

Commenced Publication in 1973

Founding and Former Series Editors:

Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

David Hutchison

Lancaster University, UK

Takeo Kanade

Carnegie Mellon University, Pittsburgh, PA, USA

Josef Kittler

University of Surrey, Guildford, UK

Jon M. Kleinberg

Cornell University, Ithaca, NY, USA

Alfred Kobsa

University of California, Irvine, CA, USA

Friedemann Mattern

ETH Zurich, Switzerland

John C. Mitchell

Stanford University, CA, USA

Moni Naor

Weizmann Institute of Science, Rehovot, Israel

Oscar Nierstrasz

University of Bern, Switzerland

C. Pandu Rangan

Indian Institute of Technology, Madras, India

Bernhard Steffen

University of Dortmund, Germany

Madhu Sudan

Massachusetts Institute of Technology, MA, USA

Demetri Terzopoulos

University of California, Los Angeles, CA, USA

Doug Tygar

University of California, Berkeley, CA, USA

Gerhard Weikum

Max-Planck Institute of Computer Science, Saarbruecken, Germany

Robert Meersman Zahir Tari (Eds.)

On the Move to Meaningful Internet Systems: OTM 2008

OTM 2008 Confederated International Conferences
CoopIS, DOA, GADA, IS, and ODBASE 2008
Monterrey, Mexico, November 9-14, 2008
Proceedings, Part II

Volume Editors

Robert Meersman
Vrije Universiteit Brussel (VUB), STARLab
Bldg G/10, Pleinlaan 2, 1050, Brussels, Belgium
E-mail: meersman@vub.ac.be

Zahir Tari
RMIT University, School of Computer Science and Information Technology
Bld 10.10, 376-392 Swanston Street, VIC 3001, Melbourne, Australia
E-mail: zahir.tari@rmit.edu.au

Library of Congress Control Number: 2008938070

CR Subject Classification (1998): H.2, H.3, H.4, C.2, H.5, D.2.12, I.2, K.4

LNCS Sublibrary: SL 3 – Information Systems and Application, incl. Internet/Web and HCI

ISSN	0302-9743
ISBN-10	3-540-88872-1 Springer Berlin Heidelberg New York
ISBN-13	978-3-540-88872-7 Springer Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

Springer is a part of Springer Science+Business Media
springer.com

© Springer-Verlag Berlin Heidelberg 2008
Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12558315 06/3180 5 4 3 2 1 0

Volume Editors

Robert Meersman
Zahir Tari

CoopIS

Johann Eder
Masaru Kitsuregawa
Ling Liu

DOA

Mark Little
Alberto Montresor
Greg Pavlik

ODBASE

Malu Castellanos
Fausto Giunchiglia
Feng Ling

GADA

Dennis Gannon
Pilar Herrero
Daniel S. Katz
María S. Pérez

IS

Jong Hyuk Park
Bart Preneel
Ravi Sandhu
André Zúquete

OTM 2008 General Co-chairs' Message

Dear OnTheMove Participant, or Reader of these Proceedings,

The OnTheMove 2008 event in Monterrey, Mexico, 9–14 November, further consolidated the growth of the conference series that was started in Irvine, California in 2002, and held in Catania, Sicily in 2003, in Cyprus in 2004 and 2005, in Montpellier in 2006, and in Vilamoura in 2007. The event continues to attract a diversifying and representative selection of today's worldwide research on the scientific concepts underlying new computing paradigms, which, of necessity, must be distributed, heterogeneous and autonomous yet meaningfully collaborative.

Indeed, as such large, complex and networked intelligent information systems become the focus and norm for computing, there continues to be an acute and increasing need to address and discuss in an integrated forum the implied software, system and enterprise issues as well as methodological, semantical, theoretical and applicational issues. As we all know, email, the Internet, and even video conferences are not sufficient for effective and efficient scientific exchange. The OnTheMove (OTM) Federated Conferences series has been created to cover the scientific exchange needs of the community/ies that work in the broad yet closely connected fundamental technological spectrum of data and web semantics, distributed objects, web services, databases, information systems, enterprise workflow and collaboration, ubiquity, interoperability, mobility, grid and high-performance computing.

OnTheMove aspires to be a primary scientific meeting place where all aspects for the development of such Internet- and Intranet-based systems in organizations and for e-business are discussed in a scientifically motivated way. This sixth edition of the OTM Federated Conferences event again provided an opportunity for researchers and practitioners to understand and publish these developments within their individual as well as within their broader contexts.

Originally the federative structure of OTM was formed by the co-location of three related, complementary and successful main conference series: DOA (Distributed Objects and Applications, since 1999), covering the relevant infrastructure-enabling technologies, ODBASE (Ontologies, DataBases and Applications of SEmanantics, since 2002) covering Web semantics, XML databases and ontologies, and CoopIS (Cooperative Information Systems, since 1993) covering the application of these technologies in an enterprise context through e.g., workflow systems and knowledge management. In 2006 a fourth conference, GADA (Grid computing, high-performAnce and Distributed Applications) was added to this as a main symposium, and last year the same happened with IS (Information Security). Both of these started as successful workshops at OTM, the first covering the large-scale integration of heterogeneous computing systems and data resources with the aim of providing a global computing space,

the second covering the issues of security in complex Internet-based information systems.

Each of these five conferences encourages researchers to treat their respective topics within a framework that incorporates jointly (a) theory, (b) conceptual design and development, and (c) applications, in particular case studies and industrial solutions.

Following and expanding the model created in 2003, we again solicited and selected quality workshop proposals to complement the more “archival” nature of the main conferences with research results in a number of selected and more “avant-garde” areas related to the general topic of distributed computing. For instance, the so-called Semantic Web has given rise to several novel research areas combining linguistics, information systems technology, and artificial intelligence, such as the modeling of (legal) regulatory systems and the ubiquitous nature of their usage. We were glad to see that in spite of OnTheMove switching sides of the Atlantic, seven of our earlier successful workshops (notably AweSOMe, SWWS, ORM, OnToContent, MONET, PerSys, RDDS) re-appeared in 2008 with a third or even fourth edition, sometimes by alliance with other newly emerging workshops, and that no fewer than seven brand-new independent workshops could be selected from proposals and hosted: ADI, COMBEK, DiSCo, IWSSA, QSI and SEMELS. Workshop audiences productively mingled with each other and with those of the main conferences, and there was considerable overlap in authors. The OTM organizers are especially grateful for the leadership, diplomacy and competence of Dr. Pilar Herrero in managing this complex and delicate process for the fifth consecutive year.

Unfortunately however in 2008 the number of quality submissions for the OnTheMove Academy (formerly called Doctoral Consortium Workshop), our “vision for the future” in research in the areas covered by OTM, proved too low to justify a 2008 edition in the eyes of the organizing faculty. We must however thank Antonia Albani, Sonja Zaplata and Johannes Maria Zaha, three young and active researchers, for their efforts in implementing our interactive formula to bring PhD students together: research proposals are submitted for evaluation; selected submissions and their approaches are (eventually) to be presented by the students in front of a wider audience at the conference, and intended to be independently and extensively analyzed and discussed in public by a panel of senior professors. Prof. Em. Jan Dietz, the Dean of the OnTheMove Academy, also is stepping down this year, but OnTheMove is committed to continuing this formula with a new Dean and peripatetic faculty.

All five main conferences and the associated workshops shared the distributed aspects of modern computing systems, and the resulting application-pull created by the Internet and the so-called Semantic Web. For DOA 2008, the primary emphasis stayed on the distributed object infrastructure; for ODBASE 2008, it has become the knowledge bases and methods required for enabling the use of formal semantics; for CoopIS 2008, the focus as usual was on the interaction of such technologies and methods with management issues, such as occur in networked organizations, for GADA 2008, the main topic was again the scalable integration

of heterogeneous computing systems and data resources with the aim of providing a global computing space, and in IS 2008 the emphasis was on information security in the networked society. These subject areas overlapped in a scientifically natural fashion and many submissions in fact also treated an envisaged mutual impact among them. As for the earlier editions, the organizers wanted to stimulate this cross-pollination by a **shared** program of famous keynote speakers: this year we were proud to announce Dan Atkins of the U.S. National Science Foundation and the University of Michigan, Hector Garcia-Molina of Stanford, Rick Hull of IBM T.J. Watson Lab, Ted Goranson of Sirius-Beta and of Paradigm Shift, and last but not least Cristina Martinez-Gonzalez of the European Commission with a special interest in more future scientific collaboration between the EU and Latin America, as well as emphasizing the concrete outreach potential of new Internet technologies for enterprises anywhere.

This year the registration fee structure strongly encouraged multiple event attendance by providing **all** main conference authors with free access or discounts to **all** other conferences or workshops (workshop authors paid a small extra fee to attend the main conferences). In both cases the price for these combo tickets was made lower than in 2007 in spite of the higher organization costs and risks!

