitkom.de/de/presse/41562_41557.aspx

Bildklassen zuzuordnen. Dieser Klassifizierer wird mit Hilfe eines Datensatzes manuell annotierter Bilder trainiert. Ein effizienter Trainingsdatensatz enthält eine möglichst eindeutige Zuordnung zwischen textueller Annotation und den beschriebenen semantischen Einheiten im Bild. Diese Zuordnung wird durch ein visuelles Wörterbuch (*Visual Dictionary*) repräsentiert. Die Einträge oder visuellen Wörter im Wörterbuch werden vom Klassifizierer mit unbekannten Bildern verglichen. Findet der Klassifizierer ein visuelles Wort in einem Bild wieder, wird dieses Bild entsprechend annotiert.

Im Rahmen der vorgestellten Masterarbeit wurde SAFIRE – ein Framework zur automatischen Annotation von Bilddaten unter Verwendung eines korrelationsfreien visuellen Wörterbuchs – implementiert. SAFIRE unterstützt die Suche nach Bilddaten auf Basis textueller Annotationen.

Zur Generierung des visuellen Wörterbuchs in SAFIRE wurde ein *lokal* annotierter Bilddatensatz gewählt. Bilddatensätze wie die häufig verwendete 'Corel Stock Photo Collection' sind global annotiert – jedem Bild sind global mehrere Stichwörter zugeordnet. Eine automatische Zuordnung zwischen globalen Annotationen und lokalen semantischen Einheiten ist hier oft nicht möglich, da i.d.R. verschiedene Merkmale korreliert sind. So zeigen zum Beispiel Bilder die einen Baum darstellen typischerweise auch ein Stück Himmel. Mit Hilfe der so generierten visuellen Wörterbücher ist es oft nicht möglich, korrelierte Objekte voneinander zu unterscheiden. In SAFIRE wurde ein Interface entworfen, welches die manuelle Segmentierung von Trainingsbildern in ihre semantischen Einheiten ermöglicht. Diese Einheiten werden anschließend vom Nutzer mit beschreibenden Stichworten versehen. So kann jedem Pixel eines Bildes eindeutig eine Annotation zugeordnet werden.

Eine manuelle Bildsegmentierung ist ein zeitaufwendiger Prozess. Um die automatische Annotation unabhängig von der Segmentierung zu gestalten, werden in SAFIRE die erzeugten Bildsegmente der Trainingsdaten in quadratische Regionen unterteilt. Diese Regionen bilden zusammen mit den Annotationen der sie umgebenden Bildsegmente die visuellen Wörter. Ein unbekanntes Bild wird analog in ein statisches Gitter quadratischer Regionen unterteilt. Der Klassifizierer vergleicht die Quadrate des unbekannten Bildes mit denen des visuellen Wörterbuchs. Bei Übereinstimmung erfolgt automatisch die Annotation des neuen Bildes.

Für den Vergleich eines visuellen Wortes mit einem unbekannten Bild ist es notwendig, die visuellen Bildinformationen numerisch zu repräsentieren. Ein häufig verwendetes Modell ist das Vektorraummodell bei dem Farb- und Texturmerkmale einzelner Bildregionen als Vektoren beschrieben werden. SAFIRE greift hierbei auf den MPEG-7 Standard zurück. Das *MPEG-7 Multimedia Description Interface*² bietet u.a. standardisierte Farb- und Textur-Deskriptoren mit deren Hilfe entsprechende Vektoren generiert werden können. Die erzeugten Vektoren beschreiben je eine quadratische Region eines Bildes. In SAFIRE wird MPEG-7 auch zur standardisierten Speicherung der Segmentierungs- und Annotationsdaten verwendet. Somit ist eine einfache Erweiterung des visuellen Wörterbuchs und der annotierten Bildkollektion möglich.

Idealerweise enthält ein visuelles Wörterbuch für eine spezifische semantische Einheit nur die visuellen Wörter, welche als repräsentativ für die beschriebene Einheit angesehen wer-

²The Moving Picture Experts Group (MPEG), <http://www.chiariglione.org/mpeg/>

den. In SAFIRE wird die Selektion repräsentativer visueller Wörter durch Clustering der Farb- und Texturvektoren erreicht. Die ermittelten Cluster-Zentren bilden die visuellen Wörter. Jeder Repräsentant erfasst dabei die typische visuelle Streuung innerhalb eines Clusters und macht das Wörterbuch somit robust gegen leichte visuelle Variationen einer semantischen Einheit. In SAFIRE wurden ein k-Means und ein Fuzzy-C-Means Clustering Ansatz implementiert deren Ergebnisse im Hinblick auf die erzielte Klassifikationsgenauigkeit verglichen wurden.

Bei der Klassifikation unbekannter Bilder verwendet SAFIRE einen k-Nearest-Neighbor basierten Ansatz. Für jede quadratische Region des zu klassifizierenden Bildes werden Farb- und Texturvektoren berechnet. Diese werden mit den Vektoren des visuellen Wörterbuchs verglichen wobei jene k Wörter ermittelt werden, deren Distanz im Vektorraum zu dem zu klassifizierenden Quadrat am geringsten ist. Basierend auf den gefundenen visuellen Wörtern berechnet der Klassifizierer einen Zuverlässigkeitswert der eine Aussage über die Qualität der geschätzten Annotation erlaubt. Eine automatische Annotation erfolgt nur, wenn der berechnete Zuverlässigkeitswert hoch ist, die geschätzte Annotation also mit hoher Wahrscheinlichkeit korrekt ist.

SAFIRE verfügt über eine Komponente zur einfachen Suche von Bildern unter Verwendung der automatisch generierten Annotationen. Verschiedene Tests wurden durchgeführt, die dem Verfahren eine hohe Klassifikationsgüte für räumlich nicht begrenzte semantische Einheiten (z.B. "Himmel", "Gras") bescheinigen. Abbildung zeigt das Ergebnis einer Suchanfrage für das Stichwort "flower". Erkennbar sind die schwarzen Quadrate, welche Regionen repräsentieren, die vom Klassifizierer annotiert werden konnten.

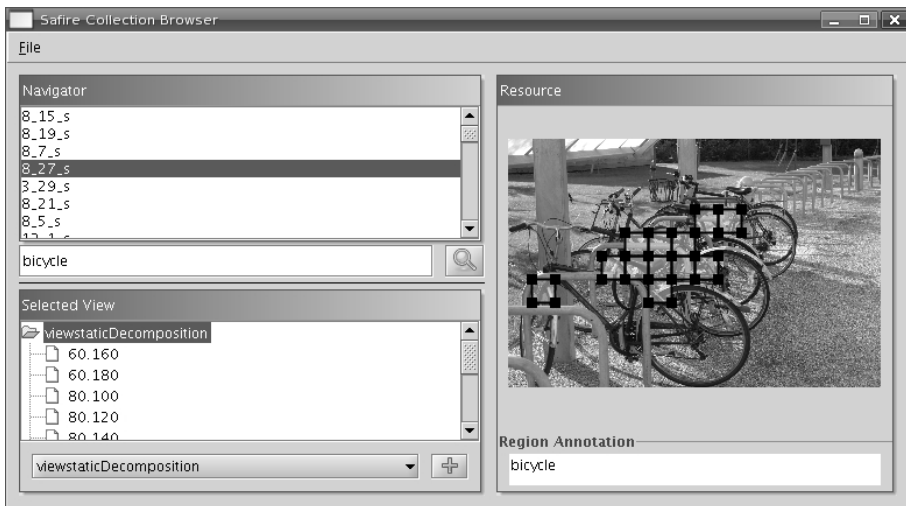


Abbildung 1: Ergebnis einer Suche in SAFIRE nach Bilder mit der Annotation "bicycle". Die schwarzen Quadrate markieren die vom Klassifizierer zuverlässig annotierten Bildregionen.

Literatur

- [Bim99] Alberto Del Bimbo. *Visual Information Retrieval*. Morgan Kaufmann Publishers, Inc., San Francisco, CA, 1999.
- [BYRN99] Ricardo Baeza-Yates und Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley, 1999.
- [CDF⁺04] Gabriella Csurka, Christopher R. Dance, Lixin Fan, Jutta Willamowski und Cédric Bray. Visual Categorization with Bags of Keypoints. In *Proceedings of the 8th ECCV, Prague*, 2004.
- [FSN⁺95] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Qian Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele und P. Yanker. Query by image and video content: the QBIC system. *Computer*, 28(9):23–32, Sept. 1995.
- [GS04] Th. Gevers und A.W.M. Smeulders. *Emerging Topics in Computer Vision*, Kapitel Image Search Engines: An Overview. Prentice Hall, 2004.
- [HNSS06] C. Hentschel, A. Nürnberger, I. Schmitt und S. Stober. SAFIRE: Towards Standardized Semantic Rich Image Annotation. In *Proceedings of the 4th International Workshop on Adaptive Multimedia Retrieval (AMR)*, 2006.
- [Ino04] Masashi Inoue. On the need for annotation-based image retrieval. In *Workshop on Information Retrieval in Context (IRiX)*, Seiten 44–46, 2004.
- [JMF99] A. K. Jain, M. N. Murty und P. J. Flynn. Data clustering: a review. *ACM Comput. Surv.*, 31(3):264–323, 1999.
- [LFM04] V. Lavrenko, S.L. Feng und R. Manmatha. Statistical models for automatic video annotation and retrieval. In *Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP '04). IEEE International Conference on*, Jgg. 3, Seiten iii–1044–7vol.3, 17-21 May 2004.
- [LZF03] F. Long, H. Zhang und D. Feng. *Multimedia Information Retrieval and Management : Technological Fundamentals and Applications*, Kapitel Fundamentals of Content-based Image Retrieval. Springer, Berlin, 1. Auflage, April 2003.
- [Mit97] Tom Mitchell. *Machine Learning*. McGraw Hill, 1997.
- [MSS02] B.S. Manjunath, Philippe Salmbier und Thomas Sikora, Hrsg. *Introduction to MPEG-7: Multimedia Content Description Interface*. John Wiley & Sons, Ltd., 2002.
- [Sch05] Ingo Schmitt. *Ähnlichkeitssuche in Multimedia-Datenbanken. Retrieval, Suchalgorithmen und Anfragebehandlung*. Oldenbourg, 2005.
- [SJ98] Simone Santini und Ramesh Jain. Beyond Query by Example. In *ACM Multimedia*, Seiten 345–350, 1998.
- [SWS⁺00] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta und R. Jain. Content-based image retrieval at the end of the early years. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(12):1349–1380, Dec. 2000.
- [VFJZ01] A. Vailaya, M.A.T. Figueiredo, A.K. Jain und Hong-Jiang Zhang. Image classification for content-based indexing. *Image Processing, IEEE Transactions on*, 10(1):117–130, Jan. 2001.

Continuous 3D Environment Sensing for Autonomous Robot Navigation and Mapping

Dirk Holz^{*,1,2}, Christopher Lörken^{2,3}

^{*}University of Applied Sciences Bonn-Rhein-Sieg
Grantham Allee 20, D-53754 Sankt Augustin, Germany
dirk.holz@smail.inf.fh-bonn-rhein-sieg.de

Type: Diploma Thesis (University of Applied Sciences Cologne)

Supervisors: Dr. Erwin Holland-Moritz¹, Dr. Hartmut Surmann²

GI-Subject: 1.4/4.3 “Robot Systems”

Abstract: Presented here is a novel approach for continuously sensing dynamic indoor environments in 3D. Based on this procedure virtual 2D maps are introduced that allow for computationally efficient navigation algorithms. Additionally, a methodology is proposed to interpret the gathered information in a way applicable for prevailing 2D and 3D mapping algorithms.

1 Introduction and Motivation

Current applications for autonomous mobile robots, like for instance in the field of service robotics, face, amongst others, the challenging task of reliably navigating in dynamic indoor environments. In order to achieve this capability it is decisive to employ robust modeling techniques that maintain internal world representations [Thr02]. Nature and complexity of these representations highly depend on the robot’s task and application space. While the creation of consistent world models requires highly accurate spatial information the focus for navigational purposes mainly lies on fast reaction times.

One of the currently most sophisticated approaches that aims at automatic model construction is applied by Surmann et. al. [SNH03]. They have extended a standard SICK LMS 200 2D laser range-finder with a horizontal opening angle of 180° (rotating mirror device) by attaching it to a rotatable axis. This allows to pitch the scanner vertically over an angular range of 120°. Taking one horizontal 2D scan at each vertical position yields a 3D model (or 3D scan) of the current scene. To construct a precise model of the environment Surmann et. al. collect such 3D scans at various locations and integrate or *register* them into a common coordinate system by applying an elaborated 6D Simultaneous Localization and Mapping (SLAM) algorithm [SNLH05]. Navigation between the scan locations is accomplished by means of 2D pose tracking and 2D obstacle avoidance, i. e. holding the scanner in a static horizontal position. The emerging course of actions results in *stop-scan-move* cycles and is thus interrupted each time a 3D scan is being acquired. Nevertheless,

¹University of Applied Sciences Cologne, Campus Gummersbach, Am Sandberg 1, D-51643 Gummersbach

²Fraunhofer Institute for Autonomous Intelligent Systems, Schloss Birlinghoven, D-53754 Sankt Augustin

³University of Osnabrück, Albrechtstraße 28, 49069 Osnabrück

the major drawback of this procedure lies in the fact that obstacles are only perceived in the two-dimensional scan plane while moving from one location to another. While this strategy may be appropriate for the task of automatic model construction it is less suitable when facing the aforementioned dynamic everyday environments.

In the following means for overcoming this limitation are being proposed by introducing a novel procedure that allows for continuously scanning the environment in 3D – even while the robot is moving. The gathered data is used both for 3D obstacle avoidance as well as for 2D and 3D mapping purposes.

2 Continuous 3D Environment Sensing

To continuously perceive spatial information about the robot's environment the laser-scanner is pitched in a nodding-like fashion. Hereby, an *area of interest* (AOI) is defined extending the idea of virtual roadways [LNHS05] to the third dimension, i.e. with respect to the robot's boundaries and thus possible areas of collision. The height of the AOI as well as the pitch rate depend on the robot's current speed and are adjusted by applying a closed loop controller with proportional and integral gain (*PI-controller*).

Furthermore *virtual 2D maps* are introduced as a computationally efficient representation of three-dimensional data – inspired by the approach introduced in [WACW04] by Wulf et.al. who, nevertheless, did not update their model continuously while moving. Hereby, the perceived data is being projected into a two-dimensional plane allowing for two modes of operation:

Virtual obstacle map - The minimal distance from the robot to the projected points is stored for each scan angle resulting in a representation that stores the information of the closest obstacles in each direction. In contrast to the previous approaches obstacles are perceived regardless of their height as long as they intersect with the virtual roadway.

Virtual structure map - Taking the maximal distances instead of the minimal results in obtaining a map that reflects the environmental boundaries like e.g. walls. Hereby small or dynamic items are implicitly filtered out.

The maps are egocentric and updated according to the robot's movement. Since the robot is meant to operate in indoor environments the floor can be assumed to be flat. Thus perceived floor points can easily be filtered out. This kind of egocentric representation is organized in a format equivalent to that of an ordinary 2D laser scan (distances ordered with the discretized angles used as indices). This allows to apply the same set of feature extraction mechanisms and behavioral navigation procedures for both representations. To deal with environment dynamics and imprecise pose estimates, points stored in a virtual map are removed after a certain transformation count is reached (e.g. after 500 transformations per point which corresponds to five complete nodding movements on average) to avoid erroneous information.

3 Navigation by means of Virtual 2D Obstacle Maps

Virtual 2D obstacle maps bare all the information necessary for performing reactive behavior-based robot control [Bro86, Ark98]. Based on this representation the robot can

determine the distance and direction to nearby obstacles and react appropriately by adjusting its driving direction and speed [Lör06]. Figure 1 depicts (left) the resulting robot trajectory with the laserscanner in a fixed horizontal position and (right) the resulting trajectory with a continuously pitching scanner. In the first case the robot was not able to perceive the obstacles while in the second they were integrated in the virtual obstacle map and thus the robot was able to avoid them successfully.

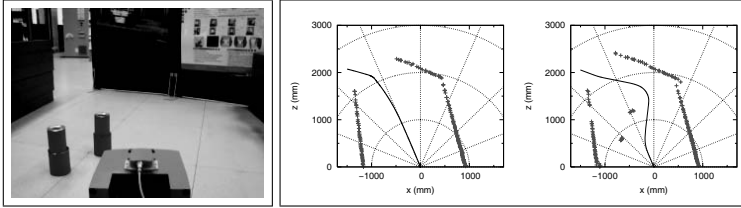


Figure 1: Obstacle avoidance in an exemplary scenario.

4 Construction of 2D and 3D Environment Maps

To construct a metric environment model based on the continuously acquired data the gathered information has to be merged into a common coordinate system. Due to the normally imprecise estimation of the robot's pose this registration is usually achieved by applying matching algorithms like the *Iterative Closest Point* (ICP) algorithm [BM92] or one of its derivatives.

Here the goal is to make the continuously acquired spatial information applicable for the usage of existing 2D and 3D matching algorithms. Regarding the case of 2D map construction the introduced virtual 2D structure maps are especially adequate for the matching process as dynamic and small obstacles are implicitly filtered out. However, to build a 3D map the continuously gathered information has to be integrated into uniform 3D point clouds as they are normally used for registration e.g. by Surmann et. al. [SNH03]. A point cloud representing one 3D laserscan is generated by transforming the continuously gathered consecutive 2D laserscans of one nodding motion by the meanwhile accumulated robot's pose shift and the scanner's pitch angle. These point clouds can afterwards be matched against each other to obtain a 3D model as shown in Figure 2.



Figure 2: Two views on a generated 3D model being constructed by matching the generated point clouds. The height of the built model corresponds to the height of the robot and thus to that of the area of interest.

By applying the presented procedures an autonomous mobile robot was able to safely navigate in dynamic indoor environments and to construct two-dimensional and three-dimensional models of the thereby visited environmental structures.

For a more detailed description of the presented approach refer to [Hol06].

References

- [Ark98] Ronald C. Arkin. *Behavior-Based Robotics*. Intelligent Robots and Autonomous Agents. MIT-Press, Cambridge, MA, USA, 1998.
- [BM92] Paul J. Besl and Neil D. McKay. A method for Registration of 3-D Shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239 – 256, February 1992.
- [Bro86] Rodney A. Brooks. A Robust Layered Control System for a Mobile Robot. *IEEE Journal of Robotics and Automation*, 2(1):14–23, March 1986.
- [Hol06] Dirk Holz. Kontinuierliche Umgebungskartographie mittels 3D-Laserscanner auf autonomen mobilen Robotern. Diploma thesis, University of Applied Sciences Cologne, Gummersbach, Germany, August 2006. Work conducted at the Fraunhofer Institute for Autonomous Intelligent Systems (FhG-AIS, St. Augustin, Germany). <http://www2.inf.fh-bonn-rhein-sieg.de/~dholz2s/holz-da.pdf>.
- [LNHS05] Kai Lingemann, Andreas Nüchter, Joachim Hertzberg, and Hartmut Surmann. About the Control of High Speed Mobile Indoor Robots. In *Proceedings of the Second European Conference in Mobile Robotics*, pages 218 – 223, Ancona, Italy, 2005.
- [Lör06] Christopher Lörken. Introducing Affordances into Robot Task Execution. Master’s thesis, University of Osnabrück, Osnabrück, Germany, November 2006. Work conducted at the Fraunhofer Institute for Intelligent Analysis and Information Systems (FhG-IAIS, St. Augustin, Germany).
- [SNH03] Hartmut Surmann, Andreas Nüchter, and Joachim Hertzberg. An autonomous mobile robot with a 3D laser range finder for 3D exploration and digitalization of indoor environments. *Journal Robotics and Autonomous Systems*, 45(3-4):181–198, 2003.
- [SNLH05] Hartmut Surmann, Andreas Nüchter, Kai Lingemann, and Joachim Hertzberg. Simultaneous Mapping and Localization of Rescue Environments. *it - Information Technology*, October 2005:282 – 291, 2005.
- [Thr02] Sebastian Thrun. Robotic Mapping: A Survey. In Gerhard Lakemeyer and Bernard Nebel, editors, *Exploring Artificial Intelligence in the New Millenium*. Morgan Kaufmann, 2002. ISBN 1-558608-11-7.
- [WACW04] Oliver Wulf, Kai O. Array, Henrik I. Christensen, and Bernardo Wagner. 2D Mapping of Cluttered Indoor Environments by Means of 3D Perception. In *IEEE/RAS International Conference on Robotics and Automation (ICRA)*, 2004.

Erzeugung von vorlesbarem Text aus L^AT_EX-Dokumenten

Marius Lauder
marius.lauder@arcor.de

Institut für Informatik, Fachbereich Physik, Mathematik & Informatik
Johannes Gutenberg-Universität Mainz
Betreuer: Prof. Dr. Herbert Göttler
Art der Arbeit: Bachelorarbeit

Das Problem

Bei Mathematikern ist L^AT_EX überaus beliebt zum Schreiben ihrer Texte, weil es insbesondere das ästhetische Layout von Formeln in einer Weise realisiert, wie es ein menschlicher Setzer kaum besser machen könnte. Allerdings müssen die einzelnen Bestandteile der Formeln durch eine L^AT_EX-eigene Auszeichnungssprache gekennzeichnet werden, die viele Sonderzeichen enthält.

Ein blinder L^AT_EX-Nutzer, der sich seinen Text von einem für Sehbehinderte entwickelten Reader-Programm vorlesen lassen will, kann dabei schnell die Übersicht verlieren, denn dieser Reader spricht auch Sonderzeichen wie das in L^AT_EX-Texten häufig vorkommende Backslash-Zeichen oder die geschweiften Klammern "expressis verbis" aus.

Das Problem stellt sich real am Fachbereich 08 der Johannes Gutenberg-Universität in Mainz, an dem ein erblindeter und L^AT_EX liebender Professor für Mathematik lehrt und forscht.

Im Rahmen einer Bachelorarbeit sollte deshalb ein Textkonverter erstellt werden, der aus einem L^AT_EX-Dokument einen Text erzeugt, wie ihn ein Mathematiker sinngemäß vorlesen würde.

Betrachtet man beispielsweise folgenden mathematischen Sachverhalt (Dieser Text ist dem *Taschenbuch der Mathematik* S. 409 [1] entnommen):

6.2.1.5 Partielles Differential

Von einer Funktion von mehreren Veränderlichen $u = f(x, y, \dots)$ kann das partielle Differential nach einer dieser Veränderlichen, z.B. nach x gebildet

werden, was durch die Gleichung

$$d_x u = d_x f = \frac{\partial u}{\partial x} dx$$

definiert ist.

– Auszug Ende –

Dieser Text könnte mit folgendem L^AT_EX-Quelltext [2] erstellt worden sein:

```
\section{Partielles Differential}
Von einer Funktion von mehreren Veränderlichen
$u = f(x,y,...)$ kann das partielle Differential
nach einer dieser Veränderlichen, z.B. nach $x$
gebildet werden, was durch die Gleichung
\begin{equation}
d_{\{x\}}u = d_{\{x\}}f = \frac{\partial u}{\partial x}dx
\end{equation}
definiert ist.
```

Der nachstehende Text könnte so einem blinden Mathematiker vorgelesen werden, der auch Wert auf eine ungefähre Vorstellung legt, wie der Text gesetzt ist:

Überschrift 6.2.1.5 Partielles Differential Überschrift Ende
Beginn Absatz Von einer Funktion von mehreren Veränderlichen
u ist gleich f runde öffnende Klammer x Komma y Komma Punkt
Punkt Punkt runde schließende Klammer kann das partielle
Differential nach einer dieser Veränderlichen, z.B. nach x
gebildet werden, was durch die Gleichung Absatz Ende In einer
Zeile In einer Spalte d mit Index x Index Ende u ist gleich d
mit Index x Index Ende f ist gleich Bruch mit Zähler partielles
D u dividiert durch partielles D x Bruch Ende d x Beginn Absatz
definiert ist. Absatz Ende

Die Lösungsidee

T_EX und L^AT_EX sind sehr flexible und mächtige Werkzeuge, die sich seit Jahrzehnten weiter entwickelt haben. So können Nutzer z.B. Makros definieren, auf die aber bei einer Konvertierung reagiert werden muss. Recherchen im Internet zeigten, dass es bereits verschiedenste Konverter für L^AT_EX gibt, die das Problem des Erstellens von Präprozessoren z.B. für Makroersetzungen gelöst haben. Andere Konverter hingegen erzeugen aus dem L^AT_EX-Quelltext beispielsweise ein HTML-Dokument, das Formeln in Form von eingebundenen Graphiken realisiert. Damit ist aber eine Weiterverarbeitung praktisch nicht mehr möglich.

Letztlich wurde der Konverter TeX4ht [3] gefunden, der L^AT_EX-Dokumente nach XHTML umwandeln kann und damit schon das Problem der Makros löst. In einem zweiten Schritt wird "nur" noch ein Umsetzer von XHTML-Quelltext in natürlichsprachlichen Text benötigt.

Wichtig hierbei ist die Verwendung des W3C-Standards *MathML* [4], der die Einbindung komplexer mathematischer Formeln in XHTML und ihre Darstellung durch einen Web-Browser gestattet.

Das Geforderte sollte also mit einem zweistufigen Verfahren geleistet werden:

- Zuerst wird unter Verwendung eines \LaTeX -nach-XHTML-Konverters der gegebene Quelltext umgewandelt.
- Im zweiten Schritt setzt der in der Arbeit entworfene XHTML2NatLang-Compiler an, der nun den XHTML-Quelltext in die gewünschte Form bringt.

Die Realisierung

Das nachstehende Schaubild zeigt das Zusammenspiel der Komponenten. In diesem Diagramm stellen die abgerundeten Rechtecke Dokumente und die übrigen Rechtecke ausführbare Programme dar. Ausgangspunkt für die Realisierung ist eine DTD ("DTD" steht für "document type definition") für XHTML, welche von zwei Perl-Programmen unter Verwendung von perlSGML [5] analysiert wird und dann sowohl die Lex-Spezifikation als auch die Yacc-Spezifikation generieren. Diese Spezifikationen als Eingabe für Lex bzw. Yacc liefern den C-Quelltext, welcher (compiliert) den XHTML2NatLang-Compiler darstellt. Der XHTML2NatLang-Compiler wiederum hat XHTML-Dokumente sowie Ersetzungstexte als Eingabe und liefert als Ergebnis einen vorlesbaren Text in natürlicher Sprache.

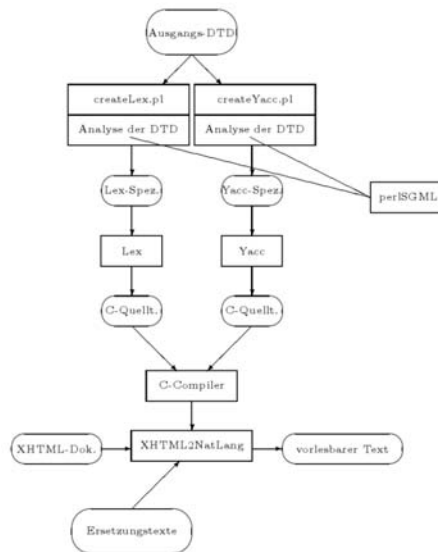


Abbildung 1: Die Umsetzung im Überblick

Da der XHTML2NatLang-Compiler mit den Unix-Werkzeugen *Lex* und *Yacc* [6] (bzw. *Bison*) erstellt wurde, mussten die entsprechenden Grammatiken generiert werden. Schnell wurde aber klar, dass dies von Hand nicht möglich ist, weil man es mit ca. 1400 Wortsymbolen und ca. 900 Produktionsregeln zu tun hat. So mussten erst noch die Programme *createLex.pl* und *createYacc.pl* geschrieben werden, um aus der XHTML-DTD mit dem DTD-Parser perlSGML die adäquaten Lex- und Yacc-Spezifikationen zu erzeugen.

Zusätzlich wurden Funktionen implementiert, die über die Verwendung von Konfigurationsdateien eine spätere Anpassung und Wartung erleichtern sollen.

In diesen Konfigurationsdateien werden für jedes in einem XHTML-Dokument vorkommende Textauszeichnungselement Ersetzungstexte gespeichert. Somit ist es z.B. denkbar eine Umwandlung nach verschiedenen Konfigurationen ablaufen zu lassen. Ein einfacher Bruch $\frac{a}{b}$ könnte so beispielsweise durch

- a durch b
- a dividiert durch b
- Zähler a dividiert durch Nenner b
- Bruch mit Zähler a dividiert durch Nenner b Ende des Bruchs
- ...

ausgedrückt werden. Auch könnte der oben angeführte erzeugte Beispieltext mit diesen Mitteln anders lauten.

Literatur

- [1] Bronstein, Semendjajew et al.: *Taschenbuch der Mathematik*, Verlag Harri Deutsch, 5. Auflage, 2001
- [2] Elke & Michael Niedermair: *LaTeX Praxisbuch*, Franzis Verlag, 2004
- [3] TeX4ht ist zu finden unter <http://www.cse.ohio-state.edu/~gurari/TeX4ht>
- [4] MathML-Standard des W3C unter <http://www.w3.org/Math>
- [5] perlSGML ist zu finden unter <http://savannah.nongnu.org/projects/perlsgml>
- [6] Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman: *Compilers: Principles, Techniques, and Tools*, Addison Wesley, 1986

EPOS: Experimental prototype for ontology based search in a multi agent system

Bernd Müller
bernd@kirx.de

Fachhochschule Aachen
Supervisor Prof. Dr. rer. nat. Heinrich Faßbender

Abstract: At the Aachen University of Applied Sciences within the scope of a scientific research project there was developed an experimental prototype to show up new technologies for the web portal of the city council of Aachen. The goal was to achieve this without developing from the scratch. For this purpose an ontology was created to represent demonstratively the available information on the existing web portal of the city council. Algorithms were implemented to access, modify and search the developed ontology. The ontology is distributed with different instances on a peer-to-peer (hereafter P2P) network with agents as nodes. In particular this paper shows the development of this search engine using state of the art tools and frameworks. It describes the design of the ontology using Protégé and how different resources of that ontology are shared and searched in a multi agent system which is built with JADE.

1 Introduction

Accepted search engines on the Internet usually collect data about information without any higher knowledge of the context of this information. So the reply to a user's request on a search engine does not give precise answers according to the user's expected reply because it is not possible to provide contexts in the user's requests. The replies typically depend on the search engines' algorithm for indexing the existing information and do not consider the user's individual view on information.

A possibility to achieve a compromise between used search algorithms and the user's view on information would be the adoption of meta information on existing information. Meta information is information about the existing data given by something that was able to categorize it in a context. That can be done by using ontologies that represent those contexts. Ontologies in computer sciences are modeling techniques with usage of non-cyclic directed graphs. So relations between different information vertices can be defined in a machine readable format with an ontology written in a certain programming language.

Another possibility for such a compromise would be the consideration of the logical and physical view of information on the internet. The information is distributed amongst vast different data formats on different nodes. The search engines crawl the different datatypes to represent their knowledge through one web interface. But the information could be

searched where it is found in the network to have logical and physical information vertices. This can be achieved by combining the search functionalities with the meta information. In particular, an improved scenario is to search directly on the nodes by using P2P networks to connect information and search functionalities directly.

In our research project we decided to use existing tools and frameworks to combine the two technologies ontologies and P2P architectures. We chose Protégé[1] to design the ontology. The JADE[2] framework was used to implement the multi agent system with the JADE addon AgentOWL[3] to access and infer the ontology. AgentOWL itself works on the Jena[4] framework. We decided to use this AgentOWL addon because it is not possible to represent ontologies in a commonly used format like RDF[5] or OWL[6] in JADE. The implementation of an ontology in JADE is quite complicated for the programmers, e.g. a slot of a class of an ontology is one whole statement written in the java source. In contrast AgentOWL provides the possibility to use an ontology in OWL- or RDF-format as knowledge base of an agent.

2 The used ontology

The ontology was developed with Protégé in OWL-format. The topics represented in the domain ontology are quotations from the city council that are available to the citizens. More precisely those quotations were modeled as services to categorize this concept. After defining the terms of the different services several vocabularies were distinguished to describe and identify the services.

In particular we designed a defined structural vocabulary to distinguish the contexts. We created different types of vocabularies that either describe or identify certain services in the conceptual categorization. This categorization is visualized in figure 1.

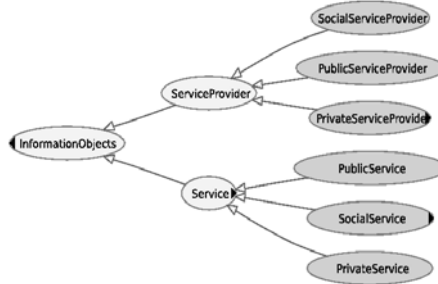


Figure 1: The ontology

3 The topology of the multi agent system

The several departments of the city council are widely spread over the whole city with different administrative responsibilities. So one point is to represent the knowledge in a domain ontology with the possibility to divide this upper domain ontology in different branches of administration and the other point is to provide this division in an architectural context of the existing network. This combination of the two different views was achieved by distributing the developed domain ontology on a P2P architecture with agents as nodes.

In the theoretical point of view there exist reactive, proactive and even social agents. With the use of JADE as agent framework it is possible to access predesigned templates of the different roles in a multi agent system. During the development of the prototype we just used the possible roles whenever they were needed and there was not a design schema about different behaviours and interaction of the agents. In practical view JADE combines those different approaches because it provides a common java framework for implementing those different roles of an agent, and also there are further addons to provide special implementations of roles like complex social behaviour. Therefore we decided to use JADE as framework for the development, especially because of its confirmation to FIPA[7] specifications.

We developed two different types of agents with different roles in the network:

- the search agent: This agent provides the entry point of a user who searches the ontology;
- the ontology agent: This agent provides the representation of the knowledge of one department in the view of an administrative of it.

At first we have a look at a typical use case in this network. A user phrases his search request through a user interface that was implemented as java applet. The user interface contacts the search agent to submit the search terms. The search agent creates an RDQL[8]-query message and sends it to the different ontology agents. The ontology agents work on the same domain ontology but they have different knowledge about it. In particular each agent has only instances of his administrative branch of the ontology. This individual

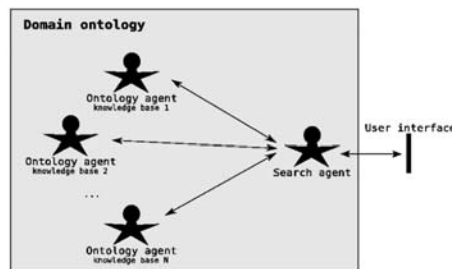


Figure 2: The topology of the MAS

branch of the domain ontology builds its agent memory. Each ontology agent queries its memory with the submitted RDQL-query and sends the results back to the search agent. The search agent purifies the results and returns them to the user interface. This topology of the multi agent system is shown in figure 2.

4 Conclusion and future work

The distribution of the information on nodes as agents in this P2P architecture shows the possibility to organize knowledge in a network in a user's view. Actually the agents are designed and implemented as autonomous components of the system. Each agent has its own control structure instead of regarding the control structure of the whole system. So the architecture is very scalable by adding and removing any ontology agents.

This prototype shows a perspective on developing a multi agent system working with ontologies as knowledge bases with state of the art tools and frameworks. In particular it shows how the two technologies, **agents** and **ontologies**, can be combined to implement data based on the users's view and the administrator's view on knowledge.

A possibility for further development is to design similar structured domain ontologies which can be searched by different search agents. Afterwards a domain search agent has to be created which contacts the several other search agents that search their domain ontologies themselves. Such developed relations between domain ontologies can be used for a domain search agent. Thus in future work several other domain ontologies that are distributed amongst peers have to be created and interconnected with an agent which organizes the interchange of knowledge in this system.

References

- [1] Protégé. *Free and Open Source ontology editor and knowledge-base framework*. <http://protege.stanford.edu>.
- [2] Java Agent DEvelopment Framework. *An Open Source platform for peer-to-peer agent based applications*. <http://jade.tilab.com>.
- [3] Michal Laclavik. AgentOWL. *Agent Library for Supporting RDF/OWL model based on Jena*. <http://agentowl.sourceforge.net>. Institute of Informatics SAS, Bratislava. Slowakei. January 2005.
- [4] Jena. *A Semantic Web Framework for Java*. <http://jena.sourceforge.net>.
- [5] RDF. *Ressource Description Framework*. <http://www.w3.org/RDF/>
- [6] OWL. *Web Ontology Language*. <http://www.w3.org/2004/OWL>.
- [7] FIPA. *Foundation for Intelligent Physical Agents*. <http://www.fipa.org/>
- [8] RDQL. *A query language for RDF*. <http://www.w3.org/Submission/RDQL>.

Ontologiebasierte Sprachvernetzung zur intelligenten Suchunterstützung von Bildinhalten

Jakob Strauch

Fachhochschule Aachen
Betreut durch Prof. H. Faßbender

jakob.strauch@web.de

Abstract: Bei steigender Bildanzahl in einem Schlagwortbasiertem Bildverwaltungssystem muss die Lücke zwischen Bildsuchenden und Bildbeschreibenden geschlossen werden. Der hier verfolgte Ansatz stützt sich auf ein zu verwaltes Sprachnetz, welches vorrangig Bedeutungen beschreibender Schlag- und Suchwörter als abstrakte Objekte miteinander vernetzt. Die daraus entstehenden semantischen Räume können als Suchunterstützung dienen, um erweiterte Suchanfragen zu stellen. Diese Suchanfragen werden je nach sprachwissenschaftlichen Beziehungsgrad unterschiedlich gewichtet, um die Relevanz eines Ergebnisses einzuschätzen.

1 Motivation

Die Ware „Fotografie“ wird vermehrt im Netz über Online Bildagenturen gehandelt, deren Kunden z.B. Zeitschriftenredakteure oder Werbeagenturen sind. Der Grund liegt in der rasanten Entwicklung der digitalen Fotografie und des weltweiten Netzes: Digitale Abbilder werden technisch immer perfekter, eCommerce hat sich als etablierte Form der Internetnutzung durchgesetzt. Viele Bildagenturen verwalten mehrere hunderttausend Fotos und vertreiben diese über ihr Portal. Beachtet man die oft unzureichende Verschlagwortung, die Verwendung unterschiedlicher Vokabulare, fehlerhafte Eingaben und die Multilingualität dieser Portale, so führt ein einfacher Textabgleich zu unbefriedigenden oder gar keinen Ergebnissen. Dazu ist ein intelligentes Suchsystem nötig, das eine Suchanfrage interpretieren kann.

2 Lösungsansatz

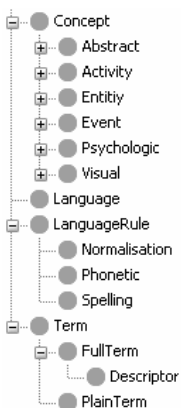
Eine erste Idee einer ontologiebasierten Websuche wurde bereits in einer vorherigen Diplomarbeit ausgearbeitet, die eine Stichwortsuche mittels einer auf das Thema zuge-

schnittenen Ontologie implementierte [1]. Aufgrund der Allgemeinheit von Fotografien¹ ist dieser Lösungsweg für eine Bildsuchmaschine auf den ersten Blick unbrauchbar. Der erste Ansatz diente jedoch als Ausgangspunkt um eine vom Thema losgelöste Ontologie zu entwickeln. Das Ergebnis ist eine Ontologie, die linguistische Zusammenhänge in Form von lexikalischen und semantischen Relationen abbildet. Das langfristige Ziel ist es, potentielle Suchwörter und tatsächliche Schlagwörter in dieses System aufzunehmen und in Relation zu setzen. Der entscheidende Unterschied liegt aber darin, dass nicht die Begriffe direkt miteinander verknüpft werden, sondern über abstrakte **Bedeutungsobjekte** gebunden werden. Im Gegensatz zu üblichem Ontologiedesign werden somit nicht direkt Begriffe in Beziehung gesetzt, sondern die Semantik dieser Begriffe. Die resultierende „Meta-Ontologie“ ist ein Netz von Bedeutungen identifiziert durch einen oder mehrere Terme. Ein weiterer wichtiger Punkt für die Verwendung einer solchen Ontologie in einem Bildsuchsystem ist die Verwendung zuvor festgelegter Assoziationsrelationen zwischen Bedeutungen. Somit können (indirekt) verknüpfte Begriffe zur Einschätzung der Relevanz eines Bildes bezüglich des Suchbegriffes dienen.

Dieses Netz dient dann einer realen Suchmaschine als Wissensbasis. Sie kann aus der Ontologie implizit Informationen erschließen, um z.B. Sucheingaben lexikalisch zu korrigieren oder auf die Mehrdeutigkeit einer Sucheingabe hindeuten. Die Verwendung einer direkten Begriff- zu-Begriff Vernetzung könnte dies nicht leisten. Begriffe müssten zum Beispiel explizit als mehrdeutig deklariert werden. Weiterhin wäre die Vernetzung mehrdeutiger Begriffe nicht sinnvoll. Es wäre nicht eindeutig, welche Bedeutung des Begriffes mit einem anderen Begriff verwandt wäre.

Der entwickelte Suchunterstützungs-Prototyp wurde als PHP-Webapplikation realisiert. Weiterhin wurde pOWL² eingesetzt, ebenfalls eine PHP-Webapplikation und API, die Datenbankbasierte Ontologien beliebiger Größe verarbeiten kann. Wie die Basis für das heterogene Wissensnetz durch ein Ontologiedesign repräsentiert wird, sehen wir im folgenden Abschnitt.

3 Ontologie Design



Einer der Grundlagen zur Entwicklung dieser Ontologie ist ein NISO Leitfaden [3] zur Erstellung kontrollierter Vokabulare, an dem auch die nachfolgende Terminologie anlehnt ist. Die Hauptklassen sind **Concept**, **Term**, **Language** sowie **LanguageRule**. Instanzen der Klasse **Language** legen fest, welche Sprachdomäne eine Instanz der Klasse **Term** verwendet. Darüber hinaus enthalten sie sprachspezifische Regeln, um z.B. Tippfehler zu erkennen oder Terme zu normalisieren. Terme sind lediglich die möglichen lexikalischen Abbildungen der abstrakten Bedeutungen (**Concept**). Ein **Concept** wird durch einen identifizierenden **Term** samt qualifizierenden Merkmal eindeutig beschrieben, Beispiel: „bank (sitzelegenheit)“. Neben diesen **Descriptor** können auch weitere Terme ein **Concept**

¹ Bilder enthalten unterschiedlichste Motive und stellen unterschiedlichste Emotionen und Situationen dar

² <http://powl.sourceforge.net>

beschreiben. Sie bilden dadurch die Synonymmenge einer Bedeutung. Ein **Term** wiederum kann auch mehrere unterschiedliche **Concepte** beschreiben, d.h. dass dieser Ausdruck mehrere Bedeutungen hat. Dazu zählt auch das vorherige Beispiel „bank“. Terme stehen nie direkt in Relation zu einander. Um einen semantischen Bezug auf Bedeutungsebene herzustellen, werden **Concepte** miteinander verknüpft. Dazu gibt es allgemeine Relationen wie z.B. Ober- bzw. Unterbegriffsrelation. Die **Concepte** sind in einer Bedeutungstaxonomie kategorisiert. Das hat den Vorteil, dass man spezifische Zusammenhänge zwischen Bedeutungen auf bestimmte Konzeptklassen einschränken kann.

Ziel ist es, Konzepte miteinander in Beziehung zu setzen, die „thematisch“ verwandt sind. So lässt sich eine assoziative Verknüpfung zwischen dem **Concept** „Soldat“ und dem **Concept** „Krieg“ herstellen, die die Suche unterstützen kann. Es ist zwingend notwendig Relationen maßgeschneidert zwischen Konzeptklassen zu definieren, um eine Entartung des Netzes zu verhindern, was sich erheblich auf die Qualität der Suchunterstützung auswirkt. Ein interessantes Hilfsmittel kann dabei auch die „Normalisierte Google Distanz“ sein, mit der man versucht mit Hilfe eines Internet Suchmaschinen Indexes die (thematische) Nähe zweier Begriffe zu bestimmen [2].

4 Semantische Interpretation des Suchausdrucks

Die Interpretation der Sucheingabe wird in mehreren Schritten durchgeführt:

1. **Identifizierung einer Terminanz** – wird der eingegebene Begriff nicht im Termpool gefunden, so handelt es sich um eine Fehleingabe oder einen unbekannten Begriff
2. **Auffinden orthografisch ähnlicher Ausdrücke** – im Fehlerfall wird hier versucht einen ähnlich geschriebenen Term anhand einer abstrakten dem Term zugehörigen Zeichenkette zu ermitteln. Diese Zeichenkette wird aus den zuvor definierten Languageurules erstellt und ist im Ähnlichkeitsfall zweier Terme identisch.
3. **Auswahl des vermutlich korrekten Ausdruckes** durch den Nutzer
4. **Identifizierung des/der Konzepte(s)** – Homographie³ Terme beschreiben mehr als ein Konzept, wie z.B. „bank“ (sitzbank/kreditinstitut). In diesem Fall muss der Nutzer die korrekte Bedeutung bestätigen.
5. **Semantischer Raum** - Ist schließlich ein Konzept identifiziert, so kann der semantische Raum⁴ dieses Konzeptes zur Sucherweiterung dienen.

Anhand des semantischen Raumes, sprich der diversen Relationen zu anderen Konzepten und deren beschreibenden Termen, kann die Suchunterstützung eingeleitet werden, wie wir im nächsten Abschnitt sehen.

³ Ein Wort mit mehreren Bedeutungen

⁴ Der semantische Raum bildet sich durch Relationen wie Synonyme, Unterbegriffe oder Assoziationen

5 Suchstrategie

Ziel ist es die Quantität und die Qualität der Suchergebnisse zu verbessern. Die **Qualität** von Suchergebnissen äußert sich durch die **Relevanz** eines Bildes zum Suchausdruck. Wie lässt sich nun die Relevanz messen? Sofern durch eine Sucheingabe ein Konzept identifiziert wurde, kann man den semantischen Raum dieses Konzeptes ausnutzen, der durch die lexikalischen (*Konzept-Term*) und semantischen (*Konzept-Konzept*) Relationen definiert ist. Da die Relationen kategorisiert und benannt sind, z.B. „*Konzept hatUnterbegriff Konzept*“ oder „*Person erfülltSozialesKonzept SozialesKonzept*“, lassen sich daraus gewichtete Suchanfragen stellen. Man bedient sich dann aller Terme, die ein verknüpftes Konzept bindet. Die Ergebnisse werden unter der Beachtung der gewichteten Anfrage zu einer Ergebnismenge zusammengefasst. Taucht ein Bild in mehreren Mengen auf, so wird dessen Relevanz addiert. Daraus ergibt sich eine sortierte Menge, die die Wichtigkeit eines Bildes über die verwendeten Schlagwörter, das ursprüngliche Suchwort und aller Wörter die thematisch mit diesem im Zusammenhang stehen. Die Beachtung aller Synonyme in den Suchanfragen schlägt somit eine Brücke zwischen den Vokabularen des Suchenden und des Beschreibenden und erhöht somit die Ergebnisquantität. Die Ergebnisqualität verbessert sich durch die Beachtung aller Wörter (Synonymmengen der Konzepte) in dem semantischen Raum des ursprünglichen Suchbegriffes.

Die Suchunterstützung sollte in einer produktiven Umgebung als Webservice dienen, der als Eingabe den ursprünglichen Suchausdruck erhält und eine Menge von Begriffen zurückliefert, die klassifiziert sind. Anhand der Klassifikation kann die individuelle Suchmaschine die Relevanzen selbst bestimmen und die konkreten Suchanfragen starten. Der hier vorgestellte Prototyp dient somit als linguistische Wissensbasis und wird zwischen der Suchanfrage und der konkreten Suche eingesetzt.

Literaturverzeichnis

- [1] Amir Pour-Heidari
Diplomarbeit „Ontologiegestützte Websuche“
Fachhochschule Aachen, 2005
- [2] Rudi Cilibrasi und Paul M. B. Vitanyi
Automatic Meaning Discovery using Google
arXiv.org - Computer Science, abstract
<http://www.arxiv.org/abs/%20cs.CL/0412098>, 2004
- [3] National Information Standards Organization
Guidelines for the Construction, Format and Management of
Monolinugal Controlled Vocabularies
<http://www.nis.org/committees/MT-info.html>, 2005

”Werkzeugunterstützung zur Einführung von Adaptionsstrukturen basierend auf einer Service-orientierten Architektur”

Sven Bartel

bartels1@iai.uni-bonn.de

Abstract: Für bestimmte [CGS05, MRS⁺06] Softwaresysteme existiert die Anforderung, Andockpunkte bereitzustellen, um Änderungen an der Software zur Laufzeit durchführen zu können. Dazu werden *Adaptionsstrukturen* zur Realisierung von Adaptivität sowie Verfahren zur Integration dieser Adaptionsstrukturen in bestehende Softwaresysteme beschrieben, die eine *Service-orientierte Architektur* in Kombination mit *Design-Pattern* [GHJV95] verwenden.

1 Einleitung

Es werden unterschiedliche Arten von Softwareveränderungen durch Adaptionsstrukturen betrachtet. Vorhandenes Verhalten kann durch neues Verhalten erweitert (*Decorator-Pattern*) und ersetzt werden (*Strategy-Pattern*). Zudem kann die Kommunikation von Ereignissen zwischen bestehenden und neuem Verhalten (*Observer-Pattern*) und die Verwendung von neuem Verhalten, dessen Schnittstelle nicht veränderbar ist, ermöglicht werden (*Adapter-Pattern*). Die Adaptionsstruktur unter Verwendung des Strategy-Pattern sowie das Verfahren zur Verwendung der Adaptionsstruktur werden exemplarisch beschrieben.