We received a total of 292 submissions for the five main conferences and 171 submissions in total for the 14 workshops. The numbers are about 30% lower than for 2007, which was not unexpected because of the transatlantic move of course, and the emergent need to establish the OnTheMove brand in the Americas, a process that will continue as we proceed in the coming years. But, not only may we indeed again claim success in attracting an increasingly representative volume of scientific papers, many from US, Central and South America already, but these numbers of course allow the program committees to compose a high-quality cross-section of current research in the areas covered by OTM. In fact, in spite of the larger number of submissions, the Program Chairs of each of the three main conferences decided to accept only approximately the same number of papers for presentation and publication as in 2006 and 2007 (i.e., average 1 paper out of 3-4 submitted, not counting posters). For the workshops, the acceptance rate varies but the aim was to stay as strict as before, consistently about 1 accepted paper for 2-3 submitted. We have separated the proceedings into three books with their own titles, two for the main conferences and one for the workshops, and we are grateful to Springer for their suggestions and collaboration in producing these books and CDROMs. The reviewing process by the respective program committees was again performed very professionally, and each paper in the main conferences was reviewed by at least three referees, with arbitrated email discussions in the case of strongly diverging evaluations. It may be worthwhile emphasizing that it is an explicit OnTheMove policy that all conference program committees and chairs make their selections completely autonomously from the OTM organization itself. The OnTheMove Federated Event organizers again made all proceedings available on a CDROM to all

participants of conferences resp. workshops, independently of their registration to a specific conference resp. workshop. Paper proceedings were on request this year, and incurred an extra charge.

The General Chairs were once more especially grateful to the many people directly or indirectly involved in the setup of these federated conferences. Few people realize what a large number of people have to be involved, and what a huge amount of work, and in 2008 certainly also financial risk, the organization of an event like OTM entails. Apart from the persons in their roles mentioned above, we therefore in particular wish to thank our 17 main conference PC co-chairs:

GADA 2008	Dennis Gannon, Pilar Herrero, Daniel Katz, María S. Pérez
DOA 2008	Mark Little, Alberto Montresor, Greg Pavlik
ODBASE 2008	Malu Castellanos, Fausto Giunchiglia, Feng Ling
CoopIS 2008	Johann Eder, Masaru Kitsuregawa, Ling Liu
IS 2008	Jong Hyuk Park, Bart Preneel, Ravi Sandhu, André Zúquete
50 Workshop PC Co-chairs	Stefan Jablonski, Olivier Curé, Christoph Bussler, Jörg Denzinger, Pilar Herrero, Gonzalo Méndez, Rainer Unland, Pieter De Leenheer, Martin Hepp, Amit Sheth, Stefan Decker, Ling Liu, James Caverlee, Ying Ding, Yihong Ding, Arturo Molina, Andrew Kusiak, Hervé Panetto, Peter Bernus, Lawrence Chung, José Luis Garrido, Nary Subramanian, Manuel Noguera, Fernando Ferri, Irina Kondratova, Arianna D'ulizia, Patrizia Grifoni, Andreas Schmidt, Mustafa Jarrar, Terry Halpin, Sijr Nijssen, Skevos Evripidou, Roy Campbell, Anja Schanzenberger, Ramon F. Brena, Hector Ceballos, Yolanda Castillo, Achour Mostefaoui, Eiko Yoneki, Elena Simperl, Reto Krummenacher, Lyndon Nixon, Emanuele Della Valle, Ronaldo Menezes, Tharam S. Dillon, Ernesto Damiani, Elizabeth Chang, Paolo Ceravolo, Amandeep S. Sidhu

All, together with their many PC members, did a superb and professional job in selecting the best papers from the large harvest of submissions.

We must all be grateful to Ana Cecilia Martinez-Barbosa for researching and securing the local and sponsoring arrangements on-site, to Josefa Kumpfmüller for many useful scientific insights in the dynamics of our transatlantic move, and to our extremely competent and experienced Conference Secretariat and technical support staff in Antwerp, Daniel Meersman, Ana-Cecilia (again), and Jan Demey, and last but not least to our apparently never-sleeping Melbourne Program Committee Support Team, Vidura Gamini Abhaya and Anshuman Mukherjee.

The General Chairs gratefully acknowledge the academic freedom, logistic support and facilities they enjoy from their respective institutions, Vrije Universiteit Brussel (VUB) and RMIT University, Melbourne, without which such an enterprise would not be feasible.

We do hope that the results of this federated scientific enterprise contribute to your research and your place in the scientific network... We look forward to seeing you again at next year's event!

August 2008

Robert Meersman
Zahir Tari

Organization Committee

OTM (On The Move) is a federated event involving a series of major international conferences and workshops. These proceedings contain the papers presented at the OTM 2008 Federated Conferences, consisting of five conferences, namely CoopIS (Cooperative Information Systems), DOA (Distributed Objects and Applications), GADA (Grid computing, high-performance and Distributed Applications), IS (Information Security) and ODBASE (Ontologies, Databases and Applications of Semantics).

Executive Committee

OTM 2008 General Co-chairs	Robert Meersman (Vrije Universiteit Brussel, Belgium) and Zahir Tari (RMIT University, Australia)
GADA 2008 PC Co-chairs	Pilar Herrero (Universidad Politécnica de Madrid, Spain), Daniel Katz (Louisiana State University, USA), María S. Pérez (Universidad Politécnica de Madrid, Spain), and Dennis Gannon (Indiana University, USA)
CoopIS 2008 PC Co-chairs	Johann Eder (University of Klagenfurt, Austria), Masaru Kitsuregawa (University of Tokyo, Japan), and Ling Liu (Georgia Institute of Technology, USA)
DOA 2008 PC Co-chairs	Mark Little (Red Hat, UK), Alberto Montresor (University of Trento, Italy), and Greg Pavlik (Oracle, USA)
IS 2008 PC Co-chairs	Jong Hyuk Park (Kyungnam University, Korea), Bart Preneel (Katholieke Universiteit Leuven, Belgium), Ravi Sandhu (University of Texas at San Antonio, USA), and André Zúquete (University of Aveiro, Portugal)
ODBASE 2008 PC Co-chairs	Malu Castellanos (HP, USA), Fausto Giunchiglia (University of Trento, Italy), and Feng Ling (Tsinghua University, China)
Publication Co-chairs	Vidura Gamini Abhaya (RMIT University, Australia) and Anshuman Mukherjee (RMIT University, Australia)
Local Organizing Chair	Lorena G. Gómez Martínez (Tecnológico de Monterrey, Mexico)
Conferences Publicity Chair	Keke Chen (Yahoo!, USA)

Workshops Publicity Chair	Gonzalo Mendez (Universidad Complutense de Madrid, Spain)
Secretariat	Ana-Cecilia Martinez Barbosa, Jan Demey, and Daniel Meersman

CoopIS 2008 Program Committee

Ghaleb Abdulla	Rania Khalaf
Marco Aiello	Hiroyuki Kitagawa
Joonsoo Bae	Akhil Kumar
Alistair Barros	Shim Kyusock
Zohra Bellahsene	Wang-Chien Lee
Boualem Benatallah	Frank Leymann
Salima Benbernou	Chen Li
Djamal Benslimane	Sanjay K. Madria
M. Brian Blake	Tiziana Margaria
Laura Bright	Leo Mark
Christoph Bussler	Maristella Matera
David Buttler	Massimo Mecella
Klemens Böhm	Ingo Melzer
Ying Cai	Mohamed Mokbel
James Caverlee	Nirmal Mukhi
Keke Chen	Jörg Müller
Francisco Curbera	Miyuki Nakano
Vincenzo D'Andrea	Wolfgang Nejdl
Umesh Dayal	Moirra Norrie
Xiaoyong Du	Werner Nutt
Marlon Dumas	Andreas Oberweis
Schahram Dustdar	Tadashi Ohmori
Rik Eshuis	Cesare Pautasso
Opher Etzion	Barbara Pernici
Renato Fileto	Beth Plale
Klaus Fischer	Frank Puhlmann
Avigdor Gal	Lakshmish Ramaswamy
Bugra Gedik	Manfred Reichert
Dimitrios Georgakopoulos	Stefanie Rinderle-Ma
Paul Grefen	Rainer Ruggaber
Amarnath Gupta	Duncan Ruiz
Mohand-Said Hacid	Kai-Uwe Sattler
Thorsten Hampel	Ralf Schenkel
Geert-Jan Houben	Jialie Shen
Richard Hull	Aameek Singh
Patrick Hung	Mudhakar Srivatsa
Paul Johannesson	Jianwen Su
Dimka Karastoyanova	Wei Tang

Anthony Tung
 Susan Urban
 Willem-Jan Van den Heuvel
 Wil Van der Aalst
 Maria Esther Vidal
 Shan Wang
 X. Sean Wang
 Mathias Weske

Li Xiong
 Jian Yang
 Masatoshi Yoshikawa
 Jeffrey Yu
 Leon Zhao
 Aoying Zhou
 Xiaofang Zhou
 Michael zur Muehlen

DOA 2008 Program Committee

Mark Baker
 Judith Bishop
 Gordon Blair
 Barret Bryant
 Harold Carr
 Gregory Chockler
 Geoff Coulson
 Frank Eliassen
 Patrick Eugster
 Pascal Felber
 Benoit Garbinato
 Jeff Gray
 Medhi Jazayeri
 Eric Jul
 Nick Kavantzaz
 Fabio Kon
 Joe Loyall
 Frank Manola
 Nikola Milanovic
 Graham Morgan
 Gero Mühl

Rui Oliveira
 Jose Orlando Pereira
 François Pacull
 Fernando Pedone
 Gian Pietro Picco
 Calton Pu
 Arno Puder
 Michel Riveill
 Luis Rodrigues
 Isabelle Rouvellou
 Aniruddha S. Gokhale
 Santosh Shrivastava
 Richard Soley
 Michael Stal
 Jean-Bernard Stefani
 Hong Va Leong
 Aad van Moorsel
 Andrew Watson
 Stuart Wheeler
 Shalini Yajnik

GADA 2008 Program Committee

Juan A. Botía Blaya
 Jemal Abawajy
 Akshai Aggarwal
 Artur Andrzejak
 Oscar Ardaiz
 Sattar B. Sadkhan Almaliky
 Costin Badica
 Mark Baker
 Pascal Bouvry

Rajkumar Buyya
 Santi Caballé Llobet
 Blanca Caminero Herraiz
 Mario Cannataro
 Jess Carretero
 Jinjun Chen
 Carmela Comito
 Toni Cortes
 Geoff Coulson

Jose Cunha
Alfredo Cuzzocrea
Ewa Deelman
Beniamino Di Martino
Marios Dikaiakos
Markus Endler
Geoffrey Fox
Maria Ganzha
Felix García
Antonio Garcia Dopico
Anastasios Gounaris
Eduardo Huedo
Félix J. García Clemente
Shantenu Jha
Liviu Joita
Francisco José da Silva e Silva
Kostas Karasavvas
Kamil Kuliberda
Jose L. Bosque
Laurent Lefevre
Ángel Lucas González Martínez
José Luis Vázquez Poletti
Francisco Luna
Rosa M. Badia
Ignacio M. Llorente

Jose M. Pea
Edgar Magana
Gregorio Martinez
Reagan Moore
Mirela Notare
Hong Ong
Neil P Chue Hong
Marcin Paprzycki
Manish Parashar
Dana Petcu
Bhanu Prasad
V́ctor Robles
Ruben S. Montero
Rizos Sakellariou
Manuel Salvadores
Alberto Sanchez
Hamid Sarbazi-Azad
Heinz Stockinger
Alan Sussman
Elghazali Talbi
Jordi Torres
Cho-Li Wang
Adam Wierzbicki
Fatos Xhafa

IS 2008 Program Committee

J.H. Abbawajy
Gail Ahn
Vijay Atluri
Manuel Bernardo Barbosa
Ezedin Barka
Elisa Bertino
Bruno Crispo
Gwenal Dorr
Huirong Fu
Clemente Galdi
Luis Javier Garcia Villalba
Helena Handschuh
Sheikh Iqbal Ahamed
James Joshi
Stefan Katzenbeisser
Hiroaki Kikuchi

Byoung-soo Koh
Klaus Kursawe
Kwok-Yan Lam
Deok Gyu Lee
Javier Lopez
Evangelos Markatos
Sjouke Mauw
Chris Mitchell
Yi Mu
Nuno Ferreira Neves
Giuseppe Persiano
Milan Petkovic
Frank Piessens
Bhanu Prasad
Carlos Ribeiro
Pierangela Samarati

Biplab K. Sarker
 Diomidis D. Spinellis
 Avinash Srinivasan
 Umberto Villano
 Liudong Xing

Shouhuai Xu
 Sang-Soo Yeo
 Xinwen Zhang
 Deqing Zou

ODBASE 2008 Program Committee

Harith Alani
 Jon Atle Gulla
 María Auxilio Medina
 Franz Baader
 Renato Barrera
 Sonia Bergamaschi
 Leopoldo Bertossi
 Mohand Boughanem
 Francisco Cantu-Ortiz
 Edgar Chavez
 Oscar Corcho
 Umesh Dayal
 Benjamin Habegger
 Bin He
 Andreas Hotho
 Jingshan Huang
 Farookh Hussain
 Vipul Kashyap
 Uladzimir Kharkevich
 Phokion Kolaitis
 Manolis Koubarakis
 Maurizio Lenzerini
 Juanzi Li
 Alexander Löser
 Lois M. L. Delcambre
 Li Ma
 Enzo Maltese
 Maurizio Marchese

Riichiro Mizoguchi
 Peter Mork
 Wolfgang Nejdl
 Erich Neuhold
 Matthew Perry
 Wenny Rahayu
 Rajugan Rajagopalapillai
 Sudha Ram
 Arnon Rosenthal
 Satya Sahoo
 Pavel Shvaiko
 Il-Yeol Song
 Stefano Spaccapietra
 Veda C. Storey
 Umberto Straccia
 Eleni Stroulia
 Heiner Stuckenschmidt
 Vijayan Sugumaran
 York Sure
 Octavian Udrea
 Michael Uschold
 Yannis Velegrakis
 Guido Vetere
 Kevin Wilkinson
 Baoshi Yan
 Laura Zavala
 Jose Luis Zechinelli
 Yanchun Zhang

OTM Conferences 2008 Additional Reviewers

Aditya Bagchi
 Adrian Holzer
 Agnes Koschmider
 Ahlem Bouchahda
 Akshai Aggarwal

Alexander Behm
 Alexander Hornung
 Alexandros Kapravelos
 Alfranio Correia Jr.
 Aliaksandr Birukou

XVIII Organization

Alvin Chan
Anh Tuan Ngyuen
Anirban Mondal
Antonio Garcia Dopico
Aries Tao
Arnd Schroeter
Bing-rong Lin
Boris Motik
Carlos Baquero
Chen Wang
Christian Meilicke
Christoph Gerdes
Cinzia Cappiello
Daniel Eichhorn
Daniel Ried
Darrell Woelk
Deok Gyu Lee
Di Wu
Dimitris Mitropoulos
Dominic Mueller
Don Baker
Eirini Kaldeli
Ela Hunt
Eli Gjorven
Elie el Khoury
Evandro Bacarin
Evgeny Kharlamov
Evi Syukur
Ezedin Barka Barka
Ezedin Barka
Fabien Duchateau
Farouk Toumani
Feng Cao
Fernando Castor Filho
Francesco Guerra
Francesco Lelli
Francisco Cantu
Frithjof Dau
G.R. Gangadharan
Gabriela Montiel-Moreno
Gail Mitchell
Genoveva Vargas-Solar
George Vasiliadis
Gero Decker
Giorgos Stoilos

Graham Morgan
Guandong Xu
Haiyang Sun
Hajo Reijers
Hakim Hacid
Hamid Motahari
Hamid Sarbazi-Azad
Harald Gjermundrod
Harald Meyer
Harry Wang
He Hu
Hideyuki Kawashima
Hiroyuki KASAI
Hong-Hai Do
Huaqing Li
Jan Schönherr
Jesus Carretero
Jiangang Ma
Jianxia Chen
Jilian Zhang
Jochem Vonk
Jonas Schulte
Joonsoo Bae
Josh Benaloh
Jutta Mlle
Kai Eckert
Kamil Kuliberda
Karen Sauvagnat
Ken. C.K. Lee
Ki Jung Lee
Konstantinos Karasavvas
Ladjel Bellatreche
Larry Moss
Laura Po
Lin Hao Xu
Linh Thao Ly
Liudong Xing
Lourdes-Angélica Martinez-Medina
Manuel Salvadores
Mao Ye
Marcin Wieloch
Marcus Schramm
Maria Auxilio Medina
María del Carmen
Suárez de Figueroa Baonza

Mariagrazia Fugini
 Massimo Benerecetti
 Maurizio Vincini
 Meng Ge
 Michael Uschold
 Mikaél Beauvois
 Minqi Zhou
 Mirko Orsini
 Mona Allani
 Nick Nikiforakis
 Nickolas Kavantzias
 Nicolas Schiper
 Noyan Ilk
 Oliver Kutz
 Ornsiri Thongoom
 Paul Khoury (El)
 Peter Graubmann
 Peter McBrien
 Pierluigi Plebani
 Ralph Bobrik
 Rares Vernica
 Raymond Pon
 Reagan Moore
 Ritu Khare
 Robert Jäschke
 Rochat Denis
 Romain Rouvoy
 Rui Joaquim
 Sasa Radomirovic
 Sattar B. Sadkhan

Serena Sorrentino
 Shantenu Jha
 Shaokun Fan
 Shengyue Ji
 Sherry Sun
 Shih-Hsi Liu
 Siang Song
 Siim Karus
 Simon Woodman
 Stefan Schlobach
 Stephan Neuhaus
 Stephanos Androutsellis-Theotokis
 Thomas Wächter
 Ton van Deursen
 Toshiyuki Amagasa
 Toumani, McBrien
 Vasileios Vlachos
 Vincenzo Maltese
 Wenjuan Xu
 Wojciech Barczynski
 Xiao Yuan Wang
 Xing Zhi Sun
 Xingjie Liu
 Yacine Sam
 Yiming Lu
 Yuan Tian
 Yue Zhang
 Yujin Tang
 Yves Younan
 Zhebglu Yang

Sponsoring Institutions

OTM 2008 was proudly sponsored by BN (Bauch & Navratil, Czech Republic), Nuevo Leon, and the City of Monterrey.



Supporting Institutions

OTM 2008 was proudly supported by RMIT University (School of Computer Science and Information Technology), Vrije Universiteit Brussel (Department of Computer Science), Technical University of Monterrey and Universidad Politécnica de Madrid.



Vrije Universiteit Brussel



TECNOLÓGICO
DE MONTERREY.