2 Annahmen, Voraussetzungen und Ziele der Diplomarbeit

Ein wesentliches Ziel der Diplomarbeit ist die Beschreibung einer Adaptionsstruktur, basierend auf einer Service-orientierten Architektur, in Verbindung mit dem Strategy-Pattern zur Realisierung dynamischer Services. Zusätzliche Ziele sind die Darstellung eines Verfahrens zur Integration der Adaptionsstruktur in einem bestehenden Softwaresystem, um Adaptivität in diesem zu ermöglichen, sowie die Implementierung eines Prototypen, mit dem die beschriebene Adaptionsstruktur durch Anwendung des beschriebenen Verfahrens in ein bestehendes Softwaresystem integriert werden kann. Es wird vorausgesetzt, dass die Adaptionsstruktur in objektorientierten Softwaresystemen zur Anwendung kommt. Das Verfahren zur Integration der Adaptionsstruktur wird detailliert, anhand des Java-Sprach-Modells, erläutert und steht exemplarisch für die Verwendung eines objektorientierten Softwaresystems.

3 Über Variabilität zur Adaptivität

Mit Variabilität wird die Möglichkeit bezeichnet, ein Softwareelement zu festgelegten Zeitpunkten im Software-Entwicklungsprozess zu variieren. Die Teile eines Softwaresystems, welche die Möglichkeit bieten, Änderungen an einem Softwaresystem zu vollziehen, werden als *Variationspunkte* bezeichnet [ED03]. An diesen Punkten ist eine Variation des Verhaltens möglich. Unterschiedliches Verhalten an einem Variationspunkt wird durch *Varianten* implementiert. Die Variation des Verhaltens durch Auswahl einer Variante aus der Menge der verfügbaren Varianten, welche an einen Variationspunkt unterstützt werden, wird als *Binden einer Variante* bezeichnet. Um Varianten an einen Variationspunkt binden zu können, müssen diese ein für den Variationspunkt geeignetes Verhalten implementieren. Übertragen auf das Strategy-Pattern entsprechen die Varianten den Implementierungen der Strategien. Durch Verwendung der Service-orientierten Architektur wird ermöglicht, die Varianten zur Laufzeit im bestehenden Softwaresystem verwenden zu können.

4 Refactoring zu einer Adaptionstruktur

Beim Refactoring zum Strategy-Pattern ist das Ziel die Extrahierung einer Strategieklassse. Analog zu der in [Ker04, SD02] beschriebenen Ausgangssituation wird als Startpunkt des Refactoring eine Methode gewählt. Jedoch wird angenommen, dass sich in dieser Methode (bezeichnet als Berechnungsmethode) nicht bedingte Logik, sondern das für die Adaptivität relevante Verhalten befindet. Dies ist das Verhalten, welches zur Laufzeit ersetzt oder erweitert werden soll. Durch statische Quellcodeanalysen wird die Refactoringausgangssituation analysiert. Auf Basis dieser Analyseergebnisse kann der Entwickler z.B. bei der Entscheidungsfindung unterstützt werden, welche Methoden neben der Berechnungsmethode variiert werden sollen. Für die Quellcodeanalysen wird ein theoretisches Modell beschrieben, das die Abhängigkeiten der Berechnungsmethode zu weiteren Methoden analysiert. Das theoretische Modell klassifiziert diese Methoden in eine der folgenden drei Mengen: Frei verschiebbare, abhängige und polymorphe Methoden.

Menge der frei verschiebbaren Methoden: Die Menge der frei verschiebbaren Methode wird rekursiv beschrieben. Zu ihr gehören alle Methoden, die ausschließlich von der Berechnungsmethode oder einer frei verschiebbaren Methode verwendet werden. Die Menge ist definiert als die Vereinigung

$$FS(m) := \bigcup_{n \in \mathbb{N}} FS_n(m) \quad (1)$$

von rekursiv definierten Mengen vom Grad $i \in \{1..n\}$. Für $i = 1$ hat die Menge den Grad Eins. Sie umfasst diejenigen Methoden, die nur von einer einzelnen Methode aufgerufen werden und ist definiert als $FS_1(m) = \{m' \in MI(m) \mid \forall m'' \in M(C) : m' \notin MI(m'')\}$. Die Menge vom Grad $i + 1$ umfasst diejenigen Methoden, die ausschließlich von Methoden vom Grad i aufgerufen werden und ist definiert als $FS_{i+1}(m) = FM(FS_i(m))$ mit $FM(A) = \{m' \in M(C) \mid \exists m_0 \in A : \forall m_1 \in M(C) \setminus A : m' \in MI(m_0) \wedge m' \notin MI(m_1)\}$.

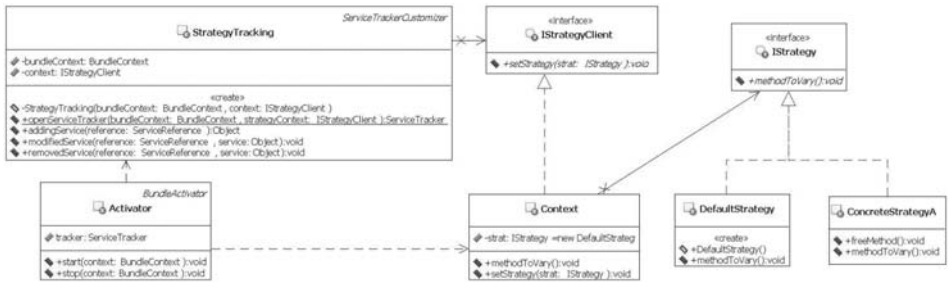


Abbildung 1: Dargestellte Adaptionstruktur durch Kombination des Strategy-Pattern mit OSGI

Menge der abhängigen Methoden: Die Menge der Methoden, die von der Berechnungsmethode und/oder von mindestens einer der frei verschiebbaren Methoden sowie von einer weiteren Methode aufgerufen werden, die weder der Berechnungsmethode entspricht noch Teil der Menge der frei verschiebbaren Methoden ist, gehört zu der Menge der abhängigen Methoden. Sie ist definiert als: Für $c \in C$ gilt

$$DS(m) = \{m' \in M(C) \mid \exists m_0 \in FS(m) : m' \in MI(m_0) \wedge m' \notin FS(m)\} \quad (2)$$

Menge der polymorphen Methoden: Bei polymorphen Methoden wird zwischen überschreibenden und überschriebenen Methoden unterschieden. Zu den überschreibenden Methoden gehören ebenfalls Methoden, die in einem Interface deklariert sind. Bei überschriebenen Methoden erhält der Benutzer die Möglichkeit, die durch polymorphe Methoden dargestellte Hierarchie mit zu verschieben, so dass diese Hierarchie in Subklassen der Klasse *Strategy* rekonstruiert wird.

Durchzuführende Refactorings: Vom Benutzer ausgewählte Methoden werden mit *Move Method* [FBB⁺99] aus der Klasse *Context* in die Klasse *Strategy* verschoben. Werden Methoden von dritten Klassen verwendet, so werden die Methoden zusätzlich delegiert, um die Strategieinstanz in der Klasse *Context* für die dritten Klassen nicht sichtbar zu machen. Dies verhindert eine Kopplung einer dritten Klasse mit der Klasse *Strategy*. Überschreibende Methoden werden ebenfalls zusätzlich delegiert, um Verhalten erhaltende Refactorings zu gewährleisten. Methoden mit einem expliziten Aufruf der Superklasse sind grundsätzlich nicht verschiebbar. Dem Benutzer kann an dieser Stelle der Hinweis gegeben werden, durch *Extract Method* [FBB⁺99] den Aufruf der Superklasse vom Rest der Methode zu extrahieren, um den Rest der Methode mit verschieben zu können.

Service-orientierte Infrastruktur in OSGI: In Abb. 1 ist die Adaptionstruktur für das Strategy-Pattern in Kombination mit der Service-orientierten Architektur *OSGI* dargestellt. Der Klasse *StrategyTracking* kann bei der Instanziierung ein Client vom Typ *IStrategyClient* übergeben werden, so dass zur Laufzeit die Methode *setStrategy* von der OSGI-Infrastruktur aufgerufen wird und eine neue Variante/Strategie setzen kann. Die Variante, die in diesem Fall zur Laufzeit adaptiert wird, ist *ConcreteStrategyA*. Eine offene Frage zur Beantwortung im Rahmen der Diplomarbeit umfasst die Selektion der Stelle im Quelltext des Client zur Durchführung der Instanziierung der Klasse *StrategyTracking*.

A Definitionen

Es wird die Notation aus [BDW99] verwendet. Sei C die Menge der Typen.

Definition A.1. (Typen) Für jedes $c \in C$ gilt: $Interface(c) \subset C$, die Menge der Interfaces, die c implementiert. $Parent(c) \subset C$, die Superklasse von c . $Children(c) \subset C$, die Menge der direkten Subklassen von c . $Ancestors(c) \subset C$, die Menge der Klassen, von denen c ableitet, entweder direkt oder indirekt.

Definition A.2. (Methoden und Felder) Sei $c \in C$, dann gilt: $M(C)$, die Menge aller Methoden. $F(C)$, die Menge aller Felder. $M_{Im}(c)$ Die Menge der Methoden, die c implementiert, d.h. c definiert einen Body für m .

Definition A.3. (Methodenaufrufe) Sei $m \in M_{Im}(c), m' \in M(C), c \in C, f \in F(C)$, dann gilt: $MI(m)$, die Menge der Methoden, die durch m aufgerufen werden, d.h. $m' \in MI(m)$ genau dann, wenn der Body von m einen Aufruf von m' enthält. Es gilt $m \in MIF(f)$ genau dann, wenn die rechte Seite der Deklaration von f die Methode m aufruft.

Definition A.4. (Feld-Referenzen) Sei $f, f' \in F(C), m \in M_{Im}(c)$ für ein $c \in C$, dann gilt: $f \in FR(m)$ genau dann, wenn der Body von m eine Referenz zu f enthält. Es gilt $f' \in FRF(f)$ genau dann, wenn die rechte Seite der Deklaration von f auf f' zugreift.

Literatur

- [BDW99] L. C. Briand, J. W. Daly und J. K. Wüst. A Unified Framework for Coupling Measurement in Object-Oriented Systems. *IEEE Transactions on Software Engineering*, 25(1):91–121, January 1999. IEEE Press, Piscataway, NJ, USA.
- [CGS05] S. Cheng, D. Garlan und B. Schmerl. Making Self-Adaptation an Engineering Reality. In *Self-Star Properties in Complex Information Systems, LNCS*, 3460:158–173, 2005. Springer-Verlag.
- [ED03] E. Visser E. Dolstra, G. Florijn. Timeline variability: the variability of binding time of variation points. In *Proceedings of the Workshop on Software Variability Management*, Seiten 119–122, 2003.
- [FBB⁺99] M. Fowler, K. Beck, J. Brant, W. Opdyke und D. Roberts. *Refactoring: Improving the Design of Existing Code*. Addison-Wesley, June 1999.
- [GHJV95] E. Gamma, R. Helm, R. Johnson und J. Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley, January 1995.
- [Ker04] Joshua Kerievsky. *Refactoring to Patterns*. Addison-Wesley, 2004.
- [MRS⁺06] H. Mügge, T. Rho, D. Speicher, J. Kuck und A. B. Cremers. Towards an infrastructure for context-sensitive intelligence. SAKS Workshop, Kassel, Germany, March 2006.
- [SD02] O. Nierstrasz S. Demeyer, S. Ducasse. *Object Oriented Reengineering Patterns*. Morgan Kaufmann, July 2002.

Service-orientierte Integration im E-Commerce

Sven Feja

svfe@informatik.uni-kiel.de

Abstract: Die Integration von Anwendungssystemen ist ein zentrales Problem des elektronischen Handels, da ERP-Systeme als IT-Herzstück des Unternehmens zunehmend durch E-Commerce-Systeme ergänzt werden. Diese Integration umfasst aber nicht mehr nur unternehmensinterne sondern auch unternehmensexterne Systeme. Daraus ergeben sich beispielsweise erhöhte Anforderungen an die Performanz der Integrationslösung.

In diesem Beitrag wird exemplarisch die Integrationslösung eines E-Commerce-Systems mit einem ERP-System vorgestellt. Die Lösung basiert auf einer Serviceorientierten Architektur unter Einsatz von Web Services und BPEL4WS-Prozessen. Ferner werden die dabei auftretenden Performanzprobleme thematisiert. Es wird eine Reduzierung der Web Service-Aufrufe über eine Bündelung angestrebt, denn eine effiziente Lösung ist ein Schlüsselfaktor für die Wirtschaftlichkeit von Handelsplattformen.

1 Einleitung

Die zunehmende Automatisierung und Verkettung von Wirtschaftsprozessen durch den Einzug des Internets und im Zuge der Globalisierung stellt hohe Anforderungen an die Anwendungslandschaften der Unternehmen. Die Zusammenführung von heterogenen Anwendungssystemen und Software-Komponenten spielte zwar auch schon in der Vergangenheit eine Rolle, jedoch waren diese Integrationen sehr speziell, statisch und auf wenige Großunternehmen beschränkt. In der heutigen Zeit sind durch die neuen Anforderungen nahezu alle Unternehmen mit Integrationsproblemen konfrontiert.

Eine entscheidende Anforderung der Integration ist die Bewältigung von Systembrüchen durch unterschiedliche Datenstrukturen, Betriebssysteme, Programmiersprachen und Protokolle zwischen den zu verbindenden Systemen. Nur durch Interoperabilität über jegliche Heterogenität kann eine Homogenisierung von Geschäftsprozessen in der verschiedenartigen Anwendungslandschaft erreicht werden. Ferner muss die Integration eine performante Verknüpfung der Systeme leisten, damit ein reibungsloser Ablauf der Geschäftsprozesse gewährleistet werden kann. Als weitere wichtige Anforderung ist die flexible Erweiterbarkeit für weitere Anwendungssysteme unterschiedlicher Plattformen zu sehen.

In diesem Beitrag soll exemplarisch die Integration eines ERP- mit einem E-Commerce-System vorgestellt werden. Der Fokus soll aber auf die performante Verknüpfung der Systeme gesetzt werden. Im folgenden Abschnitt soll kurz die Integrationslösung skizziert werden und im Anschluss wird die Bündelung der Web Service Aufrufe auf abstrakter Ebene vorgestellt. Eine ausführliche Darstellung der Integration ist in [Fej06] zu finden. Die Bündelung wird in [Fej07] näher erläutert.

2 Integration

Bei den zu integrierenden Systemen handelt es sich um das E-Commerce-System Enfinity Suite 6.0 der INTERSHOP COMMUNICATIONS AG [INT07a] und das ERP-System GODYO P/4 der GODYO AG [GOD07]. Beide Anwendungssysteme werden für mittelständische Unternehmen eingesetzt, wobei auch der Kernbereich des GODYO P/4s KMUs sind und die Enfinity Suite erst durch die INTERSHOP-Initiative „E-Commerce für den Mittelstand“ für KMUs interessant geworden ist [INT07b].

Die Integrationslösung basiert auf einer Service-orientierten Architektur unter Verwendung von Web Services und einem BPEL4WS-Prozess. Der BPEL4WS-Prozess wird in einer Komponente *Datenabgleich* realisiert. In dieser Komponente wird ein Prozess für den Datenabgleich von bspw. Artikel- und Kundendaten zwischen den Systemen erstellt. Zudem werden in dem Prozess die unterschiedlichen Datenformate durch eine als Web Service eingebundene XML-Transformationstechnik [FS06] angepasst. In dem E-Commerce- und ERP-System sind Web Services sowie Web Service-Clients zu entwickeln, welche in den BPEL4WS-Prozess eingebunden werden. Die Web Services werden asynchron realisiert, um den Workflow des einzelnen Systems nicht zu beeinflussen. Dies steigert die Performanz der Systemintegration.

3 Optimierung des Integrationsansatzes

Eine Evaluation des Integrationsansatzes hat gezeigt, dass die Anforderung der performanten Verknüpfung der kooperierenden Systeme für bestimmte Anwendungsfälle nicht vollständig erfüllt ist. Dieses Performanzproblem wird durch die besonders hohe Anzahl der Web Service-Aufrufe hervorgerufen und ist auf die Übermittlung der Daten über das HTTP-Protokoll zurückzuführen [JMH04], [KTR03]. Die (De-) Serialisierung ist ebenfalls ein Einflussfaktor auf die Performanz. Allerdings ist dieser weitaus geringer als erwartet [JMH04]. Ansätze um die (De-) Serialisierung zu verbessern, werden in [STT05], [AGGL04] und [GMNR05] vorgestellt. Auch die SOAP-Spezifikation und die daraus resultierenden langen SOAP-Nachrichten werden in [KTR03] ebenfalls als reduzierend für die Performanz von Web Services betrachtet. Es kann aber mit Hilfe des *generic Gzip Compression Algorithm* eine Komprimierung der SOAP-Nachricht um bis zu 50% erzielt werden, allerdings geht dies zu Lasten der Rechen- und Übertragungszeit. Der Optimierungsansatz in diesem Beitrag ist an das HTTP-Protokoll als „*bottle neck*“ adressiert. Es soll durch die Optimierung der Anzahl der Aufrufe von Web Services eine Performanz-Verbesserung erzielt werden.

3.1 Verbindungsarten von Web Services

Für die Verbindung der Systeme stehen grundsätzlich synchrone und asynchrone Web Services zur Verfügung. Die Auswahl der Verbindungsart ist stark vom Anwendungsfall

abhängig. Getrieben wird die Entscheidung meist von der Notwendigkeit eines sofortigen Ergebnisses einer Anfrage (wie z. B. bei Verfügbarkeitsabfragen oder Kreditkartenbuchungen) und Performanzanforderungen.

Die wesentliche Eigenschaft einer synchronen Verbindung ist die zeitliche Bindung der Systeme in eine Abarbeitungsreihenfolge, wobei vor dem Beginn einer Aufgabe die davor liegende beendet sein muss. Durch lang andauernde Aufgaben in der Abarbeitungsreihenfolge kann es zur Blockierung eines Systems kommen, was aus Kundensicht bspw. dem Anlegen eines neuen Auftrages unangenehme Wartezeiten zur Folge hat. Aus Systemsicht ist dadurch die Performanz der Systeme reduziert.

Asynchrone Verbindungen dienen grundsätzlich zum zeitlichen entkoppeln der Systeme, damit wird erreicht, dass keine Blockierung bzw. Wartezeit durch das kooperierende System entsteht. Ein Kunde kann sofort weiterarbeiten und aus Systemsicht werden vorhandene Ressourcen genutzt und somit eine Performanzsteigerung erzielt. Im Fall der Integration ist dies eine sinnvoll einsetzbare Lösung, da für die Integration keine Anfrageantwort in Echtzeit benötigt wird. Durch die asynchrone Verarbeitung der Web Service-Nachrichten wird die bereits vorgestellte Integrationslösung aus der Performanzsicht zwar beschleunigt, aber bei genauerer Betrachtung stellt z. B. der Datenabgleich bei Einspeisung von vielen Artikeln aus Produktkatalogen eine deutlich höhere Anforderung an die Performanz dar. Diesen Anforderungen soll im folgenden Abschnitt Rechnung getragen werden.

3.2 Bündelungsverfahren

Das Bündelungsverfahren ist besonders für bestimmte Anwendungsfälle von Bedeutung. Zu unterscheiden sind dabei der Batch- und Dialogbetrieb. Im Falle des Batchbetriebs werden gleichzeitig viele Daten(-sätze) von einem oder wenigen Benutzern in ein System eingepflegt wie bspw. der Import einer hohen Anzahl von Artikeln aus Produktkatalogen. Auf die vorgestellte Integrationslösung bezogen bedeutet diese, dass beim Katalogimport jeder Artikel einzeln an das kooperierende System übermittelt werden würde. Dies bringt Performanzprobleme durch die ständigen Übertragungen per HTTP-Protokoll mit sich. Beim Dialogbetrieb hingegen geben viele Benutzer eher wenige Daten(-sätzen) ein, wie es bspw. beim Anlegen von Bestellungen durch Kunden auf dem Web Frontend des INTERSHOP Enfinity der Fall ist. Die hohe Anzahl der einzelnen Aufrufe stellt auch im Dialogbetrieb ein Performanzproblem dar. Für diesen Anwendungsfall muss aber auch betrachtet werden, dass bspw. Kundenaufträge nach möglichst kurzer Zeit dem kooperierenden System übergeben werden sollten. Dies muss in den Optimierungsansatz einbezogen werden.

Folglich bedarf es der Verminderung der Anzahl der Web Service-Aufrufe für Batch- und Dialogbetrieb mit Beachtung der besonderen Anforderungen im Dialogbetrieb. Dabei ist die Bündelung besonders für Artikelimporte interessant, wie am bereits angeführten Beispiel des Imports von Artikeln aus Produktkatalogen ersichtlich wird. Aber auch die Auftrags- und Kundendaten haben ihre Anwendungsbereiche. So können z.B. Aufträge aus Sammelbestellungen gebündelt übermittelt werden oder neue Kunden von Außendienstmitarbeitern gesammelt eingetragen werden. Die Übertragung erfolgt bei Komplettierung des Bündels bzw. beim Dialogbetrieb zusätzlich noch über eine Zeitsteuerung.

4 Zusammenfassung

Der vorgestellte Integrationsansatz für ein ERP- und E-Commerce-System erreicht unter Verwendung der Service-orientierten Architektur die gestellten Anforderungen. Eine performante Verknüpfung der Systeme durch Web Services kann über eine Optimierung der Web Service-Aufrufe unter Verwendung eines Bündelungsverfahrens erreicht werden.

Literatur

- [AGGL04] Nayef Abu-Ghazaleh, Madhusudhan Govindaraju und Michael J. Lewis. Optimizing Performance of Web Services with Chunk-Overlaying and Pipelined-Send. In *International Conference on Internet Computing*, Seiten 482–485, 2004.
- [Fej06] Sven Feja. Web Service basierte Integration im E-Commerce., 02 2006.
- [Fej07] Sven Feja. Systemintegration durch Service-orientierte Architektur im elektronischen Handel und Beschaffung. In *Integration Engineering – Motivation, Begriffe, Methoden und Anwendungsfälle*, Jgg. VI of *Leipziger Beiträge zur Informatik*. Eigenverlag des Leipziger Informatik Verbunds (LIV), 2007.
- [FS06] Daniel Fötsch und Andreas Speck. XTC – The XML Transformation Coordinator for XML Document Transformation Technologies. In *DEXA'06: Proceedings of the 17th International Conference on Database and Expert Systems Applications*, Seiten 507–511, Washington, DC, USA, 2006. IEEE Computer Society.
- [GMNR05] Martin Gudgin, Noah Mendelsohn, Mark Nottingham und Hervé Ruellan. SOAP Message Transmission Optimization Mechanism. <http://www.w3.org/TR/soap12-mtom/>, 01 2005. Abruf: 15.01.2007.
- [GOD07] GODYO AG. Enterprise Resource Planning (ERP) - GODYO AG. <http://www.godyo.com/bs/produkte-und-loesungen/enterprise-resource-planning-erp>, 2007. Abruf: 15.01.2007.
- [INT07a] INTERSHOP COMMUNICATIONS AG. Intershop - E-Commerce Software der Extraklasse. http://www.intershop.de/solutions/e-commerce_software/, 2007. Abruf: 13.02.2007.
- [INT07b] INTERSHOP COMMUNICATIONS AG. Intershop - E-Commerce Software für den Mittelstand. <http://www.intershop.de/solutions/midsize>, 2007. Abruf: 14.01.2007.
- [JMH04] Mario Jeckle, Ingo Melzer und Michael Himsolt. Performance of Web Services. <http://www.mathematik.uni-ulm.de/sai/ws03/webserv/PerfWS.pdf>, 02 2004. Abruf: 15.01.2007.
- [KTR03] Jaakko Kangasharju, Sasu Tarkoma und Kimmo Raatikainen. Comparing SOAP Performance for Various Encodings, Protocols, and Connections. In *Personal Wireless Communications*, Seiten 397 – 406, 2003.
- [STT05] Toyotaro Suzumura, Toshiro Takase und Michiaki Tatsubori. Optimizing Web Services Performance by Differential Deserialization. In *ICWS '05: Proceedings of the IEEE International Conference on Web Services (ICWS'05)*, Seiten 185–192, Washington, DC, USA, 2005. IEEE Computer Society.

PRREP – Projekt Repository

Tim Lessner, Marius Hummel

Hochschule Reutlingen

Fakultät Informatik

tim.lessner@student.reutlingen-university.de

marius.hummel@student.reutlingen-university.de

Art der Arbeit: Diplomjahresprojekt

Betreuer/in der Arbeit: Prof. Dr. rer. nat. Alfred Zimmermann

Abstract: Die Arbeit beschäftigt sich mit der Konzeption und prototypischen Implementierung eines server-basierten Projektrepository auf der Basis von bestehenden Open-Source Produkten. Im Fokus steht die Integration und vor allem die programmatische Erweiterung der Produkte, zur Einbettung in den Software Engineering Prozess. Es wurde ein prototypisches Webportal zur einheitlichen Koordination und Kommunikation bei Projekten, welches ein Impact Management unter Artefakten ermöglicht, realisiert.

1 Motivation und Ziele

In einem Software Engineering Prozess entsteht innerhalb der verschiedenen Phasen eine enorme Anzahl von Artefakten in Form von Dateien. Diese unterliegen zu unterschiedlichen Graden ständigen Änderungen: Sie werden initial gespeichert, wieder verwendet, zwischen gespeichert oder verworfen. Dies führt zur Entstehung einer erheblichen Menge an Varianten der Dateien, die untereinander in einer Änderungsabhängigkeit stehen. Zur sinnvollen Organisation dieser schnell unübersichtlich werdenden Strukturen, wird eine zentrale Ablage benötigt, welche die dezentrale Arbeit an einem Artefakt ermöglicht und darüber hinaus Abhängigkeiten unter den Artefakten verwalten kann. Im Rahmen der Softwareentwicklung werden auch Sperr - und Zusammenführungsverfahren benötigt, die ein paralleles Arbeiten am gleichen Artefakt ermöglichen aber auch verhindern können. Diese Anforderungen werden mittels eines Versionierungssystems als Basis des Projekt-Repositorys realisiert. Zusätzlich existiert ein Impact Management, das die Änderungsabhängigkeiten unter den Artefakten verwaltet.

1.1 Freigabeprozess und logischer Zusammenbau von Artefakten

Um die Qualität von Artefakten zu gewährleisten, sind unter Anderem Freigabeverfahren notwendig. Artefakte sollen erst nach einer Prüfung einem immer breiter werdenden Publikum zur Bearbeitung und Weiterverwendung zur Verfügung stehen. Am Ende steht die Kundenfreigabe, das Produkt ist produktiv einsatzfähig.

—

Abbildung 1: Freigabestati von Artefakten

Neben der Freigabe besitzt jedes Artefakt einen logischen Zusammenbau der festgelegt werden kann. Ein Artefakt kann eine Version, eine Variante einer Version, eine Konfiguration, ein einfaches Projektprodukt oder ein einfacher Container sein. Der Übergang von einem zum nächsten Status ist unter Anderem von den Freigabeprozessen beeinflusst.

—

Abbildung 2: Logischer Zusammenbau von Artefakten

1.2 Impact Management

Im Repository versionierte Artefakte können ohne Beachtung der Hierarchiestruktur, in der sie versioniert sind, abhängig voneinander sein. Das Wissen über diese Abhängigkeiten ist von enormer Bedeutung für alle Projektbeteiligten, da es Rückschlüsse auf mögliche Auswirkungen von Änderungen zulässt. Es wurde ein Impact Management realisiert, welches es ermöglicht Artefakte untereinander in Relation zu setzen, um Abhängigkeiten abzubilden. Um Aussagen über die Größe und Weitläufigkeit von Auswirkungen von Änderungen auf abhängige Artefakte zu ermöglichen, werden die Relationen durch zwei Eigenschaften näher beschrieben. Die Stärke einer Relation spiegelt den Kopplungsgrad der Artefakte wider. Der Typ einer Relation dient der verbesserten Einordnung in einen semantischen Kontext, er liefert weitere Informationen die das Ausmaß der Auswirkung beschreiben.

.

Abbildung 3: Impact Management durch Definition von Relationen mit Eigenschaften

1.3 Kontext

Kompakt werden die wichtigsten Informationen als Kontext pro Artefakt zur Verfügung gestellt. Mit Hilfe des Kontexts lässt sich der Ablauf des gesamten Software Engineering Prozess verbessern, da eine Basis für die Kommunikation und somit die Arbeit im Projektverbund geschaffen wird. Der Kontext bildet die Integrationsschnittstelle für andere im SWE – Prozess verwendete Tools.

Abbildung 4: Kontext

2 Die Implementierung – Ein Überblick

Die Anwendung kann in 3 technologische Hauptkomponenten unterteilt werden, die sich wie folgt gliedern lassen: Das Versionierungssystem – Subversion - dient als Basis zur Speicherung und Verwaltung der Artefakte, die Datenbank (PostgreSQL) dient zur Speicherung der Relationen aus dem Impact Management, sowie weiterer beschreibender Daten zu den Artefakten. Die eigentliche Anwendung ist eine J2EE Applikation, diese beinhaltet die Businesslogik, die Anwendungslogik sowie die Präsentationsschicht. Zusätzlich wurde mittels Apache AXIS2 (SOAP Web Service Framework) ein „Context – Provider“ implementiert, um Fremdsystemen (z.B. CSCW, IDEs, etc.) einen Zugriff auf die zu einem Artefakt vorhandenen Daten, den Kontext zu ermöglichen.

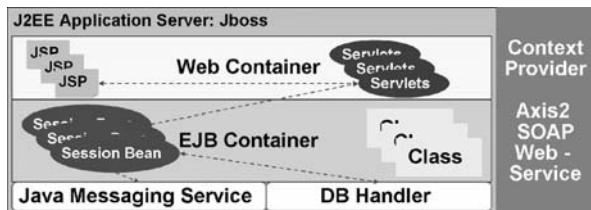


Abbildung 5: Anwendungsarchitektur

Literaturverzeichnis

- [TS05] Thomas Stark J2EE. Einstieg für Anspruchsvolle, München, Addison Wesley, 2005
- [LL06] Jochen Ludewig, Horst Lichter Software Engineering, Dpunkt, 2006
- [PG06] „PostgreSQL Database manuals“. URL:<http://www.postgresql.org/docs/manuals> [Stand: 19.01.2007]
- [JB06] “The JBoss 4 Application Server J2EE Reference”. URL: http://docs.jboss.com/jbossas/guides/j2eeguide/r2/en/html_single/ [Stand: 19.01.2007]
- [AX06] “Apache Axis2 Product Page”. ULR: <http://ws.apache.org/axis2/> [Stand 19.01.2007]
- [TM06] “TMate JavaDoc”. URL: <http://svnkit.com/kb/javadoc/index.html> [Stand:19.01.2007]
- [TM206] “TMate JavaDoc”. URL: <http://svnkit.com/kb/index.html> [Stand:19.01.2007]

Test von Serviceorientierten Architekturen (SOA)

Danilo Schmiedel, Marcel Matzat

Hochschule für Technik, Wirtschaft und Kultur Leipzig
Fachbereich Informatik, Mathematik und Naturwissenschaften
mmatzat@imn.htwk-leipzig.de
dschmied@imn.htwk-leipzig.de

Art der Arbeit: Beitrag zum Workshop „Serviceorientierte Architekturen“

Betreuer/in der Arbeit: Prof. Dr.-Ing. T. Kudraß

Abstract: Immer mehr Unternehmen entscheiden sich für die Strategie der Serviceorientierten Architektur (SOA). Ihr Ziel ist es, Produktivität und Wirtschaftlichkeit zu steigern sowie die IT-Infrastrukturen auf die Geschäftsstrategien abzustimmen. Für die erfolgreiche Umsetzung einer solchen Architektur ist es notwendig, die Aspekte der Interoperabilität, der Sicherheit, der Performance und der Benutzerfreundlichkeit zu überprüfen. Der Test einer SOA sollte sich über den gesamten Entwicklungsprozess erstrecken, die hierfür erforderlichen Testwerkzeuge sind im Idealfall zentralisiert bereitzustellen. Im Rahmen dieser Arbeit soll schließlich ein Testframework vorgestellt werden, welches versucht die oben genannten Forderungen zu erfüllen. Das Framework besteht aus den Komponenten: *CruiseControl* zur Testautomatisierung, *FitNesse* für die Testausführung und *Selenium* zur Überprüfung der grafischen Benutzerführung.

1 Testebenen

Der steigende Bedarf an flexiblen und offenen Softwarelandschaften in Unternehmen mit höchstmöglicher Flexibilität bei der Anpassung der Geschäftsprozesse an sich verändernde Marktbedingungen ist seit geraumer Zeit ein Grund für die Einführung von Serviceorientierten Architekturen. Basierend auf allgemein anerkannten Standards wie XML, Soap und WSDL verspricht SOA durch das Zusammenwirken von verschiedenen Komponenten und Services eine effiziente IT, welche alle wichtigen Geschäftsprozesse unterstützt sowie die Agilität von Unternehmen erhöht.

Das SOA-Paradigma [SD05] der lose gekoppelten miteinander agierenden Services führt sowohl zu einer Entflechtung monolithischer Softwaresysteme als auch zur Standardisierung von Schnittstellen heterogener Anwendungen. Auf der fachlichen Ebene erfolgt schließlich die Orchestrierung der einzelnen Services zu Geschäftsprozessen.

Die funktional voneinander unabhängigen Bausteine (Services) eines Geschäftsprozesses können zunächst getrennt voneinander getestet werden. Im Entwicklertest erstellt der Softwareentwickler die Testfälle, welche die funktionale Korrektheit des implementierten Services verifizieren. Entwicklertests lassen sich dabei in zwei Ebenen untergliedern: Tests auf Klassen- bzw. Unit-Ebene und Tests auf Modul-Ebene, die beiderseits durch eine Vielzahl bekannter Tools wie *JUnit* für Java oder *#Unit* für .NET unterstützt werden.

Das Zusammenspiel zwischen Services wird durch so genannte Integrationstests überprüft. Jedoch erfordert der hohe Anteil an Interaktionen zwischen einzelnen Services eine differenzierte Vorgehensweise im Vergleich zu Integrationstests für Einzelanwendungen. Die dezentrale Struktur und die Vielzahl an Services bedingen eine zentrale Bereitstellung der Testwerkzeuge sowie eine Unterstützung für die automatisierte Testdurchführung. Dabei umfasst der Begriff Testautomatisierung neben dem automatischen Testablauf auch eine automatische Initiierung, d.h. die Tests können zeit- oder ereignisgesteuert ausgeführt werden. Ein auslösendes Ereignis stellt beispielsweise die Übertragung eines geänderten Services in ein Versionsverwaltungssystem (*CVS*, *SVN*) dar. Auf der Ebene der Systemtests können die automatischen Überprüfungen schließlich die Grundlage für Service-Level-Agreements bilden. Eine frühzeitige Erkennung von Lastproblemen wird durch die Simulation realistischer Last-Szenarien ermöglicht.

Eine weitere Prüfebene repräsentieren die Akzeptanz-/Oberflächentests. Die Präsentationsschicht in serviceorientierten Architekturen wird meist durch eine Weboberfläche realisiert. Daher sollten entsprechende Testwerkzeuge neben den Entwickler- und Integrationstests zusätzlich webbasierte Oberflächentests unterstützen.

2 Testframework

Die Bedingungen, welche an ein Testframework in serviceorientierten Architekturen gestellt werden, sind vielfältig und erfordern eine offene und flexible Struktur. Im weiteren Verlauf dieser Arbeit wird ein entsprechendes Framework, bestehend aus den drei Komponenten *CruiseControl* zur Testautomatisierung, *FitNesse* für die Erstellung und Durchführung von Tests sowie *Selenium* für Oberflächentests, vorgestellt (siehe Abbildung 1).

2.1 CruiseControl

Das OpenSource-Produkt *CruiseControl* [CC07] ist eine Anwendung für "Continuous Integration" zur Automatisierung von Build- und Testprozessen in Softwareprojekten. Die Prozesse können entweder intervall-, zeit- oder ereignisgesteuert ausgelöst werden. Jeder Prozess entspricht einem Target innerhalb eines Ant-Files (Werkzeug zur automatisierten Erstellung von Softwarepaketen). Die Funktionalitäten eines Prozesses werden über Ant-Tasks definiert, welche wiederum den Aufruf der Unit- und *FitNesse*-Tests bewerkstelligen.

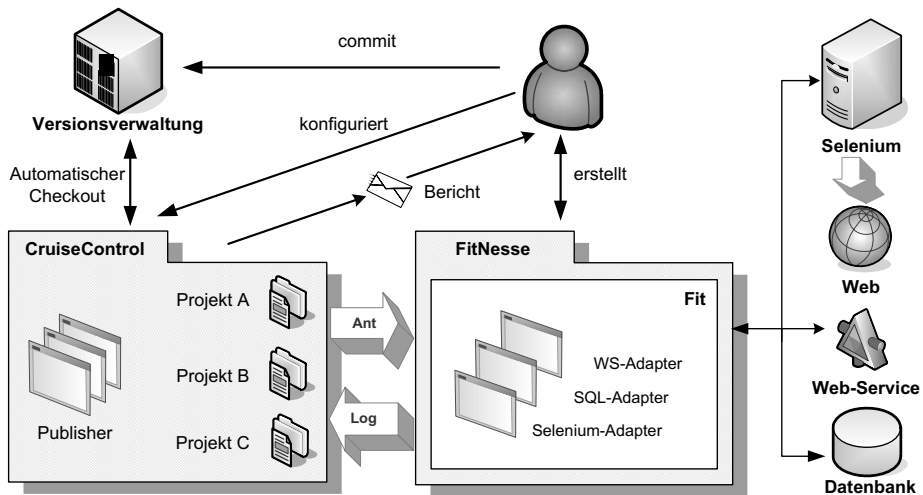


Abbildung 1: Testframework

Die Veröffentlichung der Ergebnisse erfolgt beispielsweise durch E-Mail-, FTP-, HTML- oder RSS-Publisher. Hierdurch ist etwa das Versenden von E-Mails an verschiedene Empfänger in Abhängigkeit eines Testergebnisses möglich. Ferner lassen sich die Publisher gruppieren und verschiedenen Bedingungen bzgl. des Testausgangs zuweisen. *CruiseControl* selbst wird zentral über eine Web-Schnittstelle administriert, welche das Hinzufügen und Modifizieren von Prozessen sowie die Überwachung verschiedener Prozess-Zustände ermöglicht.

2.2 FitNesse

Fit [Fit07] ist das Akronym für "Framework for Integrated Test" und wurde von Ward Cunningham (vgl. [CM05]) und James Shore entwickelt. Ursprünglich war *Fit* als Werkzeug für die Entwicklung von Akzeptanztests [Ad06] geplant, welche vorrangig aus den Vorgaben der Fachabteilung entstammen. Durch die einfache Integration benutzerdefinierter Adapter (Fixtures) ist es jedoch möglich, neben Akzeptanztests auch Integrationstests durchzuführen.

Die jeweiligen Tests werden in Form von HTML- oder Exceltabellen erstellt und anschließend von *Fit* ausgeführt. Tabelle 1 zeigt die Definition eines solchen Tests. In der ersten Kopfzeile wird die Fixture angegeben, welche die Prüfsätze an die entsprechenden Klassen, Module oder Services delegiert. In Kopfzeile zwei befinden sich jeweils Bezeichner für die Abbildung der Spalten auf die Variablen und Methoden des Adapters.

Die einzelnen Prüfsätze sind in den folgenden Datenzeilen enthalten, die je nach Ergebnis farblich hervorgehoben werden. Für den Fall, dass die eingetragenen Werte der Testperson von den Berechnungen des Adapters abweichen, erfolgt zusätzlich die Darstellung des korrekten Ergebnisses (siehe letzte Datenzeile).

Während die abgebildeten Testfälle in den Zeilen drei und vier der Tabelle 1 ein positives Ergebnis lieferten, ist das Resultat des dritten Prüfsatzes inkorrekt. Der Aufbau der Tabelle kann je nach verwendeter Fixture variieren.

example.SteuerToolAdapter		
einkommen	Steuerklasse	berechneSteuer()
25000	1	11250
50000	1	22500
25000	1	11251 / 11250

Tabelle 1: Fit-Test

Fit verfügt über drei Basis-Adapter: eine *ColumnFixture* für zeilenorientierte Integrationstests, eine *ActionFixture* für Akzeptanztests und eine *RowFixture* für Mengentests. Diese sind dahingehend erweiterbar, dass beispielsweise SQL-Queries oder Web-Services geprüft werden können. Ein Web-Service Adapter würde ein WSDL-File entgegen nehmen und daraus den clientseitigen Code generieren. Im Anschluss erfolgen nun die Tests der "Operations" eines Web-Services.

Ward Cunningham hat neben *Fit* auch an der Entwicklung von *FitNesse* [FN07] gearbeitet. Die Anwendung ist eine Wiki-basierende Administrationsoberfläche zur Erstellung, Modifikation und Ausführung von *Fit*-Tests. *FitNesse* ist hierarchisch organisiert und besitzt eine integrierte Versionsverwaltung. Innerhalb des zuvor beschriebenen *CruiseControl* erfolgt eine Integration von *FitNesse* mittels des Konsolenbasierten "TestRunner". Der Aufruf wird durch ein Ant-Task umgesetzt und mit dem *CruiseControl*-Publisher veröffentlicht.

2.3 Selenium

Das aus einem Testframework der Firma ThoughtWorks hervorgehende Selenium dient der Automatisierung von Oberflächentests für Webanwendungen. Der Testcode wird in Form von JavaScript-Code über einen Proxy in die jeweilige Web-Seite eingebettet. Selenium ist plattformunabhängig und mit allen gängigen Browsern kompatibel.

Das Testframework ist wiederum in drei Projekte untergliedert: *Selenium Core* für die JavaScript-Funktionalität, *Selenium IDE* zum Aufzeichnen der Benutzerinteraktionen und *Selenium RC* (Proxy-Server) zum Einbetten des JavaScript-Codes in die Web-Seite.

Neben der Möglichkeit, die Tests über plattformspezifische APIs zu definieren, ist es dem Benutzer ebenfalls gestattet, ein webbasiertes Prüfwerkzeug zu verwenden. Bei der Nutzung der zweiten Variante werden die Selenium-Tests in Form einer HTML-Tabelle erstellt. Dabei repräsentiert jede Zeile einen Fall mit je drei Spalten – Kommando, Ziel und Wert (siehe Tabelle 2). Für die Überprüfung stehen Kommandos für den Existenztest von Texten (*verifyTextPresent*), das Ausführen von Mausaktionen (*clickAndWait*) sowie das Ausfüllen von Formularen (*type*), zur Verfügung.

Die HTML-Elemente der zu testenden Seite können über XPath- oder DOM-Ausdrücke bzw. die ID oder den Namen referenziert werden. Die Spalte Wert enthält Parameter, die sich auf das jeweilige Kommando beziehen.

Command	Target	Value
open	/google.de	
clickAndWait	link=News	
type	name=q	Informatiktage
clickAndWait	name=btnG	
verifyTextPresent	Zukunft entwickeln	

Tabelle 2: Selenium-Test

Tabelle 2 zeigt insgesamt fünf Testfälle. Beginnend mit dem Öffnen der "google"-Seite, einem anschließenden Klick des News-Links, der Eingabe des Wertes "Informatiktage" in das Suchfeld "q" und der Bestätigung der Eingabe folgt abschließend die Überprüfung, ob der Text "Zukunft entwickeln" auf der Seite existiert.

Die Integration in *FitNesse* kann über einen Adapter erfolgen, welcher die Selenium-Befehle entgegennimmt, dann verarbeitet und darauf folgend das Ergebnis zurückliefert. In Kombination mit *CruiseControl* kann somit die Benutzerschnittstelle eines Web-Services automatisiert getestet werden.

Ein entscheidender Aspekt des vorgestellten Testframeworks ist, dass die Komponenten miteinander interagieren und nicht isoliert existieren. Dadurch kann ein durchgängiges Vorgehen erreicht werden, um letztendlich eine Erhöhung der Qualität bei gleichzeitiger Zeit- und Kostenersparnis zu erreichen.

Literaturverzeichnis

- [Ad06] Adersberger, J.: Wiki-getriebene Akzeptanztests, Java Magazin Ausgabe: 02.2006.
- [CC07] CruiseControl: Continuous integration toolkit,
<http://cruisecontrol.sourceforge.net>, 05.01.2007.
- [CM05] Cunningham, W.; Mugridge, R.: FIT for Developing Software, Prentice Hall International, 2005.
- [Fit07] Fit: Framework for Integrated Test, <http://fit.c2.com>, 06.01.2007.
- [FN07] FitNesse: The fully integrated standalone wiki, and acceptance testing framework,
<http://fitnesse.org>, 06.01.2007.
- [SD05] Richter, J.; George, T.; Gugel, T.; Heimann, T.; Lange, H.; Möllers, T: Technology Guide SOA,
http://www.sdm.de/web4archiv/objects/download/pdf/2/soa_technologyguide.pdf,
04.01.2007.

PED - Python External Decompiler

Jan C. Nordholz

17. Januar 2007

1 Motivation

Die TU Berlin hat vor etwa einem Jahr aus Teilnehmern der Vertiefungsveranstaltung für Rechnersicherheit und einem vorangegangenen Projekt, das sich mit Analyse und Test von aktuellen gängigen Firewallimplementationen beschäftigte, ein Team gebildet, das seitdem an internationalen Wettbewerben im Bereich Softwaresicherheit teilnimmt, die meist im sogenannten Capture-The-Flag-Modus ausgetragen werden. Hierbei ist das Ziel, in möglichst kurzer Zeit eine gegebene Liste von Services auf Schwachstellen zu untersuchen und daraufhin zeitgleich eine die Funktionalität des Services nicht beeinträchtigende Behebung der Lücke zu finden und einen vielversprechenden Angriffsvektor zu erzeugen, mit dem sich diese Lücke bei den konkurrierenden Teams ausnutzen läßt. Die Sprachen, in denen die Services vorliegen, decken dabei das ganze Spektrum ab, das ein aktuelles Unix-Derivat zu bieten hat: von Shellskripts zu Perl, Python, Ruby, C und natürlich auch meist ein Binary ohne Quellcode.

Im jüngsten Wettbewerb wurden wir von einer Reihe von Modulen überrascht, die als kompilierter Python-Bytecode vorlagen und mit den gängigen Bordmitteln nicht trivial zu dekompileieren, geschweige denn zu modifizieren waren. Eines der in der Nachbearbeitung als vorrangig eingestuften Ziele war also die Beschaffung oder Erstellung eines passenden Werkzeugs.

2 Vergleichbare bestehende Werkzeuge und ihre Defizite

Die Suche nach einem geeigneten Tool führte uns im Wesentlichen zu zwei Möglichkeiten: dem in Python integrierten Modul "dis" und dem separat erhältlichen Programm "decompyle". Beide wiesen jedoch Unzulänglichkeiten in Verfügbarkeit oder Funktionalität auf, so daß wir in jedem Fall in Eigeninitiative treten mußten. Wir entschieden uns nach Betrachtung der vorhandenen Tools für eine Eigenentwicklung, getauft PED[1].

2.1 Python:dis

Eigentlich bietet Python selbst die Möglichkeit, Bytecode in symbolischer Form auszugeben, in ein Array von Befehlen zu wandeln und somit auch modifizierbar zu machen – dazu dient das im Standardumfang enthaltene Modul "dis". Diese Herangehensweise hat nun aber den inhärenten Nachteil, daß das zu analysierende Modul vorher ebenfalls importiert worden sein muß; beim Import eines neuen Moduls liest das "marshal"-Modul die .PYC-Datei ein, wandelt die enthaltenen Daten in Objektstrukturen zurück und markiert gefundene Variablen und Codeblöcke entsprechend. Erst dann kann "dis" zu Werke gehen und den importierten Code analysieren und umwandeln. Module können nun aber (ähnlich dem __init-Hook in Linux DSOs) Code enthalten, der schon beim Laden des Moduls ausgeführt wird. Falls dieser Code eine blockierende Schleife enthält, läßt sich das Modul nicht mithilfe von "dis" dekompilieren, da der init-Code des Moduls die Kontrolle über den Programmfluß nicht mehr an den importierenden Codeblock abgibt.

2.2 decompyle

decompyle[2] war ursprünglich freier Code und ist daher aus dieser Zeit auch noch z.B. in der ein oder anderen Linux-Distribution[3] zu finden. Die Autoren bieten das Programm allerdings nicht mehr zum Download an und haben ihre Webseite jetzt in einen kommerziellen Service umgewandelt. Zudem kommt die Beschränkung hinzu, daß decompyle (selbst in der aktuellen Version, die auf der Webseite beworben wird) keinen Bytecode der Python-Versionen 2.4 und aufwärts analysieren kann – 2.4 ist aber dieser Tage schon praktisch Standard geworden. Insofern wäre selbst bei weiterhin freier Verfügbarkeit ohne eigene Initiative dieses Programm für uns nicht von Nutzen.