Table of Contents – Part II

Information Security (IS) 2008 International Conference

IS 2008 PC Co-chairs' Message	937

Intrusion Detection

Boosting Web Intrusion Detection Systems by Inferring Positive Signatures	938
<i>Damiano Bolzoni and Sandro Etalle</i>	
Principal Components of Port-Address Matrices in Port-Scan Analysis	956
<i>Hiroaki Kikuchi, Naoya Fukuno, Masato Terada, and Norihisa Doi</i>	
A Novel Worm Detection Model Based on Host Packet Behavior Ranking	969
<i>Fengtao Xiao, HuaPing Hu, Bo Liu, and Xin Chen</i>	

Information Hiding

Anonymous Resolution of DNS Queries	987
<i>Sergio Castillo-Perez and Joaquin Garcia-Alfaro</i>	
Steganography of VoIP Streams.....	1001
<i>Wojciech Mazurczyk and Krzysztof Szczypiorski</i>	
TrustMAS: Trusted Communication Platform for Multi-Agent Systems	1019
<i>Krzysztof Szczypiorski, Igor Margasiński, Wojciech Mazurczyk, Krzysztof Cabaj, and Paweł Radziszewski</i>	

Data and Risk Management

Computing Exact Outcomes of Multi-parameter Attack Trees	1036
<i>Aivo Jürgenson and Jan Willemson</i>	
Automatic Generation of Secure Multidimensional Code for Data Warehouses: An MDA Approach	1052
<i>Carlos Blanco, Ignacio García-Rodríguez de Guzmán, Eduardo Fernández-Medina, Juan Trujillo, and Mario Piattini</i>	

Trusted Reputation Management Service for Peer-to-Peer Collaboration.....	1069
<i>Lingli Deng, Yeping He, and Ziyao Xu</i>	

Access Control

A Model-Driven Approach for the Specification and Analysis of Access Control Policies.....	1087
<i>Fabio Massacci and Nicola Zannone</i>	
PuRBAC: Purpose-Aware Role-Based Access Control.....	1104
<i>Amirreza Masoumzadeh and James B.D. Joshi</i>	
Uncle-Share: Annotation-Based Access Control for Cooperative and Social Systems.....	1122
<i>Peyman Nasirifard and Vassilios Peristeras</i>	

Evaluation and Implementation

Verifying Extended Criteria for the Interoperability of Security Devices.....	1131
<i>Maurizio Talamo, Franco Arcieri, Giuseppe Della Penna, Andrea Dimitri, Benedetto Intrigila, and Daniele Magazzeni</i>	
Generating a Large Prime Factor of $p^4 \pm p^2 + 1$ in Polynomial Time....	1140
<i>Maciej Grześkowiak</i>	

Ontologies, Databases and Applications of Semantics (ODBASE) 2008 International Conference

ODBASE 2008 PC Co-chairs' Message.....	1151

Keynote

Artifact-Centric Business Process Models: Brief Survey of Research Results and Challenges.....	1152
<i>Richard Hull</i>	

Invited Papers

Ten Challenges for Ontology Matching.....	1164
<i>Pavel Shvaiko and Jérôme Euzenat</i>	
Dynamic Labelling Scheme for XML Data Processing.....	1183
<i>Maggie Duong and Yanchun Zhang</i>	

Real-Time Enterprise Ontology Evolution to Aid Effective Clinical Telemedicine with Text Mining and Automatic Semantic Aliasing Support	1200
<i>Jackei H.K. Wong, Wilfred W.K. Lin, and Allan K.Y. Wong</i>	

Engineering OODA Systems: Architectures, Applications, and Research Areas.....	1215
<i>Dimitrios Georgakopoulos</i>	

Semantic Matching and Similarity Measuring

Approximate Structure-Preserving Semantic Matching	1217
<i>Fausto Giunchiglia, Fiona McNeill, Mikalai Yatskevich, Juan Pane, Paolo Besana, and Pavel Shvaiko</i>	

Ontology-Based Relevance Assessment: An Evaluation of Different Semantic Similarity Measures.....	1235
<i>Michael Ricklefs and Eva Blomqvist</i>	

Equivalence of XSD Constructs and Its Exploitation in Similarity Evaluation	1253
<i>Irena Mlýnková</i>	

Design, Implementation and Evaluation of a New Semantic Similarity Metric Combining Features and Intrinsic Information Content	1271
<i>Giuseppe Pirró and Nuno Seco</i>	

Semantic Searching

Weighted Ontology for Semantic Search.....	1289
<i>Anna Formica, Michele Missikoff, Elaheh Pourabbas, and Francesco Taglino</i>	

RDF Snippets for Semantic Web Search Engines	1304
<i>Xi Bai, Renaud Delbru, and Giovanni Tummarello</i>	

Empirical Insights on a Value of Ontology Quality in Ontology-Driven Web Search	1319
<i>Darijus Strasunskas and Stein L. Tomassen</i>	

Ontology Development

Magic Rewritings for Efficiently Processing Reactivity on Web Ontologies	1338
<i>Elsa Liliana Tovar and María-Esther Vidal</i>	

Mediating and Analyzing Social Data	1355
<i>Ying Ding, Ioan Toma, Sin-Jae Kang, Zhixiong Zhang, and Michael Fried</i>	

Conceptual Synopses of Semantics in Social Networks Sharing Structured Data	1367
<i>Verena Kantere, Maria-Eirini Politou, and Timos Sellis</i>	

Ontology Maintenance and Evaluation

Sequential Patterns for Maintaining Ontologies over Time	1385
<i>Lisa Di-Jorio, Sandra Bringay, Céline Fiot, Anne Laurent, and Maguelonne Teisseire</i>	

Evaluating Automatically a Text Miner for Ontologies: A Catch-22 Situation?	1404
<i>Peter Spyns</i>	

Conceptual and Lexical Prototypicality Gradients Dedicated to Ontology Personalisation	1423
<i>Xavier Aimé, Frédéric Furst, Pascale Kuntz, and Francky Trichet</i>	

Explanation in the <i>DL-Lite</i> Family of Description Logics	1440
<i>Alexander Borgida, Diego Calvanese, and Mariano Rodriguez-Muro</i>	

Ontology Applications I

Using Ontologies for an Intelligent Patient Modelling, Adaptation and Management System	1458
<i>Matt-Mouley Bouamrane, Alan Rector, and Martin Hurrell</i>	

Context-Addressable Messaging Service with Ontology-Driven Addresses	1471
<i>Jaroslav Domaszewicz, Michal Koziuk, and Radoslaw Olgierd Schoeneich</i>	

Towards a System for Ontology-Based Information Extraction from PDF Documents	1482
<i>Ermelinda Oro and Massimo Ruffolo</i>	

Ontology Applications II

Detecting Dirty Queries during Iterative Development of OWL Based Applications	1500
<i>Ramakrishna Soma and Viktor K. Prasanna</i>	

Reusing the SIOC Ontology to Facilitate Semantic CWE Interoperability	1517
<i>Deirdre Lee, Vassilios Peristeras, and Nikos Loutas</i>	
An Ontology-Based Approach for Discovering Semantic Relations between Agent Communication Protocols	1532
<i>Maricela Bravo and José Velázquez</i>	
Semantic Query Processing	
Reference Fusion and Flexible Querying.....	1541
<i>Fatiha Saïs and Rallou Thomopoulos</i>	
Mediation-Based XML Query Answerability	1550
<i>Hong-Quang Nguyen, Wenny J. Rahayu, David Taniar, and Kinh Nguyen</i>	
Ontology Matching Supported by Query Answering in a P2P System ...	1559
<i>François-Élie Calvier and Chantal Reynaud</i>	
Using the Ontology Maturing Process Model for Searching, Managing and Retrieving Resources with Semantic Technologies	1568
<i>Simone Braun, Andreas Schmidt, Andreas Walter, and Valentin Zacharias</i>	
Author Index	1579

Table of Contents – Part I

OTM 2008 General Keynote

“The Future Internet: A Vision from European Research”	1
<i>Cristina Martinez Gonzalez</i>	

GADA + DOA + IS Keynote

E-science: Where Are We and Where Should We Go	2
<i>Daniel E. Atkins</i>	

Cooperative Information Systems (CoopIS) 2008 International Conference

CoopIS 2008 PC Co-chairs’ Message	5

Keynote

Flexible Recommendations in CourseRank	7
<i>Hector Garcia-Molina</i>	

Invited Paper

Collaborative Business Intelligence: Enabling Collaborative Decision Making in Enterprises	8
<i>Umeshwar Dayal, Ravigopal Vennelakanti, Ratnesh Sharma, Malu Castellanos, Ming Hao, and Chandrakant Patel</i>	

Web Service

Dynamic Web Services Provisioning with Constraints	26
<i>Eric Monfroy, Olivier Perrin, and Christophe Ringeissen</i>	
Timed Properties-Aware Asynchronous Web Service Composition	44
<i>Nawal Guermouche and Claude Godart</i>	
Load and Proximity Aware Request-Redirection for Dynamic Load Distribution in Peering CDNs	62
<i>Mukaddim Pathan, Christian Vecchiola, and Rajkumar Buyya</i>	

Business Process Technology

Process View Derivation and Composition in a Dynamic Collaboration Environment	82
<i>Xiaohui Zhao, Chengfei Liu, Wasim Sadiq, and Marek Kowalkiewicz</i>	
Business Provenance – A Technology to Increase Traceability of End-to-End Operations	100
<i>Francisco Curbera, Yurdaer Doganata, Axel Martens, Nirmal K. Mukhi, and Aleksander Slominski</i>	
Algorithms Based on Pattern Analysis for Verification and Adapter Creation for Business Process Composition	120
<i>Akhil Kumar and Zhe Shan</i>	

E-Service Management

Recovery of Concurrent Processes in a Service Composition Environment Using Data Dependencies	139
<i>Yang Xiao and Susan D. Urban</i>	
An Ontology-Based Approach to Validation of E-Services under Static and Dynamic Constraints	157
<i>Luigi Dragone</i>	
Data-Continuous SQL Process Model	175
<i>Qiming Chen and Meichun Hsu</i>	

Distributed Process Management

Multi-ring Infrastructure for Content Addressable Networks	193
<i>Djelloul Boukhelef and Hiroyuki Kitagawa</i>	
Online Querying of Concept Hierarchies in P2P Systems	212
<i>Katerina Doka, Athanasia Asiki, Dimitrios Tsoumakos, and Nectarios Koziris</i>	
A Multi-agents Contractual Approach to Incentive Provision in Non-cooperative Networks	231
<i>Li Lin, Jinpeng Huai, Yanmin Zhu, Chunming Hu, and Xianxian Li</i>	

Schema Matching

A Flexible Approach for Planning Schema Matching Algorithms	249
<i>Fabien Duchateau, Zohra Bellahsene, and Remi Coletta</i>	

BPEL to BPMN: The Myth of a Straight-Forward Mapping	265
<i>Matthias Weidlich, Gero Decker, Alexander Großkopf, and Mathias Weske</i>	
Boosting Schema Matchers	283
<i>Anan Marie and Avigdor Gal</i>	

Business Process Tracing

Cooperative Data Management Services Based on Accountable Contract	301
<i>Chen Wang, Surya Nepal, Shiping Chen, and John Zic</i>	
Cycle Time Prediction: When Will This Case Finally Be Finished?	319
<i>B.F. van Dongen, R.A. Crooy, and W.M.P. van der Aalst</i>	
XML Methods for Validation of Temporal Properties on Message Traces with Data	337
<i>Sylvain Hallé and Roger Villemaire</i>	

Workflow and Business Applications

A Query Language for MOF Repository Systems	354
<i>Ilia Petrov and Gabor Nemes</i>	
Towards a Calculus for Collection-Oriented Scientific Workflows with Side Effects	374
<i>Jan Hidders and Jacek Sroka</i>	
An Efficient Algorithm for Workflow Graph Structural Verification	392
<i>Fodé Touré, Karim Baïna, and Khalid Benali</i>	

Short Papers

Increasing the Efficiency of the Investments to Be Made in a Portfolio of IT Projects: A Data Envelopment Analysis Approach	409
<i>Rui de Oliveira Victorio, Antonio Juarez Alencar, Eber Assis Schmitz, and Armando Leite Ferreira</i>	
Merging Event-Driven Process Chains	418
<i>Florian Gottschalk, Wil M.P. van der Aalst, and Monique H. Jansen-Vullers</i>	
Flexible Process Graph: A Prologue	427
<i>Artem Polyvyanyy and Mathias Weske</i>	
Pattern Identification and Classification in the Translation from BPMN to BPEL	436
<i>Luciano García-Bañuelos</i>	

I-SSA: Interaction-Situated Semantic Alignment	445
<i>Manuel Atencia and Marco Schorlemmer</i>	
Awareness of Concurrent Changes in Distributed Software Development	456
<i>Claudia-Lavinia Ignat and Gérald Oster</i>	
Adapting Commit Protocols for Large-Scale and Dynamic Distributed Applications.....	465
<i>Pawel Jurczyk and Li Xiong</i>	
Semantic Interoperability in the BRITE Project: Ontologies as a Tool for Collaboration, Cooperation and Knowledge Management.....	475
<i>Timo Herborn, Ansgar Mondorf, Babak Mougouie, and Maria A. Wimmer</i>	
XML Data Integration Based on Content and Structure Similarity Using Keys.....	484
<i>Waraporn Viyanon, Sanjay K. Madria, and Sourav S. Bhowmick</i>	

Distributed Objects and Applications (DOA) 2008 International Conference

DOA 2008 PC Co-chairs' Message.....	495

Designing Distributed Systems

On the Design of a SIP-Based Binding Middleware for Next Generation Home Network Services	497
<i>Mourad Alia, Andre Bottaro, Fatoumata Camara, and Briac Hardouin</i>	
DQML: A Modeling Language for Configuring Distributed Publish/Subscribe Quality of Service Policies	515
<i>Joe Hoffert, Douglas Schmidt, and Aniruddha Gokhale</i>	
AOCI: Weaving Components in a Distributed Environment	535
<i>Guido Söldner, Sven Schober, Wolfgang Schröder-Preikschat, and Rüdiger Kapitza</i>	

Context in Distributed Systems

A Pluggable and Reconfigurable Architecture for a Context-Aware Enabling Middleware System	553
<i>Nearchos Paspallis, Romain Rouvoy, Paolo Barone, George A. Papadopoulos, Frank Eliassen, and Alessandro Mamelli</i>	

Context Grouping Mechanism for Context Distribution in Ubiquitous Environments	571
<i>M. Kirsch-Pinheiro, Y. Vanrompay, K. Victor, Y. Berbers, M. Valla, C. Frà, A. Mamelli, P. Barone, X. Hu, A. Devlic, and G. Panagiotou</i>	

A Graph-Based Approach for Contextual Service Loading in Pervasive Environments	589
<i>Amira Ben Hamida, Frédéric Le Mouël, Stéphane Frénôt, and Mohamed Ben Ahmed</i>	

High Availability

Extending Middleware Protocols for Database Replication with Integrity Support	607
<i>F.D. Muñoz-Escobí, M.I. Ruiz-Fuertes, H. Decker, J.E. Armendáriz-Íñigo, and J.R. González de Mendivil</i>	

Six-Shot Broadcast: A Context-Aware Algorithm for Efficient Message Diffusion in MANETs	625
<i>Benoît Garbinato, Adrian Holzer, and François Vessaz</i>	

Correctness Criteria for Database Replication: Theoretical and Practical Aspects	639
<i>Vaidė Zuikėvičiūtė and Fernando Pedone</i>	

Adaptive Distributed Systems

Optimizing the Utility Function-Based Self-adaptive Behavior of Context-Aware Systems Using User Feedback	657
<i>Konstantinos Kakousis, Nearchos Paspallis, and George A. Papadopoulos</i>	

Developing a Concurrent Service Orchestration Engine Based on Event-Driven Architecture	675
<i>Wei Chen, Jun Wei, Guoquan Wu, and Xiaoqiang Qiao</i>	

AKARA: A Flexible Clustering Protocol for Demanding Transactional Workloads	691
<i>A. Correia Jr., J. Pereira, and R. Oliveira</i>	

Grid computing, high performAnce and Distributed Applications (GADA) 2008 International Conference

GADA 2008 PC Co-chairs' Message	709

Scheduling Allocation

Dynamic Objective and Advance Scheduling in Federated Grids	711
<i>Katia Leal, Eduardo Huedo, and Ignacio M. Llorente</i>	
Studying the Influence of Network-Aware Grid Scheduling on the Performance Received by Users	726
<i>Luis Tomás, Agustín Caminero, Blanca Caminero, and Carmen Carrión</i>	
Q-Strategy: A Bidding Strategy for Market-Based Allocation of Grid Services	744
<i>Nikolay Borissov and Niklas Wirström</i>	

Databases in Grids

Active Integration of Databases in Grids for Scalable Distributed Query Processing	762
<i>Alexander Wöhrer and Peter Brezany</i>	
Managing Very-Large Distributed Datasets	775
<i>Miguel Branco, Ed Zaluska, David de Roure, Pedro Salgado, Vincent Garonne, Mario Lassnig, and Ricardo Rocha</i>	

Grid Applications

Peaks Detection in X-Ray Diffraction Profiles Using Grid Computing . . .	793
<i>Enrique Morales-Ramos, Miguel A. Vega-Rodríguez, Antonio Gómez-Iglesias, Miguel Cárdenas-Montes, Juan A. Gómez-Pulido, and Florentino Sánchez-Bajo</i>	
A Two Level Approach for Managing Resource and Data Intensive Tasks in Grids	802
<i>Imran Ahmad and Shikharesh Majumdar</i>	
Self-similarity and Multidimensionality: Tools for Performance Modelling of Distributed Infrastructure	812
<i>Raul Ramirez-Velarde, Cesar Vargas, Gerardo Castañon, and Lorena Martinez-Elizalde</i>	
Software Innovation for E-Government Expansion	822
<i>Stefania Pierno, Luigi Romano, Luisa Capuano, Massimo Magaldi, and Luca Bevilacqua</i>	

Data Management and Storage

Efficient Grid-Based Video Storage and Retrieval	833
<i>Pablo Toharia, Alberto Sánchez, José Luis Bosque, and Oscar D. Robles</i>	