3 Herangehensweise

Mithilfe der verfügbaren Tabellen und Beschreibungen der Bytecode-Instruktionen[4] und einiger Zeit Reverse Engineering des .PYC-Formats war es möglich, ein C-basiertes Programm zu entwickeln, das die Python- Standardbibliothek der Versionen 2.3, 2.4 und 2.5 analysieren kann, d.h. für jede Bytecode-Datei den globalen init-Code extrahieren und die anderen Objekte (Klassen und globale Funktionen) in ihre Baumstruktur zurückverwandeln kann, nested functions bzw. "lambda expressions", wie Python sie in Anlehnung an die funktionale Programmierung nennt, eingeschlossen. Für diverse Operationen wie z.B. der Import von weiteren Modulen oder das Auslesen und Überschreiben von Variablen werden die symbolischen Namen bereits wieder eingesetzt.

```

000003f0 62 36 34 65 6e 63 6f 64 65 74 12 00 00 00 73 74 |b64encodet....st|
00000400 61 6e 64 61 72 64 5f 62 36 34 64 65 63 6f 64 65 |andard_b64decode|
00000410 74 11 00 00 00 75 72 6c 73 61 66 65 5f 62 36 34 |t....urlsafe_b64|
00000420 65 6e 63 6f 64 65 74 11 00 00 00 75 72 6c 73 61 |encodet....urlsa|
00000430 66 65 5f 62 36 34 64 65 63 6f 64 65 69 00 01 00 |fe_b64decodei...|
00000440 00 74 00 00 00 00 63 02 00 00 00 05 00 00 00 05 |.t....c.....|
00000450 00 00 00 43 00 00 00 73 4a 00 00 00 74 00 00 1e |...C...sJ...t...|
00000460 7d 04 00 78 2a 00 7c 01 00 69 03 00 83 00 00 44 |}...x*...i.....|
00000470 5d 1c 00 5c 02 00 7d 02 00 7d 03 00 7c 03 00 7c |]...\..3..3...|
00000480 04 00 74 06 00 7c 02 00 83 01 00 3c 71 14 00 57 |..t...|...<q..w|
00000490 7c 00 00 69 08 00 64 01 00 69 09 00 7c 04 00 83 |...i..d...i...|
000004a0 01 00 83 01 00 53 28 02 00 00 00 4e 52 0e 00 00 |....S(....NR...|
000004b0 00 28 0a 00 00 00 74 0c 00 00 00 5f 74 72 61 6e |.(....t...._tran|
000004c0 73 6c 61 74 69 6f 6e 74 0b 00 00 00 74 72 61 6e |slationt....tran|
000004d0 73 6c 61 74 69 6f 6e 74 08 00 00 00 61 6c 74 63 |slationt....altc|
000004e0 68 61 72 73 74 05 00 00 00 69 74 65 6d 73 74 01 |harst....itemst.|
000004f0 00 00 00 6b 74 01 00 00 00 76 74 03 00 00 00 6f |...kt....vt....o|
00000500 72 64 74 01 00 00 00 73 74 09 00 00 00 74 72 61 |rdt....st....tra|
00000510 6e 73 6c 61 74 65 74 04 00 00 00 6a 6f 69 6e 28 |nslatet....join(|
00000520 05 00 00 00 52 16 00 00 00 52 11 00 00 00 52 13 |...R....R....R.|
00000530 00 00 00 52 14 00 00 00 52 10 00 00 00 28 00 00 |...R....R....(|
00000540 00 00 28 00 00 00 00 74 09 00 00 00 62 61 73 65 |..(....t....base|
00000550 36 34 2e 70 79 74 0a 00 00 00 5f 74 72 61 6e 73 |64.pyt...._trans|
:

```

(a) Hexdump der Funktion base64:_translate

```

@0x0000007e [64] LOAD_CONST ""
@0x00000081 [5a] STORE_NAME "EMPTYSTRING"
@0x00000084 [64] LOAD_CONST (function *)
"base64.py": "_translate" 2:5:5:43 2:a:5:0:0
@0x00000000 [74] LOAD_GLOBAL "_translation"
@0x00000003 [1e] SLICE+0
@0x00000004 [7d] STORE_FAST "translation"
@0x00000007 [78] SETUP_LOOP (branch @ 0x00000034)
@0x0000000a [7c] LOAD_FAST "altchars"
@0x0000000d [69] LOAD_ATTR 3
@0x00000010 [83] CALL_FUNCTION (00/00)
@0x00000013 [44] GET_ITER
@0x00000014 [5d] FOR_ITER (branch @ 0x00000033)
@0x00000017 [5c] UNPACK_SEQUENCE 2
@0x0000001a [7d] STORE_FAST "k"
@0x0000001d [7d] STORE_FAST "v"
@0x00000020 [7c] LOAD_FAST "v"
@0x00000023 [7c] LOAD_FAST "translation"
@0x00000026 [74] LOAD_GLOBAL "ord"
@0x00000029 [7c] LOAD_FAST "k"
@0x0000002c [83] CALL_FUNCTION (01/00)
@0x0000002f [3c] STORE_SUBSCR
@0x00000030 [71] JUMP_ABSOLUTE 0x00000014
:

```

(b) Aufgelöster Bytecode derselben Funktion

Abbildung 1: Python 2.4 Bibliothek, base64.pyc

4 Planung und Ausblick

Mit dem gegenwärtigen Stand lassen sich die gesetzten Ziele (Dekompilieren, Analysieren, Modifizieren und Rückübersetzen von Python-Bytecode-Modulen) im Prinzip schon bewältigen. Es bieten sich nun zwei Wege an, die Entwicklung fortzusetzen: zum einen eine bessere Integration in die Python-Umgebung und automatisierte Unterstützung beim Verändern des Bytecodes (z.B. Anpassen der relativen und absoluten Sprungadressen), so daß sich der Code ohne zusätzliche manuelle Anpassungen schnell wieder in ein Modul zurückverwandeln läßt – zum anderen ist auch der Weg zur vollständigen Dekompilation nicht mehr weit. Dank des recht simplizistischen Sprachmodells scheint ein Ansatz mit bloßer Analyse der Stackinhalte gekoppelt mit einer kleinen Grammatik zur bequemen Auflösung von try-except-Blöcken sowie Schleifen schon sehr vielversprechend. Zudem fallen im aktuellen Zustand noch die in jedem PYC-Modul enthaltenen Debug-Informationen unter den Tisch, die ein Mapping zwischen Bytecode-Offset und Zeilennummer des originären Sourcecodes darstellen. Dies ließe sich entweder direkt als Anhaltspunkt bei der Dekompilation oder anschließend im Verifikationsschritt gewinnbringend nutzen.

5 Referenzen

- [1] PED - aktuelles Development:
<http://www.pool.math.tu-berlin.de/~hesso/ped/>
- [2] Decompyle Webseite:
<http://www.crazy-compilers.com/decompyle/>
- [3] Decompyle in Linux (z.B.):
<http://packages.debian.org/decompyle/>
- [4] Python Bytecode instructions:
<http://docs.python.org/lib/bytecodes.html>

Software-Industrialisierung: Auswirkungen der automatisierten Software-Erstellung auf die Arbeit der Entwickler

Anneke Pehmöller

Fachhochschule für Technik und Wirtschaft, Berlin

annekepehmoeller@gmx.de

Abstract: Im Rahmen der Software-Industrialisierung nimmt auch die Automatisierung der Erstellung von Anwendungen immer weiter zu. Dieser Artikel beschreibt einige Auswirkungen dieser Veränderungen auf die Arbeit und Ausbildung der Entwickler.

1 Motivation

Schon 1983 hat Balzer seine Vision des Automatic Programming Paradigm [BCG83] beschrieben. Doch selbst heute, über 20 Jahre später, sind viele der Ideen nicht oder nur ansatzweise verwirklicht. Natürlich hat sich seit den 80er Jahren sehr viel in der Software-Entwicklung verändert und viele Ansätze wurden umgesetzt, ausprobiert und einige haben sich als sinnvoll und praktikabel durchgesetzt. So zeigt sich die Software-Industrialisierung heute z.B. in der vermehrten Komponenten-Entwicklung und -nutzung, Tendenz zu einer Verringerung der Fertigungstiefe, Entwicklung und Verbreitung von Standards sowie der Erweiterung des Angebots durch ergänzende Dienstleistungen. [Ta05]

MDA [MDA], generative Programmierung [KC00] und Software Factories [Gr04] sind Konzepte, die heute in aller Munde sind. Die mit diesen Ansätzen verfolgten Ziele sind z.B. Entwickler in die Lage zu versetzen immer komplexere Systeme realisieren zu können, die Produktivität zu steigern sowie die Software-Entwicklung der einer industrialisierten Fertigung näher zu bringen.

Die automatisierte Umsetzung von System-Anforderungen in Quellcode, ist mittlerweile keine Seltenheit mehr. Zwar ist häufig noch die Integration verschiedener Tools, die für unterschiedliche Aufgaben (z.B. Versionskontrolle, Aufgabenverwaltung, IDE, Dokumentationserstellung, Projektmanagement) während des Software-Entwicklungsprozess genutzt werden, das Hauptproblem. Doch gibt es bereits Ansätze, in denen von der Modellierung der Spezifikation auf einer sehr abstrakten Ebene über die automatische Generierung komplette Anwendungen erstellt werden können. Die Produkte der Firma integranova [Integra] sind hierfür ein Beispiel.

Für die folgenden Ausführungen wird davon ausgegangen, dass in der Software-Entwicklung ein Werkzeug der folgenden Art eingesetzt wird: Es wird ein einziges Tool genutzt, in dem die Anforderungen an das zu realisierende System beschrieben werden. Diese Anforderungsbeschreibung ist die Basis für alle folgenden Schritte. Aus der Definition der grafischen Benutzeroberfläche wird dieses erstellt, aus der Beschreibung der Funktionalität wird der benötigte Code generiert. Die Dokumentation wird aus der Beschreibung der Anforderung bzw. deren grafischer Darstellung erstellt. Modul- und Integrationstests werden ebenso auf Basis der anfangs beschriebenen Spezifikation erstellt und automatisiert durchgeführt. Werden Änderungen erwünscht, müssen diese nur in den Anforderungen angepasst bzw. ergänzt werden. Es wird also davon ausgegangen, dass bereits weit entwickelte Tools eingesetzt werden, die den gesamten Prozess von der Anforderungsanalyse bis zur Inbetriebnahme der Applikation unterstützen.

Dieser Artikel beschreibt nun einige Auswirkungen des vermehrten Einsatzes von Code-Generierungstools und damit zusammenhängender Automatisierung der Software-Entwicklung auf die Tätigkeit und Ausbildung von Software-Entwicklern.

Was ändert sich an der Arbeit eines Software-Entwicklers?

Die Automatisierung wirkt sich auf den gesamten Software-Entwicklungsprozess aus: Der zu erbringende Aufwand verlagert sich von der Implementierungsphase hin zur Anforderungs-Aufnahme und Analyse-Phase. Es ist also eine Verschiebung des Einsatzes innerhalb des Gesamtprozesses zu erkennen. Für die Generierung von Benutzeroberflächen zeigt dies eine vergleichende Studie, die im Auftrag der CARE Technologies S.A. von der Gartner Group durchgeführt wurde [Mo03]. Die durch weniger bzw. wegfallende manuelle Implementierung frei werdenden Ressourcen können verstärkt eingesetzt werden, um an einer Optimierung in anderen Bereichen zu arbeiten. Durch Spezialisierung kann in Projekten besser, detaillierter und mit vertieften Domänenkenntnissen mit der Fachabteilung zusammen die Anforderungen ermittelt und dokumentiert werden.

Der gesamte Requirement Engineering Prozess ist sehr komplex und wissensintensiv und stellt einen kritischen Teil innerhalb der Realisierung eines Software-Projekts dar. Die Analyse-Phase mit Modellierung und Design erhält einen besonderen Fokus innerhalb des Software-Entwicklungsprozesses: Hier wird die Basis für die Generierung der Ergebnisse gelegt. Die ständige Verifizierung der geforderten Funktionalität des Endproduktes durch den Endanwender spielt eine herausragende Rolle. Dies führt zu einem iterativen Prozess in dieser Phase. Des Weiteren unterstützt Rapide Prototyping die Realisierung und hebt die Qualität des Ergebnisses, da neue oder veränderte Anforderungen direkt berücksichtigt werden können. Es wird sowohl die Funktionalität als auch die Benutzerfreundlichkeit des Systems sichergestellt.

Zuverlässiges Requirement Management macht eine separate Qualitätsprüfung durch Tests somit geradezu überflüssig, da die Funktionalität der Module bereits in den Anforderungen detailliert beschrieben werden und diese als Grundlage für die fehlerfreie Systemgenerierung dienen. Fehler, die Menschen z.B. bei der manuellen Implementierung machen, werden bei der Code-Generierung durch eine Maschine vermieden. [Ba98] Nötig sind gegebenenfalls Integrationstests, wenn die entwickelte Software Schnittstellen zu anderen Systemen nutzt.

Wird die Dokumentation komplett aus der Spezifikation generiert, fällt auch in diesen Bereich weniger Arbeitsaufwand und außerdem sinkt die Fehleranfälligkeit. Die Dokumentation bietet dann allen Projektbeteiligten jederzeit einen Einblick in das aktuelle Produkt bzw. den Projektstatus. Es ist kein zusätzlicher Arbeitsaufwand für die Erstellung von Dokumenten mehr nötig, es werden wiederum Kapazitäten frei. Des Weiteren würde eine Tätigkeit wegfallen, die von den meisten Entwicklern ungern erledigt wird. Dokumentation wird häufig nur als unnötige Pflicht und damit Zeitverschwendung angesehen, was zumeist daran liegt, dass die Dokumentation nicht aktiv genutzt wird. Der Nutzen der erstellten Dokumentation ist also nicht ersichtlich, wenn diese nicht aktiv genutzt wird, sondern nur wegen der „Vorschrift“ erstellt wird. Die Befreiung von so einer unangenehmen Tätigkeit führt nicht nur zu freien Kapazitäten sondern auch zu einer Motivationssteigerung.

Die Wartung, also Fehlerbehebung und vor allem die Erweiterung und Verbesserung des Systems wird durch die Automatisierung erheblich vereinfacht. Die Einarbeitung in und Ergänzung oder Veränderung von Code, der zumeist von anderen, eventuell schon nicht mehr in der Firma angestellten Kollegen, geschrieben wurde, fällt weg. Denn Änderungen sollten nicht mehr direkt im Quellcode eingebracht werden, sondern auf Spezifikations-Ebene, z.B. in den (grafischen) Modellen [MSUW04]: Die Anforderungen werden angepasst und die Applikation dann komplett oder ggf. einzelne Teile davon neu generiert.

Um sich darauf verlassen zu können, dass die Ergebnisse korrekt und von höchster Qualität sind und so die Wünsche der Endanwender erfüllen, müssen allerdings gute, integrierte Tools zur Verfügung stehen, die den kompletten Entwicklungs-Prozess unterstützen. Immer wieder bestätigt sich, dass das zu Grunde gelegte Material, also die Beschreibung der Anforderungen von sehr guter Qualität sein muss, da es als Basis für alle Ergebnisse dient. Dies verdeutlicht, dass sich die Tätigkeit der Software-Entwickler innerhalb des Softwareentwicklungs-Prozess fast ausschließlich auf die Requirement-Erfassung und das Design der Software konzentrieren und alle folgenden Phasen automatisiert ablaufen. Aber dies zeigt auch, dass viele Entwickler für die Realisierung der benötigten Tools benötigt werden. Informatiker, die sich auf die Entwicklung von Transformatoren und Code-Generatoren spezialisieren, werden ebenso gebraucht.

Was ändert sich für die Ausbildung der Entwickler?

Für die Software-Entwicklung werden also vermehrt Mitarbeiter benötigt, die neben gutem technischem Wissen u.a. im Bereich Requirement Management, Modellierung und SW-Architektur auch verstärkt folgende Kenntnisse und Fähigkeiten mitbringen:

- Ausgeprägte analytische Fähigkeiten für die Erfassung der Anforderungen
- Abstraktes Denkvermögen, um in der Lage zu sein, die Anforderungen auf einer abstrakten Ebene zu modellieren. Dazu gehört auch plattform- und programmiersprachen-unabhängiges Denken. [MSUW04]
- Gute Kommunikationsfähigkeit und Kenntnisse über z.B. Interview-Methoden, um mit den Anwendern die Funktionalität erfolgreich zu erfassen.
- Spezialisierung auf Fachbereiche, um noch besser auf die Bedürfnisse der Kunden und die jeweilige Domäne eingehen zu können.

Fazit

Die fortgeschrittene Automatisierung der Software-Entwicklung bedeutet vor allem eine Verlagerung des Schwerpunkts innerhalb des Realisierungs-Prozesses. Die Tätigkeit der Software-Entwickler ist daher auf einer viel höheren, abstrakteren Ebene anzuordnen und nicht mehr auf der Ebene der tatsächlichen Implementierung (Schreiben von Quellcode). Der Fokus auf die Spezifikation- und Design-Phase bringt die beschriebene Veränderung in dem Berufsbild eines Software-Entwicklers mit sich. Des Weiteren wird klar, dass sich nicht nur Tools und Methoden ändern müssen, sondern sich die gesamte Einstellung zum Software-Entwicklungsprozess und der damit verbundenen Arbeit auf Grund der Automatisierung anpassen muss. Auf diese Veränderungen sollten sich sowohl Universitäten und Hochschulen mit ihrem Angebot einstellen als auch Unternehmen in ihren Mitarbeiter-Qualifizierungsmaßnahmen.

Literaturverzeichnis

[BCG83] Robert Balzer, Thomas E. Cheatham, Cordell Green: Software Technology in the 1990's: Using a New Paradigm in IEEE Computer, November 1983

[Ba98] Robert Balezzer: Supporting Product Line Development in Proceedings of the 10th International Software Process Workshop, IEEE Computer Society, 1998

[CE00] Krysztof Czarnecki, Ulrich W.Eisenecker: Generative Programming. Methods, Tools, and Applications, Addison-Wesley Longman, Amsterdam, 2000

[Gr04] Jack Greenfield: Softwarefabriken: Die Montage von Anwendungen mit Hilfe von Mustern, Modellen, Frameworks und Werkzeugen in ObjektSpektrum, 4/2004

[In03] Emilio Insfrán Pelozo: A Requirements Engineering Approach for Object-Oriented Conceptual Modeling, Doktorthese, Universidad Politécnica de Valencia, Valencia, Espana, Oktober 2003

[Integra] <http://www.integranova.de/>

[MDA] <http://www.omg.org/mda/>

[Mo03] Pedro Juan Molina Moreno: Especificación de interfaz se usuario: De los requisitos a la generación automática, Doktorthese, Universidad Politécnica de Valencia, Valencia, Espana, März 2003

[MSUW04] Stephen J. Mellor, Kendall Scott, Axel Uhl, Dirk Weise: MDA Distilled: Principles of Model-Driven Architecture, Addison-Wesley Longman, Amsterdam, 2004

[Ta05] Dirk Taubner: Software-Industrialisierung in Informatik Spektrum August 2005, siehe http://www.sigs.de/publications/os/2004/04/greenfield_OS_04_04.pdf

Pair Programming: Implikationen der industrialisierten Softwareentwicklung auf die universitäre Informatikausbildung

Wolfgang Reinhardt

Universität Paderborn
Institut für Informatik
wolle@uni-paderborn.de

Abstract: Die Informatikbranche steht in den nächsten Jahren einem grundlegendem Paradigmenwechsel gegenüber. Traditionelle Konzepte der Softwareentwicklung werden zunehmend durch automatisierte Ansätze abgelöst. Viele datenbankgestützte Anwendungen werden heute bereits mit MDA-basierten Prinzipien zu großen Teilen automatisiert erzeugt; in anderen Bereichen der Softwareentwicklung folgt man dem Trend zur Industrialisierung der Softwareentwicklung. Einhergehend mit der zunehmend industrialisierten Art der Softwareentwicklung kristallisieren sich in den letzten Jahren vor allem auch agilere Methodiken und eine zunehmende Fokussierung des Menschen im Softwareentwicklungsprozess heraus. Diese vornehmlich wirtschaftlichen Entwicklungen fordern von Studienabgängern neben den bekannten Kompetenzen im fachlich-methodischen Bereich vor allem auch besondere persönliche und soziale Kompetenzen. Die moderne universitäre Informatikausbildung sollte auf diese real-wirtschaftlichen Anforderungen eingehen und die teils antiquierten Wege zur Vermittlung von Softwareentwicklung überholen.

1 Status Quo

Softwareprojekte unterliegen sowohl im wirtschaftlichen wie auch im universitären Umfeld ständigem Wandel. Die zentralen Fragen der modernen Softwareentwicklung lauten:

1. Wie erhalte ich bessere Software?
2. Wie passe ich meinen Entwicklungsprozess an den ständigen Wandel an?

Unter wirtschaftlichen Gesichtspunkten muss zusätzlich nach einer möglichst effizienten Verteilung der zur Verfügung stehenden menschlichen Arbeitskraft; im wissenschaftlich-didaktischen Umfeld nach einer nachhaltigen Vermittlung grundlegender Theorien und aktueller Entwicklungen gefragt werden.

Während die Wirtschaft seit Jahren nach theoretisch wie praktisch hoch qualifizierten Fachkräften ruft (vgl. auch [DH06]), ist die Praxis der Softwaretechnik sowohl in den althergebrachten Diplom-, wie auch den recht neuen Bachelor- und Masterstudiengängen

zumeist auf einige wenige praktische Kernprojekte begrenzt (siehe z. B. [Uni04]). Grundlegende Programmierparadigmen und -konzepte werden den Studierenden in der Regel in einführenden Veranstaltungen zur Softwareentwicklung vermittelt; die Praxis dergleichen durch kleine Programmieraufgaben, die in Heimarbeit oder Übungsgruppen zu erstellen sind. Bis auf wenige Ausnahmen wird so das Bild des zurückgezogenen und allein entwickelnden Programmierers vermittelt, das in der Realität der wirtschaftlichen Softwareentwicklung längst veraltet und durch vielfältige Teamarbeit ersetzt ist.

2 Pair Programming in der Übersicht

Pair Programming (PP) ist ein Konzept der gemeinsamen softwaretechnischen Arbeit und in der industriellen Softwareentwicklung bereits seit Jahren wohl bekannt und eingesetzt [WK02, WK03]. Unter wissenschaftlichen und didaktischen Aspekten gewann PP durch seine feste Integration in die agile Methode Extreme Programming (XP) (vgl. [Bec99]) an Bedeutung und zählt mittlerweile zu einem der meist untersuchten agilen Teilprozesse.

In klassischen Projekten arbeiten Entwickler zusammen in einem Team, dessen Zusammenarbeit sich zumeist auf das Festlegen von Schnittstellen und Koordinierung von gemeinsamen Terminen beschränkt. Die eigentliche Implementierung wird allein – im so genannten *Solo Programming* – durchgeführt. Nur in Einzelfällen, wie dem Debuggen komplexer Komponenten, wird ein zweiter Entwickler hinzugezogen. PP beschränkt die Kollaboration der Entwickler nun nicht auf Einzel- und Ausnahmefälle, sondern macht sie zur Regel. Beim Pair Programming stellen zwei Entwickler gemeinsam ein Artefakt (Design, Algorithmus, Code u. a.) her. Nach außen bilden die beiden Entwickler einen zusammenhängenden Organismus, der mit einem gemeinsamen Verstand arbeitet und gemeinsam für jeden Aspekt des Artefakts zuständig ist. Ein Entwickler ist der **Driver** und hat die Kontrolle über Keyboard und Maus und entwickelt aktiv am vorliegenden Artefakt. Der andere Entwickler ist der **Observer** und denkt strategisch und langfristig – er behält das Ziel im Auge und überprüft, ob dieses auf dem momentan eingeschlagenen Weg erreichbar ist. Der Observer überprüft, ob tatsächlich die einfachste mögliche Lösung implementiert wird und nur das, was im Moment benötigt wird. Zu diesem Zweck verfolgt der Observer genau, was der Driver tut und hilft ihm u. a. bei der Einhaltung von Codestandards und dem Beseitigen von Tipp- oder Syntaxfehlern. Durch das unmittelbare Feedback zwischen den beiden Partnern wird Missverständnissen vorgebeugt und ein gleich verteiltes Wissen aufgebaut (vgl. u. a. [WK03]). Der Driver erklärt jeden Schritt, den er macht – sein Partner stellt gegebenenfalls Zwischenfragen, bringt sein Wissen und seine Erfahrung ein und macht Vorschläge zur Verbesserung. In regelmäßigen Abständen – in der Regel im Abstand von einer Stunde – tauschen die beiden Entwickler die Rollen. Der Driver wird somit zum Observer und überwacht nun seinerseits die Arbeit und gibt Anmerkungen und Verbesserungen und umgekehrt. Beide Partner sind unabhängig von ihrem Alter oder den eigenen Programmierfähigkeiten vollkommen gleichberechtigt und zu jeder Zeit in den Entwicklungsprozess eingebunden.

Als erfolgreiche industrielle Praxis führt Pair Programming unter anderem zu besserer Intra-Team-Kommunikation und somit direkt zu besserer Codequalität, weniger Fehlern

und zu leichter wartbaren Softwareprodukten. Eine Vielzahl internationaler Studien widmete sich bereits der Praktik der paarweisen Softwareentwicklung und empfiehlt den Einsatz vor allem in universitären Kontexten (siehe dazu auch [Rei06, S. 83ff.]).

3 Implikationen auf die universitäre Informatikausbildung

Die primären Ziele agiler Methoden und der konsequent paarweisen Arbeit im Pair Programming haben vor allem zunächst die Produkt- und Prozessverbesserung zum Ziel. Die beteiligten Entwickler sollen durch die angewandten Techniken der agilen Methoden aber auch mehr Freude am Prozess der Softwareentwicklung empfinden, sich so stärker mit dem Produkt identifizieren und so zu besseren Ergebnissen gelangen. Die Informatikausbildung an Hochschulen dient der Herausbildung und Verstärkung interdisziplinärer Kompetenzen bei den Studierenden (vgl. auch [Uni04]) und muss daher vor allem die subjektive Seite des Lernenden betrachten. In der didaktisch-methodischen Betrachtung des Pair Programming muss die Frage beantwortet werden, ob ein Student dauerhaft etwas aus der praktischen Erfahrung gelernt hat, dies seine subjektive Kompetenzausprägung beeinflusst und der gesamte Prozess durch einen anderen Methodeneinsatz verbessert werden kann.

Betrachtet man speziell die vielfältig geforderten *Soft Skills* und ihre zu verstärkende Vermittlung in der universitären Lehre, so zeigen sowohl die theoretischen Ideen als auch die empirischen Studien, dass Pair Programming neben methodischen vor allem soziale Kompetenzen verstärkt herausbildet. Selbstvertrauen, Überzeugungskraft, Kommunikationsfähigkeit und Verantwortungsgefühl gegenüber dem Team sind nur einige der Eigenschaften, die Entwickler durch konsequente Paararbeit verstärkt einsetzen und trainieren. Methodisch vermittelt Pair Programming umfassende Fähigkeiten in der Analyse vorliegender Problemsituationen und der Entwicklung ausgereifter Lösungsansätze. Zusammenfassend lässt sich belegen, dass speziell Sozial-, Personal- und Methodenkompetenz der Beteiligten durch Pair Programming stärker ausgeprägt werden.

4 Zusammenfassung

Die Praxis der paarweisen Softwareentwicklung entstammt der wirtschaftlichen Realität und hat sich in vielen verschiedenen Kontexten etabliert. In der universitären Vermittlung von Softwareentwicklung werden bereits Teile der Projektarbeit im Team durchgeführt; die eigentliche Programmierung wird jedoch klassischer Weise als Solo-Aktivität gelehrt. Die universitäre Informatikausbildung sollte sich die Synergieeffekte des Pair Programming (siehe [Rei06, S. 58ff.]) zu Nutze machen und so hochwertigere Softwareprodukte erwarten können. Diese sind mit Entwicklungsprozessen verbunden, die Softwareentwicklung besser vermitteln können, als jede Grundlagen- oder Theorieveranstaltung.

Die wiederkehrende Forderung obligatorischer Praxisprojekte in der universitären Ausbildung sollte gerade in der Informatik gut umsetzbar sein. Als Praxis darf dabei jedoch nicht ausschließlich die Anwendung theoretisch vermittelten Wissens verstanden, sondern

vor allem die Integration in real-wirtschaftliche Projekte mit interdisziplinären Bezügen verstanden werden. Das *Dagstuhl-Manifest zur strategischen Bedeutung des Software Engineering in Deutschland* fordert:

„Die universitäre Lehre muss international wettbewerbsfähig sein. Sie sollte international zusammenarbeiten. Trotz Grundlagenorientierung muss auch auf die gegenwärtige Praxis vorbereitet werden“ [CEG⁺06, S. 11].

Sollte die Integration in echte industrielle Projekte nicht realisierbar sein, so bilden die aus den Wirtschaftswissenschaften bekannten Planspiele einen Ansatz, um die Realitätsnähe in der universitären Softwaretechnik zu erhöhen und eine noch bessere Ausbildung beruflicher Handlungskompetenz bei den Studenten zu erwirken.

Literatur

- [Bec99] Kent Beck. *Extreme Programming Explained. Embrace Change*. Addison-Wesley, 1999.
- [CEG⁺06] A. Cremers, J. Ebert, S. Glesner, M. Glinz, M. Goedicke, G. Goos, V. Gruhn, W. Hasselbring, S. Jähnichen, S. Kowalewski, B. Krämer, S. Leue, C. Lewerentz, P. Liggesmeyer, C. Lüth, B. Paech, H. Partsch, I. Philippow, L. Prechelt, A. Rausch, W.-P. de Roever, B. Rumpe, G. Rünger, W. Schäfer, K. Schneider, A. Schürr, W. Tichy, B. Westfechtel, W. Zimmermann und A. Zündorf. *Dagstuhl-Manifest zur strategischen Bedeutung des Software Engineering in Deutschland*. In M. Broy, M. Jarke, M. Nagl und D. Rombach, Hrsg., *Perspectives Workshop - Dagstuhl Seminar Proceedings*, Schloss Dagstuhl, 2006.
- [DH06] Holger Dambeck und Tim Höfinghoff. *Fachkräftemangel: Deutsches IT-Gejammer 2.0*. <http://www.spiegel.de/wirtschaft/0,1518,455278,00.html>, Dezember 2006.
- [Rei06] Wolfgang Reinhardt. *Einfluss agiler Softwareentwicklung auf die Kompetenzentwicklung in der universitären Informatikausbildung. Analyse und Bewertung empirischer Studien zum Pair Programming*. Diplomarbeit, Universität Paderborn, 2006.
- [Uni04] Universität Paderborn. *Konzeption der Bachelor- und Masterstudiengänge Informatik an der Universität Paderborn*. <http://www.cs.uni-paderborn.de/cs/studium/konzeptionbm.html> eingesehen am 24. Juli 2006, Oktober 2004.
- [WK02] Laurie A. Williams und Robert R. Kessler. *Pair Programming: Experience the Difference*. In D. Wells und L. Williams, Hrsg., *Proceedings of the XP/Agile Universe Conference 2002*, Jgg. 2418 / 2002, Seiten 271 – 272. Springer Berlin / Heidelberg, 2002.
- [WK03] Laurie A. Williams und Robert R. Kessler. *Pair Programming Illuminated*. Addison-Wesley, Boston, MA, USA, 2003.

Entwicklung einer Verkaufsförderungsplattform für integrierte Promotions

M. Bednarz, J. Bienemann, A. Christ, S. Kather, N. Kelleter, D. Pfeiffer,
M. Röttgerding, D. Tiebe, J. Tomasoni, N. Vehring

Westfälische Wilhelms-Universität Münster
Institut für Wirtschaftsinformatik
info@muenster-vip.de

Art der Arbeit: Projektseminar (studentisches Projekt im Rahmen des Hauptstudiums)
Betreuer der Arbeit: Prof. Dr. Jörg Becker, Dr. Axel Winkelmann, Daniel Beverungen,
Christian Janiesch

Abstract: Das Projektseminar „V.I.P.“ (Verkaufsförderungsplattform für integrierte Promotions) verfolgt als Ziel die Entwicklung einer integrierten Plattform für den Austausch von Promotionstammdaten. Die Plattform soll insbesondere Coupon-Aktionen unterstützen, jedoch auch für weitere Maßnahmen der Verkaufsförderung nutzbar gemacht werden können.

1. Motivation einer Verkaufsförderungsplattform

Zur Steigerung des Umsatzes, zur Gewinnung von Neukunden und zur Intensivierung der Kundenbindung wird im Handel unter anderem auf die Durchführung von *Verkaufsförderungsaktionen* (Promotions) zurückgegriffen [Be04]. Eine Ausprägung solcher Promotions ist das *Couponing*. Hierbei werden von einem Hersteller Gutscheine (Coupons) an Kunden verteilt, die diese zu einem preisreduzierten oder unentgeltlichen Erwerb eines oder mehrerer Produkte berechtigen [Be03]. Damit der Handel diese Coupons einlösen kann, muss er über die mit der Promotion verbundenen Daten verfügen. Neben allgemeinen Daten, wie z. B. Zeitraum und Höhe des Rabattes, benötigt der Handel eine Zusammenstellung der von einer Promotion betroffenen Artikel in Form von EAN-Listen (International Article Number). In der Regel handelt es sich hierbei um Artikel einer bestimmten Marke oder einer bestimmten Produktgruppe des Herstellers. Die Problematik bei der Zusammenstellung dieser Listen besteht darin, dass die EANs eines Herstellers nicht hierarchisch aufgebaut sind und somit nicht automatisch zusammengestellt werden können. Ein weiteres Problem resultiert daraus, dass während der Laufzeit einer Promotion neue Artikel auf den Markt kommen können, die diese Promotion betreffen.. Dementsprechend müssen die EAN-Listen in den Kassensystemen der Händler aktualisiert werden, um eine korrekte Einlösung sicherzustellen. Dies setzt jedoch voraus, dass die neuen Artikel auch als solche identifiziert und den richtigen Promotions zugeordnet werden.

Der Austausch der Daten findet in der Regel in einem bilateralen Abstimmungsprozess zwischen dem Hersteller und dem Handel statt. Um diesem erhöhten Kommunikationsaufwand begegnen zu können, fehlt es aktuell an einer zentralen Instanz, über die alle relevanten Daten zusammengestellt und ausgetauscht werden können.

Ziel des Projektes ist die Konzeptionierung und prototypische Entwicklung einer Verkaufsförderungsplattform für integrierte Promotions (V.I.P.), insbesondere für das Couponing. Die für die Plattform gestalteten Prozesse sind so ausgerichtet, dass sie eine Senkung des Abstimmungsaufwands zwischen Industrie und Handel bei der Planung, Durchführung und Auswertung von Verkaufsförderungsaktionen bewirken. Ermöglicht wird dies dadurch, dass benötigte Daten multilateral abgestimmt und bezogen werden können. Auf technischer Ebene spiegelt sich dies unter anderem in der Erhöhung der Datenqualität und der automatisierten Anbindung an bestehende Systeme, z. B. Kassen- oder Warenwirtschaftssysteme, wider.

2. Konzeptionelle Beschreibung der Verkaufsförderungsplattform

Einen Überblick über die konzipierte Verkaufsförderungsplattform ermöglicht der nachfolgende Ordnungsrahmen. Er dient als Grundlage für die daraus abgeleitete Software-Architektur der Plattform.

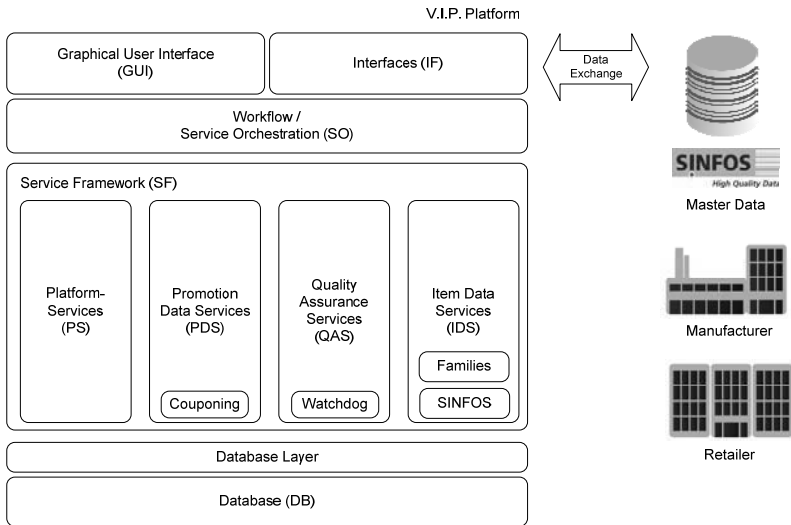


Abbildung 1: Ordnungsrahmen für eine integrierte Verkaufsförderungsplattform

Dem Entwurf liegt eine dienstorientierte Sichtweise zugrunde (sog. serviceorientierte Architektur, SOA). Der Ausgangspunkt einer SOA ist eine Ansammlung von Diensten.

Innerhalb dieser Dienste werden generische und wieder verwendbare Applikations-, Daten- oder Prozesslogiken hinter einer eindeutig definierten Schnittstelle gekapselt. Über diese Schnittstelle können andere Applikationen auf die Funktionen dieses Dienstes zugreifen [Ga05].

Die wesentlichen Dienste im Kern der Verkaufsförderungsplattform sind innerhalb eines *Service Frameworks (SF)* angesiedelt. Das Service Framework bietet Infrastrukturdienste, die die lose Kopplung von Diensten ermöglichen. Weiterhin bietet es vielfältige Möglichkeiten, Dienste zu veröffentlichen. Beispielsweise ist es denkbar, dass einzelne Dienste als Komponenten einer J2EE-Architektur oder als standardisierte Webservices angesprochen werden können. Moderne Paradigmen solcher Frameworks, z. B. aspektorientierte Programmierung, erlauben eine Konzentration auf die Funktionalität eines jeden einzelnen Dienstes, ohne dessen spätere Verwendung im Voraus einkalkulieren zu müssen.

Bei Verkaufsförderungsaktionen, insb. beim Couponing, stehen zu bewerbende Artikel im Mittelpunkt. Daraus ergibt sich für eine Verkaufsförderungsplattform die Notwendigkeit, entsprechende Funktionen zur Verwaltung und Pflege von Artikelstammdaten bereitzustellen. Auf der Verkaufsförderungsplattform geschieht dies im Rahmen der *Item Data Services (IDS)*. Neben entsprechenden Funktionen zur Verwaltung und Anzeige von Artikelstammdaten wird die effiziente Suche innerhalb von Stammdaten realisiert. Die Artikelstammdaten werden dabei aus dem Stammdatenpool *SINFOS* bezogen und in die Plattform importiert. Der SINFOS Stammdatenpool ist ein de-facto-Standard zum Austausch von Artikelstammdaten zwischen Industrie und Handel. Dieser Stammdatenpool stützt sich auf international standardisierte Inhalte, Prozess-Regeln und Funktionen und ermöglicht damit einen globalen Stammdatenaustausch [St04].

Für die Einlösung bzw. automatische Ausgabe von Coupons am Kassensystem sind Bedingungen zu hinterlegen, die sich auf die Identifikation von Artikeln stützen. Die für die Einlöse- und Ausgabebedingungen benötigten EANs müssen klassischerweise für jede Coupon-Aktion manuell aufbereitet werden. Diese Aufgaben sind nicht nur zeit- und damit kostenintensiv, sondern bringen auch Fehlerquellen mit sich. Auf der Verkaufsförderungsplattform werden diese Aufgaben durch die hierarchische Strukturierung in Artikelfamilien (*Families*) unterstützt. Die Generierung geeigneter Abfragen auf diverse Artikelattribute resultiert in der schrittweisen Verfeinerung der Artikelbasis. Eine Ausprägung einer Verfeinerung ist bspw. die Untergliederung in herstellerbezogene Markenhierarchien. Der Anwender wird bei der Erstellung der Hierarchien durch statistische Analysen auf der Artikelbasis unterstützt.

Neben den Item Data Services müssen entsprechende Funktionen zur Pflege der einzelnen Promotions vorhanden sein. Sie werden unter dem Begriff der *Promotion Data Services (PDS)* zusammengefasst und dienen im Wesentlichen der Pflege von Promotiondaten, wozu die Festlegung der zuvor beschriebenen Ausgabe- und Einlösebedingungen zählt. Darüber hinaus erfolgt in diesem Dienst die Koordination des Abstimmungsprozesses zur Durchführung einer Promotion zwischen Hersteller und teilnehmenden Händlern.

Funktionen zur Qualitätssicherung der Stammdaten werden im Dienst *Quality Assurance Services (QAS)* zusammengefasst. Ein wesentlicher Baustein ist hierbei die Identifikation neuer EANs, mittels eines sog. *Watchdogs*, und die anschließende Generierung von Vorschlägen für mögliche Anpassungen von aktuellen und geplanten Promotions. Grundidee bei der Generierung ist die Berechnung eines Ähnlichkeitsmaßes, dass die Ähnlichkeit des neuen Artikels mit den entsprechenden Promotions ausdrückt.

Im Rahmen der *Plattform Services (PS)* werden typischen Bestandteile einer Plattform abgebildet, die zu deren Betrieb notwendig sind. Hierzu zählt u. a. ein Rollenkonzept mit integrierter Benutzerverwaltung.

Auf den Diensten aufbauend bietet die Plattform Schnittstellen zur externen Kommunikation an. Zunächst zu nennen ist hier die graphische Benutzeroberfläche (*GUI*), sie ist als Webanwendung konzeptioniert und ist mit Hilfe modernen Web-Technologien umgesetzt. Des Weiteren existieren Schnittstellen (*Interfaces (IF)*) für den Import und Export von Daten. Zum einem können hier die bereits angesprochenen Artikelstammdaten aus dem SINFOS Stammdatenpool importiert werden und zum anderen wird, um eine möglichst generische Anbindung der Plattform an die Systeme von Händlern und Herstellern zu gewährleisten, eine XML-Schnittstelle bereitgestellt über die Promotionsdaten ausgelesen und geschrieben werden können.

3. Stand der Entwicklung

Die Durchführung des Projektseminars erfolgte in Zusammenarbeit mit dem Clearing-Haus *acardo technologies AG* in Dortmund und dem Institut für Wirtschaftsinformatik der Westfälischen Wilhelms-Universität in Münster und erstreckte sich über einen Zeitraum von sechs Monaten. Der praxisnahen Erhebung der Anforderungen an eine solche Plattform mittels Interviews und Prozessanalysen folgten die Erstellung eines Fachkonzepts sowie eine prototypische Realisierung. Das Projekt wird auf der CeBIT 2007 auf dem Gemeinschaftsstand der Hochschulen und Forschungseinrichtungen aus NRW präsentiert.

Literaturverzeichnis

- [Be03] Becker, J.; Vering, O.; Winkelmann, A.: *Couponing & Coupon-Clearing in Deutschland. Grundlagen und Konzepte. Handelsstudie 1.* Münster 2002, aktualisierte Version Februar 2003.
- [Be04] Becker, J.; Schütte, R.: *Handelsinformationssysteme.* 2. Aufl., Frankfurt am Main 2004.
- [Ga05] Gallas, B. E.; Schönherr, M.: *Service Management als Grundlage Service Orientierter Architekturen – Ein SOA-Testszenario.* In: *Enterprise Architecture*, Band 3. Hrsg.: S. Aier, M. Schönherr. Berlin 2005, S. 221-245.
- [St04] Stelzer, J.; Marlow, S.: *Datensynchronisation verstehen und implementieren.* Executive Guide, Version 2.1 – August 2004. Sterling Commerce. Düsseldorf 2004.

Towards a Metrics Suite for Object-Relational Mappings

Stefan Holder

Hochschule Fulda
Fachbereich Angewandte Informatik
s.holder@gmx.de

Art der Arbeit: Diplomarbeit

Betreuer der Arbeit: Prof. Stephen G. MacDonell, PhD; Prof. Dr. Peter Peinl

Abstract: The use of object-relational middleware requires the developer to specify the mapping between object model and relational schema. Supporting the developer in this considerably complex task via mapping tools is the subject of ongoing research and practice. In this paper, the foundation of a metrics suite for object-relational mappings is presented. These metrics are considered to provide a valuable means of improving mapping tool support.

1 Introduction

Object-relational (O/R) middleware is frequently used in practice to bridge the semantic gap (the so-called impedance mismatch) between object-oriented application systems and relational database management systems (RDBMSs). In this middleware approach, the transformation between objects and relations is implemented generically. This approach achieves low coupling between application and relational schema and makes it possible to change the relational schema without the need to also change the application source code.

If O/R middleware is employed, the object model needs to be linked to the relational schema. Following the so-called forward engineering approach, the developer creates the object model first and then creates the relational schema by mapping the object model to relations. Thereby, the developer is faced with the challenge to choose from a variety of mapping strategies for class associations and inheritance relationships. These mapping strategies have different impacts on the non-functional properties of application systems such as performance or maintainability [Ke97; Ph05].

Quantifying these mapping impacts via metrics is considered beneficial in the context of O/R mapping tools since these metrics enable an automated and differentiated consideration of O/R mapping impacts. This paper provides the foundation for such a metrics suite. It does so by (i) setting goals of measurement; (ii) outlining possible applications of metrics in O/R mapping tools; (iii) suggesting a novel inheritance mapping model; and (iv) proposing an initial metrics suite for inheritance mappings.

2 Measurement Goals

In order to set measurement goals, a new network of quality characteristics of O/R mappings is proposed (see Figure 1). This network extends the plain classifications of O/R mapping impacts suggested by [Ke97; Ph05]. A subset of the software quality characteristics defined by the ISO/IEC 9126 standard [ISO01] (shown as ellipses) is thereby refined by specific quality characteristics of O/R mappings (shown as rectangles).

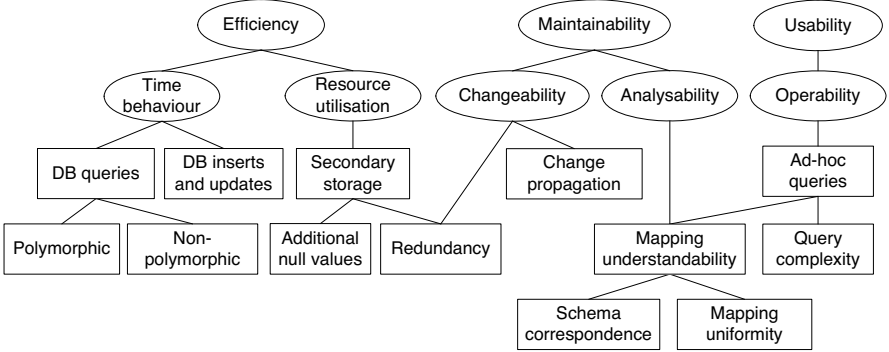


Figure 1: Quality characteristics of O/R mappings

3 Applications of Metrics in Object-Relational Mapping Tools

In order to automate O/R mapping, tools employed in practice typically use only a single inheritance mapping strategy and apply a fixed mapping strategy for one-to-one, one-to-many, and many-to-many associations, respectively [KJK93]. This approach is straightforward; however, the different impacts of alternative mapping strategies are thereby not considered.

Philippi [Ph05] therefore suggests a model-driven generation of O/R mappings. In this approach, the developer defines mapping requirements in terms of quality trade-offs. The mapping tool then automatically selects mappings that fulfil the specified requirements based on general rules of thumb regarding the impacts of mapping strategies. However, mapping impacts strongly depend on concrete object schema characteristics such as inheritance depth and the number of class attributes, not considered in [Ph05]. It is therefore suggested that such concrete schema characteristics are considered for the selection of mappings in the model-driven generation of O/R mappings. For this consideration of schema characteristics, it is proposed to apply metrics for O/R mappings.

Model-driven generation of O/R mappings is considered to ease mapping specification for the developer. At the same time, such generation of O/R mappings reduces the developer's control of the mapping process. The developer should therefore have the option to define mappings manually when required. For this manual mapping specification, it is proposed to apply metrics for O/R mappings in order to give the developer sophisticated feedback on the impacts of the chosen mappings.

4 Model for Inheritance Mapping

Current O/R middleware products such as Hibernate [Hi06] support the application of different inheritance mapping strategies to individual inheritance relationships. It is therefore reasonable that mapping metrics also consider the fine-granular application of inheritance mapping strategies. Therefore, a novel mapping model is suggested in the following, which is considered to provide a concise means for the description of inheritance mappings.

The inheritance mapping strategies *union superclass*, *joined subclass*, and *union subclass* (notation partly adapted from [Hi06]) are suggested for the mapping of individual inheritance relationships. If an inheritance relationship between superclass P and subclass C is mapped with the ‘union superclass’ strategy, the attributes of C are mapped to the table that corresponds to the most general class that is reachable from C via ‘union superclass’ inheritance relationships. This table includes a type indicator field, which contains the class type of each row. The application of the ‘joined subclass’ strategy implies that P and C are mapped to separate tables. Finally, if the ‘union subclass’ strategy is applied, all inherited and non-inherited attributes of C are mapped to an own table. An example of mixed inheritance mapping strategies is depicted in Figure 2.

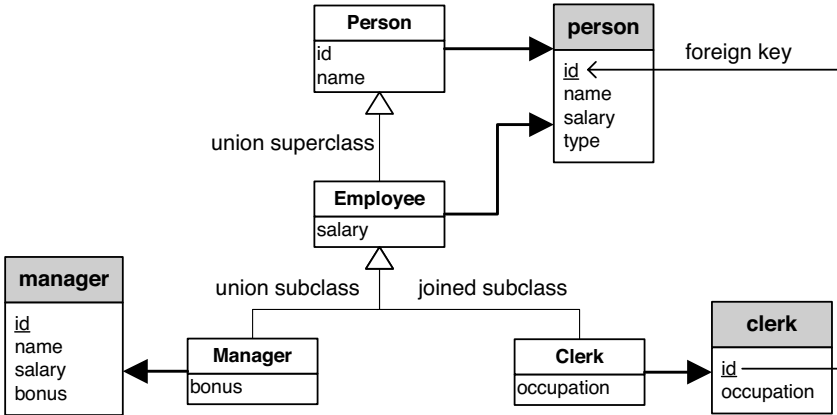


Figure 2: Example of mixed inheritance mapping strategies

5 Metrics for Inheritance Mapping

The following metrics are applicable to individual classes. This level of metric granularity is considered most suitable for supporting decision-making with regard to mapping strategies for individual associations and inheritance relationships.

Table Accesses for Type Identification (TATI): For queries that are issued against a class C , $TATI(C)$ equals the maximum number of tables that have to be accessed in order to identify the most specific class of the requested object. This metric is intended to measure retrieval performance in the context of polymorphic queries.

Number of Corresponding Tables (NCT): For a concrete class C , $NCT(C)$ equals the number of tables to which attributes of C are mapped. This metric focuses on the decrease of retrieval performance and the increase of query complexity that result from applying the ‘joined subclass’ mapping strategy.

Number of Corresponding Relational Fields (NCRF): For a class C , $NCRF(C)$ equals the number of relational fields in all tables that correspond to each non-inherited non-key attribute of C . This metric aims to assess the impact of the ‘union subclass’ mapping strategy on object model flexibility. Object model flexibility is negatively affected by the duplication of relational fields since changes in the object model then result in multiple changes to the relational schema.

Additional Null Values (ANV): If a class C , which is mapped to a table T , is at the subclass-end of a ‘union superclass’ inheritance relationship, $ANV(C)$ equals the number of non-inherited attributes in C multiplied with the number of concrete classes that are mapped to T , excluding C and all of its subclasses. Otherwise, $ANV(C) = 0$. This metric measures additional storage space in terms of null values resulting from the application of the ‘union superclass’ mapping strategy.

6 Conclusion and Further Research

It has been shown that the application of metrics in O/R mapping tools bears significant potential for supporting the developer in the considerably difficult task of mapping specification. In order to provide a foundation for O/R metrics, measurement goals have been suggested as a network of quality characteristics. Moreover, an inheritance mapping model and a metrics suite that builds on this mapping model have been proposed. It is a task for further research to investigate a more comprehensive set of metrics and to empirically evaluate the appropriateness of metrics for the manual specification and the model-driven generation of O/R mappings.

References

- [Hi06] Hibernate Open Source Software, <http://www.hibernate.org>.
- [ISO01] ISO/IEC 9126-1:2001 Software Engineering - Product Quality - Part 1: Quality Model, <http://www.iso.org>.
- [Ke97] Keller, W.: Mapping Objects to Tables: A Pattern Language. In (Buschmann, F.; Riehle, D., Hrsg.): Proc. 1997 European Conf. on Pattern Languages of Programming (EuroPLOP '97), Irsee, Germany 1997. Siemens AG, Munich, Germany, 1997; Siemens Technical Report 120/SW1/FB.
- [KJK93] Keller, A. M.; Jensen, R.; Keene, C.: Persistence Software: Bridging Object-Oriented Programming and Relational Databases. In (Buneman, P.; Jajodia, S., Hrsg.): Proc. 1993 ACM SIGMOD Int. Conf. on Management of Data, Washington, D.C., USA 1993. ACM Press, New York, USA, 1993; S. 523-528.
- [Ph05] Philippi, S.: Model Driven Generation and Testing of Object-Relational Mappings. In: Journal of Systems and Software, 77(2), 2005; S. 193-207.

Identifying Protein Domain Insertions in Large Databases

Tobias Lohe¹, January Weiner², Erich Bornberg-Bauer²

¹Institute of Computer Science, ²Division of Evolutionary Bioinformatics

University of Münster

tobias.lohe@uni-muenster.de

1 Introduction

It is widely accepted that proteins consist of protein domains which for the purpose of this article are defined as evolutionarily conserved units in a protein that often correspond to functional units and have a similar 3D structure. Since the discovery of the NADH-binding domain [RMO74] the study of protein domains became important. Today there are several databases which try to identify and classify domains in known proteins either according to their 3D structure or according to their amino acid sequence.

Studying domains and their arrangements in proteins can lead to new insights into the function and especially evolution of proteins [BBBK⁺05]. Particular forms of the modular rearrangement of proteins include circular permutations of domains [WTBB05], domain deletions [WBBB06] and the insertion of one domain (the *insert*) into another domain (the *parent*). This process leads to a disruption of the parent domain sequence as shown in Figure 1. Domain insertions were first described in [Rus94] and more extensively studied in [ASHS04] using the structure-based domain database SCOP [MBHC95] and relying on the already available domain annotations in this database.

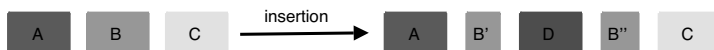


Figure 1: Schematic representation of an insertion. The domain D is inserted into the domain B of the original protein and disrupts B into the two fragments B' and B''.

However proteins with known 3D structure are by far outnumbered by proteins with known amino acid sequence and therefore their study is far from being comprehensive. Additionally the domain annotations in modern databases are known to be inaccurate in some cases, which might especially be true for insertions as they disrupt the domain sequence and structure and therefore might lead to unrecognized domain fragments. Our approach therefore attempts to identify insertions in sequence-based domain databases while being robust against inaccurate annotations.

2 Methods

For our study we chose the Pfam [BCD⁺04] protein families database which is a large collection of protein domain families generated by multiple sequence alignments and profile hidden Markov models. The version we used contained more than 8,000 domain families and covered more than 2,300,000 protein sequences.

The presented algorithm is designed to find correctly annotated insertions as well as insertions which are misannotated. For this purpose an insertion event is defined as a pair of proteins: the original protein and the *insertion protein* which contains an additional domain that was inserted into one domain of the original protein. These events are identified using a two-step approach which first searches insertion candidate pairs based on the domain arrangements and then verifies these candidates using a global sequence alignment.

Misannotations might happen in several ways, e. g. an insertion $A B C \rightarrow A B' D B'' C$ might become annotated as $A - D - C$ (B fragments not recognized) or $A B D - C$ (only first fragment B' recognized and wrongly interpreted as B). In general an insertion event $x_1 \dots x_i \dots x_n \rightarrow x_1 \dots x'_i y x''_i \dots x_n$, where the original protein consists of n domains, might have been annotated according to one of the five cases listed in Table 1.

#domains	#shared	example	domain order
n	$n - 1$	A - D - C	$x_1 \dots x_{i-1} y x_{i+1} \dots x_n$
$n + 1$	n	A B D - C	$x_1 \dots x_{i-1} z_1 z_2 x_{i+1} \dots x_n, z_1 = x_i \text{ or } z_2 = x_i$
$n + 1$	$n - 1$	A E D - C	$x_1 \dots x_{i-1} z_1 z_2 x_{i+1} \dots x_n$
$n + 2$	n	A B D B C	$x_1 \dots x_i y x_i \dots x_n$
$n + 2$	$n - 1$	A E D F C	$x_1 \dots z_1 z_2 z_3 \dots x_n$

Table 1: Possible cases for the annotation of a general insertion event $x_1 \dots x_i \dots x_n \rightarrow x_1 \dots x'_i y x''_i \dots x_n$ in Pfam. The column #domains gives the total number of domains for the insertion protein. The number of shared domains between the original protein and the insertion protein is given in column #shared. The expected domain string for the insertion protein is given in the domain order column, while an example annotation is given in the example column.

The first step of the algorithm searches the Pfam database for protein pairs that meet the requirements of one of the cases in Table 1. Therefore all proteins with n domains are compared to all proteins with $n, n + 1$ or $n + 2$ proteins to identify pairs that have the appropriate number of shared domains and the correct domain order.

These pairs are verified in a second step using the Needleman-Wunsch global sequence alignment algorithm [NW70] as depicted in Figure 2. In order to prevent false positives resulting from very short and repeated domains it is also checked whether the parent domain B is at least 25 amino acids long and if parent and insert domain are different. For all events which match these requirements the proportion of similar amino acids is computed using the global alignment. This similarity value can be used for later analysis.

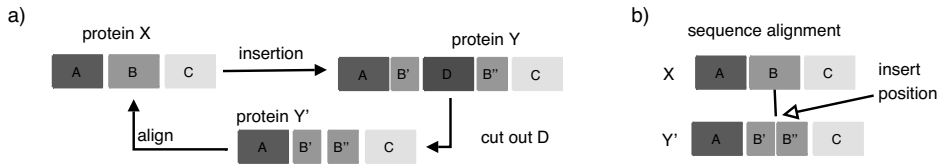


Figure 2: Process of verifying insertions where domain *D* is inserted into protein *X* yielding the insertion protein *Y*. a) During the verification process *D* is identified and cut out. The sequence of the resulting protein *Y'* is then globally aligned to the original protein *X*. b) If the insert position in the alignment corresponds to a domain in the original protein, an insertion event has been found.

3 Results

In total 26,260 multi-domain proteins were analyzed using the presented algorithm. There were 3,778 (14.4%) insertion candidates found ignoring the computed sequence similarity and 136 (0.5%) insertion events identified using an experimentally determined sequence similarity threshold of 50%. Some examples for insertions identified by the presented algorithm are shown in Figure 3. The events have to be further analyzed to verify if they really correspond to an evolutionary insertion or whether they are mere artifacts.

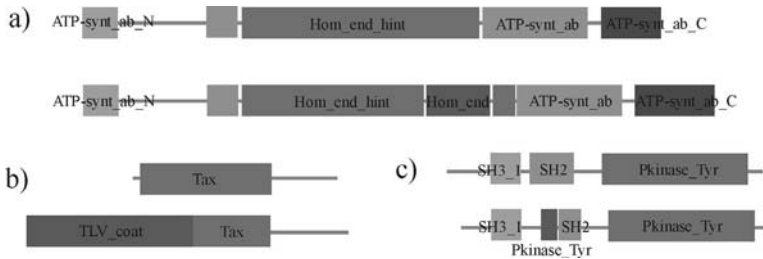


Figure 3: Examples for insertions found by the presented algorithm with the inserted domain in red. a) The insertion of the Hom_end domain is correctly annotated in Pfam. b) The TLV_coat domain might have been inserted here, but the sequence before TLV_coat is missing, so we cannot be sure. c) The small Pkinase_Tyr was inserted into the SH2 domain but not annotated as an insertion.

4 Conclusion

The analysis of protein domains is important to understand protein evolution. Domain insertions are of particular interest as they might be a mechanism for proteins to acquire new functionality very fast by getting inserted a new domain, which might also prove useful for protein engineering [CHP⁺00]. The presented algorithm is able to identify insertion events in large sequence-based protein domain databases and is robust against misannotations.

References

- [ASHS04] R. Aroul-Selvam, Tim Hubbard, and Rajkumar Sasidharan. Domain Insertions in Protein Structures. *Journal of Molecular Biology*, 338(4):633–641, 2004.
- [BBBK⁺05] Erich Bornberg-Bauer, Francois Beaussart, Sarah K. Kummerfeld, Sarah A. Teichmann, and January Weiner. The Evolution of Domain Arrangements in Proteins and Interaction Networks. *Cellular and Molecular Life Sciences*, 62(4):435–445, 2005.
- [BCD⁺04] Alex Bateman, Lachlan Coin, Richard Durbin, Robert D. Finn, Volker Hollich, Sam Griffiths-Jones, Ajay Khanna, Mhairi Marshall, Simon Moxon, Erik L. L. Sonnhammer, David J. Studholme, Corin Yeats, and Sean R. Eddy. The Pfam Protein Families Database. *Nucleic Acids Research Database Issue*, 32:D138–D141, 2004.
- [CHP⁺00] Bruno Collinet, Mireille Hervé, Frédéric Pecorari, Philippe Minard, Olivier Eder, and Michel Desmadril. Functionally Accepted Insertions of Proteins within Protein Domains. *The Journal of Biological Chemistry*, 275(23):17428–17433, 2000.
- [MBHC95] Alexey G. Murzin, Steven E. Brenner, Tim Hubbard, and Cyrus Chothia. SCOP: A Structural Classification of Proteins Database for the Investigation of Sequences and Structures. *Journal of Molecular Biology*, 247(4):536–540, 1995.
- [NW70] Saul B. Needleman and Christian D. Wunsch. A General Method Applicable to the Search for Similarities in the Amino Acid Sequence of Two Proteins. *Journal of Molecular Biology*, 48:443–453, 1970.
- [RMO74] Michael G. Rossmann, Dino Moras, and Kenneth W. Olsen. Chemical and Biological Evolution of Nucleotide-binding Protein. *Nature*, 250:194–199, 1974.
- [Rus94] Robert B. Russell. Domain Insertion. *Protein Engineering*, 7(12):1407–1410, 1994.
- [WBBB06] January Weiner, Francois Beaussart, and Erich Bornberg-Bauer. Domain Deletions and Substitutions in the Modular Protein Evolution. *FEBS Journal*, 273:2037–2047, 2006.
- [WTBB05] January Weiner, Geraint Thomas, and Erich Bornberg-Bauer. Rapid Motif-Based Prediction of Circular Permutations in Multi-Domain Proteins. *Bioinformatics*, 21(7):932–937, 2005.

Vom Informations- zum Wissensmanagement: Der Knowledge Bus als flexibler Lösungsansatz

Wolfgang Reinhardt

Universität Paderborn
Institut für Informatik
wolle@uni-paderborn.de

Tobias Nelkner

Universität Paderborn
Institut für Informatik
tobin@uni-paderborn.de

Michael Mlynarski

Universität Paderborn
Institut für Informatik
funky@uni-paderborn.de

Abstract: Ein professionelles Informationssystem reicht heutzutage nicht aus, um die vielfältigen und umfangreichen Dokumente eines Unternehmens effizient verwalten zu können. Erst der Einsatz von Wissensmanagement-Techniken hilft diese mit der Unternehmensstruktur und seinen Prozessen in Verbindung zu setzen. Im Rahmen eines Projektes an der Universität Paderborn wurde ein Architekturkonzept entwickelt, welches Informations- und Wissensmanagement in einer flexiblen Form verbindet. Die Nutzung des Systems soll im Unternehmen erhebliche Mehrwerte in der Verwaltung der eigenen Dokumente bringen. Mit Hilfe von Metadaten, Taxonomien und der mehrschichtigen Softwarearchitektur ist ein System entstanden, dessen Möglichkeiten weit über traditionelles Dokumentenmanagement hinaus gehen.

1 Das Problem

Die heutige Informationsgesellschaft hat mit einem schwerwiegenden Problem umzugehen: es existieren zu viele Informationen, um diese sinnvoll zu nutzen und in explizites Wissen zu transformieren. In einem produktiven Unternehmensumfeld, welches sich durch effektive Informationsverarbeitung in kurzen Zeitabständen auszeichnet ist es neben der reinen Extraktion relevanter Informationen besonders wichtig diese so zu abulegen und verwalten, dass die Mitarbeiter Zugriff auf sie haben. Die in einem Unternehmen zu verarbeitenden Informationen verdoppeln sich etwa pro Jahr ein mal (vgl. [GI07]). Zudem entstehen Informationen in den verschiedensten Quellen und werden in unterschiedlichen informationsverarbeitenden Systemen abgelegt.

MoKEx (Mobile Knowledge Experience) ist eine internationale Projektserie, die sich interdisziplinär mit neuen Wegen der Wissensorganisation in mobilen Lernkontexten beschäftigt (vgl. [HHM⁺06]). Die Projekte sind eine Zusammenarbeit verschiedener industrieller Partner sowie Universitäten in Deutschland und der Schweiz und verbindet erfolgreich universitäre Informatikausbildung mit praktischen Erfahrungen durch reale Aufgabenszenarien (vgl. auch [MvW00, COR01, Den04]). Während sich MoKEx 1 in diesem Kontext mit der Auflösung bekannter Restriktionen im Rahmen mobilen E-Learnings beschäftigte (vgl. den Mobile Delivery Server [ABH⁺06]), widmete sich MoKEx 2 dem Problem der redundanten Datenhaltung im Wissensmanagement. Das Ziel des Projekts MoKEx 2 war es einerseits die divergierenden Anforderungen der verschiedenen industri-

ellen Partner zu befriedigen und andererseits eine Architektur zu schaffen, die flexibel und intelligent verschiedene informationsverarbeitende Systeme verbindet und die gemeinsame Nutzung der Daten ermöglicht.

2 Ein Lösungsansatz

Der im Rahmen von MoKEx 2 entwickelte Lösungsvorschlag ist der Knowledge Bus (KNB), welcher eine offene Schnittstelle zwischen verschiedenen Frontend- und Backend Systemen ist. Er vereint so die vielfältige Informations- und Datenverwaltung der einzelnen Teilsysteme zu einem Wissensmanagement System. Ziel war es, eine Lösung zu schaffen, welche in der Lage ist, mehrere Backend Systeme (Dokumentenmanagementsystem, Lernmanagementsystem, etc.) durch eine flexible Schnittstellenstruktur miteinander zu verknüpfen und die verwalteten Daten als einen transparenten Wissenspool zur Verfügung zu stellen. Damit stellt der KNB den Middle Tier da, welcher zum EIS-Tier die Schnittstellendefinitionen für Backend Systeme bereitstellt und zum Client-Tier die Schnittstellendefinitionen für Frontend Systeme hält. Die sich daraus ergebenden Aufgaben des KNB beinhalten vor allem das *Transaktionsmanagement*, um die Persistenz der verschiedenen Daten und Transaktionen sicherzustellen und des Weiteren eine flexible und mehrgliedrige *Rechteverwaltung*, um zum einen die Sicherheit der Daten gegenüber der Benutzerschnittstelle zu gewährleisten, zum anderen aber auch, um der Vielfalt der Rechtekonzepte der einzelnen Backend Systeme gerecht zu werden, was im Rahmen der Transparenz gegenüber Endbenutzern in der Integration des Single-Sign-On Konzepts Anwendung findet.

Die prinzipielle Architektur des Systems ist so konzipiert, dass jedes Backend System angebunden werden kann, welches eine Schnittstelle zur eigenen Datenbasis zur Verfügung stellt. Darüber hinaus stehen für Frontend Systeme mit verschiedenen Standardschnittstellen für Protokolle wie SOAP, flexible Anbindungsmöglichkeiten zur Verfügung, so dass Desktop Applikationen wie auch webbasierte Lösungen zum Einsatz kommen können. Um den Mehrwert für Endbenutzer weiter zu erhöhen, wurde über die reine Verknüpfung einzelner Systeme hinaus, die Möglichkeit geschaffen, vom Benutzer generierte Datenobjekte in Taxonomien und Prozessschritte einzuordnen, unabhängig von der Benutzung eines speziellen Backend Systems.

Um zum einen die für die verwaltenden Aufgaben notwendigen Daten und zum anderen die Metadaten zu benutzergenerierten Objekten zu organisieren, wurde ein mehrdimensionales Metadatenkonzept entwickelt. Den LOM Standard erweiternd, müssen folgende Daten abgebildet werden:

- Kontextuelle Metadaten der Besitzer der einzelnen Datenobjekte
- Metadaten zur Einordnung von Datenobjekten in Taxonomien
- Metadaten zur Einordnung von Datenobjekten in Prozessstrukturen
- Manuell eingegebene, das Datenobjekt beschreibende Metadaten

- Automatisch generierte, kontextbezogene Metadaten

Jedes eingespeiste Datenobjekt, wird mit den Informationen des Besitzers verknüpft, womit Rückschlüsse darauf gezogen werden können, in welchem Rahmen dieses Objekt entstanden ist und in welchem Kontext es eingesetzt wird, wozu ebenfalls die automatische Einordnung des Objekts in die Unternehmenstaxonomie gehören kann. Die Taxonomieeinordnung und die Prozessschritte, zu denen ein Objekt gehört und welche jeweils angegeben werden können, vereinfachen die Arbeit in prozessorientierten Arbeitsabläufen, was eine Forderung aller Industriepartner darstellte. Abschließend werden manuell erfasste, als auch automatisch generierte Metadaten zu allen Objekten hinterlegt, wobei zu beachten ist, dass restriktionsgemäß zu allen eingepflegten Objekten ein Mindestsatz an Metadaten erfasst werden muss. Diese objektzugehörigen Metadatensätze werden in einer fest integrierten, XML-basierten Datenbank gehalten, die im Gesamtsystem die vorhandene Wissensbasis widerspiegelt und daher auch intern die Basis zum Suchen und Finden von Informationen darstellt.

3 Zusammenfassung und Ausblick

Mit Mobile Knowledge Experience meinen wir wirklich den Einsatz von Wissensmanagement in mobilen, sich ständig ändernden Umgebungen. Anhand der Analyse mehrerer Unternehmen haben wir ein Konzept für den effektiven Einsatz von modernen Techniken sowie mehrstufigen Metadaten, Ontologien und Taxonomien entwickelt. Die neue und innovative Vorgehensweise der Organisation der täglichen Unternehmenstätigkeiten (wie die Erstellung, Ablage und Suche von Datenobjekten) kann in mittleren und großen Firmen erhebliche Mehrwerte bringen (vgl. [Tak95]). Vor allem die Simplizität des Systems gewährleistet es mit Hilfe von nur drei Operationen (Erstellung, Ablage und Suche) wichtiges Unternehmenswissen schnell und effektiv zu nutzen. Weiterhin kann man den Wissensverarbeitungsprozess anhand der Versionsverwaltung (semantische und einfache Änderungen) nachverfolgen. Durch Aufteilung des Systems in mehrere Schichten und Module ist eine enorme Flexibilität gewährleistet. Die Verknüpfung der Datenobjekte mit vorhandenen Unternehmensstrukturen oder laufenden Geschäftsprozessen wie auch Projekten bewirkt, dass die Objekte besser klassifiziert und dadurch gefunden werden.

Ideen wie die zentrale Datenverwaltung (mit Hilfe des Knowledge Bus) haben sich bereits in anderen Gebieten der Informatik (wie Netzwerkverwaltung, EAI oder SOA) bewertet. Mit der im MoKEx 2 Projekt entwickelten Anwendung haben wir das theoretische Konzept in Form eines prototypisch laufenden Systems umgesetzt. In der ersten Testphase und der Präsentation bei den Projektpartnern konnte man die erheblichen Mehrwerte des Systems praktisch nachvollziehen.

Die Weiterentwicklung des Systems wird sich auf die Integration weiterer, unterschiedlicher Backend-Systeme konzentrieren. Das mehrstufige Metadaten-Konzept soll um verschiedene Ansätze zur automatischen Metadaten-Extraktion erweitert werden. Zusätzlich wird an die Integration eines professionellen BPMS (Business Process Management Sys-

tem) gedacht. Die möglichen weiteren Einsatzgebiete des Systems, wie bspw. die Nutzung von E-Mails oder Bookmarks als Datenobjekte, stehen noch vollkommen offen.

Literatur

- [ABH⁺06] Dieter Averbeg, Julien Boppert, Kai Holzweißig, Tobias Loke, Tim Riemann und Johannes Magenheim. Mobile Delivery Server - MDS. A holistic solution for resolving problems and limitations in mobile E-Learning scenarios. In *Proceedings of the 1st International Conference on Mobile Communications and Learning (MCL2006)*, Mauritius, April 23–28 2006.
- [COR01] CORPORATE The Joint Task Force on Computing Curricula. Computing Curricula 2001. *Journal on Educational Resources in Computing (JERIC)*, 1(3), 2001.
- [Den04] Peter J. Denning. Great Principles in Computing Curricula. In *Proceedings of the 35th SIGCSE Technical Symposium on Computer Science Education, SIGSCE 2004*, Seiten 336–341, New York, NY, USA, 2004. ACM Press.
- [GI07] Gesellschaft für Informatik e.V. Informatiktage 2007: Workshops. <http://www.gi-ev.de/informatiktage/informatiktage-2007/workshops/>.
- [HHM⁺06] Knut Hinkelmann, Kai Holzweißig, Johannes Magenheim, Fabian Probst und Wolfgang Reinhardt. Linking Communities of Practice with Learning Communities in Computer Science Education. In *Proceedings of the 19th IFIP World Computer Congress (WCC2006)*, Santiago de Chile, August 20–25 2006.
- [MvW00] F. Mulder und T. van Weert. Informatics Curriculum Framework 2000 for Higher Education Edited by International Federation for Information Processing (IFIP) / UNESCO. <http://www.ifip.org/pdf/ICF2001.pdf>, 2000.
- [Tak95] Nonaka Takeuchi. *The Knowledge creating company*. Oxford University Press, New York, 1995.

How does a project-related Personal Information Collection reflect the View of a Project?

Jan Parchmann

Fakultät für Informatik
Universität Karlsruhe (TH)
Am Fasanengarten 5
76131 Karlsruhe
jan.gi@online.de

Abstract: This paper gives a description of an observation of seven people working at a current project. The aim was to find out how project-related Personal Information Collections (pr-PICs) were created and changed over time during a project. Sometimes, the view of a project changes since new aspects became important that were unknown before. This might change pr-PICs, too.

Different people not only have different strategies to organize information they encounter during a project, but also have different strategies to keep their organizations up to date.

1 Introduction

Working at projects—like “develop an internet presentation for a company” or “create a job description for a new employee” (work-related), “redesign a DVD on-screen menu” (study-related), or “planning a wedding” (private)—people encounter much information, in the form of paper, e-mail, electronic documents, and so on, which they mostly want and need to keep—for example for reference, or as a to-do reminder. They not only have to keep information, but also have to re-find it later [JM03]. Personal Information Management (PIM) is a wide area of research that deals with individual behavior of organizing collections of information. This includes information gathering as well as received information, storage and retrieval of information, and how to organize and manage all information in a personal environment [L88].

This study results from the graduate course “PIM – Personal Information Management” given at the “Royal Institute of Technology” in Stockholm, Sweden by Affiliate Associate Professor William P. Jones from the “School of Library and Information Sciences” at the University of Washington.

2 Related Work

Previous research focused on important aspects of PIM, which is a broad term for the strategies of collecting, storing, and organizing digital information in a personal environment [L88].

One general problem for users is cognitive overload when handling information. The supply-related overload, caused by pushed and pulled information, should be decreased by providing an environment as an activity space, supporting a cognitive workflow, which is adapted to the habits and needs of the person acting in it [K00]. One way to improve is to reduce the decision, of keeping information or not keeping it, to a single detection task. There are ways to reduce costs and the likelihood of those keeping mistakes [J04].

Working at different projects means to handle information for different projects. People do have problems creating their virtual workspaces, and when solving these problems they are often on their own, receiving little or no support from the outside [K96]. [BS04] described how PIM behavior evolves over time by presenting longitudinal data. [JPGB05] reasons that structures of a pr-PIC, like folder hierarchies, are already information in their own.

There is also a need to improve problems given by information fragmentation along applications [JM03]. Some approaches try to combine necessary tools for managing information in single platforms, like [K03] with UMEA, with quite well results. To reduce information fragmentation without taking the user her applications away, a Universal Labeler was created which is described in [JBFM05]. This tool is a step towards the integration of information regardless of its form (e-document, paper, web page), and towards the integration of information management with the management of tasks and projects.

3 Methods

Methodology used in the “Keep Found Things Found” (KFTF) project to study how people organize project-related information were provided by [J05], and were used in modified form to collect data for this study.

Seven participants were observed during a period of twelve weeks working on current projects to show how they structure encountered information, and how these structures changes over time. The study consisted of seven sessions with each participant, divided into introductory sessions (to get the participants general usage with information to know, and to find and be introduced to a suitable project to observe), intervening sessions (to observe changes in structure of their information collections concerning the selected project), and final sessions (to wrap up, and to compare results with a chosen back-up project to avoid tempering findings).

The observed projects were either work-related or study-related or private, and varied from small to huge projects, had moderate to sophisticated difficulties, and familiarity of the participants varied from familiar to unknown.

4 Results

Even though several different approaches and strategies for handling encountered information were observed during this study, and not even each participant followed a consistent strategy herself during the whole project, a couple of similarities and basic differences could be found.

Those who were very familiar with the topic of their project from the very beginning were used to create their structure in advance, and filled it up with information over time. This top-down strategy was very organized, and the pr-PIC always gave a representation of the whole project. For these people there should be a possibility to plan their project with today's known project planning tools, but to combine these results with their structure to file information. Tools were project planning, email, and file system goes together would be a good base for those people. To not always re-create structures for new but similar projects it could be helpful to just copy structures without files from previous projects.

The second group—those participants who were more or less unskilled in this exactly project, and did not know neither what kind of information nor how much they would encounter—embarked on the strategy of divide and conquer, and built their structure bottom-up. During the project, they added new sub-folder by ad-hoc decisions when a part of their collection became too messy. Since the structure of their pr-PIC grew over time, it always gave an overview of the current state of the project.

The last group was quite messy in organizing encountered information. They just piled up documents, and changed their workspace after it became too messy. To keep attention to currently important and necessary files, they created sub-folders called “old”, or “new”, respectively. Therefore, they always had just the information they needed and which were important at a glance. This kind of maintaining costs time and de-structures workspace. A file browser with several different hierarchal views could be better to re-find archived information. Each file in a folder, for example, could be tagged as “current”, “project XY”, or “archived”. One view of the folder shows all information and files, another only current or project XY-related, et cetera. Another approach could be to show just recently used documents, and hide those who were not used for a longer period. To re-show longer unused documents users just have to move a time-slider. Alternatively, they could delete unnecessary documents if there was an easy way to restore them, e.g. by getting a previous folder view, as it is common in version control systems.

Participants were asked to draw mental representations of their projects, and it was astonishing that none of these could be translated into file, or email structure of their operating systems, or email tools. Mental representations seem to be more unstructured, or rather structured in a not hierarchical way. The common hierarchical organization possibilities are not supporting this fact well.

Satisfaction with handling information varies among different participants following different strategies. Especially those who were familiar with their project were mostly dissatisfied with possibilities given by their system and tools.

5 Discussion

All described conclusions are under consideration. They refer to the collected data that cannot be representative for all people working at projects and dealing with information. Furthermore, all participants were knowingly observed during the selected project. The back-up projects showed that biases—mostly in terms of acting explicitly organized—could not be excluded. However, the back-up projects showed that all participants generally organize the way they were used to in the observed projects.

Prototypes could be developed from the successions given in the result section to verify usability and acceptance by users. There should be no need to force people using one of the presented strategies, but all strategies should be supported well by tools and operating systems.

References

- [BS04] Boardman, R.; Sasse, M.A.: “Stuff goes into the computer and doesn’t come out” A Cross-tool Study of Personal Information Management. ACM SIGCHI Conference on Human Factors in Computing Systems (CHI 2004).
- [J04] Jones, W.: Finders, keepers? The present and future perfect in support of personal information management. First Monday, Volume 9, Issue 3, March 2004.
- [J05] Jones, W.: Methodology used in the KFTF project to study how people organize project-related information. Personal communication via email, March 9, 2005.
- [JBFM05] Jones, W.; Bruce, H.; Foxley, A.; Munat, C.F.: The Universal Labeler: Plan the Project and Let Your Information Follow. Inclusion at ASIST ’05, 2005
- [JM03] Jones, W.; Maier, D.: Report from the Session on Personal Information Management. Workshop of the Information and Data Management Program, National Science Foundation Information, Seattle, WA, 2003.
- [JPGB05] Jones, W.; Phuwanartnurak, A.J.; Gill, R.; Bruce, H.: Don’t Take My Folders Away! Organizing Personal Information to Get Things Done. CHI 2005, Portland, Oregon.
- [K96] Kaptelinin, V.: Creating Computer-Based Work Environments: An Empirical Study of Macintosh Users. ACM SIGCPR/SIGMIS’96 Conference, Denver, Colorado, 1996.
- [K03] Kaptelinin, V.: UMEA: Translating Interaction Histories into Project Contexts. CHI 2003, Ft. Lauderdale, Florida.
- [K00] Kirsh, D.: A few thoughts on cognitive overload. *Intellectica*, 2000/1, 30, pp. 19-51.
- [L88] Lansdale, M.: The psychology of personal information management. *Applied Ergonomics* 19,1 (1988), pp. 55–66.

Supporting Presentation Reuse with a Presentation Repository System

Stefan Scheidewig

Friedrich-Schiller-Universität Jena
Fakultät für Mathematik und Informatik
thalis@web.de

1 Introduction

Nowadays, everyone who gives presentations on a more or less regular basis is confronted with the problem of managing a magnitude of presentation documents. This is equally true for instructors at universities as for practitioners in industry. Typically, new presentations are not developed from scratch, but existing slides are chosen, adapted to a new context, put in new combinations and supplemented by new slides. Today, this is an ad-hoc process. Authors are offered little support for accomplishing this task. The aim of this work is to address this problem with the design and implementation of a repository, that supports presentation authors and enables systematic reuse of existing materials.

To clarify the purpose of this repository system, we imagine an example user Bob, who has to create a presentation document for a meeting. Because the creation of presentations from scratch is time-consuming and since Bob has given a similar presentation before, he decides to reuse this very presentation. The first problem he is confronted with is the retrieval of this presentation document. Since he doesn't quite remember the file name, it takes him a few minutes of search on his hard disk to locate the file. Bob then decides to add some slides of a colleague's presentation to the existing talk. To get the colleague's presentation document, he has to contact this person. After some time, he gets the slides as attachment of an e-mail. The last slide of this presentation holds the bibliography, which is quite out of date. He searches his presentation documents for a presentation he has recently created, holding an updated bibliography slide. But he cannot find it and has to update this slide manually.

This whole working process can be enhanced with the help of a presentation repository system. Every colleague populates the repository stock with its presentation documents. If Bob wants to reuse an older presentation, he can search the presentation stock by keywords to find the appropriate presentation. Because every presentation and slide is automatically linked to the existing content, the user can search similar presentations and slides within the stock. So, Bob is able to retrieve the slides from his colleague without a lag of time. In addition, Bob can easily find an updated bibliography slide by searching for the newest version within the bibliography group the bibliography slide is linked to.

By regarding these scenarios, we can derive several requirements of a presentation repository system. For the retrieval of presentations and slides within the stock, the user should have three options: Either the user builds up a logical repository tree similar to a file system or searches the presentations with the help of keywords. The third option are content linking and grouping information that point to similar presentations or slides.

Of course, the repository content should be provided with meta data to refine searches and give the user additional information about the entries. But next to the search and display of presentations and slides, the user should also be capable of reusing the existing slides and templates in new presentation documents by composition. To facilitate this composition process the user can group consecutive slides into logical sections.

Furthermore, this repository system should also provide functionalities for document versions, multilingualism and workflow mechanisms with an integrated communication function to reduce the external communication overhead. It is evident that a presentation content management system has to support as much presentation document formats as possible. An overview of these requirements is depicted in Table 1.

2 Related Work

This repository system implemented in this work has the working title *PreReS - Presentation Repository System* and can be described by the generic term content management system. To compare PreReS with existing solutions, we selected two content management systems capable of handling presentation documents. The first one is a closed source system named *SlideServer* [AG] by *Chamaeleon AG*. This is a web-accessible content management based on the programming language *Perl* and exclusively handles PowerPoint presentation documents. The second content management system compared with PreReS is an open source software system by the firm *Alfresco* [Alf]. Alfresco is intended as enterprise content management system especially supporting workflow mechanisms not only for presentation documents. What features PreReS, SlideServer and Alfresco support regarding the requirements is summarized in Table 1.

3 Architecture and Implementation

3.1 System Architecture

The system described above is intended to be accessed by many people, thus a design after the client/server paradigm was chosen. Figure 1 depicts the system architecture. Every action within the system needs access authorization. Checking the validity and creating or deleting users is done by the *user/group manager*. A verified user can build its own repository tree structure to manage the set of presentations. This tree is also persistently stored within the back-end database by the *tree manager*.

The user works with the slide thumbnails of a presentation. She can update the meta data, which she can later use to recover this presentation. Via drag and drop, the user is also able to build up new presentations. A text manipulation within this program is not intended and must be performed with external applications. With the information which slides from which presentations should be used, the new presentation is produced by the *presentation transformer* by loading the appropriate master templates and slides from the source documents and creating a new presentation in the desired destination format. If wished by the user the header, footer, date and the page numbers are also adjusted by the system.

The *repository manager* is responsible for storing and loading the presentation objects from the database. Due to performance reasons all presentation objects are stored at the client side via object serialization. Only if the working place is changed by the user or some of the files are deleted, the missing presentation objects are reloaded from the server.

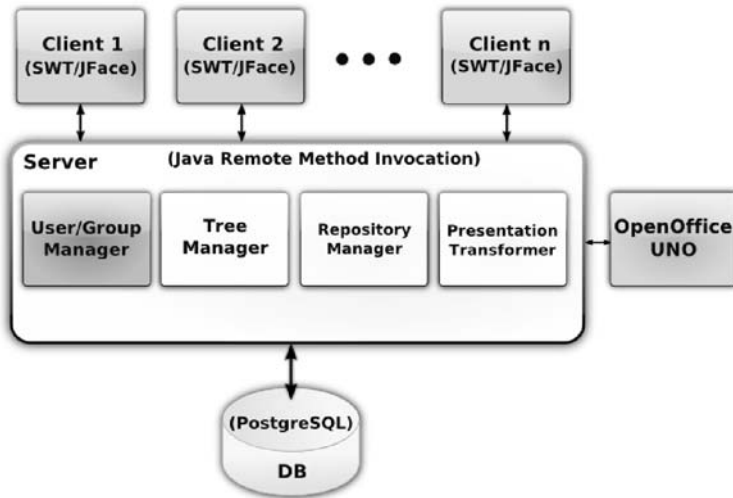


Figure 1: System Architecture and Implementation

3.2 Implementation

In order to achieve platform independence, Java was used as programming language. The client GUI (*Graphical User Interface*) should run on every major platform. To be fast enough and to make graphical work with the slide pictures possible, we decided to use the SWT/JFace [Ecl] libraries. Thereby, the look and feel of the operating system can be maintained in conjunction with fast handling. The server was implemented as a Java RMI [Micb] (*Remote Method Invocation*) server object, with access to a PostgreSQL database system. The main part of the server, the presentation transformer, which has to disassemble

the presentations, uses OpenOffice.org UNO [Mica] (*Universal Network Objects*). This requires an OpenOffice application to run in the background.

4 Summary and Future Work

The aim of the work described in this paper was to design a presentation repository that supports authors in managing and reusing existing presentation materials. Currently, the concept presented here is being implemented. Table 1 shows the current status of development where the \approx symbol represents work in progress. For the future, we are looking at possibilities to improve the search function by augmenting the information stored in the repository with semantic annotations stemming from an ontology.

Feature	PreReS	SlideServer	Alfresco
Repository tree	✓	✓	✓
Full text search	✓	✓	✓
Meta data	✓	✗	✓
Versions	✓	✗	✓
Thumbnail visualization	✓	✓	✗
Composition (<i>Content Reuse</i>)	\approx	✓	✗
Context linking	\approx	✗	✗
Grouping of slides	\approx	✗	✗
Sectioning	\approx	✓	✗
Offline access	✓	✗	✗
Platform independent	✓	✗	✓
Multilingualism	✗	✓	✓
Workflow	✗	✗	✓
Communication	✗	✗	✓
Supports PowerPoint format	✓	✓	✓
Supports OpenDocument format	✓	✗	✓
Supports StarOffice format	✓	✗	✓
Supports Office Open XML	✗	✗	✗
Supports Latex sources	✗	✗	✗
Export to PDF	✓	✓	✓
Export to SWF	✓	✗	✓
Export to HTML	\approx	✗	✓

Table 1: Comparison between the features of *PreReS*, *SlideServer* and *Alfresco*

References

- [AG] Chamaeleon AG. *Homepage of Chamaeleon AG's SLIDEServer*. <http://www.slideserver.eu/features/index.html>.
- [Alf] Alfresco. *Homepage of Alfresco*. <http://dev.alfresco.com/>.
- [Ecl] Eclipse.org. *Homepage of SWT: The Standard Widget Toolkit*. <http://www.eclipse.org/swt/>.
- [Mica] Sun Microsystems. *OpenOffice.org API Project Homepage*. <http://api.openoffice.org/>.
- [Micb] Sun Microsystems. *Remote Method Invocation Home*. <http://java.sun.com/javase/technologies/core/basic/rmi/index.jsp>.

Wissensmanagement

- Herausforderung Prozesswissen in Konzernen -

Carolin Wall

Carolin.wall@yahoo.de

Bedingt durch die Organisationsstrukturen in Konzernen sind Prozesse über verschiedene Abteilungen und Aufgabenträger weit verteilt. Bei Umstrukturierungen führt dies häufig zu Verschiebungen der Prozeßgrenzen, die teilweise eine Neudefinition von Schnittstellen zur Folge haben und zu höheren Reibungsverlusten zwischen den Kommunikationspartnern führen. Da der Großteil des Prozesswissens als Erfahrungswert vorliegt und sich wesentlich in der Kommunikation sowie einer effizienten Zusammenarbeit zwischen den unterschiedlichen Rolleninhabern niederschlägt, soll erörtert werden, welche IT-technischen Mittel unterstützend in der Kommunikation eingesetzt werden können.

1 Bestandsaufnahme

Das Besondere im Umfeld von Konzernstrukturen ist, dass häufiger als in mittelständischen Unternehmen oder gewachsenen Familienbetrieben Umstrukturierungen erfolgen. Diese Reorganisationsen sind besonders nach Fusionen oder bei einer Neuausrichtung der Geschäftsfelder eine beliebte Maßnahme, um die strategischen Ziele umzusetzen und Synergien zu erzielen. Das identifizierte Verbesserungspotential liegt in der Optimierung von Prozessabläufen und wird durch eine stärkere Detaillierung und Spezialisierung der Aufgabenbereiche angestrebt.¹ Bedingt durch die Organisationsstrukturen in solchen großen Unternehmen wie in Konzernen sind Prozesse über verschiedene Abteilungen verteilt und mehrere Mitarbeiter mit Aufgaben eines Prozesses betraut. Mit der größeren Anzahl an Schnittstellen zwischen Zuständigen und Verantwortlichen wächst der Kommunikationsaufwand². Für einen guten Prozessablauf ist das reibungslose Zusammenwirken aller Prozessbeteiligten ein wesentlicher Erfolgsfaktor. Dabei ist der Faktor Mensch bei der Interaktion nicht zu unterschätzen und sein Bedürfnis nach Beständigkeit bzw. die Scheu vor Veränderungen zu bedenken.³

¹ Vgl. dazu [HL05], S. 251: fragmentierte Prozesse führen zu eng definierten und spezialisierten Aufgaben, die meist mit einer funktionsorientierten Organisationsstruktur einhergehen.

² Vgl. Studie wonach Manager im Durchschnitt 90% ihrer Zeit mit Kommunikation verbringen [SK99], S.136.

³ Vgl. dazu [Fa01], S.272: Schwierigkeiten bei Wissensaustausch durch innere Widerstände der Beteiligten und Angst vor Neuerungen.

Mit der Neuordnung geht oft eine Verschiebung der Prozeßgrenzen einher und ein veränderter Ablauf, der dazu führen kann, dass Schnittstellen neu definiert oder präzisiert werden müssen. Sowohl bei gut etablierten Standardprozessen als auch beim Einsatz von Frameworks wie ITIL oder CroBIT kann es durch die Umbesetzung von Rollen und Ansprechpartnern zu Irritationen der Betroffenen führen. Nicht selten etablieren sich Abläufe neben den offiziellen Prozessen und der „Weg an den Prozessen vorbei“ wird als bewährte Abkürzung täglich in Anspruch genommen.

Während die Prozesse Gefahr laufen Papierkonstrukte zu bleiben, gelten in der täglichen Zusammenarbeit unter Kollegen oft andere Gesetzmäßigkeiten. Das Vertrauen auf langjährige und damit bewährte Wege ist oft erfolgsversprechender als sich an „verkomplizierten“ Abläufen entlang zu hangeln. Dabei kann der Wert von gewachsenen Interaktionsmustern nicht hoch genug eingeschätzt werden. Abgesehen von spontanen Kommunikationsanlässen wie bei Gesprächen beim Mittagessen oder auf dem Flur, wo durch zufällige Begegnungen persönliche Bindungen gefestigt und meist auch fachliche Informationen ausgetauscht werden, ist mit einem bekannten Gesprächspartner leichter zu verhandeln, als sich an die Eigenheiten eines neuen Ansprechpartners zu gewöhnen.

2 Informationen und spezifisches Wissen

Im Unternehmensumfeld als verteiltes Wissenssystem⁴ ist der Erfolgsfaktor nicht allein von Vorhandensein bzw. Zugängigsein der Information abhängig sondern vor allem von der Nutzung des Wissens und der persönlichen Bewertung der Information durch den einzelnen Mitarbeiter für seine auszuführende Tätigkeit. „Informationen werden von uns Menschen aufgenommen und vor dem Hintergrund unserer Erfahrungen, bestehendem Wissen und durch Kombination versucht einzuordnen. Dieser sehr komplexe Vorgang [...] aufgrund der aufgenommenen Information und deren cognitive Einordnung [entsteht] neues Wissen“⁵ S. 304

Da dieses Wissen sich oft aus Erfahrungswerten ableitet wird verständlich, dass die Einarbeitungszeit z.B. von neuen Mitarbeitern in Prozessabläufe relativ hoch ist. Durch die Spezialisierung und der damit einhergehenden Erhöhung von Schnittstellen werden die Abläufe komplex und selten sind die Zusammenhänge und Abhängigkeiten innerhalb eines Prozesses für den Einzelnen in seiner Gesamtheit zu erfassen.

Die verfügbaren Informationen wie elektronische Telefonverzeichnisse, Organigramme und Prozessbeschreibungen sind zwar nützlich, decken aber nicht alle Informationen ab, die zur Ausübung der Aufgabe erforderlich sind. Abgesehen von einem geringen Detaillierungsgrad der generischen Modelle und fehlende personelle Spezifizierungen der Beschreibungen z.B. durch explizite Zuordnung von Rolle zu Mitarbeiter oder Offenlegung von Verantwortlichkeiten und Entscheidungskompetenzen steigt der

⁴ Siehe auch [HL05], S. 250 ff.

⁵ Siehe [SK99], S. 304.

Anpassungsaufwand mit dem Grad der Detaillierung bei Veränderungen enorm an, und das Vertrauen in die Aktualität der Informationen leidet.

3 Ansatzpunkte und Ideen zur Verbesserung

Die Schwierigkeiten von virtueller bzw. verteilter Kommunikation via Email oder Chat bestehen darin, dass viele Komponenten der realen Begegnung, die über andere Kanäle als Sprache transportiert werden, bei der Übertragung in die Datenwelt verloren gehen. So wird bei Videokonferenz zwar neben dem akustischen zusätzlich der visuelle Eindruck der Gesprächsrunde simuliert, aber die Fokussierung z.B. auf den Wortführer oder eine Präsentation machen den entfernten Konferenzteilnehmern nur einen Realitätsausschnitt zugänglich.

Denkbare Ansatzpunkte könnten darin bestehen:

- die emotionale Befindlichkeit des Gespräch/Chatpartners sichtbar zu machen,
- Umfeld bzw. Umwelteinflüsse und zufällige Begegnungen in der virtuellen Welt zu simulieren,
- Netzwerkstrukturen und Kommunikationswege zu visualisieren.

Die meisten Abstimmungen erfolgen telefonisch oder per Mail, wobei zunehmend der Einsatz von Instantmessaging (Chat) sich auch im Unternehmensumfeld durchsetzt. Die Vorteile der Textübertragung in Echtzeit bestehen darin, dass die Antwortaktivität zeitlich vom Empfänger gesteuert werden kann und er dies in seinen Arbeitsablauf integrieren kann ohne in seiner derzeitigen Beschäftigung gestört und unterbrochen zu werden. Viele der technischen Implementierungen von Instantmessaging verfügen über ein Status-Icon, wodurch für den Sender ersichtlich wird, ob der Ansprechpartner derzeit verfügbar ist oder, falls diese Informationen hinterlegt wurden, sich der Empfänger gerade in einem Meeting befindet oder nicht gestört werden möchte. Eine wesentliche Komponente der persönlichen Kommunikation, die augenblickliche emotionale Befindlichkeit unseres Gegenübers durch Mimik, Gestik und Körpersprache einzuschätzen, steht bei virtuellen Begegnungen allerdings nicht zur Verfügung. Hier könnte bereits eine Verbesserung erreicht werden, indem für den Anrufenden oder Chatpartner die aktuelle selbsteingeschätzte Befindlichkeit des Kommunikationspartners sichtbar wird. Ein denkbares Realisierungsmodell könnte eine erweiterte Ansicht des Statusbuttons darstellen, zB. durch farbliche oder gestalterische Differenzierung. Obwohl hinsichtlich Form und Farbe von einem allgemeingültigen (evtl. Kulturabhängigen) Verständnis ausgegangen werden kann, wäre es von Vorteil in einem Workshop die "Stimmungs-Symbole" gemeinsamen zu erarbeiten, um gleichzeitig die Gelegenheit zu nutzen bestehende Kommunikationsprobleme zu thematisieren bzw. Spielregeln im Umgang mit den Emotionssignalen festzulegen.