Data Transformation Services over Grids with Real-Time Bound Constraints	852
<i>Alfredo Cuzzocrea</i>	
Towards a High Performance Implementation of MPI-IO on the Lustre File System	870
<i>Phillip Dickens and Jeremy Logan</i>	
 New Tendencies and Approaches	
The Grid as a Single Entity: Towards a Behavior Model of the Whole Grid	886
<i>Jesús Montes, Alberto Sánchez, Julio J. Valdés, María S. Pérez, and Pilar Herrero</i>	
A Reference Model for Grid Architectures and Its Analysis	898
<i>Carmen Bratosin, Wil van der Aalst, Natalia Sidorova, and Nikola Trčka</i>	
Distributing Orthogonal Redundancy on Adaptive Disk Arrays	914
<i>J.L. Gonzalez and Toni Cortes</i>	
Author Index	933

IS 2008 PC Co-chairs' Message

On behalf of the Program Committee of the Third International Symposium on Information Security (IS 2008), it was our great pleasure to welcome the participants to the IS 2008 conference, held in conjunction with OnTheMove Federated Conferences (OTM 2008), November 9–14, 2008, in Monterrey, Mexico. Information security is a challenging and rapidly evolving research area with an increasing impact on the overall ICT sector. The objective of the symposium was to stimulate research on information security and to encourage interaction between researchers from all over the world. In response to the call for papers, 33 submissions were received, from which 14 were carefully selected for presentation in five technical sessions. Each paper was peer reviewed by at least three members of the Program Committee or external reviewers. The symposium program covered a broad range of research topics: intrusion detection, information hiding, data and risk management, access control, and finally evaluation and implementation. We thank all the authors who submitted valuable papers to the symposium. We would like to express our gratitude to the members of the Program Committee and to the external reviewers for their constructive and insightful comments. We are also indebted to the many individuals and organizations that have contributed to this event, and we would like to thank in particular Springer. Last but not least, we are grateful to the OTM organizing committee and chairs for their help in all aspects of the organization of this symposium. We hope that you enjoyed the Third International Symposium on Information Security at Monterrey, Mexico, and that you found it a stimulating forum for the exchange of ideas, results and recent findings.

August 2008

Jong Hyuk Park
Bart Preneel
Ravi Sandhu
André Zúquete

Boosting Web Intrusion Detection Systems by Inferring Positive Signatures^{*}

Damiano Bolzoni¹ and Sandro Etalle^{1,2}

¹ University of Twente, Enschede, The Netherlands
`damiano.bolzoni@utwente.nl`

² Eindhoven Technical University, The Netherlands
`s.etalles@tue.nl`

Abstract. We present a new approach to anomaly-based network intrusion detection for web applications. This approach is based on dividing the input parameters of the monitored web application in two groups: the “regular” and the “irregular” ones, and applying a new method for anomaly detection on the “regular” ones based on the inference of a regular language. We support our proposal by realizing Sphinx, an anomaly-based intrusion detection system based on it. Thorough benchmarks show that Sphinx performs better than current state-of-the-art systems, both in terms of false positives/false negatives as well as needing a shorter training period.

Keywords: Web application security, regular languages, anomaly detection, intrusion detection systems.

1 Introduction

In the last decade, the Internet has quickly changed from a static repository of information into a practically unlimited on-demand content generator and service provider. This evolution is mainly due to the increasing success of so-called web applications (later re-branded web services, to include a wider range of services). Web applications made it possible for users to access diverse services from a single web browser, thereby eliminating reliance on tailored client software.

Although ubiquitous, web applications often lack the protection level one expects to find in applications that deal with valuable data: as a result, attackers intent on acquiring information such as credit card or bank details will often target web applications. Web applications are affected by a number of security issues, primarily due to a lack of expertise in the programming of secure applications. To make things worse, web applications are typically built upon multiple technologies from different sources (such as the open-source community), making it difficult to assess the resulting code quality. Other factors affecting the (in)security of web applications are their size, complexity and extensibility. Even

^{*} This research is supported by the research program Sentinels (<http://www.sentinals.nl>). Sentinels is being financed by Technology Foundation STW, the Netherlands Organization for Scientific Research (NWO), and the Dutch Ministry of Economic Affairs.

with high quality components, the security of a web application can be compromised if the interactions between those components are not properly designed and implemented, or an additional component is added at a later stage without due consideration (e.g., a vulnerable web application could grant an attacker the control of another system which communicates with it).

An analysis of the Common Vulnerabilities and Exposures (CVE) repository [1] conducted by Robertson et al. [2] shows that web-related security flaws account for more than 25% of the total number of reported vulnerabilities from year 1999 to 2005 (this analysis cannot obviously take into account vulnerabilities discovered in web applications developed internally by companies). Moreover, the Symantec 2007 Internet Security Threat Report [3] states that most of the *easily exploitable vulnerabilities* (those requiring little knowledge and effort on the attacker side) are related to web applications (e.g., SQL Injection and Cross-site Scripting attacks). Most of the web application vulnerabilities are SQL Injections and Cross-site Scripting. These statistics show that web applications have become the Achilles' heel in system and network security.

Intrusion detection systems (IDSs) are used to identify malicious activities against a computer system or network. The growth of web applications (and attacks targeting them) led to adaptations of existing IDSs, yielding systems specifically tailored to the analysis of web traffic (sometimes called *web application firewalls* [4]). There exist two kinds of intrusion detection systems: *signature*- and *anomaly-based*. Here we focus on anomaly detection systems: as also argued by Vigna [2], signature-based systems are less suitable to protect web-services; among the reasons why anomaly-based systems are more suitable for protecting web applications we should mention that (1) they do not require any a-priori knowledge of the web application, (2) they can detect polymorphic attacks and (3) they can protect custom-developed web applications. On the negative side, anomaly-based systems are generally not easy to configure and use. As most of them employ mathematical models, users usually have little control on the way the system detects attacks. Often, system administrators prefer signature-based IDSs over anomaly-based ones because they are – according to Kruegel and Toth [5] – easier to implement and simpler to configure, despite the fact they could miss a significant amount of real attacks. Finally, anomaly-based systems usually show a high number of false positives [6], and – as we also argued in [7] – a high number of false positives is often their real limiting factor. These issues make the problem of protecting web servers particularly challenging.

Contribution. In this paper we present a new approach for anomaly detection devised to detect data-flow attacks [8] to web applications (attacks to the work flow are not taken into consideration) and we introduce Sphinx, an anomaly-based IDS based on it. We exploit the fact that, usually, most of the parameters in HTTP requests present some sort of regularities: by considering those regularities, we divide parameters into “regular” and “irregular” (whose content is highly variable) ones; we argue that, for “regular” parameters, it is possible to exploit their regularities to devise more accurate detection models. We substantiate this with a number of contributions:

- We introduce the concept of “positive signatures”: to carry out anomaly-detection on the “regular” parameters, we first infer human-readable regular expressions by analyzing the parameter content, then we generate *positive signatures* matching *normal* inputs.
- We build a system, Sphinx, that implements our algorithm to automatically infer regular expressions and generate positive signatures; positive signatures are later used by Sphinx to build automaton-based detection models to detect anomalies in the corresponding “regular” parameters. For the parameters we call “irregular”, Sphinx analyzes their content using an adapted version of our NIDS POSEIDON [9] (as it would not be “convenient” to generate a positive signature).
- We extensively benchmark our system against state-of-the-art IDSs such as WebAnomaly [10], Anagram [11] and POSEIDON.

We denote the generated signatures as “positive signatures”, following the idea that they are as flexible as signatures but match positive inputs (in contrast with usual signatures used to match malicious inputs). Differently from mathematical and statistical models, positive signatures do not rely on any data frequency/presence observation or threshold. As shown by our benchmarks, positive signatures successfully detect attacks with a very low false positive rate for “regular” parameters.

Our new approach merges the ability of detecting new attacks without prior knowledge (common in anomaly-based IDSs) with the possibility of easily modifying/customizing the behaviour of part of the detection engine (common in signature-based IDSs).

Sphinx works with *any* web application, making custom-developed (or close-source) ones easily protected too. By working in an automatic way, Sphinx requires little security knowledge from system administrators, however expert ones can easily review regular expressions and make modifications.

We performed thorough benchmarks using three different data sets; benchmarks show that Sphinx performs better than state-of-the-art anomaly-based IDSs both in terms of false negatives and false positives rate as well as presenting a better learning curve than competing systems.

2 Preliminaries

In this section, we introduce the definitions and the concepts used in the rest of the paper.

Anomaly-based systems. For the purpose of this paper, we assume the presence of an application A that exchanges information over a network (e.g., think of a web server connected to the Internet running web applications). An input is any finite string of characters and we say that S is the set of all possible inputs.

Anomaly-based IDSs are devised to recognize regular activity and make use of a model $M_A \subseteq S$ of *normal* inputs: if an input $i \notin M_A$ then the IDS raises an alert. Typically, M_A is defined implicitly by using an abstract model M_{abs}

We can discard the request version and – for the sake of exposition – the method. Of interest to us is the presence of a path, a number of parameter names and of their respective values (in Figure 1 the parameter names are “name”, “file” and “sid” and their respective values are “New”, “Article” and “25”). The set of parameters is finite. A value can be any string (though, not all the strings will be accepted by the web application). Since no type is defined, the semantic of each parameter is implicitly defined within the context of the web application and such parameters are usually used in a consistent manner (i.e., their syntax is fixed). In the sequel, we refer to the natural projection function: Given an input i $path?p_1 = v_1 \& p_2 = v_2 \& \dots \& p_n = v_n$, we define $p_n(i) = v_i$ as the function extracting the value of parameter p_n from input i .