Die Simulation von zufälligen Begegnungen in virtuellen Welten hat eher Spielcharakter und wird sich daher weniger im unternehmerischen Umfeld durchsetzen lassen, wobei bereits Ansätze existieren, wenn man zB. die Interaktionsmöglichkeit zwischen Nutzern bedenkt, die gleichzeitig auf Ressourcen oder Informationen zugreifen.

Netzwerke, wie sie sich bei Interessensgemeinschaften im Internet ausgebildet haben, könnten auch übertragen auf die Strukturen innerhalb eines Unternehmens Vorteile bieten. So ist eine Visualisierung der Verknüpfungen von Kontakten vergleichbar zu Freundschaftsattributen hilfreich, um neue Kontakte zwischen Mitarbeitern herzustellen, die möglicherweise zu einer Verkürzung der Kommunikationswege führen könnte.

Ein weitere Aspekt ist die Untersuchung der Kommunikation hinsichtlich Kommunikationspartner und Inhalt. Die Idee anhand von festgelegten Schlüsselworten eine Kategorisierung des Mailverkehr und evtl. sogar von Telefonaten durchzuführen, könnte eine effiziente Möglichkeit der Kommunikationsanalyse bieten. Daraus ließe sich ein relativ genaues Abbild der Kommunikationswege und Interaktionen ableiten und zum anderen könnte eine automatisierten Auswertung zu Vergleichen mit bestehenden Prozessmodellierungen⁶ herangezogen werden, wodurch Diskrepanzen zwischen Plan- und Ist-Prozessen schneller bei geringerem Aufwand identifiziert und ggf. angepasst werden können. Fraglich bleibt, ob ein solches Szenario praktisch umgesetzbar ist, da einerseits die Qualität der Auswertung stark von einer unternehmensweit einheitlichen Klassifizierung jeder einzelnen Mail abhängt als auch im betrieblichen Umfeld aus Sicht des Arbeitnehmerschutzes vertreten durch den Betriebsrat vermutlich gegenüber einer solchen technischen Realisierung deutliche Vorbehalte bestehen.

Literaturverzeichnis

- [Fa01] Fank, M.: Einführung in das Informationsmanagement. Oldenbourg, München, 2001.
- [HL05] Heinrich, L.; Lehner, F.: Informationsmanagement, Oldenbourg, München, 2005.
- [No07] Novak, J.: Multiperspektivische Wissensvisualisierung für Wissensaustausch in heterogenen Community-Netzwerken. In (Oberweis, A. et al., Hrsg.): eOrganisation 8. internationale Tagung Wirtschaftsinformatik, Band 2. Universitätsverlag Karlsruhe, 2007; S. 611-628.
- [SK99] Schwarzer, B.; Krcmar, H.: Wirtschaftsinformatik. Schäffer-Poeschel, Stuttgart, 1999.
- [VG01] Voß, S.; Gutenschwager, K.: Informationsmanagement. Springer, Berlin, 2001.

⁶ Die „Verwendung von Referenzmodellen zur Prozessmodellierung erfordert viel Zeit und Know-how bzgl. Modellierungsmethodik“ [SK99], S. 110. Dieser Nachteil könnte durch eine automatische Generierung der Abhängigkeiten überwunden werden, vorausgesetzt es läßt sich eine vergleichbare Qualität erreichen.

Wiederverwendbarkeit von Prozessspuren in prozessintegrierten Werkzeugen

Michael Comanns
Softlab Group (BMW Group Company)
michael.comanns@softlab.de

Abstract: Industrielle Designprozesse zeichnen sich im Allgemeinen durch eine hohe Komplexität aus und stellen große Ansprüche an das Fachwissen und die Erfahrung der Design-Ingenieure. Eine Verbesserung solcher Designprozesse durch rechnergestützte Informationssysteme ist ein viel versprechender Ansatz zur Steigerung der Wettbewerbsfähigkeit industrieller Unternehmungen. Ein rechnergestütztes Informationssystem muss in der Lage sein Wissen um die Designprozesse zu sammeln, zu vergleichen und zur Verfügung zu stellen. Dies geschieht durch Anleitung innerhalb von prozessintegrierten Software-Werkzeugen mittels der Interpretation von Prozessfragmenten. Der Fokus dieser Arbeit liegt auf der Vergleichbarkeit und Wiederverwendbarkeit von solchen Prozessspuren.

1 Einleitung

Wie lässt sich Wissen definieren, formalisieren und klassifizieren? Gerade in hochtechnisierten Branchen stellt die Erfahrung der Mitarbeiter oft das größte Potenzial eines Unternehmens dar. Informationssysteme sollen Produktionszyklen verkürzen und die Qualität der Produkte erhöhen. Ein populärer Ansatz hierfür ist die Zerlegung komplexer (Design-)Aktivitäten in Prozesse und Prozessfragmente. Die Herausforderung für moderne Informationssysteme ist es, solch einen prozessorientierten Ansatz zu unterstützen. Des weiteren ist es unabdingbar als Teil von solchen Systemen eine Reihe von Werkzeugen zur Verfügung zu stellen, welche eine Entwicklung von komplexen und nicht-deterministischen (Design-)Aktivitäten überhaupt erst möglich machen. Diese Werkzeuge müssen in das System integriert, und an die entwickelten Modelle und Methoden angepasst sein. Ein solches System baut auf einer *Experience Base* aus gesammeltem Wissen auf.

Ziel ist es ein bestehendes Informationssystem dahingehend zu erweitern, dass es nicht nur in der Lage ist Wissen in Form von vordefinierten Prozessen zu verarbeiten, sondern auch eine Möglichkeit zu schaffen Arbeitsschritte untereinander zu vergleichen. Der Schwerpunkt dieser Arbeit liegt ganz klar auf der Unterstützung des Benutzers in seinen Entscheidungsfindungsprozessen und nicht in der Automatisierung von Abläufen oder einer Form von Fehlerkorrektur. Diese Arbeit gliedert sich wie folgt, zunächst werden Hintergrund und Problemstellung beleuchtet. Danach wird der Lösungsansatz umschrieben und im letzten Kapitel werden die gewonnen Ergebnisse präsentiert.

2 Hintergrund und Problemstellung

2.1 Wissenstransformation und Wissensverwendung

Wissen und Wissensmanagement, als abstrakte Begriffe, sind kritische Faktoren in modernen Entwicklungsprozessen. Der Begriff Wissensmanagement umfasst ein sehr breites Spektrum. Wissen kann die Expertise von Mitarbeitern sein, die Formalisierung eines Prozesses und vieles mehr. Eine formale Abgrenzung des Begriffes ist offensichtlich notwendig. Ein verbreitetes und anerkanntes Modell ist das SECI-Modell von Nonaka [NT95]. Dieses Modell beschreibt Wissensmanagement als Wissenserschaffung und Wissenstransformation. Grundsätzlich wird dort Wissen in *Explizites* und *Implizites* Wissen unterteilt. Implizites Wissen bezeichnet Wissen in Form von Erfahrungswerten oder erlernten Fähigkeiten. Explizites Wissen dagegen bezeichnet strukturiertes Wissen, wie beispielsweise formalisierte Prozesse. SECI steht nun für die einzelnen möglichen Transformationsschritte *Socialization*, *Externalization*, *Combination* und *Internalization*. Das Hauptaugenmerk dieser Arbeit liegt auf dem Schritt der *Socialization* und somit einer automatisierten Transformation von implizitem Wissen. Dies bedeutet implizites Wissen soll nicht mehr ausschließlich durch Expertise von menschlichen Akteuren repräsentiert werden, sondern durch Komponenten eines Informationssystems den handelnden Entwicklern zur Verfügung gestellt werden.

2.2 PRIME-Rahmenwerk

Das PRIME Rahmenwerk ist ein vom Lehrstuhl für Informatik 5 im Rahmen des IMPROVE-Projekts der Deutschen Forschungsgesellschaft (DFG) entwickeltes Rahmenwerk zur prozessintegrierten Unterstützung von Designprozessen innerhalb der kunststoffverarbeitenden Industrie. Das verwendete Environment-Metamodell basiert auf dem NATURE-Modell aus [JRSD99]. Die grundlegenden Elemente des Modells sind *Contexts*, *Situations* und *Intentions*. Eine detaillierte Erklärung der einzelnen Elemente kann [PWD⁺99] entnommen werden. Die Prozessunterstützung der Benutzer erfolgt über teilweise oder vollständig prozessintegrierte Werkzeuge. Die Benutzer werden im Rahmen der *Method Guidance* [JLW99] durch vorgefertigte Prozessfragmente geführt. Die Idee dieser Arbeit ist es das Rahmenwerk um den Aspekt des *Method Advising* zu erweitern und so den Benutzer in seinen Entscheidungsfindungsprozessen zu unterstützen.

3 Lösungsansatz

In dieser Lösung soll die Wiederverwendung von Wissen vor dem Hintergrund des SECI-Modells verwendet und in das PRIME Rahmenwerk eingebettet werden. Der Benutzer und seine Entscheidungen stehen im Zentrum dieses Systems. Eine vollständige Automatisierung ist nicht erwünscht. Der Benutzer möchte lernen und nicht angeleitet werden. Dieser

Aspekt stellt eine klare Abgrenzung zwischen *Method Advising* und *Method Guidance* dar. Im Letzteren werden von Experten verifizierte Modelle gebraucht um den Benutzer anzuleiten. Im *Method Advising* werden die Erfahrungen von Benutzern gespeichert, ausgewertet und interpretiert um anderen Nutzern die Möglichkeit des Lernens zu geben.

Für diese Wiederverwendung von bereits gemachten Erfahrungen (*Traces*) ist es notwendig eine Metrik auf diesen zu definieren und einen *Retrieve*-Algorithmus zu entwerfen, welcher zu einer gegebenen Anfrage des Benutzers eine Menge von möglichen fortführenden Maßnahmen ausgibt. In diesem Fall existieren zwei atomare Elemente, *Products* und *ExecutableContexts*. Jede *Situation* definiert sich über die ihr zugeordnete Menge von *Products* und jeder *Context* ist entweder ein Vektor von *Contexts* oder ein *ExecutableContext* und somit atomar. *Traces* sind geordnete Tupel aus *Situations* und *Contexts* die mit einer *Situation* beginnen und enden, und in denen auf eine *Situation* immer ein *Context* folgt. In dieser Arbeit werden, statt einer Distanzfunktion, Ähnlichkeitsfunktionen auf dem Wertebereich $[0, 1]$ verwendet. Die verschiedenen Ähnlichkeitsfunktionen für die einzelnen Elemente können [Com06] entnommen werden. Die Ähnlichkeit für *Situations* α_S basiert auf dem *Jaccard Koeffizient*, während die Ähnlichkeit für *Contexts* α_C auf einem selbstentwickelten Homomorphismus und der *Levenshtein Distanz* basiert [ZADB06]. Aus diesen beiden Ähnlichkeitsfunktionen läßt sich die Ähnlichkeit zweier *Traces* wie folgt bestimmen.

$$\alpha_T(t_1, t_2) = \frac{1}{2} * (\Lambda_S(t_1, t_2) + \Lambda_C(t_1, t_2))$$

Die Funktion $\alpha_T : T \times T \rightarrow [0, 1]$ bestimmt somit die relative Ähnlichkeit zweier *Traces* auf dem Intervall $[0, 1]$. Λ_S ist die gemittelte Ähnlichkeit aller *Situations* und Λ_C die gemittelte Ähnlichkeit aller *Contexts* eines *Trace*. Basierend auf der definierten Metrik und den Ähnlichkeitsfunktionen, lässt sich der FIST-Algorithmus, für (Finding Similar Trades), definieren. Ausgangsbasis für den Algorithmus ist die aktuelle *Situation* in der sich ein Benutzer befindet, sowie alle *Contexts* und *Situations* die ihn zu dieser geführt haben. Der Algorithmus erhält also als Eingabe den aktuellen *Trace* t_a des Benutzers, einen Ähnlichkeits-Schwellenwert λ , sowie einen Scoping-Faktor κ . Diese beiden Faktoren stellen wichtige Skalierungsgrößen des Algorithmus dar. Der Algorithmus identifiziert zunächst alle *Situations* innerhalb der *Experience Base*, deren Ähnlichkeit zur aktuellen *Situation* größer als λ ist. Die *Traces*, die diese *Situations* enthalten werden auf die durch den Scoping-Faktor κ spezifizierte Länge gekürzt. Die so entstandenden *Traces* werden nun mit dem aktuellen *Trace* des Benutzers verglichen und alle die, deren Ähnlichkeit größer λ ist werden der Ergebnismenge hinzugefügt.

4 Ergebnisse

Gegenstand der vorliegenden Arbeit war die Untersuchung von Methoden und Ansätzen zur Erarbeitung und Wiederverwendung von Wissen und Erfahrung. Diese Arbeit ist angesiedelt im Bereich der prozessorientierten Rahmenwerke und der erfahrungsbasierten

Systeme. Ziel war die Entwicklung eines Werkzeugs zur erfahrungsbasierten Wiederverwendung von Prozessspuren. Die technische Umsetzung, sowie ein Fallbeispiel können [Com06] entnommen werden. Ziel dieser Arbeit war es das PRIME Rahmenwerk um den Aspekt des *Method Advising* zu erweitern um so den Schritt der Socialization zu realisieren. Ziel des Systems ist nicht die Automatisierung von Designprozessen. Der Benutzer bleibt weiterhin im Zentrum des Systems. *Method Advising* soll im Gegensatz zur *Method Guidance* den Benutzer nicht anleiten, sondern ihn in seinem Entscheidungsfindungsprozess unterstützen.

Die Validierung des Werkzeugs hat mehrere Aspekte aufgezeigt. Ähnlichkeit-Schwellenwert und Scoping-Faktor stellen sehr wichtige Parameter dar, die sich direkt auf die Güte des Ergebnisses auswirken. Der Benutzer ist angehalten an dieser Stelle sinnvoll mit diesen Parametern umzugehen um brauchbare Ergebnisse zu bekommen. Des weiteren hat die Validierung die Möglichkeiten des *Method Advising* aufgezeigt. Innerhalb von vollständig prozessintegrierten Werkzeugen wie dem Fließbildwerkzeug ist *Method Advising* mindestens gleich mächtig zur *Method Guidance* oder kann diese sogar überflügeln. Bei der Verwendung von lose prozessintegrierten Werkzeugen stellt das *Method Advising* zur Zeit sogar die einzige Möglichkeit dar den Benutzer direkt bei der Durchführung komplexer Prozesse zu unterstützen. Die Validierung hat aber auch die Grenzen dieses Prinzips aufgezeigt. *Method Advising* ist nicht in der Lage eine Fehlerkorrektur vorzunehmen, da die Grundannahme für das Funktionieren eine möglichst fehlerarme *Experience Base* ist. Die Güte, sowie die Granularität der Ergebnisse hängt direkt von der Güte und der Granularität des Environment-Modell ab. Für das *Method Advising* sind deshalb sorgfältig modellierte Prozessfragmente genauso wichtig wie für die *Method Guidance*. Beide Systeme, *Method Guidance* und *Method Advising*, sollen auch nicht miteinander konkurrieren, sondern sich gegenseitig ergänzen um dem Benutzer eine bestmögliche Unterstützung zu liefern.

Literatur

- [Com06] Michael Comanns. Werkzeugunterstützung für erfahrungsbasierte Wiederverwendung von Prozessspuren. Diploma thesis, RWTH Aachen University, 2006.
- [JLW99] Matthias Jarke, Thomas List und Klaus Weidenhaupt. A Process-Integrated Conceptual Design Environment for Chemical Engineering. In *International Conference on Conceptual Modeling / the Entity Relationship Approach*, Seiten 520–537, 1999.
- [JRSD99] Matthias Jarke, Colette Rolland, Alistair Sutcliffe und Ralf Dömges, Hrsg. *The NATURE of Requirements Engineering*. Shaker, 1999.
- [NT95] Ikujiro Nonaka und Hirotaka Takeuchi. *The Knowledge-Creating Company*. Oxford University Press, 1995.
- [PWD⁺99] K. Pohl, K. Weidenhaupt, R. Dömges, P. Haumer, M. Jarke und R. Klamma. PRIME: Towards Process-Integrated Environments. *ACM Transactions on Software Engineering and Methodology*, 8:4:343–410, 1999.
- [ZADB06] Pavel Zezula, Guiseppe Amato, Vlatislav Dohnal und Michal Batko. *Similarity Search, The Metric Space Approach*. Springer, 2006.

Entwicklung eines Spiels für Mobile Endgeräte mit Unterstützung wie GPS und UMTS auf Basis der Java2 Micro-Edition

Michaela Frenzel

Private FernFachhochschule Darmstadt

michaelafrenzel@gmx.de

Art der Arbeit: Diplomarbeit
Betreuer/in der Arbeit: Dipl.-Inform. (FH) Michael Best

1 Motivation

1.1 Entwicklungsziel

Mobile Endgeräte spielen in der heutigen Zeit eine wichtige Rolle, wobei mittlerweile nicht nur Handys als mobile Endgeräte zählen. Auch die Verwendung von PDA's und speziell auch Navigationssystemen haben in den letzten Jahren immer mehr an Bedeutung hinzugewonnen. Kaum ein Autofahrer ist heute ohne Navigationssystem unterwegs (sei es integriert oder als externes Gerät).

Die Funktionalität von Navigationsgeräten lässt sich jedoch auch auf „spielerische“ Art und Weise nutzen, was Bestandteil der Diplomarbeit ist. Als Vorlage dient das bekannte Brettspiel „Scotland Yard“ oder auch „Schnitzeljagd“, welches bekannter weise von mehreren Spielern auf einem größeren und möglichst nicht zu übersichtlichem Outdoor-Terrain (z.B. ein Park von mindestens 200 x 200 Metern) gespielt wird

Bisherige Umsetzungen von z.B. Scotland Yard für mobile Endgeräte sind auf viel mehr Aktivität der Mitspieler ausgelegt. So können die Spieler nicht ihre exakten Positionen (also GPS-Positionen) angeben, sondern lediglich konkrete Angaben von Strassen oder Plätzen. Das hat den Nachteil, dass die Spieler eine grundlegende (Straßen)-Kenntnis ihrer Spielumgebung benötigen und das Spiel auf offenem Gelände wie Parks nur sehr eingeschränkt nutzbar ist. Außerdem müssen die Positionen auf Kommunikationswegen wie SMS, MMS oder Anrufen verbreitet werden, was für die Handyrechnungen der Spielteilnehmer nicht positiv zu werten ist. Somit stellt sich als besondere Herausforderung an die Lösung, die Datenmengen möglichst effizient zu halten und zu verbreiten.

2 Architektur und Technologien

2.1 Überblick

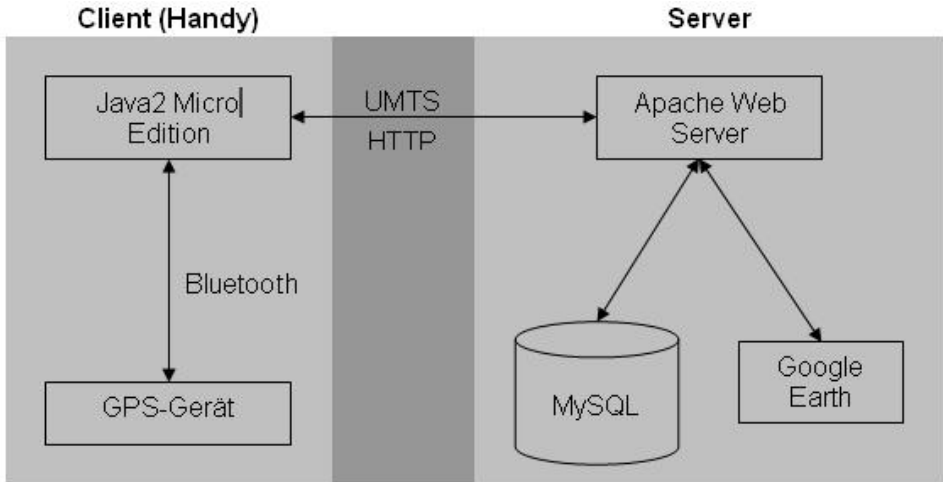


Abb.: 2.1

Die obige Abbildung soll als Überblick der Architektur und verschiedenen Technologien der Umgebung dienen. Die Architektur und die einzelnen Technologien werden in den folgenden Abschnitten genauer beschrieben.

2.2 Server

Der Server dient als Plattform für die einzelnen laufenden, aber auch abgeschlossenen Spiele. Er stellt die Kommunikation zwischen den Spielteilnehmern sicher, speichert die entsprechenden Datensätze in einer Datenbank und stellt zusätzlich noch eine Internet-Oberfläche zur Verfügung, mit der die Spieldaten zu einem späteren Zeitpunkt ausgewertet werden können.

Für die Implementierung des Servers werden folgende Komponenten bzw. Technologien verwendet:

- Apache Web Server (Server-Software)
- PHP (Skriptsprache, die vorwiegend bei Webseiten verwendet wird)
- MySQL (Open Source Datenbank)
- Google Earth (Open Source Tool)

2.3 Client

Der Client bildet das eigentliche Spiel ab. Dieser baut eine Verbindung zum Server auf und übermittelt und empfängt die notwendigen Daten. Diese Daten werden dann zusätzlich aufbereitet und dem Spielteilnehmer zur Verfügung gestellt.

Für die Implementierung des Clients werden folgende Technologien verwendet:

- Java2 Micro-Edition
- Bluetooth
- GPS
- UMTS

2.4 Kommunikation zwischen Client und Server

Die Kommunikation der Clients untereinander erfolgt über den Web Server. Dieser wird vom Client aus mittels HTTP (UMTS) angesprochen und bekommt dann die Daten (GPS-Position) des Clients übergeben. Die Daten werden dann in der MySQL Datenbank gespeichert. Zusätzlich werden die für den Client benötigten Daten aus der Datenbank ermittelt und ebenfalls per HTTP an den Client zurückgesendet. Der Client empfängt diese Daten und bereitet diese dann entsprechend auf und stellt sie dem eigentlichen Spieler dann zur Verfügung.

Da die Datenübertragung zum Server und der Datenempfang vom Server preiseffizient gestaltet werden soll, wurde UMTS als Technologie gewählt, da die Preise von UMTS-Paketen in den letzten Jahren immer mehr gesunken sind. Über das HTTP-Protokoll werden lediglich die benötigten Daten an den Server übermittelt (im Aufruf der PHP-Seite als Parameter). Der Server wertet die Daten aus und speichert diese entsprechend in der Datenbank und gibt lediglich eine geringe Menge an Daten an den Client zurück (einzelne Zeichen, zur Bestätigung des Erfolgs beim Speichern in der Datenbank, Fehlermeldungen). Es werden keine grafischen Elemente übermittelt, sodass hier eine kostengünstige Variante vorliegt. Die eigentliche Aufbereitung der Daten nimmt dann wieder der Client selbst vor, sodass auch hier keine Verbindung zum Internet mehr vorliegt.

3 Ausblick

Ziel dieser Diplomarbeit ist es, eine Spielumgebung zu entwickeln, die die Möglichkeiten des GPS-Empfangs mit einem Handy nutzt. Es soll dargestellt werden, dass sehr verschiedene Technologien durchaus miteinander verbunden werden können.

Die Nutzung kann nicht nur im „spielerischen“ Bereich liegen, sondern auch für andere Ideen genutzt werden (z.B. Aufzeichnen von Wander- oder Fahrradrouten und entsprechende grafische Darstellung mit Google Earth).

Literaturverzeichnis

- [Br06] Breymann, Ulrich (2006): Java ME – Anwendungsentwicklung für Handys, PDA und Co., München / Wien: Carl Hanser Verlag
- [Fr04] Franke, Jochen (2004): Dynamische Webseiten mit PHP, Poing: Franzis Verlag GmbH
- [CPM04] Converse, Tim; Park, Joyce; Morgan, Clark (2004): PHP 5 and MySQL Bible, Indianapolis, Indiana (USA): Wiley Publishing, Inc.

Apache HTTP Server Project: <http://www.apache.org>
MySQL AB: <http://dev.mysql.com/doc>
Google Earth: <http://earth.google.com>
Bluetooth: <https://www.bluetooth.org/>
PHP: <http://de.php.net/>

Kommunikationssensitiver dynamischer Lastausgleich in Multi-Clustern mit Raumfüllenden Kurven

Mathias Goldau

Mathias.Goldau@gmx.de

Abstract: Mittlerweile betreiben viele Forschungsinstitute Mini-Clusterrechner mit typischerweise 10-30 Knoten. Liegen solche Clusterrechner gekoppelt vor, spricht man von einem Multi-Cluster. Ebenso existieren auch viele eng gekoppelte Cluster-Applikationen wie zum Beispiel zahlreiche zelluläre Automaten. Will man nun solche Anwendungen auf einem Multi-Cluster ausführen kann dies eine starke Belastung des Verbindungsnetzwerkes, welches die einzelnen Cluster koppelt, bewirken. Auch steht das Verbindungsnetzwerk meist nicht exklusiv für den Multi-Clusterbetrieb zur Verfügung, so dass andere Anwendungen ebenso Netzwerkressourcen in Anspruch nehmen können. Wünschenswert wäre daher eine Möglichkeit dynamisch auf Kommunikationsengpässe reagieren zu können. Der erste Ansatz für einen solchen Lastausgleich sind Partitionierungsschemata. Die Diplomarbeit untersuchte in wie weit sich Raumfüllende Kurven für einen solchen Lastausgleich eignen. Dazu gehe ich zunächst auf einige weitverbreitete Partitionierungsmethoden wie Striping, Checkerboard und METIS ein. Anschließend erfolgt die Auswahl der Hilbert Kurve als viel versprechende Raumfüllende Kurve, der mit einem diffusionsbasierten Lastausgleich kombiniert wird.

1 Einleitung

Parallele eng gekoppelte Anwendungen arbeiten auf einem großem Problemraum, welcher in viele Teile partitioniert wird. Leider erfolgt bei solchen Anwendungen die Berechnung der Partitionen nicht unabhängig voneinander, da von Zeit zu Zeit Randwerte der Partitionen kommuniziert werden müssen. Oftmals erfolgt die Berechnung der Lösung in mehreren Schritten oder Iterationen, welche aus Berechnung und Kommunikation bestehen. Ein nächster Schritt kann erst begonnen werden, wenn der vorherige von *allen* Knoten abgeschlossen wurde.

Die hohe Belastung des Verbindungsnetzwerkes eines Multi-Clusters resultiert aus überlagerter Kommunikation von mehreren Knoten aus verschiedenen Clustern. Wenn mehrere Knoten eines Clusters, jeweils mit Knoten aus anderen Clustern kommunizieren, so müssen sich die Kommunikationspaare die zur Verfügung stehenden Netzwerkressourcen des Verbindungsnetzwerkes teilen. Ebenso können Multi-Cluster-externe Applikationen das Verbindungsnetzwerk benutzen und die ohnehin schon knappen Netzwerkressourcen beanspruchen. Das Resultat ist eine hoch dynamische Veränderung der Netzwerkressourcen des Verbindungsnetzwerkes, welches einer dynamischen Anpassung bedarf.

In einem Ansatz nach Otero [OCBL04] kann man auf solche Kommunikationsengpässe reagieren, wenn Problemraumelemente von einer kommunikativ überlasteten Partition zu weniger belasteten Partionen verlagert werden. Dabei geht man je Iterationsschritt von einer Berechnungsphase und einer nachfolgenden Kommunikationsphase aus. Wenn Knoten, welche lange für Kommunikation brauchen, durch eine verkürzte Berechnungsphase früher beginnen können, führt das zu einer kürzeren Gesamtzeit.

Partitionierungsstrategien sind der erste Ansatzpunkt, um diesen Ansatz dynamisch zu realisieren. Im Folgenden wird kurz auf populäre Methoden wie Striping, Checkerboard und METIS eingegangen, die hinsichtlich eines kommunikationssensitiven dynamischen Lastausgleichs beleuchtet werden. Dabei zeigt sich, das Striping hierfür äußerst gut geeignet ist. Lediglich das schlechte Verhältnis von Randelementen zur Partitonsgröße: Surface / Volume [HW02, Zum00] einer Striping-Partition ist von großen Nachteil.

In letzter Zeit wurden auch Raumfüllende Kurven [Sag94] unter anderem auch als Partitionierungsstrategie bekannt und untersucht [SW03, Zum03]. In meiner Arbeit gehe ich der Frage nach, ob sich Raumfüllende Kurven für eine Implementation des Ansatzes von Otero [OCBL04] eignen. Leider gibt es eine enorme Vielzahl an Raumfüllenden Kurven und alternativen Indizierungsmethoden.

Aufgrund folgender Eigenschaften ist die Hilbert Kurve jedoch am Besten geeignet:

- Für beliebige Dimensionalität verfügbar, beispielsweise \mathbb{R}^d [AN00]
- Gutes Surface / Volume Verhältnis für induzierte Partitionen [HW02, MJFS01]
- Existenz effizienter Algorithmen für Umrechnung: Problemraum \leftrightarrow Indexraum. [Law00, But71]

Eine weitere wichtige Eigenschaft ist die Anwendbarkeit der Raumfüllenden Kurve auf mindestens $n \times \dots \times n$ Problemräume. Gerade aber bei der Hilbert Kurve ist dies mit $2^k \times 2^k$ nicht gegeben. Allerdings bietet meine Arbeit eine leicht modifizierte Hilbert-Kurve an, welche dies leistet.

2 Algorithmus

Jede Raumfüllende Kurve kann auch als eine Perlenkette durch den Problemraum betrachtet werden. Dabei entsprechen die “Perlen” den Problemraumelementen. Um nun eine Partitionierung mit einer solchen Raumfüllenden Kurve zu gewinnen, unterteilt man diese Perlenkette in gerade soviel Intervalle wie es Knoten im Multi-Cluster gibt. Daraus folgt, das nun die Problemraumelemente an die benachbarten Intervalle verschoben werden können.

Wenn eine solche induzierte Partition kommunikativ überlastet ist, so können einfach die Intervallgrenzen angepasst werden. *Wann* und *wieviele* Problemraumelemente verschoben werden, kann nun nach verschiedenen Lastausgleichs-Verfahren [Buy99] erfolgen. Zentrale Verfahren haben den Nachteil, dass jeder Knoten aus einem Cluster die Lastinforma-

tionen einer zentralen Instanz übermitteln muss. Damit das Verbindungsnetzwerk, welches die Cluster koppelt, nicht noch stärker belastet wird, sind für Multicluster besonders verteilte Verfahren, wie diffusionsbasierte Lastausgleichsverfahren geeignet.

Eine durch Rechenlast oder Kommunikation überlastete Partition kann somit leichter Elemente an benachbarte Partitionen übergeben. Daher wird zum einen die Rechenlast reduziert und zum anderen, falls auch eine kommunikative Überlastung vorliegt, kann die Kommunikation wegen der reduzierten Rechenlast eher beginnen.

3 Ergebnisse

Meine Arbeit bietet zwei wesentliche Ergebnisse:

1. Raumfüllende Kurven eignen sich nicht für *kommunikationssensitiven* dynamischen Lastausgleich in Multi-Clustern.
2. Hingegen eignen sie sich sehr gut für *berechnungssensitiven* Lastausgleich in Multi-Clustern.

Da die Randelemente einer Partition, bezüglich des Hilbert-Indexes, nahezu beliebig in dem zugehörigem Intervall verteilt sein können, kann man solche Partitionen nicht ohne weiteres verkleinern. Wenn man beispielsweise Randelemente einer Partition über das schwache Verbindungsnetzwerk kommuniziert werden, so lohnt es sich nicht diese Elemente auf Clusterpartitionen zu verteilen. Denn alle Partitionen, die solche speziellen (langsamen) Randelemente kommunizieren, haben dadurch längere Kommunikationsphasen. Somit existieren letztendlich zu wenige Partitionen, welche Berechnungslast aufnehmen könnten, um die kommunikativ überlasteten Partitionen zu entlasten. Das heisst, Raumfüllende Kurven bieten keine Abschirmung von Kommunikationsengpässen, wie zum Beispiel Striping.

Die induzierten Partitionen von der Hilbert Kurve bieten aber eine zu Striping überlegenes Surface zu Volume Verhältnis. Simulationen mit einem Netzwerksimulator [OMN06] ergaben daher, dass die Partitionierung mit der Hilbert Kurve, trotz optimiertem Stripings, zu kürzeren Iterationsschritten führen kann. Denn nur, wenn viele Rechner und viele Cluster beteiligt sind kann die Hilbert Kurve mit weitaus geringerem Kommunikationsaufwand punkten. Dies ist insbesondere bei Zweierpotenzen 2^k gegeben.

4 Zusammenfassung

Raumfüllende Kurven eignen sich zwar nicht für dynamischen *kommunikationssensitiven* Lastausgleich in Multi-Clustern, wohl aber für den normalen *berechnungssensitiven* Lastausgleich. Ebenso muss man mit eventuell entarteten Partitionen rechnen, welche ein schlechtes Surface zu Volume Verhältnis aufweisen. Ein vielversprechenderer Ansatz sind heterogene Partitionierungsstrategien, welche hierarchisch strukturiert sind.

Literatur

- [AN00] Jochen Alber und Rolf Niedermeier. On Multidimensional Curves with Hilbert Property. *Theory of Computing Systems*, 33(4):295–312, 2000.
- [But71] Aethur R. Butz. Alternative Algorithm for Hilbert Space-Filling Curve, 1971.
- [Buy99] Rajkumar Buyya. *High Performance Cluster Computing: Architectures and Systems*. Prentice Hall PTR, 1999.
- [HW02] Jan Hungershöfer und Jens-Michael Wierum. On the Quality of Partitions based on Space-Filling Curves. 2002.
- [Law00] J. Lawder. Calculation of Mappings between One and n-dimensional Values Using the Hilbert Space-Filling Curve, 2000.
- [MJFS01] Bongki Moon, H. V. Jagadish, Christos Faloutsos und Joel H. Saltz. Analysis of the Clustering Properties of the Hilbert Space-Filling Curve. *Knowledge and Data Engineering*, 13(1):124–141, 2001.
- [OCBL04] Beatriz Otero, Jos M. Cela, Rosa M. Badia und Jess Labarta. A Domain Decomposition Strategy for GRID Environments. *Lecture Notes in Computer Science*, 3241:353 – 361, 2004.
- [OMN06] OMNETPP.ORG. OMNeT++ Discrete Event Simulation Tool, 2006.
<http://www.omnetpp.org>.
- [Sag94] Hans Sagan. *Space - Filling Curves*. New York, Springer, 1994.
- [SW03] Stefan Schamberger und Jens-Michael Wierum. Graph Partitioning in Scientific Simulations: Multilevel Schemes versus Space-Filling Curves, 2003.
- [Zum00] Gerhard Zumbusch. On the Quality of Space-Filling Curve Induced Partitions, 2000.
- [Zum03] Gerhard Zumbusch. *Parallel Multilevel Methods: Adaptive Mesh Refinement and Load-balancing*. Teubner, 2003.

Challenges across two worlds: The concerns of IT Change Managers to deal with SAP WebAS Java and WebAS ABAP

Matthias - Claudius Kneissl (Galileo Group AG)

University of Passau
Information Management
matthias.kneissl@galileo-group.de

Type of work: Academic Research Partnership with Galileo Group AG

Abstract: Since release 6.40 the SAP Netweaver platform consists of two different stacks, WebAS ABAP and WebAS Java. Although both stacks can be installed within one system, the technology and development processes are completely different. This article discusses the differences regarding IT Change Management of ABAP and Java and outlines potential problems, which have to be solved to ensure an ITIL compliant change management process for software factories.

1 Introduction

With the release of the Netweaver platform, SAP changed its strategy and offers parallel an ABAP and Java stack. Both technologies are platform independent. While the development process of ABAP is centred to the mainframe architecture, Java has a classical client - server development process. In a typical enterprise IT scenario the WebAS Java is used for web based portal technologies, whereas the master data and the business logic reside in the ABAP stack. Other possible and common examples for the usage of both stacks are Business Intelligence or the SAP Process Integration engine (PI), formerly known as exchange infrastructure (XI). Due to the current development and plans of several companies to migrate their J2EE developments on the WebAS Java, it is expected, that the importance of this technology and diffusion rate will increase.

The close linkage between WebAS ABAP and WebAS Java and their usage scenarios often enforce changes, which have to be performed in more than one system. The proper management of changes in different technologies, the linkage of them and their timely delivery gets more difficult with a complex system landscape and various system technologies. In enterprise IT environments, especially the differences of the change management process in WebAS Java and WebAS ABAP come up and have to be solved to ensure a streamlined IT software factory management process.

2 Organizational Differences

Regarding the scenario of a software factory and an ITIL compliant configuration and change management process, functional change requests have to be approved by the Change Advisory Board and have to get a confirmed release date for productive setting. The technology specific change requests inherit the release date. In case of different technologies, usually the responsible persons for the changes are different. One of the most important ITIL principles is, that no change should be performed without a proper assignment.

In the WebAS ABAP stack, this way of handling change requests is not a problem. A functional change request, which is technology independent, is linked to one or more SAP ABAP change requests. Those requests can be assigned to the particular responsible person. The tasks, which can be created under this change request, may be assigned to the different developers working on the change request. After the release of the different tasks, the technology specific ABAP change request can be released. A change in the coding or a customizing can only be performed after a change request and task has been created.

However, the corresponding WebAS Java workflow is totally different. The basic rule, that no change should be performed without a change request assignment is not in place. Software developers can change source codes, create local activities and can bundle those afterwards to a change request. The problem with this workflow is, that activities can be created without an assignment and subordinated to change requests. Those will go through the workflow oriented development cycle as long as the WebAS Java change management service is in place. While in the WebAS ABAP stack, the assignment to a change request can be performed by creating a task for a particular developer, this is not possible in Java. This means, that the management and assignment of change requests can not be done in the change management service of Java, but has to be done outside and somehow linked. It is an organizational challenge to introduce a handling procedure for WebAS Java change requests and their specific linkage to the overall technology unspecific change request.

One of the great challenges will be the definition of a technology independent development process to link the different steps to the actions, which have to be performed on system level. Without the support of a change management tool, which can handle those actions, it is likely to be a quite error-prone process.

3 Technical Differences

3.1 Linkage of change requests to releases and responsible persons

In the ABAP stack, several changes can be bundled into a single change request. This request usually is assigned to the responsible user and can consist of different tasks, which can be assigned to different developers. In the Java stack, it is currently not to pre-assign a change request to a responsible change manager, as the request is created upon releasing one or more activities. Without proper change management logic, it becomes difficult to streamline the software development cycle.

3.2 Integrated workflow concept in WebAS Java

As long as the WebAS Java change management service is used, sources are managed within tracks. Those contain a workflow as well as the specific sources. The development process is workflow oriented and strictly linear. The problem, which comes up is, that the WebAS ABAP development process does not have a built-in workflow. While the delivery process in Java is managed by a workflow, the ABAP delivery process is normally managed by adding transport requests to the import queue of the particular system. The proper and correct definition of a workflow on the ABAP side, which is adequate to the Java track based approach, is a challenge for IT managers, which has to be solved organizationally. Furthermore, the specific functional environments of the WebAS Java have to be mapped to the WebAS ABAP functional environments.

3.3 Differences in the functional environments

For the standard ABAP driven development procedure, three functional environments are usually used (development, quality and productive system). Within the Java development workflow, four functional environments are provided. In case of this standard scenario, either the ABAP development process has to be extended by a functional environment, or the development process in ABAP has to be adapted. The track concept in WebAS Java can only deliver one system per step. In case of a more complex system landscape with more than one productive system, several tracks with the integrated workflow have to be connected properly, to provide an automated delivery of more than one productive system. A challenge for each organisation will be the adoption of the ABAP development process to this workflow and the execution of the specific steps within a small timeframe to avoid inconsistencies within each technology.

3.4 Different workflow concept for Process Integration

The management of XI change requests can also be handled by the SAP track construct. A big challenge is the fact, that in contrast to the Java typical four system landscape, process integration development workflows are slightly adapted and provide a three system landscape. Changes in XI often lead to change requests in WebAS ABAP and WebAS Java. While for the Java part a workflow is provided, this is not the case for the ABAP part. This limitation is a big challenge, to manually align the steps in the ABAP part to the Java part. A possibility to bundle technology specific change requests to a functional change request and a streamlined management of the functional change would be very important at this point.

4 Conclusion and Future Work

The comparison of the two different technologies shows, that IT change managers have to handle a high challenge to fulfil ITIL compliance and a proper configuration and change management if both stacks are used and are linked to each other. This scenario is likely to be quite common, as SAP offers more and more web based applications like employee self services or an enterprise portal. The linkage of three different SAP specific workflows (i.e. ABAP, Java, PI) and maybe also another development workflow for legacy systems having is a great challenge. The next step will be the definition of a technology independent process with several scenarios, mapping the technology dependent workflow to the technology independent workflow.

References

- [EEE05] Hochstein, Axel/Zanrekow, Rüdiger/Brenner, Walter: ITIL as Common Practice Reference Model for IT Service Management: Formal Assessment and Implications for Practice, Proceedings of the 2005 IEEE International Conference on E-Technology, E-Commerce and E-Service, Honkong, 29.03.2005, Volume 00, IEEE, 2005.
- [ITL00a] Office of Government Commerce (OGC): ITIL - The key to Managing IT services - Best Practice for Service Support, 1. edition, TSO Publishing, United Kingdom (UK), 2000.
- [ITL00b] Office of Government Commerce (OGC): ITIL - The key to Managing IT services - Best Practice for Service Delivery, 1. edition, TSO Publishing, United Kingdom (UK), 2000.
- [ITL00c] Office of Government Commerce (OGC): ITIL - The key to Managing IT services - Best Practice for ICT Infrastructure Management, 1. edition, TSO Publishing, United Kingdom (UK), 2000.
- [ITL00d] Office of Government Commerce (OGC): ITIL - The key to Managing IT services - Best Practice for Application Management, 1. edition, TSO Publishing, United Kingdom (UK), 2000.
- [KN05] Kösegi, A.; Nerding R.; SAP Änderungs und Transportmanagement, Galileo Press, Bonn 2006
- [KTB05] Kessler, K.; Tillert, P.; Dobrikov, P.: Java Programmierung mit dem SAP Web Application Server, Galileo Press, Bonn, 2006
- [StOr05] Stumpe, S; Orb, J.: SAP Exchange Infrastructure, 1.Auflage, Galileo Press, Bonn, 2005

Konzeption und Realisierung eines Frameworks für Gruppenkommunikationsanwendungen auf mobilen Endgeräten unter Nutzung von IMS-Diensten

Andreas Pinhammer

Andreas.Pinhammer@gmx.de

Abstract: Ein Merkmal vieler Gruppenkommunikationsanwendungen ist der Zugriff auf gemeinsame Daten, wie etwa Kontakt- und Gruppeninformationen. Im Rahmen der Diplomarbeit wurde ein Framework entwickelt, welches den Zugriff auf diese Daten und deren Visualisierung für Symbian-basierte Mobiltelefone vereinfacht. Die Verwendung des Frameworks wurde anhand einer Anwendung für CoSIMS, ein auf IMS basierendes Gemeinschaftsprojekt der Deutsche Telekom Laboratories und des Fraunhofer Instituts für Offene Kommunikationssysteme (FOKUS), demonstriert.

1 Motivation

Virtuelle Gemeinschaften, so genannte Communitys, sind *der* Trend der letzten Jahre gewesen. Eine Community ermöglicht einer Gruppe von Anwendern miteinander Informationen und Erfahrungen auszutauschen. Populäre Onlinecommunitys wie XING, MySpace oder StudiVZ haben innerhalb kürzester Zeit die breite Masse der Internetanwender erreicht. Zusammen mit anderen Entwicklungen gehören Communitys zum sogenannten *Web 2.0*.

Im Mittelpunkt einer Community steht die Kommunikation zwischen ihren Mitgliedern. Diese wird von meist mehreren Gruppenkommunikationsdiensten ermöglicht. Bisher kann die Mehrzahl dieser Dienste nur über das Web verwendet werden. Unterwegs ist der Anwender von seiner Community abgeschnitten, obwohl Mobiltelefone technisch längst in der Lage dazu sind, solche Dienste zu ermöglichen.

Die Deutsche Telekom Laboratories und FOKUS möchten Community-orientierte Dienste auch in Mobilnetzen ermöglichen. Dafür entwickeln sie gemeinsam CoSIMS [Fra], eine auf dem IP Multimedia Subsystem (IMS) [Nok04] aufbauende Dienstplattform für Communitys.

2 Aufgabenstellung

Das Ziel der Diplomarbeit [Pin06] war die Entwicklung eines Frameworks, welches die Erstellung von Gruppenkommunikationsanwendungen erleichtert. Der Schwerpunkt lag

dabei auf dem Zugriff auf Daten, wie sie beispielsweise in Form von Kontaktinformationen auftreten können. Als Zielplattform wurde das auf dem Symbian-Betriebssystem basierende S60 (2nd Edition, Feature Pack 3) [Nok02] gewählt, die derzeit am weitesten verbreitete Plattform für Smartphones. Die Anwendung des Frameworks sollte anhand einer CoSIMS-Anwendung demonstriert werden.

Als Grundlage für die Anforderungen an das Framework dienten drei Szenarios, die Anwendungsmöglichkeiten für Community-orientierte Dienste und Gruppenkommunikationsfunktionalität im Alltag und Berufsleben aufzeigten. Eines der Szenarios stellte darüber hinaus auch die Grundlage für die Beispielanwendung dar.

3 Framework

Das Framework unterteilt sich in die Datenschicht, welche den Zugriff auf die Daten kapselt, und die Bereitstellung von UI-Komponenten, welche die Daten visualisieren. Die Umsetzung des Frameworks erfolgte in Symbian C++ [Nok06]. Der typische Aufbau einer Frameworkanwendung wird in Abbildung 1 gezeigt.

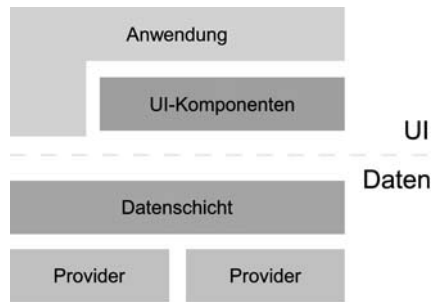


Abbildung 1: Schichtendiagramm einer Framework-Anwendung

Kern der Datenschicht ist die Abstraktion von der eigentlichen Datenquelle. Anwendungen können somit unabhängig von dieser implementiert werden. Die eigentlichen Daten werden von Plugins, sogenannten Provider, bereitgestellt.

Die Datenschicht wurde als Client-Server-Architektur umgesetzt. Ein datenhaltender Prozess (Server) beantwortet die Anfragen der einzelnen Anwendungen (Clients). Client-Server-Architekturen werden unter Symbian häufig zur Interprozesskommunikation verwendet. Ein direkter Zugriff auf Speicherbereiche eines anderen Threads oder Prozesses ist nicht möglich, da diese innerhalb unabhängiger Adressräume arbeiten.

Die UI-Komponenten stellen die von der Datenschicht gelieferten Informationen dar. Die Komponenten entsprechen dabei dem *Look-and-Feel* der S60-Plattform und können den Bedürfnissen der jeweiligen Anwendung angepasst werden.

4 CoSIMS-Anwendung

Die CoSIMS-Anwendung ermöglicht es Anwendern, von CoSIMS bereitgestellte Dienste wie Gruppenmanagement und Präsenz zu nutzen. Für Symbian ist jedoch derzeit kein IMS-Stack verfügbar. Einige CoSIMS-Dienste sind aber als Webdienste realisiert, auf die mittels SOAP [W3C03] zugegriffen werden kann. Dienste für die direkte Kommunikation zwischen den Teilnehmern, also beispielsweise Instant-Messaging, werden über herkömmliche Telefondienste wie etwa SMS simuliert.

Ursprüngliches Ziel bei der Implementierung der Anwendung war es, die auf dem Mobiltelefon vorhandene Kontakthanwendungen zu erweitern, so dass der Anwender gleichermaßen auf lokale und externe Kontaktdaten zugreifen kann. Es zeigte sich jedoch recht früh, dass eine Erweiterung der Kontakthanwendung nicht möglich war. Die entwickelte Anwendung ersetzt daher die vorhandene Kontakthanwendung. Dabei werden Eigenschaften der Plattform beim Start von Programmen ausgenutzt, um den Aufruf der ursprünglichen Anwendung zu verhindern und statt dessen die neue Anwendung zu starten. Die Bedienung der neuen Anwendung entspricht dabei weitgehend der ursprünglichen Anwendung. Abbildung 2 zeigt einen Screenshot.

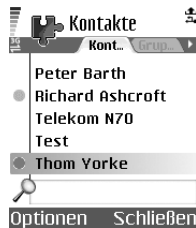


Abbildung 2: Screenshot der CoSIMS-Anwendung

Die Daten werden von zwei Providern bereitgestellt. Der erste Provider ermöglicht den Zugriff auf die lokale Kontaktdatenbank des Smartphones. Der zweite Provider ist für den Zugriff auf die externen Kontakt- und Präsenzdaten zuständig. Die Kommunikation zwischen Mobiltelefon und den CoSIMS-Diensten erfolgt indirekt über einen separaten Relayserver, der zusätzlich auch eine Testumgebung mit eigener Datenbank bereitstellt. Für die Kommunikation zwischen dem Mobiltelefon und dem Relayserver kann dadurch ein im Vergleich zu SOAP kompakteres, proprietäres Format verwendet werden. Die Umsetzung des Relayservers erfolgte in der Skriptsprache Python. Abbildung 3 gibt eine Übersicht über die beteiligten Komponenten.