Exploiting regularities. Intuitively, it is clear that the more “predictable” the input of the application A is, the easier it is to build a model M_A satisfying the desiderata (1), (2) and (3). For instance, if we knew that A accepted only – say – strings not containing any “special character” (a very predictable input), then building M_A as above would be trivial.

Our claim is that, in the context of web applications, it is possible to exploit the regularities which are not present in other settings to define and build M_A based on the inference of regular automata, which leads to the definition of an IDS that is more effective (yet simpler) than state-of-the-art systems.

Commonly, anomaly-based IDSs build (and use) a single model M to analyse network traffic. Our proposal takes advantage of the fact that requests to web applications present a fixed syntax, consisting of a sequence of *parameter* = *value*, and instead of building a single model to analyse the input, it builds an *ad hoc* model M_n for each parameter p_n (in practice, we create a separate model for many – not all – parameters). As already observed by Kruegel and Vigna in [14], this allows it to create a more faithful model of the application input. The idea is that of defining M_A implicitly by electing that $i \in M_a$ iff for each parameter n we have that $p_n(i) \in M_n$ (or that $p_n(i)$ is empty).

Regular and irregular parameters. So we first divide the parameters in two groups: the *regular parameters* and the *irregular parameters*. The core of our idea is that for the *regular* parameters it is better to define M_n as a *regular language* rather than using state-of-the-art anomaly-based systems. By “better” we mean that this method yields (a) lower false positive rate, (b) same (or higher) detection rate (c) a shorter learning phase. We support our thesis by presenting an algorithm realizing this.

For each regular parameter, we build a model using a combination of abstraction and regular expression inference functions that we are going to explain in the following section: We call this the *regular-text* methodology, following the intuition it is devised to build the model for parameters which are usually filled by data having a well-defined format (e.g., integer numbers, dates, user session cookies etc.). For the *irregular* parameters we use classical anomaly-based techniques, i.e., n-gram analysis: We call this the *raw-data* methodology, since it is meant to be more suitable for building the model of parameters containing e.g., pieces of blogs or emails, images, binary data etc.

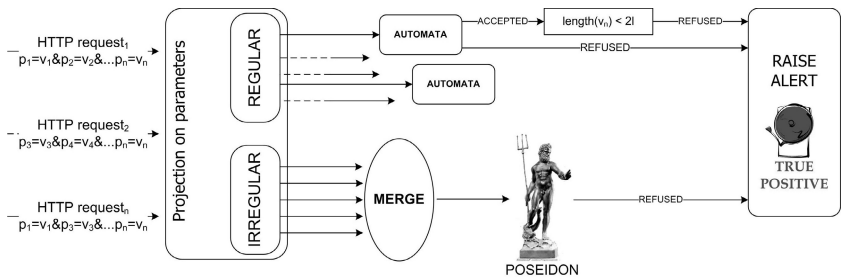


Fig. 2. Sphinx’s internals

4 Sphinx’s Detection Engine

To substantiate our claims, we built Sphinx: An anomaly-based intrusion detection systems specifically tailored to detect attacks in a web application data flows. Let us see how it works: building the intrusion detection system involves the following steps.

4.1 Building the Model

We first outline how we build the model M_A of the application given a *training set* DS ; DS is a set of inputs (i.e., HTTP requests), which we assume does not contain fragments of attacks. Typically, DS is obtained by making a dump of the application input traffic during a given time interval, and it is cleaned (i.e., the malicious traffic is removed) off-line using a combination of signature-based intrusion detection techniques and manual inspection.

During the *first* part of the training, we discover the set of parameters used by the web application: DS is scanned a first time and the parameters $\{p_1, \dots, p_n\}$ are extracted and stored. We call $DS_n = \{p_n(i) \mid i \in DS\}$ the training set for the parameter p_n (i.e., the projection of DS on the parameter p_n).

In the second step we divide the parameters into two classes: the *regular* ones (for which we use the new anomaly detection algorithms based on regular expressions) and the irregular ones. In practice, to decide which parameters are the “regular” ones, in the sequel we use a simple a-priori syntactic check: If at least the 10% of the samples in DS_n contains occurrences of more than 5 distinct non-alphanumeric characters, we say that p_n is an irregular parameter, otherwise it is a regular one. This criterion for separating (or, better, defining) the regular parameters from the irregular ones is clearly arbitrary. Simply, our benchmarks have shown that it gives good results. We impose a minimum amount of samples (10%) to present more than 5 distinct non-alphanumeric characters to prevent Sphinx’s engine from classifying a parameter as “irregular” because of few anomalous samples. An attacker could in fact exploit this to force the system to classify any parameter as “irregular”.

In the last step of the training phase we build a model M_n for each of the regular parameter p_n , using the training set DS_n . The *irregular* parameters, in turn, are again grouped together and for them we build a unique model: We could also build a single model per irregular parameter, but this would slow down the learning phase, which is already one of the weak spots of classical anomaly detection techniques.

4.2 The Regular-Text Methodology

This represents the most innovative aspect of our contribution. The *regular-text* methodology is designed to build a simple model of the “normal” input of the regular parameters. This model is represented by a regular expression and anomalies are detected by the derived finite automaton. We illustrate this methodology by presenting two algorithms realizing it: The first one is called the *simple regular expression generator* (SREG) and it is meant to illustrate the fundamental principles behind the construction of such regular language, the second one is called *complex regular expression generator* (CREG), and can be regarded as a further development of the first one. Here we should mention that standard algorithms to infer regular expressions (see [15] for a detailed overview) cannot be used for intrusion detection because they infer an expression matching exactly the strings in the training data set only, while we need to match a “reasonable” superset of it.

Simple regular expression generator. Here we introduce our first algorithm. We have a training set DS_n (the training set relative to parameter n) and we want to build a model M_n of the parameter itself, and a decision procedure to determine for a given input i whether $p_n(i)$ is contained in M_n or not.

Our first algorithm to generate M_n is based on applying two abstractions to DS_n . The first abstraction function, abs_1 , is devised to abstract all letters and all digits (other symbols are left untouched), and works as follows:

$$abs_1(c_1 \dots c_n) = abs_1(c_1), \dots, abs_1(c_n)$$

$$abs_1(c_i) = \begin{cases} "a", & c_i \in \{ "a", \dots, "Z" \} \\ "1", & c_i \in \{ "0", \dots, "9" \} \\ c_i & otherwise \end{cases}$$

Thus abs_1 abstracts alphanumeric characters while leaving non-alphanumeric symbols untouched (for the reasons we clarified in Section 3). The reason for this choice is that, in the context of web applications, the presence of “unusual” symbols (or a concatenation of them) could indicate the presence of attack payloads.

The second abstraction we use is actually a contraction:

$$abs_2(c_1 \dots c_n) = \begin{cases} abs_2(c_2 \dots c_n) & \text{if } c_1 = c_2 = c_3 = "a" \text{ or } "1" \\ c_1 c_1 abs_2(c_3 \dots c_n) & \text{if } c_1 = c_2 \neq c_3 \text{ and } c_1 = "a" \text{ or } "1" \\ c_1 \cdot abs_2(c_2 \dots c_n) & \text{if } c_1 \neq c_2 \text{ or } c_1 = c_2 \text{ and } c_1 \neq "a" \text{ and } c_1 \neq "1" \end{cases}$$

Table 1. Some examples of applying abstractions abs_1 and abs_2 on different inputs

Input	$abs_1(i)$	$abs_2(i')$
11/12/2007	11/11/1111	11/11/11
addUser	aaaaaaa	aa
C794311F-FC92-47DE-9958	a111111a-a11-11aa-1111	a11a-a11-11aa-11

Intuitively, abs_2 collapses all strings of letters (resp. digits) of length greater or equal to two onto strings of letters (resp. digits) of length two. Again, symbols are left untouched, as they may indicate the presence of an attack. Table 1 provides some examples of application of abs_1 and abs_2 on different input strings.

These two abstraction algorithms are enough to define our first model, only one detail is still missing. If the samples contained in DS_n have maximum length say l , then we want our model M_n to contain strings of maximum length $2l$: an input which is much longer than the samples observed in the training set is considered anomalous (as an attacker could be attempting to inject some attack payload).

Definition 1. Let DS_n be a training set. Let $l = \max\{|x| \mid x \in DS_n\}$. We define the simple regular-text model of DS_n to be

$$M_n^{simple} = \{x \mid |x| \leq 2l \wedge \exists y \in DS_n \text{ } abs_2(abs_1(x)) = abs_2(abs_1(y))\}$$

During the *detection* phase, if $p_n(i) \notin M_n^{simple}$ then an alert is raised. The decision procedure for checking whether $i \in M_n^{simple}$ is given by the finite automaton $\alpha_{M_n^{simple}}$ that recognizes M_n^{simple} . Building $\alpha_{M_n^{simple}}$ is almost straightforward. It is implemented using a (unbalanced) tree data-structure, therefore adding a new node (i.e., a previously unseen character) costs $O(l)$, where l is the length of the longest observed input. The complexity of building the tree for n inputs is therefore $O(n \cdot l)$. The decision procedure to check, given an input i , if $p_n(i) \in M_n$ has complexity $O(l)$. To simplify things, we can represent this automaton as a regular expression.