5 Fazit

Die Diplomarbeit hat gezeigt, dass ein Gruppenkommunikationsframework auch auf einem stark limitierten Gerät wie einem Smartphone realisierbar und in gewohnter UI-Umgebung integrierbar ist. Das Framework vereinfacht die Umsetzung von Kommuni-

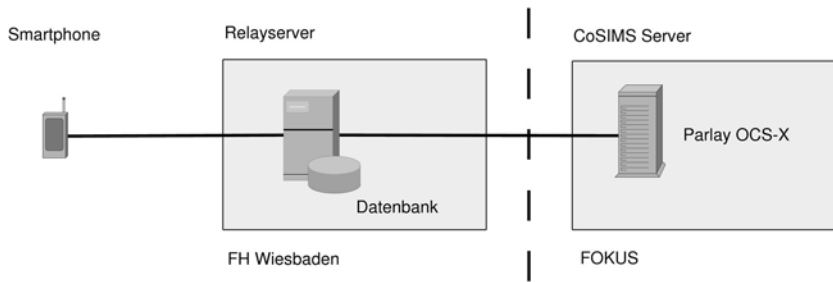


Abbildung 3: Kommunikation zwischen Smartphone und Webdiensten

kationsanwendungen erheblich. Die Integration in die Plattform stieß jedoch im Rahmen der CoSIMS-Anwendung auf Grenzen. Mit aktuelleren Versionen der S60-Plattform (ab 3rd Edition) ist das Ersetzen der Telefonbuchanwendung nur noch in Zusammenarbeit mit Symbian bzw. dem Smartphone-Hersteller möglich.

Betreuer

Prof. Dr. Peter Barth, Fachhochschule Wiesbaden
 barth@informatik.fh-wiesbaden.de

Dipl. Ing. (FH) Roland Schwaiger, Deutsche Telekom Laboratories

Literatur

- [Fra] Fraunhofer Institut für Offene Kommunikationssysteme. *Community based Services on IMS (CoSIMS)*. <http://www.fokus.fraunhofer.de/bereichsseiten/projekte/CoSIMS>.
- [Nok02] Nokia Corporation. *Designing Applications for Smartphones – Series 60 Platform Overview*, April 2002.
- [Nok04] Nokia Corporation. *IP multimedia – a new era in communication*, Januar 2004.
- [Nok06] Nokia Corporation. *S60 Platform: Comparison of ANSI C++ and Symbian C++*, Mai 2006.
- [Pin06] Andreas Pinhammer. *Konzeption und Realisierung eines Frameworks für Gruppenkommunikationsanwendungen auf mobilen Endgeräten unter Nutzung von IMS-Diensten*. Diplomarbeit, Fachhochschule Wiesbaden, Dezember 2006.
- [W3C03] W3C. *SOAP Version 1.2 Part 0: Primer*, Juni 2003.

Lebenszyklusorientierte Performance-Instrumentierung verteilter Anwendungen

Master-Arbeit an der Fachhochschule Wiesbaden
in Kooperation mit dem Cork Institute of Technology

Jan Schäfer - jan.schaefer@informatik.fh-wiesbaden.de

Abstract: Leistungstests während der Entwicklungs- und Testphase, Service Level Management im Betrieb und Leistungsvalidierung nach Modifikationen sind notwendige Vorgänge im Lebenszyklus verteilter Anwendungen. Performance-Instrumentierung ist damit notwendige Voraussetzung für verschiedene Aktivitäten im Lebenszyklus von Softwaresystemen. Diese Arbeit verbindet Performance-Instrumentierung mit der Methodik der Model Driven Architecture zur Integration des Performance-Aspekts in den Softwareentwicklungsprozess. Dadurch können Entwickler die Leistungsüberwachung schon während der Entwurfsphase im UML-Modell der Anwendung planen, ohne Details der später verwendeten Instrumentierungstechnologien zu kennen. Für die im Modell platzierten “Sensoren” wird im Anschluss Instrumentierungscode generiert, der zur Laufzeit Performance-Messdaten liefert, die für Überwachung und Management der Anwendung verwendet werden können. Diese Arbeit ermöglicht eine lebenszyklusorientierte Performance-Instrumentierung bis hin zur Rückführung von Messdaten in das Anwendungsmodell.

1 Motivation und Zielsetzung

Die nachfolgend präsentierte Master-Arbeit entsteht im Rahmen des Forschungsprojekts *Effiziente Tool-gestützte Instrumentierung verteilter Anwendungen* (eMIVA) [Fac06] und wird kooperativ betreut von Prof. Dr. Reinhold Kröger von der Fachhochschule Wiesbaden und Dr. Jeanne Stynes vom Cork Institute of Technology (Irland).

Nicht nur für Dienstleister und Dienstanutzer gewinnt die Leistungsüberwachung im Rahmen des *Service Level Managements* (SLM) [SMJ00] an Bedeutung. Zu professioneller Softwareentwicklung gehört neben dem Test auf funktionale Korrektheit auch die Planung und kontinuierliche Überwachung der Leistungsdaten einer Anwendung. Die Überwachung benötigt eine Sensorik (durch *Instrumentierung*), die die gewünschten Messdaten aus der Anwendung heraus zur Laufzeit liefern kann. Für Software gibt es zwei wesentliche Arten der Instrumentierung: Die hier verwendete Quellcode-Instrumentierung bezeichnet den Prozess des Einfügens von zusätzlichen Anweisungen in den Quellcode einer Anwendung. Daneben existiert auch eine Binärcode-Instrumentierung beziehungsweise eine Instrumentierung von Bytecode (z.B. der JVM [Shi06]).

Oft wird die Leistungsanalyse einer Anwendung erst bei der Installation oder im Betrieb nach Auftreten von Leistungsengpässen vorgenommen. Dazu kommt, dass viele der Leis-

tungstests bezüglich des Anwendungsverhaltens wenig aussagefähig sind, da entweder zu wenige oder nicht korrelierte Instrumentierungspunkte vorgesehen werden. Gründe hierfür können sowohl fehlende Kenntnis geeigneter Instrumentierungstechnologien seitens der Entwickler als auch technische Einschränkungen der Anwendung sein. Häufig empfinden Entwickler den zusätzlichen Aufwand für die Erstellung und Pflege von nicht-funktionalen Aspekten als Last. Auch werden durch das manuelle Hinzufügen von Instrumentierungscode zum Quellcode verschiedene Aspekte der Anwendung vermischt, so dass eine isolierte Betrachtung nachfolgend (z.B. zur Fehlersuche) erschwert wird. Insgesamt wird es als notwendig angesehen, den Performance-Aspekt frühzeitig im Entwicklungsprozess zu berücksichtigen [JDZ00] [Ang96] [B. 01]. Während Methoden des *Software Performance Engineering* (SPE) [Con02] dazu analytische oder simulative Modelle zur Beurteilung erzeugen, steht hier Leistungsmessung in der entwickelten Software im Fokus. Ziel der Arbeit ist es, durch entsprechende Methoden und Werkzeuge Entwickler während des Softwarelebenszyklus bei der Instrumentierung zu unterstützen.

2 Vorgehensweise und Ergebnisse

Da die Instrumentierung schon in der Entwurfsphase ansetzen soll, wird die *Unified Modelling Language* (UML) als Basis verwendet, da sie heute die meistgenutzte Modellierungssprache im Bereich Softwareentwicklung ist. UML bietet mittels UML-SPT [Obj05] die Möglichkeit, nur zeitliche Bedingungen aber keine korrelierten Instrumentierungsvorgänge zu modellieren. Deshalb wird für diese Arbeit eine Erweiterung in Form eines UML-Instrumentierungsprofils definiert, das zwei Stereotypen namens `Log` und `Measure` für diese Aufgabe zur Verfügung stellt. `Log` ermöglicht das Einfügen von Logpunkten in das Modell, z.B. zur Erzeugung von Statusmeldungen an kritischen Stellen, `Measure` dient der zeitlichen Vermessung von internen Abläufen.

Mit Hilfe dieser Stereotypen erfolgt die plattformunabhängige Definition von Instrumentierungspunkten in der Entwurfsphase der Anwendung, unabhängig von den eingesetzten Technologien. Dabei erfolgt ein Rückgriff auf sogenannte Instrumentierungsmuster, die Beziehungen zwischen verschiedenen Instrumentierungspunkten bezeichnen (z.B. RPC-Muster bestehend aus vier Punkten). So wird das UML-Modell einer verteilten Anwendung durch ein Instrumentierungsmodell ergänzt. Aus diesem Gesamtmodell kann dann unter Verwendung eines Code-Generators der instrumentierte Quellcode erzeugt werden. Die so instrumentierte Anwendung liefert zur Laufzeit Messdaten, die für Monitoring und Management der Anwendung oder als Grundlage für eine detailliertere Instrumentierung des Modells verwendet werden können. Basierend auf diesen Messdaten könnten auch Grenzwerte (*Constraints*) modelliert werden, die über Änderungen des Anwendungsverhaltens im Rahmen von Regressionstests Aufschluss geben. Der Schwerpunkt der Arbeit liegt auf der modellbasierten Instrumentierung und anschließenden Code-Generierung.

Der modellbasierte Instrumentierungsprozess wird in Abbildung 1 veranschaulicht. Das UML-Instrumentierungsprofil wird mit dem UML-Werkzeug MagicDraw 12.0 CE definiert und ins XMI-Format exportiert. Aus dem instrumentierten Modell wird mit Hilfe des *openArchitectureWare* (OAW) [ope06] [Obj03] Code-Generator-Frameworks instrumen-

tierter Java-Quellcode generiert. Hierzu werden *Templates* für die Erstellung des grundlegenden Java-Codes als auch des Instrumentierungscodes benötigt. Für den grundlegenden Java-Code wird die *JavaBasic-Cartridge* (CJB) der *Fornax Platform* (FNX) [For07], und für den Instrumentierungscode werden eigens entwickelte Templates verwendet. Der Ablauf der Code-Generierung wird in einem *Workflow* definiert. Dafür müssen UML-Profil und Anwendungsmodell im XMI-Format vorliegen. Für nicht instrumentierte UML-Elemente ruft der Code-Generator die JavaBasic-Templates auf, für die anderen die Instrumentierungs-Templates. Der dadurch entstehende Quellcode ist somit automatisch instrumentiert, so dass kein Eingriff von Entwicklern nötig ist. Für den generierten Code des `Log`-Stereotyps wird derzeit die *Jakarta Commons Logging* (JCL)-API [Apa06] verwendet, für den `Measure`-Stereotyp die *Application Response Measurement* (ARM)-API [Ope03].

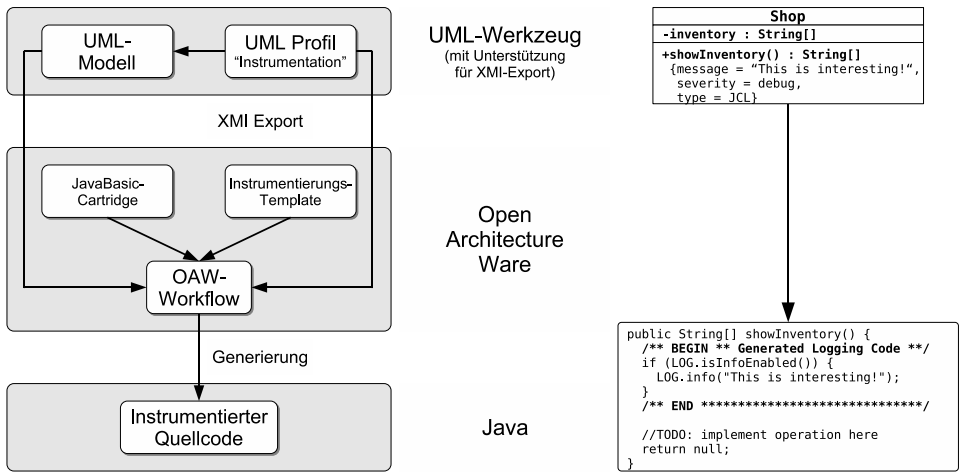


Abbildung 1: Der modellbasierte Instrumentierungsprozess

Die Performance-Instrumentierung von UML-Modellen mit Hilfe eines UML-Instrumentierungsprofils ist im aktuellen Zustand der Arbeit bereits möglich. Bisher können mit den beiden Stereotypen jedoch nur UML-Operationen in Klassendiagrammen erweitert werden, so dass die Granularität der Instrumentierung auf Beginn und Ende eines Methodenaufrufs beschränkt ist. An der Anwendbarkeit auf Sequenz- und Aktionsdiagrammen wird derzeit gearbeitet. Die Anwendung des Ansatzes in einem gemeinsamen Projekt mit Lufthansa Systems und tang-IT ist in Vorbereitung.

Literatur

- [Ang96] C. Anglano. Performance Modeling of Heterogeneous Distributed Applications, 1996.
- [Apa06] The Apache Software Foundation. *The Jakarta Project Commons Logging*, 2006. <http://jakarta.apache.org/commons/logging/>.
- [B. 01] B. Theelen, J. Voeten, L. van Bokhoven, P. van der Putten, A. Niemegeers and G. Jong. Performance Modeling in the Large: A Case Study, 2001.
- [Con02] Connie U. Smith and Lloyd G. Williams. Performance and Scalability of Distributed Software Architectures: An SPE Approach, 2002.
- [Fac06] Fachhochschule Wiesbaden - Labor für verteilte Systeme. *Effiziente Tool-gestützte Instrumentierung verteilter Anwendungen (eMIVA)*, 2006. <http://wwwvs.informatik.fh-wiesbaden.de/projekte/emiva.html>.
- [For07] Fornax Platform. *Fornax Platform - An Open Platform for Model Driven Software Development*, 2007. <http://www.fornax-platform.org/>.
- [JDZ00] J. Freiheit J. Dehnert und A. Zimmermann. Workflow Modeling and Performance Evaluation with Colored Stochastic Petri Nets, 2000.
- [Obj03] Object Management Group. *OMG's Model Driven Architecture Guide Version 1.0.1*, 2003. <http://www.omg.org/cgi-bin/apps/doc?omg/03-06-01.pdf>.
- [Obj05] Object Management Group. *UML Profile for Schedulability, Performance, and Time*, 2005. <http://www.omg.org/cgi-bin/apps/doc?formal/05-01-02.pdf>.
- [Ope03] The Open Group. *ARM 4.0 Java Language Binding Technical Standard 4.0*, October 2003. <http://www.opengroup.org/arm/uploads/40/3945/C037.pdf>.
- [ope06] openArchitectureWare.org. *openArchitectureWare - A Modular MDA/MDD Generator Framework*, 2006. <http://www.openarchitectureware.org/>.
- [Shi06] ShiftOne. *JRat the Java Runtime Analysis Toolkit*, 2006. <http://jrat.sourceforge.net/>.
- [SMJ00] Rick Sturm, Wayne Morris und Mary Jander. *Foundations of Service Level Management*. SAMS Publishing, April 2000.

Ein Kommunikations-Framework für die Selbstorganisation von Software-Systemen

Thorsten Stein

Labor für Verteilte Systeme
Fachhochschule Wiesbaden
Kurt-Schumacher-Ring 18
65197 Wiesbaden

tstei001@stud.informatik.fh-wiesbaden.de

Abstract: Die immer höher werdenden Anforderungen an die Erbringung von rechnergestützten Dienstleistungen werden in komplexen und hochdynamischen Architekturen im Kontext von Selbstmanagement durch spezielle Softwarekomponenten, sog. Selbstmanagern, überwacht. Im klassischen Fall liegt der Fokus eines solchen Selbstmanagers auf der Optimierung eines Systems nach lokalen Kriterien. Im Rahmen der hier vorgestellten Diplomarbeit wurde ein Framework zur selbstorganisierten Koordination einzelner Selbstmanagerinstanzen entwickelt, das die Optimierung eines aus mehreren Komponenten bestehenden Systems nach übergeordneten Zielen erlaubt. In diesem Rahmen wurden verschiedene Koordinationsstrategien evaluiert und prototypisch in einem Framework umgesetzt.

1 Motivation und Zielsetzung

Unternehmensweite Anwendungen werden immer komplexer und damit ebenfalls die Abläufe innerhalb aktuell für diesen Bereich verwendeter Service-orientierter Architekturen (kurz: *SOA*) [ZDGH05]. Gleichzeitig spielt das Antwortzeitverhalten angebotener Dienstleistungen eine immer größere Rolle für den geschäftlichen Erfolg eines Unternehmens. Die Anforderungen an eine Dienstleistung werden formal in speziellen Vereinbarungen, sog. *Service Level Agreements* (SLAs), beschrieben [Lu03]. Bei der Verletzung eines SLAs drohen dem Anbieter eines Dienstes i.d.R. Konventionalstrafen. Daher ist die Implementierung eines effektiven *Service Level Managements* (SLM) [SMJ00, Le01, HAN99] notwendig. Aufgrund der hohen Komplexität und Dynamik vieler SOA-Umgebungen können Management-Aufgaben, d.h. permanente Überwachung und Optimierung von Abläufen innerhalb der Umgebung, nur noch bedingt von einem menschlichen Operator gewährleistet werden. Daher gilt es, Möglichkeiten zu finden, die in einer SOA-Umgebung existierenden Softwaresysteme in die Lage zu versetzen, möglichst selbstständig für die Einhaltung der in einem SLA als *Service Level Objectives* (SLOs) beschriebenen Ziele Sorge zu tragen [HMG05, KC03, ST05]. Im Rahmen dieser Ziele sollen die einzelnen Softwaresysteme in der Lage sein,

sich mit Hilfe einer Manager-Komponente selbsttätig zu rekonfigurieren. Die dadurch erfolgende Automatisierung von Management-Aufgaben wird durch den Oberbegriff *Selbstmanagement* ausgedrückt. Da in SOA-Umgebungen unterschiedliche Dienste kooperativ in einem Workflow an der Erfüllung einer Aufgabe mitwirken, genügt es nicht, wenn das Selbstmanagement lokal für eine einzige Komponente betrieben wird. Vielmehr ist es nötig, lokale Management-Bemühungen zu koordinieren und somit einen insgesamt akzeptablen Systemzustand zu erreichen. Da das für die Erbringung der Gesamtleistung vereinbarte SLA nicht verletzt werden darf, müssen die in dem SLA enthaltenen SLOs auf alle beteiligten Softwaresysteme heruntergebrochen werden. Der Anteil eines Softwaresystems an einem SLO wird als das *individuelle Service Level Objective* (iSLO) bezeichnet. Wenn ein Softwaresystem nicht in der Lage ist, sein iSLO zu erfüllen, sollte das iSLO erweitert werden können. Da die Gesamtheit aller iSLOs das globale SLO nicht überschreiten darf, muss in diesem Fall das iSLO eines anderen Softwaresystems im gleichen Maße reduziert werden. Diese Aufgabe soll von den zuständigen Managern übernommen werden. Im Labor für Verteilte Systeme an der Fachhochschule Wiesbaden wurde ein Selbstmanager zur Durchführung von lokalen Selbstmanagement-Aufgaben entwickelt. Das Ziel dieser Arbeit ist die Erweiterung des Selbstmanagers um Mechanismen zur Koordination mit anderen Selbstmanagern.

2 Architektur und Ergebnisse

Zur Abwicklung der Kommunikation und der Koordination von Selbstmanagern wurde ein Framework auf Basis der P2P-Kommunikationsplattform JXTA¹ implementiert. Das Framework besteht aus Teilsystemen für *Kommunikation und Gruppenmanagement* sowie *Koordination*. Aufgabe von *Kommunikation und Gruppenmanagement* ist es, die JXTA-eigenen Mechanismen zur Nachrichtenübermittlung sowie zur Bildung von Peer-Gruppen abzubilden [CK01] und dem Teilsystem *Koordination* eine Schnittstelle zu dem unterlagerten JXTA-Framework zur Verfügung zu stellen. Dadurch wird es möglich, das JXTA-Framework zu einem späteren Zeitpunkt gegen einen anderen Mechanismus austauschen zu können, ohne dass die übrigen Softwarekomponenten angepasst werden müssen. Das Teilsystem *Koordination* implementiert Mechanismen zur Koordination der Selbstmanager. Diese ermöglichen es einzelnen Managern, ein iSLO zu erweitern bzw. zu reduzieren. Hierzu wurden drei Möglichkeiten betrachtet: *Auktionen*, *Markt-Mechanismen* und *Mechanismen zur sozialen Interaktion*. Das Teilsystem *Koordination* enthält Mechanismen zur Abwicklung von Transaktionen auf einem virtuellen Marktplatz und Mechanismen zur Durchführung von Auktionen [WWW01]. Von den möglichen Auktionsvarianten *englische Auktion*, *holländische Auktion*, *Ausschreibung* und *Vickrey-Auktion* [Kr02] wurde die englische Auktion implementiert, als Variante wurde noch eine Auktion im Stil des Online-Auktionshauses eBay² vorgesehen. Die Ermittlung des Auktionators wird von einem Wahl-Mechanismus übernommen. Dieser Wahl-Mechanismus besteht aus einer abgewandelten Implementierung des Bully-Algorithmus [Gm82]. Damit das aus diesen beiden Teilsystemen bestehende Framework an den existierenden Selbstmanager angebunden werden konnte, wurden ein spezielles

¹ <http://www.jxta.org>

² <http://www.ebay.de>

Erweiterungsmodul sowie einige anhängende Komponenten entworfen und implementiert. Abbildung 1 gibt einen Überblick über die entworfene und realisierte Gesamtarchitektur. Ein sog. Strategie-Adapter ist direkt an den Selbstmanager angebunden und ist für die Weitergabe der durch den Selbstmanager ermittelten Leistungswerte des überwachten Systems an das Strategie-Modul zuständig. Zusätzlich empfängt das Strategie-Modul noch eine Datenstruktur, welche zur Berechnung des zur Erreichung des aktuellen Management-Zustands benötigten Aufwands verwendet werden kann. Diese Werte werden von dem Strategie-Modul in Form der sog. *Compliance* [Sc07] berechnet. Das Strategie-Modul trifft basierend auf den empfangenen Leistungswerten und der Compliance strategische Entscheidungen und teilt das Ergebnis dem Strategie-Adapter mit, welcher dann für die korrekte Umsetzung innerhalb des Selbstmanagers zuständig ist. Im Rahmen der Strategie-Entscheidungen kann das iSLO bei Bedarf erweitert bzw. reduziert werden. Der hierzu notwendige Mechanismus (der sog. *SalesAgent*) wird über einen Plug-In-Mechanismus an das Strategie-Modul angebunden und kann somit flexibel ausgetauscht werden. Damit je nach Szenario eine dynamische Anpassung auch zur Laufzeit möglich ist, wurde für die hierzu benötigte Logik eine Reihe von externen Tcl³-Skripten vorgesehen.

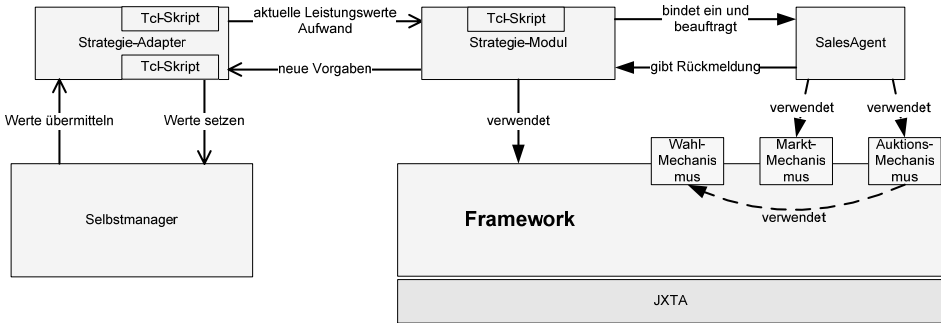


Abbildung 1: Architekturüberblick

Durch die implementierten Koordinations-Mechanismen sind die Selbstmanager in der Lage, selbsttätig über die Erweiterung bzw. Reduktion der iSLOs zu verhandeln.

3 Fallbeispiel

Zur Bewertung des Nutzens des entwickelten Frameworks wurden mehrere Experimente zu konkreten Fallbeispielen ausgewertet. Anhand eines Beispiel-Workflows mit mehreren Diensten wurde die Funktionalität des Ansatzes verifiziert. Die beteiligten Selbstmanager konnten sich mit Hilfe des implementierten Frameworks gegenseitig durch die Abgabe von Teilen ihrer iSLOs unterstützen und somit für eine bedarfsgerechte Verteilung sorgen. In weiteren Tests wurden Reaktionsgeschwindigkeit und Skalierbarkeit des Ansatzes untersucht. Nächste Schritte sehen die Evaluation des Ansatzes anhand einer realen Anwendung vor.

³ <http://www.tcl.tk/>

Literaturverzeichnis

- [ZDGH05] Zimmermann, O., Doubrovski, V., Grundler, J., Hogg, K.: Service-oriented architecture and business process choreography in an order management scenario: rationale, concepts, lessons learned, ACM Press, 2005
- [Lu03] Ludwig, H.: Web services QoS: external SLAs and internal policies or: how do we deliver what we promise?, Web Information Systems Engineering Workshops, 2003.
- [SMJ00] Sturm, R., Morris, W., Jander, M.: Foundations of Service Level Management, SAMS Publishing, 2000.
- [Le01] Lewis, L.: Managing Business and Service Networks, Kluwer Academic Publishers, 2001.
- [HAN99] Hegering, Abeck, Neumair: Integriertes Management vernetzter Systeme, dpunkt-Verlag, 1999
- [HMG05] Herrmann, K., Muehl, G., Geihs, K.: Self-Management: The Solution to Complexity or Just Another Problem?, IEEE Distributed Systems Online 6(1), 2005.
- [KC03] Kephart, J.O., Chess, D.M.: The Vision of Autonomic Computing, Computer 36(1), 2003.
- [ST05] Salehie, M., Tahvildari, L.: Autonomic Computing: Emerging Trends and Open Problems, ACM Press, 2005
- [CK01] Chockler, G.V., Keidar, I., Vitenberg, R.: Group communication specifications: a comprehensive study, ACM Comput. Surv. 33(4), 2001
- [WWWM01] Wellman, M., Walsh, W., Wurman, P., Mason, M.J.: Auction protocols for decentralized scheduling, Games and Economic Behavior 35, 2001
- [Kr02] Krishna, V.: Auction Theory, Academic Press, 2002
- [Gm82] Garcia-Molina, H: Elections in Distributed Computer Systems, IEEE Transactions on Computers, C-31, 1982.
- [Sc07] Schmid, M.: Ein Ansatz für das Service Level Management in dynamischen Architekturen. Workshop Selbstorganisierende, Adaptive, Kontextsensitive verteilte Systeme (SAKS), Accepted for Publication, 2007.

Reducing Time and RAM Requirements in Content-Based Image Retrieval using Retrieval Filtering

Jens Forster and Thomas Deselaers

Human Language and Pattern Recognition Group, Computer Science Department,
RWTH Aachen University, D-52056 Aachen
{forster,deselaers}@i6.informatik.rwth-aachen.de

Abstract: In this paper we present and evaluate how computationally cheap comparison measures can be applied in content-based image retrieval applications to reduce the time- and memory requirements. The time requirements are reduced by applying filtering techniques and evaluating computationally costly distance measurements only to a suitably chosen subset of images. The RAM requirements are reduced by keeping only those features in memory that are absolutely required. It is shown that runtime- and memory efficiency is greatly improved with hardly any changes in retrieval quality.

Introduction

With the ubiquity of cameras and the ever-increasing necessity of digital images in medicine, the amount of images stored in databases is growing quickly. Access to these data is commonly achieved using textual meta data. Content-Based Image Retrieval (CBIR) systems are an alternative and allow accessing image databases by image content rather than by textual information. A problem with CBIR systems is that they require computationally expensive operations and large amounts of memory to allow for acceptable results. The aim of a CBIR system is to find visually similar images for a given query image.

To find visually similar images, typically features are extracted from each image in the database and from a given query image. Then, the features of the query image are compared to the features of each database image and thus the most similar images from the database can be determined. Some of these distance measures, e.g. the Image Distortion Model (IDM) [KGN04], provide good results in terms of error rates but have high computational costs which do not allow for interactive use in the context of large databases.

The concept of *filtered retrieval*, i.e. use a computationally cheap distance function to preselect images for the computationally more costly distance function, is well-known in the database and data exploration community, e.g. [FBF⁺94] and [SH94] propose to use a lower dimensional distance function as a filter for a higher dimensional quadratic distance function.

Filtered Retrieval in FIRE

We integrated the concept of filtered retrieval into the Flexible Image Retrieval Engine (FIRE)¹ [DKN04]. In FIRE, an *image* X is a set of *features* X_1, \dots, X_M representing certain aspects of the images. Normally, FIRE calculates the distance between a query image Q and a given image X contained in the database as a weighted sum over all individual feature distances. This calculated distance is used to assign a similarity score $S(Q, X)$ to

¹<http://www-i6.informatik.rwth-aachen.de/~deselaers/fire.html>

the database image.

$$S(Q, X) = \exp(-d(Q, X)) = \exp\left(-\sum_{m=1}^M w_i \cdot d_m(Q_m, X_m)\right) \quad (1)$$

where $d(Q, X)$ denotes the total distance between Q and X , $d_m(Q_m, X_m)$ the distance of Q and X wrt. feature m and w_i the weighting factor of feature m .

In filtered retrieval, we avoid calculating all distances, by first calculating the distance for one feature type for all images and keeping only the best matching images for subsequent steps in which the images are compared using the remaining features and distance functions. The order in which feature distances are evaluated is crucial since later steps are influenced by earlier steps. If several different features are used, the aim is to reduce the number of images to be considered in subsequent steps from feature to feature:

Let the database \mathcal{B} consist of N images: $\{X_1, \dots, X_n, \dots, X_N\}$, each image X_n is represented by M features $\{X_{n1}, \dots, X_{nm}, \dots, X_{nM}\}$. Then the distance according to the first feature is evaluated for all N database images. The N_1 best matching images are considered for the evaluation of the second feature. This is repeated with decreasing N_m until all M features were used. To be able to calculate scores (Eq 1), we set distances that were not evaluated to a value slightly higher than the highest calculated distance.

Now, not every feature for each image is required to process a query, e.g. if image X has been ruled out after the evaluation of feature m it is not necessary to keep features $m+1, \dots, M$. Therefore it is possible to remove some of the features from RAM and to load features only on-demand which may lead to a large reduction in memory requirements if it is possible to select small features first and load large ones only when requested. The gain in memory and speed is biggest if the features that are evaluated first are small features and allow for efficient distance calculation and the features evaluated last need more memory and computation time for distance calculation.

Databases

ImageCLEF² is an evaluation of CBIR techniques which is part of the Cross Language Evaluation Forum (CLEF)³ [DWK⁺05, DWN07, CMD⁺05]. In the context of this evaluation two medical image retrieval tasks are defined using the following databases:

The *IRMA 10,000* database was used in ImageCLEF 2005 and consists of 9,000 training images subdivided into 57 classes and 1,000 test images. The images have been chosen randomly from daily routine at the RWTH University Aachen Hospital.

The ImageCLEF 2005 and 2006 medical retrieval database consists of more than 50,000 images together with medical case descriptions and various other meta data

Our group participated in ImageCLEF 2005 and 2006 and obtained very good results. However, for the experiments, high computing times were required on machines with much RAM. Given the announcement that the databases are growing over the next years and that faster evaluation of the experiments is desired we use the proposed techniques for these tasks and present the outcomes in the following section.

Experimental results

First we evaluate the reduction of runtime on the IRMA task as here the image distortion model is required to obtain good results. Therefore we use Euclidean distance filtering

²<http://ir.shef.ac.uk/imageclef/>

³<http://clef.iei.pi.cnr.it/>

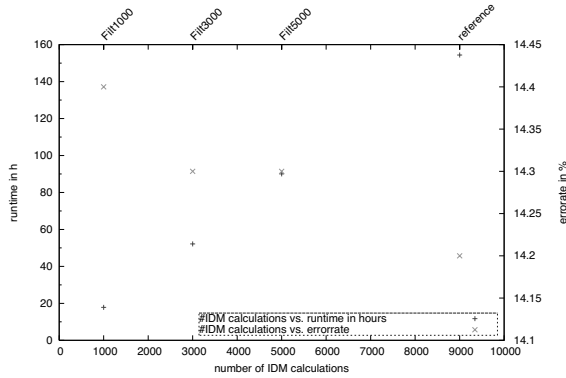


Figure 1: Time and error rate results on the IRMA 10,000 database

on 32×32 thumbnails of the images and apply the image distortion model distance to a subset only. The IDM distance function has been weighted 5 to 1. In the reference experiment filtered retrieval has not been used. In the following experiments the number of IDM distance calculations has been varied from 5,000 to 1,000. The achieved results depicted in Figure 1 show that the runtime of overall experiment has been reduced from about 154 hours to about 18 hours while increasing the error rate by only 0.2 %, i.e. two images more are classified incorrectly. On the ImageCLEF medical retrieval dataset, no class information is given but we have relevance judgements for a set of query images and measure the performance in mean average precision (MAP). Experimental results are given in Table 1. A reduction from 2.069GB to 0.7995GB peak RAM is observed while hardly changing the retrieval results.

Summary

The taken approach lead to very good results. In total a runtime reduction up to a factor of 8.6 and a reduction of peak RAM requirement up to 66% have been achieved with hardly any changes in retrieval performance. Hence it is now possible to apply CBIR to large databases in near real-time tasks. Nevertheless experiments show that the feature sequence needs to be well-chosen to keep retrieval performance. A more detailed description of the taken approaches and further results can be found in [For07].

Table 1: Experimental results to RAM reduction on the ImageCLEF2006 med database

	reference	partial loading		
		FIL10P	FIL01P	FIL014P
run				
32 x 32 image feat.	*	2.(15000)	1.(30000)	1.(30000)
color hist.	*	1.(30000)	2.(15000)	2.(15000)
32 x 32 image feat.	*	-	-	3.
peak memory usage in GB	2.069	0.4811	0.5404	0.7995
mean average precision	0.1470	0.0653	0.0746	0.1024

References

- [CMD⁺05] Paul Clough, Henning Mueller, Thomas Deselaers, Michael Grubinger, Thomas Lehmann, Jeffrey Jensen, and William Hersch. The CLEF 2005 Cross-Language Image Retrieval Track. In *Workshop of the Cross-Language Evaluation Forum (CLEF 2005)*, LNCS, page in press, Vienna, Austria, September 2005.
- [DKN04] Thomas Deselaers, Daniel Keysers, and Hermann Ney. Features for Image Retrieval – A Quantitative Comparison. In *DAGM 2004, Pattern Recognition, 26th DAGM Symposium*, number 3175 in LNCS, pages 228–236, Tübingen, Germany, September 2004.
- [DWK⁺05] Thomas Deselaers, Tobias Weyand, Daniel Keysers, Wolfgang Macherey, and Hermann Ney. FIRE in ImageCLEF 2005: Combining Content-based Image Retrieval with Textual Information Retrieval. In *Workshop of the Cross-Language Evaluation Forum (CLEF 2005)*, volume 4022 of LNCS, pages 652–661, Vienna, Austria, September 2005.
- [DWN07] Thomas Deselaers, Tobias Weyand, and Hermann Ney. Image Retrieval and Annotation Using Maximum Entropy. In *Evaluation of Multilingual and Multi-modal Information Retrieval – Seventh Workshop of the Cross-Language Evaluation Forum, CLEF 2006*, LNCS, page to appear, Alicante, Spain, September 2007.
- [FBF⁺94] C. Faloutsos, R. Barber, M. Flicker, J. Haffner, W. Niblack, D. Petkovic, and W. Equitz. Efficient and Effective Querying by Image Content. *IIIS*, 3(3–4):231–262, July 1994.
- [For07] Jens Forster. Reducing Time and RAM Requirements in Content-Based Image Retrieval using Retrieval Filtering. Studienarbeit, RWTH Aachen University, January 2007.
- [KGN04] Daniel Keysers, Christian Gollan, and Herman Ney. Classification of Medical Images using Non-linear Distortion Models. *BVM 2004, Bildverarbeitung für die Medizin 2004*, pages 366–370, March 2004.
- [SH94] Harpreet S. Sawhney and James L. Hafner. Efficient Color Histogram Indexing. In *Image Processing 1994, ICIP-1994, IEEE International Conference*, volume 2, pages 66–70, November 1994.

Semantikgetriebene automatische Verfahren für die kartografische Generalisierung von 3D Gebäudemodellen und ihre Visualisierung

Tassilo Glander

Hasso-Plattner-Institut
Universität Potsdam
tassilo.glander@hpi.uni-potsdam.de

Art der Arbeit: Masterarbeit
Betreuer der Arbeit: Prof. Dr. Jürgen Döllner

Abstract: Raumbezogene Visualisierung und insbesondere virtuelle 3D-Stadtmodelle haben in letzter Zeit eine wachsende Aufmerksamkeit durch Wissenschaft und Öffentlichkeit erfahren. Während ein Ziel aus Sicht der Computergrafik die effiziente und qualitativ hochwertige Darstellung komplexer 3D-Modelle ist, werden in vielen Anwendungen zunehmend „intelligente“ Darstellungen gefordert, die die Semantik der Objekte mit bei der Level-of-Detail-Berechnung und zur Detail-&-Context-Visualisierung heranziehen.

Mit den Generalisierungsalgorithmen für 3D-Gebäudemodelle soll ein Betrag zur qualitativ hochwertigen 3D-Visualisierung geschaffen werden. Der Karten-Charakter steht dabei im Vordergrund: Karten sind auf das Wesentliche reduzierte graphische Darstellungen von multivariaten räumlichen Daten, bei denen es nicht um eine fotoreale Wiedergabe geht. Stattdessen unterstützen sie den Menschen bei der Bewältigung von Aufgaben wie Orientierung und Wegfindung, aber auch ganz allgemein der Exploration, Analyse und Kommunikation räumlicher Informationen. Durch den im Vergleich zur Realität geringeren Raum zur Darstellung entsteht der Zwang zur Skalierung, insbesondere aber auch zur Abstraktion des Modells: Man benötigt eine Generalisierung, die bezüglich eines Stadtmodells Gebäudegeometrie vereinfacht, dabei aber wichtige strukturelle Charakteristika erhält.

Bei der Untersuchung existierender Verfahren für die 3D-Gebäudegeneralisierung zeigten sich zwei Tendenzen: Zum einen wurden in der Algorithmischen Geometrie und Kartografie bereits leistungsstarke Ansätze für die automatisierte Anwendung von Generalisierungs-Operatoren wie Verdrängung und Hervorhebung entwickelt, sie lassen sich jedoch auf 3D nur selten und wenn, dann unbefriedigend übertragen. Zum anderen gibt es Verfahren speziell für 3D-Gebäudemodelle, die sich allerdings bisher nur auf die Vereinfachung von Einzelgebäuden konzentrieren und damit das Prinzip semantikgetriebener, also „kartografischer“ Generalisierung nicht berücksichtigen, weil sie Ordnungsstrukturen unberücksichtigt lassen.

Im Rahmen der vorgestellten Masterarbeit wurden Generalisierungsalgorithmen für 3D-Gebäudemodelle analysiert und implementiert. Aus einem gegebenen virtuellen 3D-Stadtmodell, bestehend aus Straßen- und Gebäudemodellen, erzeugen sie ein generalisiertes Bebauungsmodell. Dabei zeichnen sich die verwendeten Ansätze durch eine gemeinsame Generalisierung einer Menge von Gebäudemodellen aus. Aus der Darstellung von Städten in klassischen zweidimensionalen Karten wurden die Verein-

fachung und die Aggregation benachbarter Gebäude als bedeutendste Elemente einer Generalisierung abgeleitet. Zusätzlich werden wichtige Gebäude, die der Orientierung dienen, nicht vereinfacht, sondern als so genannte Landmarken hervorgehoben. Die dreidimensionale Darstellung wird genutzt, um ein der Realität angenähertes charakteristisches Höhenprofil zu visualisieren.

Das erste erforschte Verfahren besteht in einer einfachen Blockgeneralisierung, die zunächst aus den sich schneidenden Straßen eine Zellstruktur aufbaut. Die Zellen werden als Grundrisse genutzt, die dann auf die gewichtete Durchschnittshöhe der in der Zelle enthaltenen Gebäude zu Blöcken extrudiert werden. Landmarken werden abschließend aus den Blöcken ausgeschnitten und unverändert platziert.

Das zweite erforschte Verfahren erstellt eine komplexe Blockgeneralisierung, die sich an existierende kartografische Generalisierungsverfahren anlehnt. Dabei werden zunächst die Gebäudegrundrisse mittels morphologischer Operationen zusammengefasst, dann im Krümmungsraum vereinfacht und schließlich ebenfalls wieder auf die gemittelte Höhe extrudiert. Hier bleibt die Struktur der Gebäude besser erhalten.

Neben diesen beiden $2\frac{1}{2}$ D-Ansätzen wurden in der Arbeit voll dreidimensional arbeitende Verfahren erprobt. Im einfachsten Fall liefert die Approximation der Gebäudegeometrie innerhalb einer Zelle durch ihre konvexe Hülle ein aggregiertes Bebauungsmodell. Ein weiteres 3D-Verfahren erzeugt eine geschlossene Blockoberfläche, indem die Dächer der Gebäude einer Zelle mittels Constrained Delaunay Triangulation neu verbunden werden. Die erhaltene Dachoberfläche wird an den Rändern Richtung Boden zu Fassadenflächen extrudiert, so dass auch hier ein Block entsteht, der von den Einzelgebäuden abstrahiert, dabei aber die ursprünglichen Dachformen erhält.

Auf der Suche nach innovativen Verfahren wurden in der Masterarbeit darüber hinaus Ansätze für zwei neuartige Generalisierungsverfahren entwickelt. Das eine Verfahren wendet 3D Alpha-Shapes auf die Gebäudegeometrie an, um enge Zwischenräume zu überbrücken bzw. zu verfüllen und so nahe beieinander stehende Häuser zu aggregieren. Ein weiteres Verfahren erreicht Vereinfachung und Aggregation, indem die Gebäudegeometrie zunächst gerastert und in den 3D-Bildraum übertragen wird. Hier können mittels morphologischer Operationen sowie der Anwendung klassischer Filter wie dem Gaußschen Weichzeichner lokale Feinheiten eliminiert und zusammengefasst werden. Anschließend werden die Rasterdaten durch Anwendung von Marching Cubes zurück in ein polygonales Modell gewandelt.

In der Masterarbeit werden die Implementierungen der verschiedenen Verfahren erläutert und ihre Anwendbarkeit anhand von Ausschnitten des Berliner 3D-Stadtmodells getestet. Dabei werden die jeweiligen Ergebnisse diskutiert und durch Abbildungen dokumentiert. Insgesamt konnte durch die extensive Erprobung von geometrischen Algorithmen zur Generalisierung von Bebauungsmodellen eine breite Palette von bisher nicht auf 3D-Gebäude angewendeten Techniken getestet werden, so dass im Ergebnis ein Repertoire von Generalisierungsvarianten geschaffen wurde.

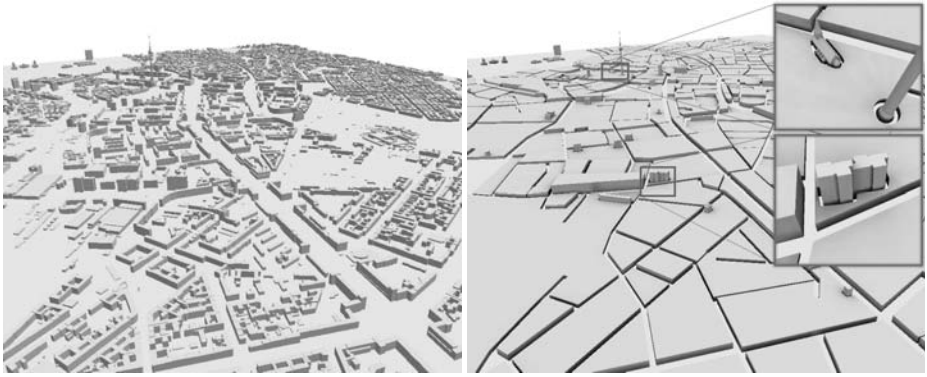


Abbildung 1: Die Abbildungen zeigen die Anwendung der einfachen Blockgeneralisierung unter der Berücksichtigung von Landmarken (links: Ausgangsmodell, rechts: generalisiertes Modell)

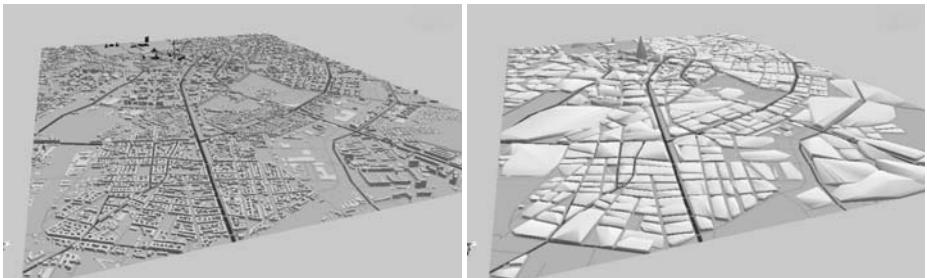


Abbildung 2: Die Abbildungen zeigen die Anwendung des Konvexe-Hülle-Verfahrens auf das Stadtmodell von Berlin Mitte Ost (links: Ausgangsmodell, rechts: generalisiertes Modell)

Literatur

- [And05] K.-H. Anders. Level of detail generation of 3d Building groups by aggregation and typification. *Proceedings of 22nd International Cartographic Conference, La Coruña, Spain*, 2005.
- [EM94] Herbert Edelsbrunner und Ernst P. Mücke. Three-dimensional alpha shapes. *ACM Trans. Graph.*, 13(1):43–72, 1994.
- [EPK06] B. Elias, V. Paelke und S. Kuhnt. Kartographische Visualisierung von Landmarken. *Aktuelle Entwicklungen in Geoinformation und Visualisierung, GEOVIS 2006, 5./6. April 2006, Potsdam, Kartographische Schriften Band 10*, 2006.
- [FM02] A. Forberg und H. Mayer. Generalization of 3D Building Data Based on Scale-Spaces. *Proceedings of the Symposium on Geospatial Theory, Processing, and Applications. Ottawa*, Seiten 225–230, 2002.

- [FR02] A Forberg und Jagdish Lal Raheja. Generalization of Three-Dimensional (3D) Settlement Structures Based on Scale-Spaces and Structure Recognition. 2002.
- [HG97] P.S. Heckbert und M. Garland. Survey of polygonal surface simplification algorithms. *SIGGRAPH 97 Course Notes*, 25, 1997.
- [HHK⁺95] Taosong He, Lichan Hong, A. Kaufman, A. Varshney und S. Wang. Voxel based object simplification. *vis*, 0:296, 1995.
- [Kad02] M. Kada. Automatic Generalisation of 3D Building Models. *GIS-Geo-Information-Systems. Journal for Spatial Information and Decision Making*, 9(2002):30–36, 2002.
- [Kad05] M. Kada. 3D Building Generalisation. *Proceedings of 22nd International Cartographic Conference, La Coruña, Spain*, 2005.
- [LM04] Jagdish Lal und Liqiu Meng. Building recognition using artificial neural network. 2004.
- [LYAC04] Z. Li, H. Yan, T. Ai und J. Chen. Automated building generalization based on urban morphology and Gestalt theory. *International Journal of Geographical Information Science*, 18(5):513–534, 2004.
- [May98a] H. Mayer. Model-Generalization of Building Outlines Based on Scale-Spaces and Scale-Space Events. *International Archives of Photogrammetry and Remote Sensing*, 32:530–536, 1998.
- [May98b] H. Mayer. Three Dimensional Generalization of Buildings Based on Scale-Spaces. *Report, Chair for Photogrammetry and Remote Sensing, Technische Universität München*, 1998.
- [Reg01] N. Regnault. Contextual Building Typification in Automated Map Generalization. *Algorithmica*, 30(2):312–333, 2001.
- [Ses] M. Sester. *Maßstabsabhängige Darstellungen in digitalen räumlichen Datenbeständen*. Dissertation, Habilitationsschrift-Reihe C 544, Verlag der Bayerischen Akademie der Wissenschaften, München, 2001.
- [Ses00] M. Sester. Generalization Based on Least Squares Adjustment. *International Archives of Photogrammetry and Remote Sensing*, 33:931–938, 2000.
- [Spe01] R. Spence. *Information visualization*. Addison-Wesley Harlow, 2001.
- [TS04] F. Thiemann und M. Sester. Segmentation of Buildings for 3D-Generalisation. *Proceedings of the ICA Workshop on Generalisation and Multiple Representation, Leicester, UK*, 2004.
- [War00] C. Ware. *Information visualization: perception for design*. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, 2000.

Human Detection and Character Recognition in TV-Style Movies

Alexander Kläser
Fachhochschule Bonn Rhein Sieg
alex.klaeser@gmx.de

1 Objective

The objective of this master thesis is the recognition of human characters in TV-style video sequences. In order to recognize the same person at different time instances in a video sequence, the outward appearance of the person has to be described and learned with an appropriate model. The diversity in which humans can appear makes the task of human detection and character recognition to a particularly challenging problem. TV-style movies provide an uncontrolled, realistic working environment for human detection and character recognition. Possible applications range from surveillance (e.g., intrusion detection) and security applications (e.g., person identification) to image retrieval or semi-automatic image annotation (e.g., automatic labeling of faces in personal photo albums).

2 Approach

A supervised approach is proposed that can be split into three stages. After a pre-processing stage, humans are detected at different time instances in a given movie. Then, the identity of each detected person is computed based on learned models. A more detailed overview on the different stages is given in the following.

Pre-processing. Extracted full-frames of video sequences are considered since no temporal information is included. In order to structure the large amount of data, the extracted frames are grouped into shots. The shot detection is based on the comparison of color histograms of successive frames.

Human Detection. The human detection combines detectors of upper body parts which are *face*, *head*, and *head+shoulders*. They are divided into *frontal* and *side* view (i.e., six detectors in total). The body part (view) detector in this work is based on the approach introduced by Dalal and Triggs [DT05]. Their system samples densely overlapping SIFT-like [Low04] feature points which are trained with a linear Support Vector Machine.

In order to obtain more robust results, single body part detections are combined applying knowledge about geometric relations between the body parts. This is done by learning a Gaussian model similar to the work of Mikolajczyk et al. [MSZ04].

In this way, humans are detected in varying views (frontal and side) and in resolutions varying from distant up to close-up views. Partly occluded people are detected as well. For the training of the body part detectors, an annotation data set is created that consists of roughly 330 training annotations for each body part (of which about $\frac{1}{3}$ are for the side and $\frac{2}{3}$ for the frontal view).

Human Character Recognition. Our approach for the character recognition was motivated by the *Bag-of-Features* method [DWF⁺04, AR02] that has been used in previous work for object classification tasks. The Bag-of-Features method extracts feature points (i.e., image points that are described not necessarily by their color/intensity values, but by their local neighborhood based on, e.g., gradient information) from a set of training images. In their feature space, the feature points are grouped by a clustering algorithm. Based on the resulting clusters (all clusters together are referred to as *code book* and one cluster is referred to as *visual word*), occurrence histograms are generated for each body part image. A classifier is then trained on these histograms. Occurrence histograms reflect how many feature points are assigned to each visual word.

Our approach is build on SIFT- [Low04] and CIE L*u*v* color-based code books that are obtained by clustering with *k*-means. A non-linear multi-class Support Vector Machine (SVM) is learned on occurrence histograms for five main characters and on one category “Others” that consists of all other characters. The trained Support Vector Machines (or SVM models) are then used to predict the identity of a detected person. Probabilistic votes of connected body parts (i.e., body parts that belong to one and the same person) are combined for a more stable prediction. The training data is generated from an annotation data set in which the name of the corresponding character is noted for each body part. Since a supervised approach is applied and since the focus lies on learning character identities for a particular video (e.g., a complete TV/cinema movie), example annotations for this particular movie are created. Based on these annotation data, codebooks are generated and SVM models are learned. The codebooks and the SVM models are then applied subsequently on the entire video sequence.

In this way, particular (human) characters are recognized at different points in time in a given video sequence. Our method takes into account that humans change in pose, perspective, and distance to the camera, that illumination conditions change, and the same character can wear different clothes.

3 Results

Figure 1 shows performance plots for the different body part detectors. The plots show the ROC curve for each single detector as well as for each detector in combination with the other detectors, i.e., employing the model for the geometric relations of the body parts.

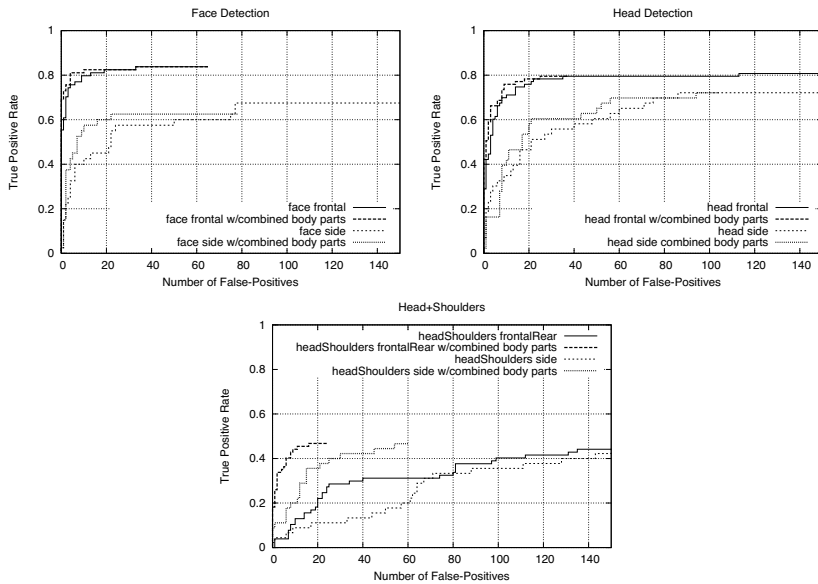


Figure 1: The performance of the (single as well as combined) body part detectors plotted as ROC curve with the number of false positives.

The frontal face detector performs best. It can be seen that the combination of the different detectors can always improve the performance, especially for detectors with an initial low performance.

Figure 2 shows some examples of typical correct detections and character labelings. As example video sequence, the first episode of *Buffy the Vampire Slayer Season 5* has been used. Different orientations and combinations of the body parts are detected successfully (see Figure 2 bottom right, direction is indicated through arrows in the face detections). If the size of a person in an image is too small, his faces cannot be detected. However, in order to detect the person, the detection of head and head+shoulders suffices (see Figure 2 bottom left). Partly occluded and overlapping people are detected correctly as well (see Figure 2 top row).

In the experiments carried out on our annotation set, the character recognition accuracy achieved 81.3% (over all character classes and all body parts using leave-one-out cross validation). For the training of the BoF-SVMs ca. 90 example annotations for each body part have been used for the class “Others” and for the main character Buffy, and ca. 40 examples have been used for the remaining four main characters. Among the cases where the character identification failed, mislabeling of other characters as belonging to one of the main character classes were more common than vice versa. Characters with for example blond hairs were several times classified as Buffy, whereas Buffy herself would be identified correctly in most cases. This is certainly due to the general nature of the class “Others”.

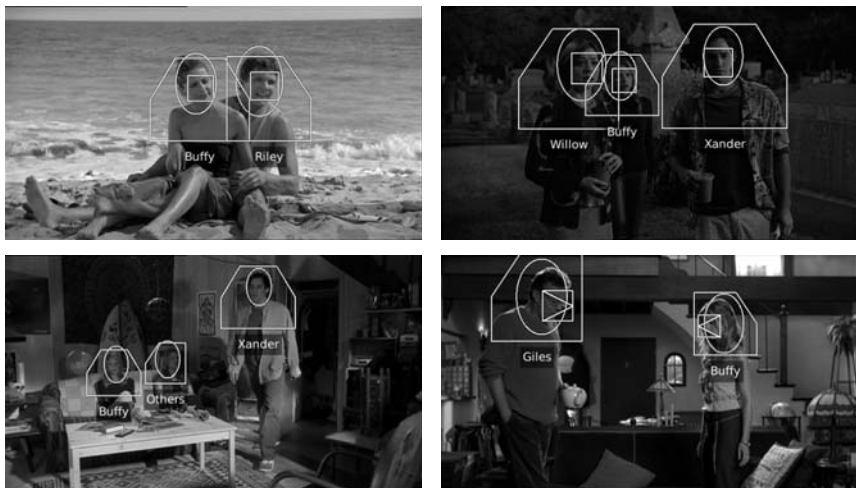


Figure 2: Typical examples for correct final detections and character identifications.

Conclusion. Good detection and recognition results were obtained with a system that combines different body part detectors and a Bag-of-Features based character recognition. Especially the combination of different body parts helps to improve the overall human detection performance. The character identification module worked well on the test sequence, although less sophisticated methods for the clustering (k -means) and the feature extraction (dense sampling) has been employed. Again, the combination of the votes from the different body parts helps to improve the final results.

References

- [AR02] Shivani Agarwal and Dan Roth. Learning a sparse representation for object detection. In *Proceedings of the 7th European Conference on Computer Vision*, volume 4, pages 113–130, 2002.
- [DT05] Navneet Dalal and Bill Triggs. Histograms of Oriented Gradients for Human Detection. In Cordelia Schmid, Stefano Soatto, and Carlo Tomasi, editors, *International Conference on Computer Vision & Pattern Recognition*, volume 2, pages 886–893, INRIA Rhône-Alpes, ZIRST-655, av. de l’Europe, Montbonnot-38334, June 2005.
- [DWF⁺04] Chris Dance, Jutta Willamowski, Lixin Fan, Cedric Bray, and Gabriela Csurka. Visual categorization with bags of keypoints. In *ECCV International Workshop on Statistical Learning in Computer Vision*, 2004.
- [Low04] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.
- [MSZ04] Krystian Mikolajczyk, Cordelia Schmid, and Andrew Zisserman. Human detection based on a probabilistic assembly of robust part detectors. In *European Conference on Computer Vision*, volume I, pages 69–81, 2004.

Curvature Correction for Camera-based Document Processing

David C. Schneider, Freie Universität Berlin (dc.schneid@googlemail.com)

The term *document processing* subsumes technologies for handling printed documents with computers. Ambitious projects pursued by major players of the industry, such as Google's undertakings in book-scanning, as well as large international conferences like the ICDAR prove that this is a prospering area of research. A major topic in document processing is the issue of digitizing printed documents which is a precondition to all further processing. Its core problem, optical character recognition (OCR), is widely considered solved today—if one condition holds: The input image to OCR must be of high quality which is usually the case if the image is acquired from a flat bed scanner. In the recent years, however, digital cameras have become a widespread equipment, and, with prices falling and sensor resolutions increasing, they are becoming more and more attractive as image providers for document processing. Digital cameras have two major advantages over scanners: They are highly portable, and they are fast. These benefits facilitate new applications of document processing: The work described here, for example, was conducted within a research project that aims at developing a wearable reading aid for the visually impaired, combining a digital camera, image processing, OCR, and text-to-speech conversion. However, the images taken with digital cameras “in the field” typically do not meet all quality requirements needed for error free OCR.

This sets up the topic for the thesis at hand: To contribute to closing the gap between a digital camera image of a printed document and an image suitable as input to OCR. While recent OCR software is tolerant towards uneven lighting and noise, geometric distortions have proven to greatly affect the recognition result. An intricate type of distortion frequently occurring in practical application is the non-linear distortion imposed by the curvature of book pages. While the thesis describes several document processing algorithms, its major contribution is an advancement in curvature correction. Note that correction of perspective distortion is implicit in the curvature correction algorithm to be presented.

In the following, only the two most important algorithms introduced in the thesis are described in detail. For the sake of brevity, several preprocessing methods can only be listed in this abstract. These are (1) a fast stochastic algorithm for detecting the overall orientation of a document in an image, (2) a segmentation algorithm for separating elements of the document such as headlines, columns, etc. using distance transforms, and (3) a method for identifying text and document boundaries.

The approaches to the curvature problem described in the literature can be divided into two groups: Those requiring additional hardware and software-only methods. In the first group, a geometric shape with known properties is projected onto the document, the projection is recognized in the digital image and a distortion model is derived from the dif-

ferences between projection and ideal shape. Examples are [Pil01] and [CZZT05]. In the second group, to which my approach belongs, it is recognized that most printed text documents already have intrinsic properties that are known *a priori* so that a projection is not necessary to compute a deviance: For the vast majority of documents the lines of text can be assumed to be parallel. The algorithms thus aim at recognizing the text baselines, i.e. the imaginary lines most characters “rest on”. From these a distortion model is derived which is ultimately used to correct the image. The approaches differ in their strategy for baseline recognition, in their choice of the distortion model and the methods used for correction; see, for example, [WA02], [CDL03] and [ZT05].

My algorithm for **baseline recognition** is based on the fact that along a text baseline there are many sharp transitions from dark (ink) pixels to light (paper) pixels orthogonal to the baseline direction. Assuming that the depicted document is well-oriented in the image (i.e. that text lines roughly go from left to right), the Y component of the image gradient is a suitable measure for these transitions. The algorithm splits up the Y-gradient image into vertical stripes. For each stripe, an oriented measure $f(y)$ of baseline strength is computed for each Y coordinate y in the stripe; details of the formula are provided in the thesis. From local maxima of f a series of feature points is obtained for each stripe. The feature points of incident stripes then serve as control points for cubic spline models of the text baselines. Note that deciding which points to connect is far from trivial: There may be erroneous feature points, the baselines themselves may have different lengths and the splines have to follow an unknown curvature. To cope with these issues, the algorithm first computes hypotheses of ideal line progression based on the already recognized parts of the lines. When selecting the actual points to connect, those with a better fit to a hypothesis are preferred. The result of baseline extraction is a set of cubic splines.

Typically, the approaches to **curvature correction** in the literature directly use the spline models of the text baselines to construct their distortion model. For example, Zhang and Tan ([ZT05]), whose work I used as a starting point, use a two dimensional Gordon surface which can be easily derived from a set of splines (see e.g. [PT97]). Then they rectify the image by resampling it in the surface’s coordinate system. This approach has a major drawback: The distortion model can only be derived if the text baselines fit its requirements, e.g. the requirements of a Gordon surface. In many practical applications this is simply not the case: For example, to rectify an entire page, splines and hence baselines must stretch over the entire page which is not the case in multi column documents. Thus it is desirable to detach baselines from the distortion model in order to obtain greater robustness over a wide range of documents.

In the following I refer to the coordinate system of the input image as the XY-system and to that of the distortion model as the UV-system. In my algorithm, the model is obtained in a two step process. First, a vector field $\mathcal{V} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ is defined, mapping each point \vec{p} in the XY-system to a unit vector \vec{u} pointing in the direction of a projection of a U-parallel line into the XY-system. To clarify this idea, note that the splines modelling text baselines can be seen as such projections of U-parallel lines into the XY-system. Hence the value of \mathcal{V} is known for all points that lie on some baseline spline S :

$$\mathcal{V}(S(p)) = \frac{S'(p)}{\|S'(p)\|} \quad (1)$$

where p is a valid parameter value for the spline and $S'(\cdot)$ is its first derivative. Values of \mathcal{V} at locations offside the line splines can be obtained as follows. Since \mathcal{V} maps to unit vectors, it can also be described by a mapping $\mathcal{V}^* : \mathbb{R}^2 \rightarrow [0 \dots \pi)$ of points to angles:

$$\mathcal{V}^*(S(p)) = \arccos \left(\frac{S'(p) \circ (1 \ 0)}{\|S'(p)\|} \right) \quad (2)$$

whereas \circ denotes the scalar product. Clearly, \mathcal{V} and \mathcal{V}^* are related by

$$\mathcal{V}(S(p)) = \begin{pmatrix} \cos \mathcal{V}^*[S(p)] \\ -\sin \mathcal{V}^*[S(p)] \end{pmatrix} \quad (3)$$

This vector-to-scalar representation \mathcal{V}^* can be used as basis for scattered data interpolation schemes using values of \mathcal{V}^* at locations on the line splines as known data points. In my work I used a scheme based on a Delaunay triangulation and barycentric interpolation of unknown points as described, for example, in [Ami02].

Using the interpolation scheme we can evaluate the vector field at arbitrary points. By numerically approximating *flow lines* of the field we can obtain arbitrary XY-projections of U-parallel lines—completely independent of length and location of the baseline splines we used to obtain the vector field. By using flow lines as the basis for the distortion model the aspired detachment of baselines and distortion model is achieved.

In the second step the actual model is constructed. While the flow lines could be used to build up a Gordon surface or another spline based model, I chose an alternative approach for reasons of computation speed: The nonlinear mapping is approximated with multiple linear ones. Therefor, a dense grid G of points equidistantly distributed along flow lines is computed. Let $G_{i,j}$ denote the i -th point on the j -th flow line in G . Let R_{XY} be a quadrangle in the XY-system with corner points

$$\langle G_{i,j}, G_{i+1,j}, G_{i+1,j+1}, G_{i,j+1} \rangle \quad (4)$$

R_{XY} corresponds to a rectangle R_{UV} in the UV-system with corners

$$\langle (ci, dj), (c(i+1), dj), (c(i+1), d(j+1)), (ci, d(j+1)) \rangle \quad (5)$$

where c and d are scaling factors. The relation between arbitrary points in R_{XY} and R_{UV} is described by a homography whose matrix can be computed from the four corresponding corner points (see, e.g., [FP03]). The overall rectification is computed by linearly mapping a large number of quadrangles in the XY-system to rectangles in the UV-system. The choice of grid density allows a fine grained adjustment of quality and speed—the more rectangles are used, the better the nonlinear mapping is approximated.

All methods described were implemented in C++. The algorithm's capacity of improving OCR results was tested with the commercial OCR package "Abby FineReader Professional 8". Depending on the strength of curvature, up to 25 percent increase of recognition rate could be achieved. The algorithm was also compared to the program's own curvature correction method and outperformed it by up to 21 percent.

References

- [Ami02] Isaac Amidror. Scattered Data Interpolation Methods for Electronic Imaging Systems: A Survey. *Journal of Electronic Imaging*, 11 (2):157–176, 2002.
- [CDL03] Huaigu Cao, Xiaoqing Ding, and Changsong Liu. Rectifying the Bound Document Image Captured by the Camera: A Model Based Approach. In *Proceedings of the Seventh International Conference on Document Analysis and Recognition*, 2003.
- [CZZT05] Kok Beng Chu, Li Zhang, Yu Zhang, and Chew Lim Tan. A Fast and Stable Approach for Restoration of Warped Document Images. In *Proceedings of the Eighth International Conference on Document Analysis and Recognition*, 2005.
- [FP03] David Forsyth and Jean Ponce. *Computer Vision – A Modern Approach*. Prentice Hall, 2003.
- [Pil01] Maurizio Pilu. Undoing Page Curl Distortion Using Applicable Surfaces. In *Proceedings of the IEEE International Conference on Image Processing*, 2001.
- [PT97] Les A Piegl and Wayne Tiller. *The Nurbs Book*. Springer, 1997.
- [WA02] Changhua Wu and Gady Agam. Document Image De-Warping for Text/Graphics Recognition. In *Proceedings Of Joint IAPR 2002 and SPR 2002*, 2002.
- [ZT05] Li Zhang and Chew Lim Tan. Warped Image Restoration with Applications to Digital Libraries. In *Proceeding of the Eighth International Conference on Document Analysis and Recognition*. IEEE, 2005.

Reliable Real-time Detection and Tracking of Multiple Persons in Work Safety Relevant Environments

Moritz Vieth

moritz.vieth@smail.inf.fh-brs.de
Bonn-Rhein-Sieg University of Applied Sciences,
Department of Computer Science,
Grantham-Allee 20, 53757 Sankt Augustin

1 Introduction

Several hundreds of accidents involving the use of hand-fed machines such as circular saws occur each year, resulting in non-reversible injuries of hands, fingers or thumbs [1]. This contribution presents a vision-based approach to the detection of hazardous hand movements in real workshop environments. At first, an overview camera which observes the table saw and the area in front of it detects all humans in its visual field and tracks them. The person closest to the table saw is taken as the focus of attention. The head region of that person is detected since it encompasses the most prominent feature of an operator when working at a machine. The result of this processing step is a reliable identification of the operator's head position in front of the machine [2]. The knowledge of the head and shoulder position determines the search space for the other body parts. In the next step of the processing, the shoulder and upper/lower arm regions are detected. Knowing the location of these body part regions, a body posture detection and analysis is performed. The results of this processing step related to the overview camera system are knowledge about the position and orientation of the body of the operator and if he/she is within the operation area¹.

The objective of this thesis within the entire project context is to ensure a fast and reliable detection and tracking of multiple persons within the visual field of the overview camera. Furthermore, interfaces have to be provided to allow further processing of the acquired data by other sub-systems.

¹Def. of operating area: An area around the table saw from which an operator is able to reach into the saw blade, within the visual field of the overview camera.

2 Person Detection and Tracking

One important step to assess a possible hazardous situation is to determine if a person is within the outer observing area². Thus, it is important that the presence of a human is identified quickly and reliably, and that it can be tracked in subsequent processing steps. To achieve this goal, an Attentive Vision approach integrating model knowledge of the human body and its inheritant constraints is applied to detect multiple humans in the outer observation area. The underlying idea is that humans should be detected some time before entering the extended operation area³. This improves the reliability of subsequent processes. A coarse-to-fine strategy ensures a rapid initial detection of a person, whose position and posture can then be tracked. The detection of a person is divided into three general steps: foreground/background segmentation, head detection and shoulder and arm detection.

2.1 Foreground/Background Segmentation

In real workshop environments the background of a stationary overview camera can and will be subject to changes of illumination as well as of the overall appearance (e.g. objects being moved). To compensate for this, the background image is constantly updated using Gaussian Mixture Models (GMM's) as described in [2]. To prevent persons to be integrated in the background image, the areas of the image in which a person has been detected in the last frame excluded from the update process. Furthermore, a region of interest (ROI) for each detected person is specified in order to reduce the search space for subsequent processing.

2.2 Head Detection

The head detection analysis step exploit multiple image features to restrict and improve the detection process. Skin colour, prominent edges and motion cues are evaluated to determine the head position. This is done for all initial processing steps when this person has not been in the visual field of the camera before. If so, the head detection processes are expanded to a head tracking process in which past positions of the head are evaluated during the processing of the most recent frame. For the evaluation of the skin color areas (like hands or even wood), which are determined as proposed in [3], whether they contain faces or not, a Case-Based Reasoning (CBR) algorithm as proposed by Zhu et. al. [6] is applied. This ensures a reliable detection of faces which is robust against illumination changes. Once a face has been classified, a surrounding ellipsoid as ROI is calculated for all head areas.

²Def. of Observing area: An area similar to the operation area, but extended by a semicircle of 4 meter radius.

³Def. of extenden operation area: An area similar to the operation area, but extended by a semicircle of 1 meter radius.

2.3 Shoulder and Arm Detection

Based on the position of a detected head, the position of the shoulder and upper arm region of a person can be computed. The approach used in this system is proposed in [4]. Using a steerable filter scheme and model knowledge assumptions, the outline of the shoulder and upper arm region can be detected, leading to a robust estimation of the hand positions (see figure 1).

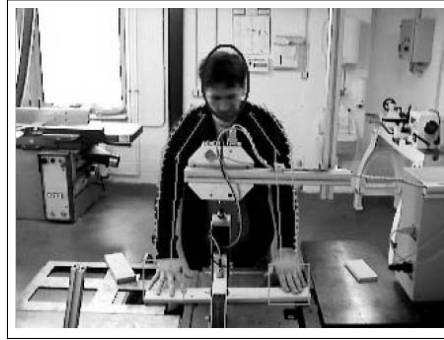


Figure 1: Result of the head detection approach (blue ellipsoid) and shoulder and arm detection algorithm. The green bounding boxes surrounding the hands indicate the estimated hand positions. These areas are used for more detailed analysis steps.

3 Skeletonisation and Body Posture Analysis

An important information regarding work safety is not only the knowledge about the position of a person in the image, but rather its posture, especially that of persons in the *operating area*. The reconstruction of this posture is an ongoing research topic in our department [5]. For this, a 2D skeleton model of the person is needed as input. This skeleton is computed for the person in the *operating area*, combining the information obtained from prior processing steps and model knowledge assumptions.

4 Conclusion and Future Work

Preliminary results have shown that this work provides a suitable approach for a reliable solution of the main objective of this thesis, the detection and tracking of multiple persons in work safety relevant environments, which poses a far greater challenge than working in lab environments. Persons are detected and tracked properly even under illumination changes and with a moving and cluttered background.

The benefits of this approach are not only relevant for table and panel saw security applications, but can also be transferred to other work safety relevant applications.

This project, which is done in cooperation with the German Board for Work Safety, is the first successful attempt of applying a vision based computer system for securing work

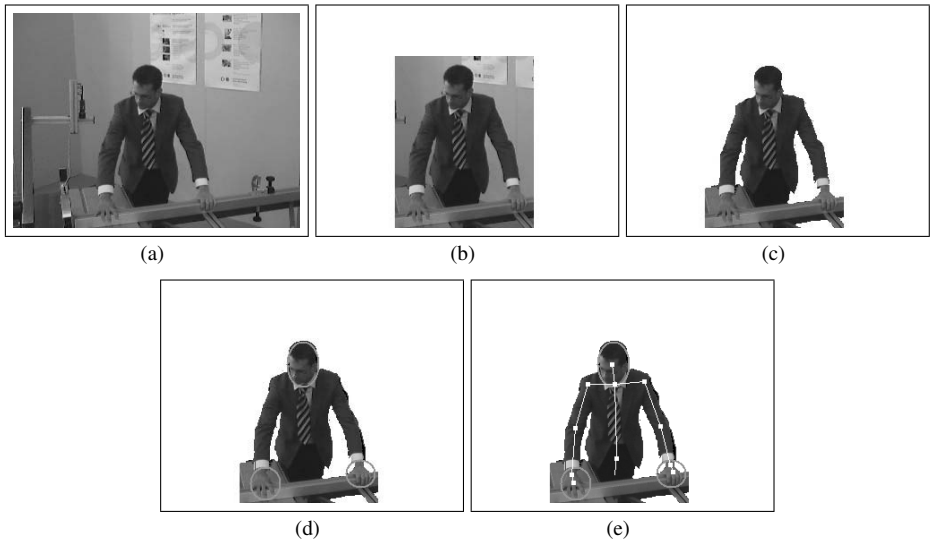


Figure 2: The person detection pipeline: (a) Full frame image, (b) ROI surrounding the operator, (c) Background Segmentation, (d) Head-Arm Detection, (e) Skeletonisation

safety relevant areas.

A final evaluation of the performance however needs to be performed to further identify strengths and weaknesses of the presented approach.

References

- [1] Reinert, D., et.al. : Finger and Hand Protection on Circular Table and Panel Saws, *4th Int. Conf. on Safety of Industrial Automated Systems*, Elect. Publishing, Chicago, USA (2005).
- [2] Barth, A., Hergers, R. : Robust head detection and tracking in cluttered workshop environments using GMM, *27th Annual meeting of the German Association for Pattern Recognition*, Vienna, Austria, (2005)
- [3] R. Hergers, K. Derpanis, W. J. MacLean, G. Verghese, M. Jenkin, E. Miliotis, A. Jepson, J.K. Tsotsos : SAVI: an actively controlled teleconferencing system, *Journal Image and Vision Computing*, Vol. 19(11), pp. 793-804 (2001).
- [4] Hahne, S., Hergers, R., Barth, A., Reinert, D. : Real-Time detection and Tracking of Human Body parts in Real Working Environments, *REM 2006 Research and Education in Mechatronics KTH*, Stockholm, Schweden (2006).
- [5] Krausz, B. : Posture Reconstruction, *Semester Project, Department of Computer Science, University of Applied Sciences Bonn-Rhein-Sieg*, Sankt Augustin, Germany (2006).
- [6] Zhu, Z., Liao, W. and Ji, Q.: Robust Visual Tracking Using Case-Based Reasoning with Confidence, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2006).

Reliable Detection and Tracking of Hands in Work Safety Relevant Environments

Oliver Zilken

Oliver.Zilken@smail.inf.fh-bonn-rhein-sieg.de

1 Introduction

Several hundreds of accidents involving the use of hand-fed machines such as circular saws occur each year, resulting in non-reversible injuries of hands, fingers or thumbs [1]. This contribution presents a vision-based approach to the detection of hazardous hand movements in real workshop environments. A contactless detection approach of the hand region and the fingers has been investigated to provide comprehensive protection on circular table and panel saws. The hazard area¹ is observed and evaluated by a camera system, which is located right above the saw blade. It exploits the information about the scale, orientation and shape of an expected hand extremity provided by a preceding processing step, and recognizes the relative position of the fingers of the operator. When the fingers are too close to the saw blade, an alarm is computed, which will be used to prevent the operator of getting injured.

2 Hand Detection

In the beginning, a precedent processing steps report hands in the visual field of a second camera system. This other camera system is located in front of the table saw, with a height of 2 meters and observing a large overview area in front of the saw. It can detect and track the body shape of an operator and gives alarm, if the operator moves his arm within a small range to the sawblade (This other system will not be further described here, as it is part of another work). At this point the main camera system, which is located perpendicular to the saw table and observes an area about four times as big as the hazard are, is activated in order to evaluate the exact position of the hand and fingers.

An initial detection of the hand and the fingers of the operator is computed. The detection process is divided into several processing steps. After a preprocessing step, in which the original input image is undistorted with the intrinsic parameters of the camera using Heikkilä and Silvén's method from [6], the following tasks are computed.

¹Def. of hazard area: An area of 12 cm surrounding the saw blade, as defined by the German Board of Work Safety [1].

2.1 Background Segmentation

For the segmentation of the hand region, the search space should be reduced as much as possible. It is assumed that the hand is moving into the observation area with a certain velocity, so that motion cues might support the segmentation process. First evaluations have shown that on the one hand Gaussian Mixture Models (GMM) [2] provide an efficient and adaptive foreground/background segmentation. On the other hand they are not very performant and it takes a couple of frames until an object from the background is classified as an foreground object. For efficiency reason due to the real time requirements, an adaptive differential image based motion analysis approach has been developed. It is basically based on the method shown in [7](p. 310), and is improved by adding high level feedback

$$BG_n = \frac{w_a I_n + \sum_i^N w_i BG^{(n-i)}}{w_c} + FB_{n-1} \quad (1)$$

It considers up to $N = 8$ preceding images. All preceding images are computed with different weights w_a, w_i and w_c . In addition, the classification result of the last frame (FB_{n-1}) is used as high level feedback to improve the background image BG_n . Figure 1(c) shows the result of this segmentation step.

2.2 Skin Color Recognition

In this project, skin color recognition is used as a supporting feature for the later hand classification. Veznevets et.al. showed in [3] that many of the currently used skin color algorithms produce a high rate of false positives and false negatives (ca.9-30%). The best performance for this application showed the work presented by Singh et. al. in [4] which had an accuracy of over 95%. It is based on a combination of the three color space RGB, YCbCr and HSI. The results of the three color spaces are then combine using different weights w_1, w_2 and w_3 for each result:

$$PI_{skin}(x, y) = w_1 * I_{rgb}(x, y) + w_2 * I_{ycbcr}(x, y) + w_3 * I_{hsv}(x, y) \quad (2)$$

Figure 1(c) shows the resulting graylevel skin color propability map image PI_{skin} . Black color means the lowest propability for this pixel containing skin color, white color the highest propability.

2.3 Hand Classification

The final classification of hand regions is the most challenging part of this project. Many different approaches for this task have been evaluated [15][13] [10] [11] [14] [5]. No existing approach could deliver a robust hand classification, when a skin color similar wooden workpiece is present in the image. So another algorithm has been used, which was originally developed for face recognition. This object detector, first published by Viola and

Jones in [8] and improved by Lienhart et.al. in [9] uses a cascade of ada-boosted classifiers working with haar-like features. This classification cascade has been trained using 500 positive and 3500 negative example images. Using this trained data, the classification is fast, exact and robust against changes in illumination. In order to improve the classification result, hand region candidates are verified using the skin color probability map from the previous step. If a region has been successfully classified as a hand region, then this region is passed to next frame as high level feedback for the background segmentation. Figure 1(d) shows the result of this classification step.

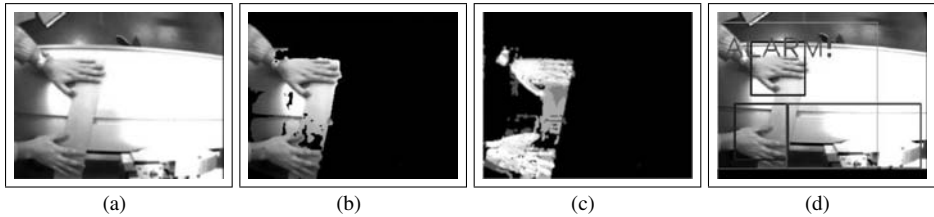


Abbildung 1: Processing steps of the hand detection application: (a) Full frame image, (b) background segmented image, (c) skin color probability map image, (d) output image including the classified hand regions (blue) and the hazard area around the rotating saw blade (red).

2.4 Hand Tracking

For efficiency reasons, the rectangular hand region is tracked assuming a reliable and robust detection and classification in previous steps. For tracking, a mixture of motion estimation methods and a camshift algorithm [12] is used.

3 Conclusion

First evaluations have shown that this system approach provides a reliable solution for the detection of persons and their body parts in work safety relevant environments. The hand classification algorithm is fast and reliable enough to even distinguish between the hand of an operator and hand-coloured wooden workpieces. The benefits of this approach are not only relevant for table and panel saw security applications, but can also be transferred to other work safety relevant applications. This project, which is done in cooperation with the German board for work safety, is the first attempt of applying a vision based computer system for securing work safety relevant areas. A final evaluation of the performance however needs to be performed to further identify strengths and weaknesses of the presented approach.

Acknowledgments

This work has been supported by the German Hauptverband der Berufsgenossenschaften (HVBG), which is gratefully acknowledged.

Literatur

- [1] Reinert, D., et.al. : Finger and Hand Protection on Circular Table and Panel Saws, *4th Int. Conf. on Safety of Industrial Automated Systems*, Elect. Publishing, Chicago, USA (2005).
- [2] Barth, A., Herpers, R. : Robust head detection and tracking in cluttered workshop environments using GMM, *27th Annual meeting of the German Association for Pattern Recognition*, Vienna, Austria, (2005)
- [3] Vezhnevets, V., Saszonov, V., Andreeva, A.: A Survey on Pixel-Based Skin Color Detection Techniques, Graphics and Media Laboratory, Moscow State University (2002)
- [4] Singh, S.Kr., Chauhan, D.S., Vatsa, M., Singh, R.: A Robust Skin Color Based Face Detection Algorithm, *Tamkang J. of Science and Engineering*, Vol.6, No. 4, pp. 227-234 (2003).
- [5] Stefan Hahne: Modell- und bildbasierte Detektion und Verfolgung der Arme in Videosequenzen, *Diplomarbeit, Fachbereich Informatik der Fachhochschule Bonn-Rhein-Sieg* 2005.
- [6] J. Heikill, O. Silven: A four-step camera calibration procedure with implicit image correction, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition 1997*, pp. 1106 - 1112.
- [7] David A. Forsyth, Jean Ponce: Computer Vision - A modern approach, *Pearson Prentice Hall*, 2003, ISBN 0-13-191193-7.
- [8] P. Viola, M. Jones: Rapid object detection using a boosted cascade of simple features, *CVPR, 2001*, citeseer.ist.psu.edu/viola01rapid.html.
- [9] R. Lienhart, Luhong Liang, A. Kuranov: A detector tree of boosted classifiers for real-time object detection and tracking, *ICME '03: Proceedings of the 2003 International Conference on Multimedia and Expo*.
- [10] M. Kolsch, M. Turk: Fast 2D Hand Tracking with Flocks of Features and Multi-Cue Integration, *CVPRW '04: Proceedings of the 2004 Conference on Computer Vision and Pattern Recognition Workshop*, Volume 10, pp. 158-165.
- [11] Y. Ma, F. Pollick, W. Hewitt: Using B-Spline Curves for Hand Recognition, *ICPR '04: Proceedings of the 17th International Conference on Pattern Recognition*, Volume 3, pp.274-277.
- [12] Jose Dias, P. Nande, N. Barata, A. Correia: O.G.R.E. - Open Gestures Recognition Engine, *SIBGRAPI '04: Proceedings of the XVII Brazilian Symposium on Computer Graphics and Image Processing*, pp. 33-40.
- [13] C. Vogler, D. Metaxas: Adapting Hidden Markov models for ASL recognition by using three-dimensional computer vision methods, *SMC'97*, 1997.
- [14] Konstantinos Derpanis: Vision based Gesture Recognition within a Linguistics Framework, *Master thesis, York University, Toronto, Ontario, Canada*, 2003.
- [15] C. Demant, B. Streicher-Abel, P. Waszkewitz: Industrial Image Processing, *Springer Verlag*, 1999, ISBN 3-540-66410-6.

Verwundbarkeiten von Web-Anwendungen

Christopher Holm, Christopher Schwardt

3holm@informatik.uni-hamburg.de, 3schwardt@informatik.uni-hamburg.de

Abstract: Web-Applikationen werden heute immer wichtiger, für viele Firmen sind sie die Haupteinnahmequelle, für einige auch die Einzige. Gleichzeitig werden diese Applikationen auch immer umfangreicher und komplexer. Diese Entwicklung führt dazu, dass der Code immer schwerer zu überblicken ist und auch ein *Audit* der gesamten Applikation den Zeit- und Kostenrahmen vieler Unternehmen sprengen würde. Durch die Zusammenarbeit vieler Menschen schleichen sich häufig logische Fehler in die Anwendung. Diese sollen hier jedoch nicht thematisiert werden. Wir konzentrieren uns hingegen auf gängige Implementationsschwachstellen. Häufig könnten derartige Sicherheitslücken durch eine bessere Ausbildung und Sensibilisierung der Programmierer vermieden werden.

1 Cross Site Scripting (XSS)

Eine der heutzutage am bekanntesten und mit am weitesten verbreiteten Sicherheitslücken in Web-Applikationen ist *Cross-Site Scripting*. XSS bezeichnet das Einfügen eigenen Codes, meist *JavaScript*, in fremde Seiten.

Die große Gefahr hierbei liegt darin, dass JavaScript mächtig genug ist, den gesamten Seiteninhalt einer für XSS anfälligen Seite zu verändern. Mit Hilfe von *iframes* und dem seit Web 2.0 so bekannt gewordenen *XMLHttpRequest* Objekt können sogar komplexe Aktionen im Namen des Benutzers ausgeführt werden, ohne, dass dieser etwas davon mitbekommt. Ausserdem können so *Cookies* dieser Domain ausgelesen und verändert werden. Somit kann der Angreifer das Verhalten der Seite nach Belieben ändern, Aktionen im Namen des Opfers ausführen oder sensitive Daten auslesen.

XSS ist eine der wenigen Methoden, um die *Same Origin-Policy* zu umgehen. Diese besagt, dass nur Daten, die von der gleichen Domain stammen, ausgelesen werden dürfen. Eine Subdomain darf nur auf Daten ihrer eigenen Subdomain, Subsubdomains von ihr und der explizit nur über ihr liegenden Domains zugreifen, nicht jedoch auf Daten benachbarter Subdomains.

Letztendlich sind die Möglichkeiten nur durch die Kreativität des Angreifers beschränkt. Es bleibt noch abzuwarten bis die ersten Würmer die im Moment so beliebten Online-Communities heimsuchen.

2 Session Riding/Cross Site Request Forgery (CSRF)

Cross-Site Request Forgery wurde etwa 2000/2001 bekannt und ist somit eine noch relativ neue Schwachstelle, zieht nun jedoch langsam immer mehr Aufmerksamkeit auf sich. Der neuere Name, *Session Riding*, beschreibt die Sicherheitslücke etwas plastischer, nämlich das Weiterbenutzen einer bestehenden Session.

Ziel einer solchen Attacke ist es, Aktionen auf einer Webseite durchzuführen, dies aber im Namen eines fremden Benutzers.

Das Standardszenario ist folgendes:

Der Benutzer wird auf eine Seite gelockt, die sich unter der Kontrolle des Angreifers befindet. Auf dieser wird im Hintergrund die Webseite mit gezielten Parametern geladen, auf der der Angreifer nun eine Aktion im Kontext des Opfers durchführen will. Unter der Voraussetzung, dass der Benutzer zu diesem Zeitpunkt auf dieser Seite angemeldet ist, wird sein Browser bei der Anfrage einen Cookie mitsenden, so dass die Aktion im Namen des Benutzers ausgeführt werden kann.

So können zum Beispiel, falls keine Vorkehrungen gegen solche Angriffe getroffen wurden, Passwörter oder andere Benutzereinstellungen ohne Wissen des Angegriffenen geändert oder auch Nachrichten im Namen des Benutzers verschickt werden, je nachdem, welche Services die angegriffene Seite anbietet.

Leider wird dieser Schwachstelle nicht so viel Aufmerksamkeit geschenkt und so ist eine derartige Lücke heutzutage, einfach auf Grund mangelnder Bekanntheit, verbreiteter als zum Beispiel SQL-Injection-Verwundbarkeiten (siehe unten), gegen die in Programmiersprachen teilweise bereits standardmäßig Vorkehrungen getroffen werden, wie *Magic Quotes*.

3 Session Hijacking

Beim sogenannten *Session Hijacking*, einer Angriffstechnik, die meist, aber nicht ausschließlich, auf unverschlüsselte Kommunikationsverbindungen angewendet wird, übernimmt der Angreifer mit Hilfe gültiger Session- oder Authentifizierungs-Informationen des Benutzers dessen laufende Sitzung.

Die Angriffsmöglichkeiten sind vielfältig und reichen von der Übernahme einer *HTTP*-Session (z.B. Zugriff auf den passwortgeschützten Bereich einer Webanwendung) bis hin zur Entwendung einer *TCP*-Verbindung (z.B. *Telnet*-Hijacking).

Zuerst sammelt der Angreifer mit Hilfe von *Sniffing*, *Man-in-the-middle*- oder sonstigen Angriffen ausreichend Daten, um die Session übernehmen zu können. Dann „entführt“ er sozusagen die Sitzung des Angegriffenen und führt sie selbst weiter.

Für den Angegriffenen erscheint die Übernahme meist wie jeder andere hin und wieder vorkommende Verbindungsverlust und er ist nicht in der Lage, sie als Angriff zu identifizieren.

4 Session Fixation

Session Fixation ist eine Technik, mit der man die Session eines Benutzers übernehmen kann. Sie basiert darauf, dem Benutzer, dessen Session entwendet werden soll, eine dem Angreifer bekannte Session-ID zukommen zu lassen, die dieser dann verwendet. Dies kann auf unterschiedliche Art und Weise geschehen.

Die einfachste Methode ist es, dass sich der Angreifer eine beliebige Session-ID aussucht und das Opfer dazu bringt, auf die gewünschte Seite zuzugreifen, jedoch über einen vom Angreifer erstellten Link. Der leichteste Weg ist die Mitgabe der Session-ID über einen *GET*-Parameter. Die Web-Applikation erstellt nun eine neue Session mit der angegebenen Session-ID und sobald sich der Benutzer angemeldet hat, kann die Session durch die Kenntnis der Session-ID bernommen werden.

Viele Session-Management-Implementationen verhalten sich glücklicherweise nicht so. Sie akzeptieren nur die von ihnen erstellten Sessions-IDs.

Dies ist jedoch auch nicht viel schwerer zu umgehen, da der Angreifer lediglich eine neue Session erstellen muss, bevor er den gefälschten Link erzeugt. Somit muss sich der Angegriffene innerhalb des Session-Timeouts anmelden, damit keine neue Session angelegt wird.

Eine sehr effektive Methode, es dem Angreifer möglichst schwer zu machen, ist es, die Session-ID nur über Cookies zu transportieren, da durch die Same Origin-Policy das Injizieren eines Session-ID-Cookies in den Browser des Opfers deutlich erschwert wird.

Im Gegensatz zum Session Hijacking (siehe voriger Abschnitt) stellt die Verschlüsselung des Datenverkehrs keinen Schutzmechanismus dar.

5 Directory Traversal

Beim sogenannten *Directory Traversal* nutzt ein Angreifer Schwachstellen in HTTP-Servern oder darauf laufenden Applikationen, um auf Dateien zuzugreifen, die eigentlich nicht für ihn bzw. den Webserver zugreifbar sein sollten.

Webserver liefern ihre Dateien aus einem sogenannten *DocumentRoot* heraus. Oberhalb dieses Verzeichnisses sollte der Webserver den Zugriff auf Dateien und Ordner verweigern, da unter Umständen wichtige Konfigurationsdateien oder andere Daten ausgelesen werden könnten.

Der Webserver sollte seinen Dateisystemzugriffen immer seine *DocumentRoot* voranstellen: ein Zugriff auf `http://www.example.com/index.php`, der für den Server selbst einen Zugriff auf `/index.php` bedeutet (wobei `/` hier das *DocumentRoot* darstellt), muss im Dateisystem ausgeführt werden als Zugriff auf `/var/www/index.php`. Dies ändert jedoch nichts an der Verwundbarkeit.

Fügt ein Angreifer Strings wie „../“ oder Umschreibungen dieser Zeichenkette in seine Serveranfrage ein, so kann er aus dieser Beschränkung ausbrechen.

6 Remote File Inclusion (RFI)

Diese Schwachstelle, für die (fehlkonfiguriertes) PHP besonders anfällig ist, ermöglicht es dem Angreifer, beliebigen Code in eine Applikation einzubinden. Meist bezeichnet *Remote File Inclusion* das Einbinden von Dateien des Angreifers, die auf dessen Webserver oder dem eines kostenlosen Webhosters liegen, welche dann vom angegriffenen Server ausgewertet werden, als wären sie Teil der dort laufenden Applikation. Da die Auswertung serverseitig passiert, werden die Attacken mit allen Rechten der ausgenutzten Anwendung ausgeführt und haben Zugriff auf die entsprechenden Daten wie Dateien und Umgebungsvariablen des Servers, Benutzersessions, etc. Passiert das Auswerten clientseitig, so fällt der Angriff meist in die bereits vorgestellte Kategorie Cross Site Scripting (XSS).

7 SQL Injection

SQL Injection ist eine Schwachstelle, die, unabhängig von der Programmiersprache, auf interaktiven Webseiten mit Datenbankbindung auftreten kann. Werden die Benutzereingaben nicht ausreichend geprüft und in der Datenbankabfrage verwendet, so kann ein Angreifer SQL-Code einschleusen. Auf diese Weise ist nicht nur das Ausspähen der Datenbankinhalte möglich - da in SQL-Datenbanken lange Zeit brauchbares Benutzermanagement fehlte und es sich auch nach dessen Einführung nur langsam durchsetzt, läuft selbst Software, die eigentlich nur Lesezugriff auf die Datenbank benötigen würde, mit Schreibrechten und jeder Angreifer ist prinzipiell in der Lage, die zur Anwendung gehörigen Datenbankinhalte komplett zu löschen, überschreiben oder verändern. Wenn der Angreifer zum Beispiel die Datenbankstruktur einer Software herausgefunden hat, die die Datenbank zur Benutzerverwaltung nutzt, so kann er sogar bei schlecht konfigurierten Webservern in der Lage sein, sich Administratorrechte zu beschaffen.

Literatur

- [HGM04] Hoglund, Greg und Gary McGraw, *Exploiting software (How to break code)*, Addison-Wesley Professional, 2004.
- [Als06] Alshanetsky, Ilia (2006), *Die Gefahren von XSS und CSRF*, <http://phpsolmag.org/de/phpsolmag/download.html> (aufgerufen 12. Dezember 2006)
- [Hus04] Huseby, Sverre H., *Innocent Code: A Security Wake-up Call for Web Programmers*, Wiley & Sons, 2004.
- [Kap] Kapoor, Shray (no date), *Session Hijacking - Exploiting TCP, UDP and HTTP Sessions*, <http://www.infosecwriters.com/-text/resources/-pdf/-SKapoor/-SessionHijacking.pdf> (aufgerufen 12. Dezember 2006)
- [Jak06] Edge, Jake, (2006), *Remote file inclusion vulnerabilities*, <http://lwn.net/Articles/203904/> (aufgerufen 12. Dezember 2006)

Virtualisation of a SIM-Card using Trusted Computing

Subscriber Authentication with virtual software SIM Credentials on Trustworthy Operating Platforms supporting Multiple-Trusted-Engines

Michael Kasper
PFFH Darmstadt
kasper.michael@quadrat-o6.de

1 Introduction

During the last decade, mobile communication systems became one of the fastest growing and most demanding telecommunication technology worldwide. With its phenomenal success it is rapidly becoming one of the most popular equipment for human communication. In this regard, enterprise IT-Managers and Mobile Network Operators are strongly demanding for a new security approach next to future mobile handsets and smartphones [iGR06].

In order to meet these requirements, the Trusted Computing Group's Mobile Phone Work Group (TCG MPWG) has published a specification draft, which offers new potentials for implementing trust in mobile computing platforms by introducing a hardware-based trust anchor supporting multiple trusted engines on behalf of specific stakeholders [TCG06a, TCG06b]. This trust-anchor is called Mobile Trusted Module (MTM) and has comparable properties and features with a Trusted Platform Module [TCG04, TCG05].

The TCG MPWG specification promises the development of more secure mobile phone applications, by integrating this architecture into upcoming mobile phones as a set of tamper resistant trusted engines. Therefore, this specification offers so-called shielded capabilities and protected functionality to the trusted mobile device and each trusted engine.

2 Subscriber Authentication with vSIMs based on Trusted Computing

The primary goal of this thesis is to develop a software replacement for a Subscriber Identity Module (SIM) based on Trusted Computing in order to access a mobile cellular network and its offered services. Therefore, we introduce a virtual software SIM with comparable usage and security characteristics like the traditional smartcard-based solution. Additionally, running a virtual SIM as trusted and protected software on a mobile device

allow significant expansion of services by introducing new usage scenarios and business models, cost reduction and more flexibility. Unlike other virtual SIM systems [Imp], our approach demonstrates for the first time the substitutability of a SIM card with an adequate trusted software module supported and protected by a trustworthy operating system.

In this context, we consider the TCG MPWG specification from a slightly different point of view as it is envisaged by the TCG. Even though, SIM-based authentication is the established and proposed means for user authentication and securely accessing mobile cellular networks, an alternative is coming up with the advent of the TCG technology in mobile devices.

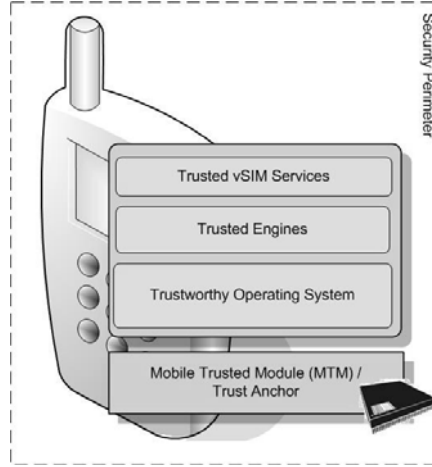


Figure 1: Trusted Mobile Platform with trusted vSIM services

Due to the capabilities of a mobile trusted platform to support multiple trusted engines with protected storage, strong isolation and secure communication within a defined security perimeter [TCG06a, TCG06b], the trusted platform is able to take over the SIM functionality. Figure 1 schematically shows the layout of such a trusted platform. It holds a virtual software SIM module which substitutes the traditional smartcard and its functionality.

2.1 Scenario

The model under consideration is illustrated in Figure 2 and involves four significant entities: the user (U), the trusted mobile platform (TP), the Mobile Network Operator (MNO), and the Point-of-Sale/Point-of-Presence (POS). In this scenario, U wants to establish a long-time relationship with the MNO (Step 1), in order to use the mobile network infrastructure and its offered services.

Instead of purchasing a physical SIM card, the MNO provides the user, or more precisely

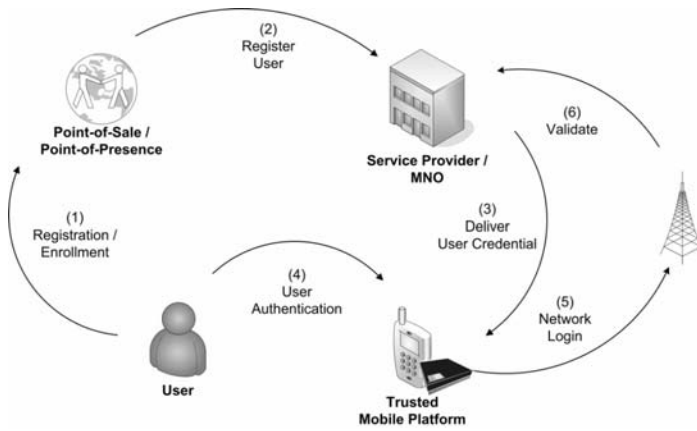


Figure 2: Generic Trusted Mobile Scenario

the trusted vSIM service, with a virtual software SIM credential (Step 3). Every time the user wants to access the mobile network, U authenticates itself to TP (Step 4), which uses the vSIM credential to perform network authentication (Step 5,6).

2.2 Conceptual Models for Subscriber Authentication with vSIMs

Based on this scenario, we design three intergraded conceptual models for authentication in mobile cellular networks using trusted computing. With these models we show how a traditional SIM-Card could be replaced by a software emulation, which runs within isolated environments, protected and supported by trusted engines. We call this software emulation a "trusted virtual Subscriber Identity Module" - or in short - vSIM.

Furthermore, we discuss how user enrollment and key delivery mechanisms could be carried out efficiently.

2.2.1 Model "One"- Subscriber Authentication with compatibility to GSM - Authentication

The approach for model "One" is straightforward to actual GSM authentication standard [ETS97a]. It enables subscriber access in mobile cellular networks based on Trusted Computing with compatibility to regular GSM authentication. This model could be implemented in conventional GSM clients without any technological changes at the GSM infrastructure and at the GSM authentication protocol.

The main task of the trusted vSIM is to take over the functional range of a traditional SIM card, with no additional duties and responsibilities regarding to GSM 11.11 SIM specification [ETS97b]. The cryptographic algorithms A3 and A5, responsible for user authentication and key generation are implemented within the trusted vSIM core services.

2.2.2 Model "Two" - Subscriber Authentication with Remote Attestation for Basic Network Access

A more comprehensive approach compared with the precedent one is presented by this model. Additionally to model "One", we integrate remote attestation for basic network access. A variant of this method has been described in [KS06]. Beside the main task

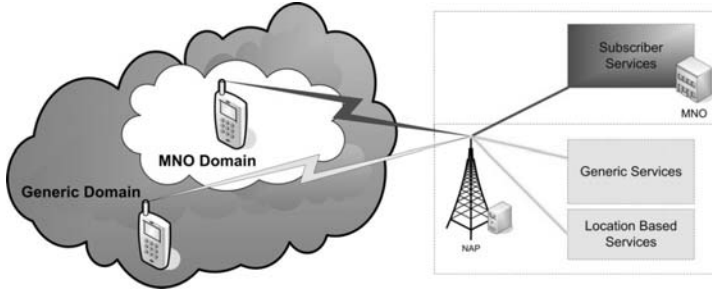


Figure 3: Generic- and MNO Network Access

of SIM substitution, it provides device-authenticated access to the generic domain, user-authenticated access to the subscriber subdomain, mutual authentication between the *MNO* and *TP*, finer-grained functional restriction (e.g. SIM-lock), and dynamic down-/upgrade of services.

As shown in Figure 3, all devices inside a generic domain are able to use generic services of the mobile communication network. A trusted platform which is located in the *MNO* domain has access to both specific subscriber-authenticated services and generic services.

Such generic services, for instance, may be location-based information or WLAN-based internet access. A trusted platform attached to the generic domain uses a generic credential based on remote attestation mechanisms, to gain basic network access. The assignment to the subscriber domain of *MNO* is then done by performing a user-specific authentication process using a vSIM credential.

2.2.3 Model "Three" - Generalized Subscriber Authentication in Network Infrastructures based on Trustworthy Operating Platforms

Finally, we introduce to an authentication model that captures the different aspects of the precedent models from a more abstract point of view. Here, we present a proposal for user- and device authentication using vSIM credentials in generic network infrastructures. This model is based on Trusted Computing and like the predecessor it supports remote attestation and mutual authentication. In contrast to the previous models, we are using more generalized assumptions and specifications. This concerns particularly the structure and abilities of a vSIM credential and the trusted vSIM services. Thus, it provides an opportunity for supporting manifold scenarios for authentication and authorization beside the mobile domain.

3 Prototypical Implementation on Trustworthy Operating Platforms

The prototypical implementation of the trusted engines TE_U , TE_{MNO} and TE_{DM} , and the specified vSIM services are realized as an extension to the existing Turaya Computing Platform. Turaya is an implementation of the EMSCB security architecture [EMS06, SSP04, PER]. It provides fundamental security mechanisms and a protected execution environment, which meet the requirements of the TCG MPWG specification.

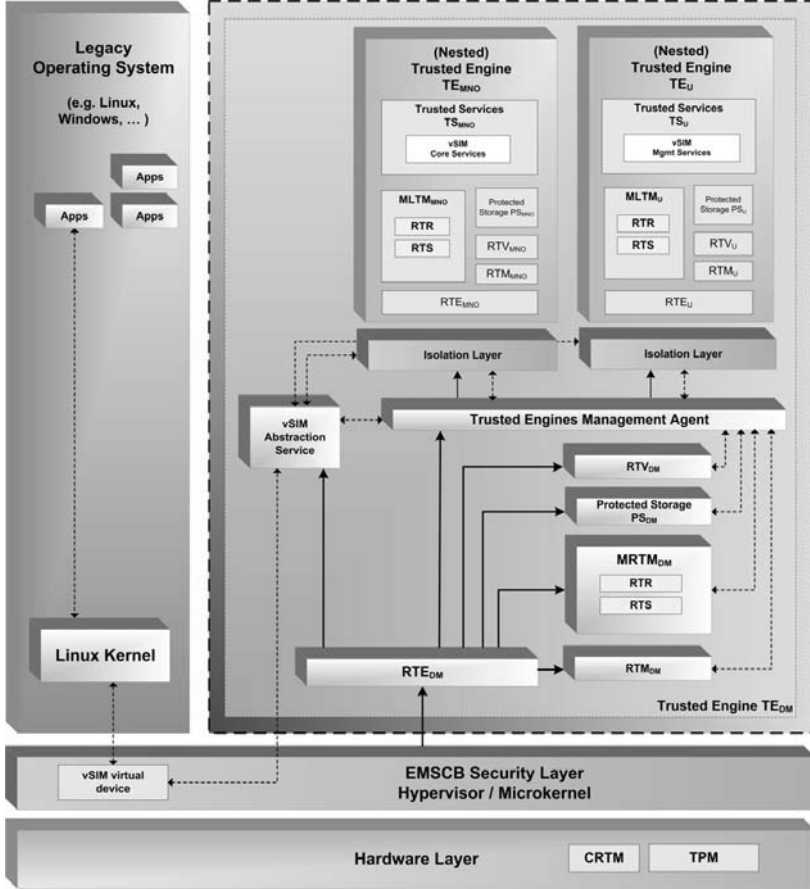


Figure 4: Trustworthy Operating Platform with Multiple-Trusted-Engines

Figure 4 shows our approach, where a hypervisor/microkernel executes a legacy operating system in coexistence with a running instance of the EMSCB-based security architecture. The latter controls a virtual machine with several trusted engines and services compliant to the TCG requirements [TCG06a, TCG06b]. Each engine holds a set of trusted services on behalf of a specific stakeholder, which are executed within isolated execution environments [PER, BBH⁺98, Hoh96] on top of the security kernel or nested inside

TE_{DM} . In particular, TE_U is responsible for vSIM management and user authentication and TE_{MNO} holds the trusted vSIM core services (e.g. the GSM security algorithms A3/A8 [ETS97a]).

4 Conclusion and Further Work

We have introduced to vSIM credentials as a means for subscriber authentication based on the TCG MPWG technology. It offers a real alternative to SIM-based solutions, if an equal degree of security and usage characteristics are reached. On a trustworthy operating platform, like the EMSCB platform, an adequate level may be achieved.

Furthermore, we have realized significant parts of TCG MPWG architecture upon a standard TPM 1.2 trust-anchor. Using a vSIM as a trusted and protected software allows expansion to a much wider field of authentication and identification management systems on standard PC platforms [DEPY03]. Supporting online transactions by vSIM authentication may be one thinkable approach. However, there are some privacy and security challenges associated with this implementation on a desktop computer, which needs a further research.

References

- [BBH⁺98] R. Baumgartl, M. Borriss, Cl.-J. Hamann, M. Hohmuth, L. Reuther, S. Schönberg, and J. Wolter. Dresden Realtime Operating System (DROPS). In *Workshop of System-Designed Automation*, (SDA'98), 1998.
- [DEPY03] Jane Dashevsky, Edward C. Epp, Jose Puthenkulam, and Mrudula Yelamanchi. SIM Trust Parameters. *Intel Developer Update Magazine*, 2003.
- [EMS06] EMSCB. European Multilaterally Secure Computing Base - Towards Trustworthy Systems with Open Standards and Trusted Computing. <http://www.emscb.com>, 2006.
- [ETS97a] ETSI. Digital cellular telecommunications system (Phase 2); Security related network functions (GSM 03.20 version 4.4.1). Technical report, European Telecommunications Standards Institute, 1997.
- [ETS97b] ETSI. GSM 11.11 - Digital cellular telecommunications system (Phase 2+); Specification of the Subscriber Identity Module - Mobile Equipment (SIM - ME) interface. Technical report, European Telecommunications Standards Institute, 1997.
- [Hoh96] Michael Hohmuth. *Linux-Emulation auf einem Mikrokernel*. PhD thesis, TU Dresden, Fakultät Informatik, Lehrstuhl Betriebssysteme, 1996.
- [iGR06] iGR. Worldwide Wireless and Mobile Market Forecast, 2005-2010. Technical Report 022006-02, iGillottResearch Inc., 2006.
- [Imp] Implementa GmbH. Virtual SIM System for Carrier Class GSM Gateways. <http://www.implementa.com>.

- [KS06] Nicolai Kuntze and Andreas U. Schmidt. Trusted Computing in Mobile Action. In *Peer reviewed Proceedings of the ISSA 2006 From Insight to Foresight Conference*. Information Security South Africa (ISSA), 2006.
- [PER] PERSEUS. The Perseus Security Framework. <http://www.perseus-os.org>. Applied Data Security Group, Ruhr-University Bochum.
- [SSP04] Ahmad-Reza Sadeghi, Christian Stübke, and Norbert Pohlmann. European Multilateral Secure Computing Base - Open Trusted Computing for You and Me. *Datenschutz und Datensicherheit (DUD)*, pages 548–554, 2004.
- [TCG04] TCG. TCG Specification Architecture Overview, Specification Revision 1.2, 2004.
- [TCG05] TCG. Trusted Platform Module Specification (TPM), Version 1.2, Revision 94. <https://www.trustedcomputinggroup.org/specs/TPM>, 2005.
- [TCG06a] TCG. TCG MPWG Mobile Reference Architecture. Specification Version 1.0, Draft 27, November 2006.
- [TCG06b] TCG. TCG MPWG Mobile Trusted Module Specification, Version 0.9, 2006.

Efficient implementation of rainbow table based cryptanalysis tools on general purpose microprocessors

Andreas Koltes

University of Glasgow
Computer Security and Cryptography
koltes@ieee.org

Type of work: Academic Research Report
Supervisor: Dr. John T. O'Donnell

Abstract: In 2003 Philippe Oechslin described a space- and time-efficient time-memory trade-off for speeding up attacks against symmetric and one-way cryptosystems. This technique consists of an incomplete index structure over the key space which is able to invert cipher functions if ciphertext as well as plaintext are known. It is an attractive technique for attacks on hash-based password authentication systems as well as weak encryption systems that allow guessing a plaintext belonging to a known ciphertext. We propose several algorithms making high performance implementations on cheap and widely available general purpose microprocessors possible. These algorithms and techniques allow for speedups of up to 8 times over existing implementations.

1 Introduction

In [Oec03] Philippe Oechslin proposed a space-efficient time-memory trade-off based on previous work published in [Hel80] and [Den82]. The technique builds an incomplete index structure over the key space of a cryptosystem allowing for the inversion of an attacked cipher function with only a fraction of the computational effort required for an exhaustive search. As the term “incomplete” suggests, it is not guaranteed to find the key corresponding to a given ciphertext because the described technique is a probabilistic method.

Shortly after the publication of [Oec03] tools supporting the method started circulating. The first of these was the RainbowCrack suite published in 2003 by Zhu Shuanglei. Many tools have been implemented based on the original source code, but they give generally poor performance.

2 Hot spot operations

The hot spot operations, which are likely to dominate performance, when stepping through the rainbow chains as described in [Oec03] can be grouped into the following categories:

- Computation of a ciphertext from the fixed plaintext and a variable key
- Reducing a ciphertext resulting in a key space index
- Expanding a key space index to the actual key in the chain

Improving the performance of the cipher operation heavily depends on the attacked cryptosystem. For many block ciphers there are SIMD implementations (e.g. [Bih97] for DES). For hashing algorithms operating on vectors over finite number sets there are efficient SIMD matrix implementations. Because these improvements are highly algorithm-specific we focus on improvements on the last two groups of operations listed.

The second point of the list silently assumes that the ciphertext space is larger than the actual key space for practical applications. The actual key space does not necessarily need to be a contiguous set of natural numbers. On the one hand there may be the need to represent keys of different lengths. On the other hand it might be unnecessary to cover all keys theoretically supported by the cryptosystem forming the key space K (e.g. including only characters which can easily be input using a keyboard). Because of this a typical key space is a set of pairs of lengths and keys representable as $\tilde{K} := (l, k) \subseteq \mathbb{F}_\kappa \times \mathbb{F}_{\sigma^\kappa}$ with κ being the maximum length of a key supported by the cryptosystem. In this context σ is the number of characters in the alphabet $(\Sigma_k)_{k \in \mathbb{F}_\sigma}$. For practical applications it can be assumed that $\exists m \in \mathbb{N} : \forall k \in \mathbb{F}_\sigma : \Sigma_k \in \mathbb{F}_{2^m}$.

3 Efficient implementation strategies

For high efficiency of a rainbow table, the family of surjective functions $f_n : C \rightarrow I$ used to reduce elements of the ciphertext space $C := \mathbb{F}_\gamma$ to elements of the key index space $I := \mathbb{F}_\ell$ should preserve the (usually uniform) probability distribution of the cipher function. Since C and I are finite sets, in most cases these distributions can only be approximated. The following property for f_n guarantees an optimal discrete approximation:

$$\forall i \in I : |\{c | f_n(c) = i\}| \in \left\{ \left\lfloor \frac{|C|}{|I|} \right\rfloor, \left\lceil \frac{|C|}{|I|} \right\rceil \right\}$$

The obvious way to do this is to define $f_n(c) := (c+n) \bmod |I|$. Most currently available tools use these functions. However, on typical architectures this is inefficient, because Integer division operations are among the slowest operations a CPU can perform (see Table 1).

On the other hand most general purpose CPUs feature high speed combinational integer multipliers supporting multiplications of N -bit integers into $2N$ -bit results. In 2005 [Rob05] proposed an adaptive derivation algorithm based on [MPPZ88] and [GM94] for precomputation of factors making the replacement of arbitrary N -bit divisions by N -bit multiplications followed by shifts and fast correction operations possible. Precomputation is necessary only once if they share a common divisor ($|I|$ in our case). Computation of the remainders is possible by another multiply operation followed by a subtraction.

Operation	Data types	Intel NetBurst 64-bit	AMD 64-bit
Addition / Subtraction	$\mathbb{F}_{2^{64}} \times \mathbb{F}_{2^{64}} \rightarrow \mathbb{F}_{2^{64}}$	0.5-1	1
Multiplication	$\mathbb{F}_{2^{32}} \times \mathbb{F}_{2^{32}} \rightarrow \mathbb{F}_{2^{64}}$	10-18	3
	$\mathbb{F}_{2^{64}} \times \mathbb{F}_{2^{64}} \rightarrow \mathbb{F}_{2^{128}}$	10-18	5
Division	$\mathbb{F}_{2^{64}} \times \mathbb{F}_{2^{32}} \rightarrow \mathbb{F}_{2^{32}}$	56-80	39
	$\mathbb{F}_{2^{128}} \times \mathbb{F}_{2^{64}} \rightarrow \mathbb{F}_{2^{64}}$	56-80	71

Table 1: Typical average latencies of integer arithmetic operations on general purpose 64-bit CPU architectures measured in clock cycles assuming pure register operation

For practical applications it can be assumed that $\exists m \in \mathbb{N} : |C| = 2^m$. By using an alternative family of functions $g_n(c) := \lfloor ((c+n) \bmod |C|) \cdot |I|/|C| \rfloor$ the costly modulo operation by $|I|$ is avoided and replaced by much faster multiply and shift operations. Since n and $|C|$ are constant, the remaining modulo operation can be substituted by a fast conditional subtraction once again roughly doubling the speed over the optimised traditional functions.

Transformation of key indexes is done by a surjective function $s : I \rightarrow \tilde{K}$, which basically is a base transformation function. Determining the length of the actual key can be done by constructing a table $(t_n)_{n \in T}$ mapping the covered key lengths to the lowest associated key index, computed as $t_n := \sum_{n > k \in T} \sigma^k$. In this context T is the set of the key lengths covered by the table. The ordering of the keys in the key space is determined by two rules:

- Shorter keys have lower indexes than longer keys
- Keys of equal lengths are sorted in lexicographic order of the character indexes

$(t_n)_{n \in T}$ is parsed starting with index $\max T$ because the number of keys increases exponentially with the key length. The corresponding table entry is subtracted from the key space index and the difference converted into the key by transforming its base to σ and looking up the corresponding characters in (Σ_k) . Since the transformation involves divisions by the constant σ the application of [Rob05] is possible again. Further speedup can be achieved by using lookup tables facilitating lookups of multiple characters at once.

Operation	32-bit	32-bit on AMD64	64-bit
<i>rtgen 1.2</i>			
LANMAN hash computation	1,225,490	1,225,490	N/A
MD5 hash computation	1,677,852	1,572,327	N/A
LANMAN full step	892,857	1,020,408	N/A
MD5 full step	1,116,071	1,157,407	N/A
<i>Reference implementation</i>			
LANMAN hash computation	7,431,767	14,354,837	14,324,197
MD5 hash computation variable	5,227,198	8,375,207	8,355,187
MD5 hash computation fixed	7,405,197	8,723,594	8,745,421
Ciphertext reduction	101,735,831	87,909,613	321,766,319
LANMAN full step	4,189,067	7,255,012	7,946,579
MD5 full step	4,124,699	5,294,584	5,559,972

Table 2: Typical throughputs on common general purpose CPUs measured in operations per second

4 Experimental results

The optimisation approaches proposed were tested in a reference implementation. Table 2 shows typical throughputs for smaller tables ($\sigma = 26, T = \{1..7\}$). For comparability reasons the reduction functions used were compatible with the ones used by the RainbowCrack suite. Therefore the speeds could still be faster using the improved functions proposed. The 32-bit CPU used was an Intel Pentium 4 2.80 GHz, the 64-bit CPU an AMD Athlon 64 3200+ both featuring 512 KB level 2 cache.

5 Conclusion and future work

We proposed several concrete implementations of the fairly abstract algorithms and concepts presented in [Oec03] and showed that it is possible to implement rainbow table operations in a highly efficient way on general purpose CPUs. In combination with fast implementations of hash and encryption algorithms respectively, these improvements reach speeds of up to nearly 8 times faster than generally available implementations of [Oec03].

Further speed improvements could probably be achieved by developing fast transformation functions mapping ciphertexts onto plaintexts without taking the indirection of computing a key space index. Since it might be required to change the mapping of ciphertexts to keys such a solution would probably be incompatible with tools readily available.

References

- [Bih97] Eli Biham. A Fast New DES Implementation in Software. In *Fast Software Encryption: 4th International Workshop, FSE'97, Haifa, Israel, January 1997. Proceedings*, pages 260–272, 1997.
- [Den82] Dorothy E. Denning. *Cryptography and Data Security*, page 100. Addison-Wesley, June 1982.
- [GM94] Torbjörn Granlund and Peter L. Montgomery. Division by invariant integers using multiplication. In *Proceedings of the ACM SIGPLAN 1994 conference on Programming language design and implementation*, pages 61–72, 1994.
- [Hel80] Martin E. Hellman. A Cryptanalytic Time-Memory Trade-Off. *IEEE Transactions on Information Theory*, IT-26:401–406, 1980.
- [MPPZ88] Daniel J. Magenheimer, Liz Peters, Karl Pettis, and Dan Zuras. Integer Multiplication and Division on the HP Precision Architecture. *IEEE Transactions on Computers*, 37(8):980–990, 1988.
- [Oec03] Philippe Oechslin. Making a Faster Cryptanalytic Time-Memory Trade-Off. In *Advances in Cryptology - CRYPTO 2003*, pages 617–630, 2003.
- [Rob05] Arch D. Robison. N-Bit Unsigned Division via N-Bit Multiply-Add. In *Proceedings of the 17th IEEE Symposium on Computer Arithmetic*, pages 131–139, 2005.

Anwendung der "Common Criteria for Information Technology Security Evaluation" (CC) / ISO 15408 auf ein SOA Registry-Repository

M.Sc. Jan Trobitius

Hochschule Darmstadt
Fachbereich Informatik
jan@trobitius.de

Art der Arbeit: Masterthesis

Betreuerin der Arbeit: Prof. Dr. Uta Störl, Prof. Dr. Inge Schestag

Abstract: Das Thema Sicherheit wird zunehmend ein Entscheidungskriterium für Käufer von IT-Systemen. Für die Hersteller dieser Systeme bedeutet dies, dass diesen Aspekten erhöhte Aufmerksamkeit gewidmet werden muss. Um die Sicherheitseigenschaften von IT-Systemen bewerten zu können, gibt es mit den Common Criteria (CC) einen international standardisierten Kriterienkatalog. Dieser schreibt ein formalisiertes Vorgehen bei der Entwicklung eines IT-Systems vor, der durch vielfältige Dokumentationen belegt werden muss. Im Zuge dieser Arbeit sollten für einen grossen Hersteller von seriengefertigten IT-Produkten die Konzepte der CC aufgearbeitet werden und auf eine aktuelle Neuentwicklung - ein Registry-Repository für eine Serviceorientierte Architektur (SOA) - angewendet werden. Dazu wurden exemplarisch die von der CC geforderten Dokumentationen angefertigt. Dabei wurde klar, dass der Aufwand für eine CC-konforme Entwicklung des Systems ein erhebliches Ausmass annimmt. Es wird deutlich, dass sich dieser Aufwand nur dann rentiert, wenn potentielle Kunden explizit eine CC-Konformität anfordern oder wenn gesetzliche Vorgaben für das vorgesehene Einsatzgebiet des Systems diese Konformität obligatorisch machen. Es zeigt sich aber auch, dass das formalisierte Vorgehen der CC dabei helfen kann, Lücken und Unsicherheiten bereits bei der Entwicklung erkennen und vermeiden zu können. Dadurch kann die CC dazu beitragen, die Sicherheit von IT-Systemen zu verbessern.

1 Motivation

Globalisierte und durch das Internet vernetzte Wirtschaftsabläufe, aber auch aktuelle politische Geschehnisse rücken den Sicherheitsaspekt auch in der IT-Welt immer mehr in den Mittelpunkt des Interesses. Um die Sicherheitseigenschaften von Systemen beurteilen zu können, gibt es bereits seit längerem unterschiedliche Kriterien, wie das amerikanische Orange Book oder die europäische ITSEC. Die "Common Criteria For Information Technology Security Evaluation" (CC) sind das Ergebnis internationaler Anstrengungen, die bestehenden Kriterien zu harmonisieren und einen weltweit einheitlichen Standard zur Bewertung von Sicherheitseigenschaften von IT-Systemen zu definieren. Die CC geben einem potentiellen Kunden ein Mittel an die Hand, mit dem er die Sicherheitseigenschaften von seriengefertigten Softwareprodukten mit den eigenen Anforderungen vergleichen kann, anstatt ausschließlich individuelle Lösungen beauftragen zu müssen. Aus diesem Grund ist eine CC-Zertifizierung für den Hersteller von seriengefertigten Softwareprodukten ein nicht zu unterschätzendes Marketing-Argument. Zusätzlich machen gesetzliche Vorgaben sowie behördliche Richtlinien bei der Software-Beschaffung eine CC-Zertifizierung von Produkten, die in einem solchen Umfeld eingesetzt werden sollen, obligatorisch.

Für Hersteller von seriengefertigten Software-Produkten stellt eine CC-Zertifizierung ihrer Produkte eine interessante Möglichkeit dar. Dazu sollte im Rahmen der Masterarbeit ein Einblick in die Konzepte der CC gewonnen und der Aufwand hinter einer CC-Evaluierung untersucht werden.

2 Die Common Criteria

Die durch die Common Criteria definierten Konzepte stellen einen nützlichen Leitfaden für die Entwicklung eines IT-Produktes oder -Systems mit Sicherheitsfunktionalität dar. Die Philosophie hinter der CC bietet Sicherheit basierend auf einer aktiven Untersuchung (Evaluierung) des Produkts oder des Systems, welches geprüft werden soll. Die Gültigkeit und Plausibilität der Dokumentation und des resultierenden Systems wird durch erfahrene Evaluatoren bewertet. Dabei wird die Prüftiefe und Rigorosität durch eine Vertrauenswürdigkeitsstufe vorgegeben, die durch den Hersteller festgelegt wird. Die Prüfung und Bewertung der Dokumentation und des Systems wird durch ein akkreditiertes Prüflabor nach einer im CC-Standard genau vorgegebenen Art und Weise durchgeführt [CC05]. Am Ende einer erfolgreichen Durchführung der Prüfungen wird für das entsprechende Produkt ein Zertifikat erteilt.

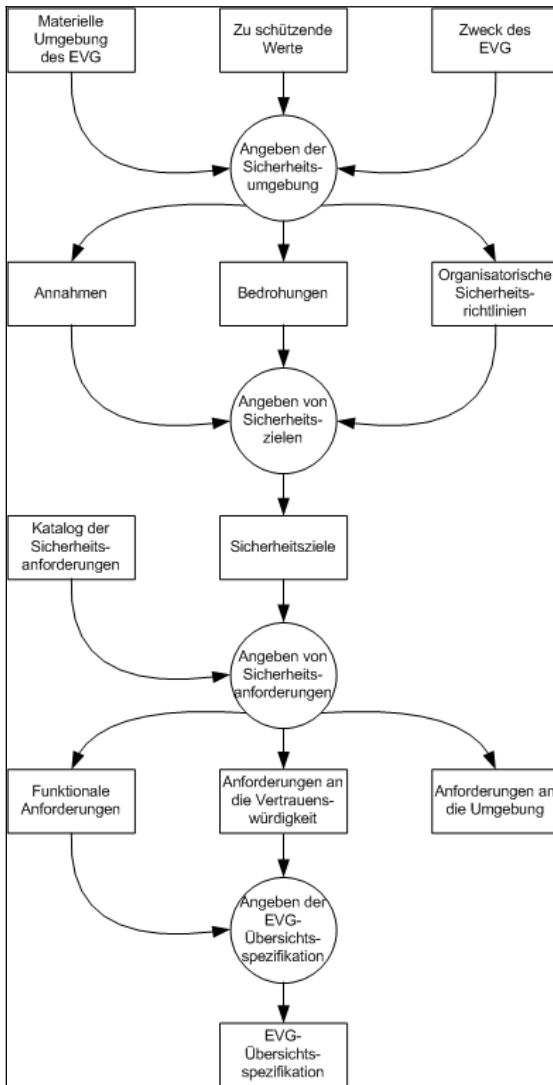


Abbildung 1: Herleitung von Anforderung und Spezifikation [CC99]

In nebenstehender Übersicht wird das durch die CC vorgeschriebene formalisierte Vorgehen bei der Entwicklung eines IT-Systems dargestellt. Grundlage einer Evaluierung sind Dokumente mit vorgegebener Struktur, in denen - ausgehend von der Sicherheitsumgebung und den zu schützenden Werten - nachvollziehbar Sicherheitsziele abgeleitet werden, die wiederum durch aus einem CC-Katalog ausgewählte funktionale Sicherheitsanforderungen erfüllt werden. Die wichtigsten Dokumente sind das implementierungsunabhängige Schutzprofil, sowie die implementierungsspezifische Sicherheitsvorgabe. [AR06], [DH03]

Die CC-Zertifizierung eines Produktes folgt einer erfolgreichen, weitgehend dokumentenbasierten Evaluierung, wobei neben Schutzprofil und Sicherheitsvorgabe weitere, vom Entwickler anzufertigende Dokumentationen des Systems auf Konsistenz, Angemessenheit und Vollständigkeit geprüft werden.

3 Ergebnis und Fazit

Im Rahmen der Masterarbeit wurden die Konzepte der CC auf ein bestehendes System angewendet. Aufbauend auf einer umfangreichen Einarbeitung in die CC-Vorgehensweise und deren Konzepte wurde ein Schutzprofil entwickelt, in dem die Sicherheitseigenschaften eines generischen SOA Registry-Repository betrachtet werden. Außerdem wurde eine Sicherheitsvorgabe erstellt, in der exemplarisch zwei Sicherheitsfunktionen des betrachteten konkreten SOA Registry-Repository detailliert untersucht werden.

Bei dieser Entwicklung zeigte sich, dass die Erstellung und Pflege dieser CC-Dokumente eine sehr zeitaufwändige Entwicklung darstellt. Gerade wenn diese Dokumente nachträglich für ein bereits bestehendes System entwickelt werden, nimmt der Aufwand für diese Anfertigung ein solches Ausmaß an, dass sich eine CC-Zertifizierung nur rentiert, wenn diese eine explizite Kundenanforderung ist oder wenn gesetzliche Vorgaben für das vorgesehene Einsatzgebiet dies voraussetzen.

Auf der anderen Seite wurde bei der Erstellung der erforderlichen Dokumente eine potentielle Sicherheitslücke aufgedeckt. Durch die vorgegebenen Sicherheitsanforderungen im CC-Katalog, die dort jeweils mit Abhängigkeiten vorliegen, konnte ein nicht berücksichtigtes Detail identifiziert werden. Als Ergebnis der Betrachtung konnte also festgehalten werden, dass die Anwendung der CC-Konzepte helfen kann, sicherheitskritische Fehler bereits bei der Entwicklung (oder auch bei nachträglicher Betrachtung) zu entdecken. Für potentielle Kunden bieten die öffentlich verfügbaren CC-Sicherheitsvorgaben die Möglichkeit, seriengefertigte Softwareprodukte im Hinblick auf Sicherheitseigenschaften mit den eigenen Anforderungen zu vergleichen. Damit können unter Umständen teure individuelle Entwicklungen vermieden werden.

Literaturverzeichnis

- [AR06] Ragen, Alex: *Manager's Guide to the Common Criteria, Version 1.10*. <http://www.alexragen.com>, September 2006.
- [BS06] Bundesamt für Sicherheit in der Informationstechnik (BSI): *IT Sicherheit auf Basis der Common Criteria - ein Leitfaden*. http://www.bsi.de/cc/cc_leitf.pdf, zuletzt besucht am 23. September 2006
- [CC99] CC Editorial Board: *Gemeinsame Kriterien für die Prüfung und Bewertung der Sicherheits von Informationstechnik, Version 2.1 : Teil 1: Einführung und allgemeines Modell*. August 1999.
- [CC05] CC Management Board (CCMB): *Common Methodology for Information Technology Security Evaluation - Evaluation Methodology*. Version 2.3, August 2005.
- [DH03] Herrmann, Debra S.: *Using the Common Criteria for IT Security Evaluation*. Auerbach, 2003.
- [IS04] International Standards Organisation: *Information technology - Security techniques - Guide for the production of Protection Profiles and Security Targets*. ISO Technical Report 15446, Juli 2004.

GI-Edition Lecture Notes in Informatics

- P-1 Gregor Engels, Andreas Oberweis, Albert Zündorf (Hrsg.): Modellierung 2001.
- P-2 Mikhail Godlevsky, Heinrich C. Mayr (Hrsg.): Information Systems Technology and its Applications, ISTA'2001.
- P-3 Ana M. Moreno, Reind P. van de Riet (Hrsg.): Applications of Natural Language to Information Systems, NLDB'2001.
- P-4 H. Wörn, J. Mühling, C. Vahl, H.-P. Meinzer (Hrsg.): Rechner- und sensor-gestützte Chirurgie; Workshop des SFB 414.
- P-5 Andy Schürr (Hg.): OMER – Object-Oriented Modeling of Embedded Real-Time Systems.
- P-6 Hans-Jürgen Appelrath, Rolf Beyer, Uwe Marquardt, Heinrich C. Mayr, Claudia Steinberger (Hrsg.): Unternehmen Hochschule, UH'2001.
- P-7 Andy Evans, Robert France, Ana Moreira, Bernhard Rumpe (Hrsg.): Practical UML-Based Rigorous Development Methods – Countering or Integrating the extremists, pUML'2001.
- P-8 Reinhard Keil-Slawik, Johannes Magenheimer (Hrsg.): Informatikunterricht und Medienbildung, INFOS'2001.
- P-9 Jan von Knop, Wilhelm Haverkamp (Hrsg.): Innovative Anwendungen in Kommunikationsnetzen, 15. DFN Arbeitstagung.
- P-10 Mirjam Minor, Steffen Staab (Hrsg.): 1st German Workshop on Experience Management: Sharing Experiences about the Sharing Experience.
- P-11 Michael Weber, Frank Kargl (Hrsg.): Mobile Ad-Hoc Netzwerke, WMAN 2002.
- P-12 Martin Glinz, Günther Müller-Luschnat (Hrsg.): Modellierung 2002.
- P-13 Jan von Knop, Peter Schirmbacher and Viljan Mahni_ (Hrsg.): The Changing Universities – The Role of Technology.
- P-14 Robert Tolksdorf, Rainer Eckstein (Hrsg.): XML-Technologien für das Semantic Web – XSW 2002.
- P-15 Hans-Bernd Bludau, Andreas Koop (Hrsg.): Mobile Computing in Medicine.
- P-16 J. Felix Hampe, Gerhard Schwabe (Hrsg.): Mobile and Collaborative Business 2002.
- P-17 Jan von Knop, Wilhelm Haverkamp (Hrsg.): Zukunft der Netze –Die Verletzbarkeit meistern, 16. DFN Arbeitstagung.
- P-18 Elmar J. Sinz, Markus Plaha (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2002.
- P-19 Sigrid Schubert, Bernd Reusch, Norbert Jesse (Hrsg.): Informatik bewegt – Informatik 2002 – 32. Jahrestagung der Gesellschaft für Informatik e.V. (GI) 30.Sept.-3.Okt. 2002 in Dortmund.
- P-20 Sigrid Schubert, Bernd Reusch, Norbert Jesse (Hrsg.): Informatik bewegt – Informatik 2002 – 32. Jahrestagung der Gesellschaft für Informatik e.V. (GI) 30.Sept.-3.Okt. 2002 in Dortmund (Ergänzungsband).
- P-21 Jörg Desel, Mathias Weske (Hrsg.): Promise 2002: Prozessorientierte Methoden und Werkzeuge für die Entwicklung von Informationssystemen.
- P-22 Sigrid Schubert, Johannes Magenheimer, Peter Hubwieser, Torsten Brinda (Hrsg.): Forschungsbeiträge zur "Didaktik der Informatik" – Theorie, Praxis, Evaluation.
- P-23 Thorsten Spitta, Jens Borchers, Harry M. Sneed (Hrsg.): Software Management 2002 – Fortschritt durch Beständigkeit
- P-24 Rainer Eckstein, Robert Tolksdorf (Hrsg.): XMIDX 2003 – XML-Technologien für Middleware – Middleware für XML-Anwendungen
- P-25 Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Commerce – Anwendungen und Perspektiven – 3. Workshop Mobile Commerce, Universität Augsburg, 04.02.2003
- P-26 Gerhard Weikum, Harald Schöning, Erhard Rahm (Hrsg.): BTW 2003: Datenbanksysteme für Business, Technologie und Web
- P-27 Michael Kroll, Hans-Gerd Lipinski, Kay Melzer (Hrsg.): Mobiles Computing in der Medizin
- P-28 Ulrich Reimer, Andreas Abecker, Steffen Staab, Gerd Stumme (Hrsg.): WM 2003: Professionelles Wissensmanagement – Erfahrungen und Visionen
- P-29 Antje Düsterhöft, Bernhard Thalheim (Eds.): NLDB'2003: Natural Language Processing and Information Systems
- P-30 Mikhail Godlevsky, Stephen Liddle, Heinrich C. Mayr (Eds.): Information Systems Technology and its Applications
- P-31 Arslan Brömme, Christoph Busch (Eds.): BIOSIG 2003: Biometric and Electronic Signatures

- P-32 Peter Hubwieser (Hrsg.): Informatische Fachkonzepte im Unterricht – INFOS 2003
- P-33 Andreas Geyer-Schulz, Alfred Taudes (Hrsg.): Informationswirtschaft: Ein Sektor mit Zukunft
- P-34 Klaus Dittrich, Wolfgang König, Andreas Oberweis, Kai Rannenber, Wolfgang Wahlster (Hrsg.): Informatik 2003 – Innovative Informatikanwendungen (Band 1)
- P-35 Klaus Dittrich, Wolfgang König, Andreas Oberweis, Kai Rannenber, Wolfgang Wahlster (Hrsg.): Informatik 2003 – Innovative Informatikanwendungen (Band 2)
- P-36 Rüdiger Grimm, Hubert B. Keller, Kai Rannenber (Hrsg.): Informatik 2003 – Mit Sicherheit Informatik
- P-37 Arndt Bode, Jörg Desel, Sabine Rathmayer, Martin Wessner (Hrsg.): DeLFI 2003: e-Learning Fachtagung Informatik
- P-38 E.J. Sinz, M. Plaha, P. Neckel (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2003
- P-39 Jens Nedon, Sandra Frings, Oliver Göbel (Hrsg.): IT-Incident Management & IT-Forensics – IMF 2003
- P-40 Michael Rebstock (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2004
- P-41 Uwe Brinkschulte, Jürgen Becker, Dietmar Fey, Karl-Erwin Großpietsch, Christian Hochberger, Erik Maehle, Thomas Runkler (Edts.): ARCS 2004 – Organic and Pervasive Computing
- P-42 Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Economy – Transaktionen und Prozesse, Anwendungen und Dienste
- P-43 Birgitta König-Ries, Michael Klein, Philipp Obreiter (Hrsg.): Persistence, Scalability, Transactions – Database Mechanisms for Mobile Applications
- P-44 Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): Security, E-Learning, E-Services
- P-45 Bernhard Rumpe, Wolfgang Hesse (Hrsg.): Modellierung 2004
- P-46 Ulrich Flegel, Michael Meier (Hrsg.): Detection of Intrusions of Malware & Vulnerability Assessment
- P-47 Alexander Prosser, Robert Krimmer (Hrsg.): Electronic Voting in Europe – Technology, Law, Politics and Society
- P-48 Anatoly Doroshenko, Terry Halpin, Stephen W. Liddle, Heinrich C. Mayr (Hrsg.): Information Systems Technology and its Applications
- P-49 G. Schiefer, P. Wagner, M. Morgenstern, U. Rickert (Hrsg.): Integration und Datensicherheit – Anforderungen, Konflikte und Perspektiven
- P-50 Peter Dadam, Manfred Reichert (Hrsg.): INFORMATIK 2004 – Informatik verbindet (Band 1) Beiträge der 34. Jahrestagung der Gesellschaft für Informatik e.V. (GI), 20.-24. September 2004 in Ulm
- P-51 Peter Dadam, Manfred Reichert (Hrsg.): INFORMATIK 2004 – Informatik verbindet (Band 2) Beiträge der 34. Jahrestagung der Gesellschaft für Informatik e.V. (GI), 20.-24. September 2004 in Ulm
- P-52 Gregor Engels, Silke Seehusen (Hrsg.): DELFI 2004 – Tagungsband der 2. e-Learning Fachtagung Informatik
- P-53 Robert Giegerich, Jens Stoye (Hrsg.): German Conference on Bioinformatics – GCB 2004
- P-54 Jens Borchers, Ralf Kneuper (Hrsg.): Softwaremanagement 2004 – Outsourcing und Integration
- P-55 Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): E-Science und Grid Ad-hoc-Netze Medienintegration
- P-56 Fernand Feltz, Andreas Oberweis, Benoit Otjacques (Hrsg.): EMISA 2004 – Informationssysteme im E-Business und E-Government
- P-57 Klaus Turowski (Hrsg.): Architekturen, Komponenten, Anwendungen
- P-58 Sami Beydeda, Volker Gruhn, Johannes Mayer, Ralf Reussner, Franz Schweiggert (Hrsg.): Testing of Component-Based Systems and Software Quality
- P-59 J. Felix Hampe, Franz Lehner, Key Pousttchi, Kai Rannenber, Klaus Turowski (Hrsg.): Mobile Business – Processes, Platforms, Payments
- P-60 Steffen Friedrich (Hrsg.): Unterrichtskonzepte für informatische Bildung
- P-61 Paul Müller, Reinhard Gotzhein, Jens B. Schmitt (Hrsg.): Kommunikation in verteilten Systemen
- P-62 Federrath, Hannes (Hrsg.): „Sicherheit 2005“ – Sicherheit – Schutz und Zuverlässigkeit
- P-63 Roland Kaschek, Heinrich C. Mayr, Stephen Liddle (Hrsg.): Information Systems – Technology and its Applications

- P-64 Peter Liggesmeyer, Klaus Pohl, Michael Goedicke (Hrsg.): Software Engineering 2005
- P-65 Gottfried Vossen, Frank Leymann, Peter Lockemann, Wolffried Stucky (Hrsg.): Datenbanksysteme in Business, Technologie und Web
- P-66 Jörg M. Haake, Ulrike Lucke, Djamshid Tavangarian (Hrsg.): DeLFI 2005: 3. deutsche e-Learning Fachtagung Informatik
- P-67 Armin B. Cremers, Rainer Manthey, Peter Martini, Volker Steinhage (Hrsg.): INFORMATIK 2005 – Informatik LIVE (Band 1)
- P-68 Armin B. Cremers, Rainer Manthey, Peter Martini, Volker Steinhage (Hrsg.): INFORMATIK 2005 – Informatik LIVE (Band 2)
- P-69 Robert Hirschfeld, Ryszard Kowalczyk, Andreas Polze, Matthias Weske (Hrsg.): NODe 2005, GSEM 2005
- P-70 Klaus Turowski, Johannes-Maria Zaha (Hrsg.): Component-oriented Enterprise Application (COAE 2005)
- P-71 Andrew Torda, Stefan Kurz, Matthias Rarey (Hrsg.): German Conference on Bioinformatics 2005
- P-72 Klaus P. Jantke, Klaus-Peter Fährnrich, Wolfgang S. Wittig (Hrsg.): Marktplatz Internet: Von e-Learning bis e-Payment
- P-73 Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): "Heute schon das Morgen sehen"
- P-74 Christopher Wolf, Stefan Lucks, Po-Wah Yau (Hrsg.): WEWoRC 2005 – Western European Workshop on Research in Cryptology
- P-75 Jörg Desel, Ulrich Frank (Hrsg.): Enterprise Modelling and Information Systems Architecture
- P-76 Thomas Kirste, Birgitta König-Riess, Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Informationssysteme – Potentiale, Hindernisse, Einsatz
- P-77 Jana Dittmann (Hrsg.): SICHERHEIT 2006
- P-78 K.-O. Wenkel, P. Wagner, M. Morgens-tern, K. Luzi, P. Eisermann (Hrsg.): Land- und Ernährungswirtschaft im Wandel
- P-79 Bettina Biel, Matthias Book, Volker Gruhn (Hrsg.): Softwareengineering 2006
- P-80 Mareike Schoop, Christian Huemer, Michael Rebstock, Martin Bichler (Hrsg.): Service-Oriented Electronic Commerce
- P-81 Wolfgang Karl, Jürgen Becker, Karl-Erwin Großpietsch, Christian Hochberger, Erik Maehle (Hrsg.): ARCS '06
- P-82 Heinrich C. Mayr, Ruth Breu (Hrsg.): Modellierung 2006
- P-83 Daniel Huson, Oliver Kohlbacher, Andrei Lupas, Kay Nieselt and Andreas Zell (eds.): German Conference on Bioinformatics
- P-84 Dimitris Karagiannis, Heinrich C. Mayr, (Hrsg.): Information Systems Technology and its Applications
- P-85 Witold Abramowicz, Heinrich C. Mayr, (Hrsg.): Business Information Systems
- P-86 Robert Krimmer (Ed.): Electronic Voting 2006
- P-87 Max Mühlhäuser, Guido Röbling, Ralf Steinmetz (Hrsg.): DELFI 2006: 4. e-Learning Fachtagung Informatik
- P-88 Robert Hirschfeld, Andreas Polze, Ryszard Kowalczyk (Hrsg.): NODe 2006, GSEM 2006
- P-90 Joachim Schelp, Robert Winter, Ulrich Frank, Bodo Rieger, Klaus Turowski (Hrsg.): Integration, Informationslogistik und Architektur
- P-91 Henrik Stormer, Andreas Meier, Michael Schumacher (Eds.): European Conference on eHealth 2006
- P-92 Fernand Feltz, Benoît Otjacques, Andreas Oberweis, Nicolas Poussing (Eds.): AIM 2006
- P-93 Christian Hochberger, Rüdiger Liskowsky (Eds.): INFORMATIK 2006 – Informatik für Menschen, Band 1
- P-94 Christian Hochberger, Rüdiger Liskowsky (Eds.): INFORMATIK 2006 – Informatik für Menschen, Band 2
- P-95 Matthias Weske, Markus Nüttgens (Eds.): EMISA 2005: Methoden, Konzepte und Technologien für die Entwicklung von dienstbasierten Informationssystemen
- P-96 Saartje Brockmans, Jürgen Jung, York Sure (Eds.): Meta-Modelling and Ontologies
- P-97 Oliver Göbel, Dirk Schadt, Sandra Frings, Hardo Hase, Detlef Günther, Jens Nedon (Eds.): IT-Incident Mangament & IT-Forensics – IMF 2006

- P-98 Hans Brandt-Pook, Werner Simonsmeier und Thorsten Spitta (Hrsg.): Beratung in der Softwareentwicklung – Modelle, Methoden, Best Practices
- P-99 Andreas Schwill, Carsten Schulte, Marco Thomas (Hrsg.): Didaktik der Informatik
- P-100 Peter Forbrig, Günter Siegel, Markus Schneider (Hrsg.): HDI 2006: Hochschuldidaktik der Informatik
- P-101 Stefan Böttinger, Ludwig Theuvsen, Susanne Rank, Marlies Morgenstern (Hrsg.): Agrarinformatik im Spannungsfeld zwischen Regionalisierung und globalen Wertschöpfungsketten
- P-102 Otto Spaniol (Eds.): Mobile Services and Personalized Environments
- P-103 Alfons Kemper, Harald Schöning, Thomas Rose, Matthias Jarke, Thomas Seidl, Christoph Quix, Christoph Brochhaus (Hrsg.): Datenbanksysteme in Business, Technologie und Web (BTW 2007)
- P-104 Birgitta König-Ries, Franz Lehner, Rainer Malaka, Can Türker (Hrsg.) MMS 2007: Mobilität und mobile Informationssysteme
- P-105 Wolf-Gideon Bleek, Jörg Raasch, Heinz Züllighoven (Hrsg.) Software Engineering 2007
- P-106 Wolf-Gideon Bleek, Henning Schwentner, Heinz Züllighoven (Hrsg.) Software Engineering 2007 – Beiträge zu den Workshops

The titles can be purchased at:

Köllen Druck + Verlag GmbH

Ernst-Robert-Curtius-Str. 14 · D-53117 Bonn

Fax: +49 (0)228/9898222

E-Mail: druckverlag@koellen.de