Complex regular expression generator. The simple SREG algorithm is effective for illustrating how regular expressions can be useful in the context of anomaly detection and how they can be used to detect anomalies in regular parameters. Nevertheless we can improve on SREG in terms of FPs and FNs by using an (albeit more complex) algorithm, which generates a different, more complex model.

The algorithm goes through two different phases. In the *first phase* each DS_n is partitioned in groups with common (shared) prefixes or suffixes (we require at least 3 shared characters, to avoid the generation of useless regular expressions).

Table 2. Examples of how SREG works on different input sets

Training sets	$abs_2(abs_1(i))$	SREG
01/01/1970	11/11/11	1(1/1(1/11/11)/1/11)
30/4/85	11/1/11	
9/7/1946	1/1/11	
41E44909-C86E-45EE-8DA1	11a11-a11a-11aa-1a1	1(1(a11-a11a-11aa-1a1 1a1a1a-11a-11-a11a 1a1aa1-a11-11a-11aa) a11a1a-11a-11-a11a)
0F786C5B-940B-4593-B96D	1a11a1a-11a-11-a11a	
656E0AB4-B221-422F-92AC	111a1aa1-a11-11a-11aa	

Table 3. Examples of pattern selection, generated regular expressions for samples and the resulting one

Training set	Symbol Pattern	Shared Pattern	Intermediate Regular Expressions	Resulting Regular Expression
alias@atwork.com 3l1t3@hack.it info@dom-ain.org	[. @ .] [. @ .] [@ - .]	[@ .]	$(a^+ [.]^+ @ (a^+).(a^+)$ $((a 1)^+ [.]^+ @ (a^+).(a^+)$ $(a^+) @ (a^+ [-])^+.(a^+)$	$((a 1)^+ [.]^+ @ (a^+ [-])^+.(a^+)$

In the *second phase*, the algorithm generates a regular expression for each group as follows. First, it applies abstractions abs_1 and abs_2 on each stored body (we call the *body* the part of the input obtained by removing the common prefix or suffix from it: prefixes and suffixes are handled later). Secondly, it starts to search for common symbol patterns inside bodies. Given a string s , we define the symbol pattern of s the string obtained by removing all alphanumerical characters from s .

As bodies could contain different symbol patterns, non-trivial patterns (i.e., patterns of length greater than one) are collected and the pattern matching the highest number of bodies is selected. If some bodies do not match the selected pattern, then the same procedure is repeated on the remaining bodies set till no more non-trivial patterns can be found.

For each non-trivial symbol pattern discovered during the previous step, the algorithm splits each body into sub-strings according to the symbol pattern (e.g., $s = s' - s'' - s'''$ is split in $\{s', s'', s'''\}$ w.r.t. the pattern). Corresponding sub-strings are grouped together (e.g., having strings s_1, s_2 and s_3 the algorithm creates $g1 = \{s'_1, s'_2, s'_3\}$, $g2 = \{s''_1, s''_2, s''_3\}$ and $g3 = \{s'''_1, s'''_2, s'''_3\}$) and a regular expression is generated for each group. This regular expression is the modified according to some heuristics to match also similar strings. Regular expressions are then merged with the corresponding symbol pattern, and any previously found prefix or suffix is eventually added (e.g., $re = (prefix)re_{g1} - re_{g2} - re_{g3}$). Table 3 depicts an example of the different steps of CREG.

Finally, for bodies which do not share any non-trivial symbol pattern with the other members of the group, a dedicated general regular expression is generated. Table 4 shows some examples of generated regular expressions for different sets of strings.

Given the resulting regular expression (i.e., the positive signature), we build a finite automaton accepting the language it represents. The automaton is built in such a way that it accepts (as in the case of SREG) only strings of length less

Table 4. Examples of how CREG works on different input sets

Training sets	Pattern	Resulting Regular Expression
01/01/1970 30/4/85 9/7/1946	[/ /]	$(1^+/1^+/1^+)$
addUser deleteUser viewUser	N/A	$(a^+)User$
41E44909_C86E_45EE_8DA1 0F786C5B-940B-4593-B96D 656E0AB4-B221_422F-92AC	Cannot find non-trivial patterns $\Rightarrow$ general regular expression	$((a 1)^+ [- .]^+)$

than $2l$. In the detection phase, if an input is not accepted by the automaton, an alert is raised.

Effectiveness of positive signatures. Because of the novelty of our approach, let us see some concrete examples regarding the potential of positive signatures.

Think of a signature such as “id=1⁺”, accepting numeric values: the parameter *id* is virtually protected from any data-flow attack, since only digits are accepted. When we consider more complex signatures, such as “email=((a|1)⁺ [. | -]⁺ @ (a⁺ [-]⁺ . (a⁺)⁺)” (extracted from Table 3), it is clear that common attack payloads would be easily detected by the automaton derived from the regular expression, as they require to inject different symbol sets and in different orders.

One could argue that it could be sufficient (and simpler) to detect the presence of typical attack symbols (having them classified and categorized) or, better, the presence of a symbol set specific to an attack (e.g., “'”, “,” and “-” for a SQL Injection). However, a certain symbol set is not said to be harmful per se, but it must be somehow related to the context: In fact, the symbol “,” used in SQL Injection attacks, can also be found in some representations of real numbers. Positive signatures, in contrast with usual state-of-the-art anomaly detection approaches, provide a sort of context for symbols, thus enhancing the detection of anomalies.

For instance, by May 2008, CVE contains more than 3000 SQL Injection and more than 4000 Cross-site Scripting attacks but only less than 250 path traversal and less than 400 buffer overflow attacks (out of a total of more than 30000 entries). Most of the SQL Injections happen to exploit “regular” parameters (integer-based), where the input is used inside a “SELECT” statement and the attacker can easily add a crafted “UNION SELECT” statement to extract additional information such as user names and their passwords. The same reasoning applies to Cross-site Scripting attacks. Sphinx’s positive signatures can significantly enhance the detection of these attacks.

4.3 The Raw-Data Methodology

The raw-data methodology is used to handle the “irregular” parameters. For them, using regular automata to detect anomalies is not a good idea: The input is so heterogeneous that any automaton devised to recognize a “reasonable” super set of the training set would probably accept any input. Indeed, for this kind of heterogeneous parameters we can better use a classical anomaly detection engine, based on statistical content analysis (e.g., n-gram analysis). In the present embodiment of Sphinx we use our own POSEIDON [9] (which performs very well in our benchmarks), but we could have used any other anomaly-based NIDS. POSEIDON is a 2-tier anomaly-based NIDS that combines a neural network with n-gram analysis to detect anomalies. POSEIDON originally performs a packet-based analysis: Every packet is classified by the neural network, then, using the classification information given, the real detection phase takes place based on statistical functions considering the byte frequencies and distributions (the n-gram analysis). In the context of web applications, we are dealing with

streams instead of packets, therefore we have adapted POSEIDON to the context. POSEIDON requires setting a threshold value to detect anomalous inputs: to this end, in our tests we have used the automatic heuristic provided in [7].

4.4 Using the Model

When the training phase is completed, Sphinx switches to detection mode. As a new HTTP request comes, parameters are extracted applying the projections $p_1, \dots, p_n$. Sphinx stores, for each parameter analyzed during the training phase, information regarding the model to use to test its input. For a parameter that was labelled as “regular”, the corresponding automaton is selected. If it does not accept the input, then an alert is raised. If the parameter was labelled “irregular”, the content is analyzed using the adapted version of POSEIDON (which uses a single model for all the irregular parameters). If the content is considered to deviate from the “normal” model, an alert is raised. Sphinx raises an alert also in the case a parameter has never been analyzed before and suddenly materializes in a HTTP request, since we consider this eventuality as an attempt to exploit a vulnerability by an attacker.

Editing and customizing positive signatures. One of the most criticized disadvantages of anomaly-based IDSs lies in their “black-box” approach. Being most of anomaly-based IDS based on mathematical models (e.g., neural networks), users have little or no control over the detection engine internals. Also, users have little influence on the false positive/negative rates, as they can usually adjust some threshold values only (the link between false positives/negatives and threshold values is well-known [16]).

Signatures, as a general rule, provide more freedom to customize the detection engine behaviour. The use of positive signatures opens the new possibility of performing a thorough tuning for anomaly-based IDSs, thereby modifying the detection models of regular parameters. Classical anomaly-based detection systems do not offer this possibility as their models aggregate collected information, making it difficult (if not impossible) to remove/add arbitrary portions of data.

Positive signatures allow IT specialists to easily (and quickly) modify or customize the detection models when there is a need to do so (see Table 5), like when (1) some malicious traffic, that was incorporated in the model during the training phase, has to be purged (to decrease the false negative rate) and (2) a new (previously unseen) input has to be added to the model (to decrease the false positive rate).

Table 5. Some examples of signature customization

Positive Signature	Problem	Action
$id=d^+ \mid (a^+ \mid \cdot \mid \cdot \mid \cdot \mid \cdot)^+$	The payload of a SQL Injection attack was included in the training set	The IT specialist manually modifies the signature $\Rightarrow id=d^+$
$date=1^+/1^+/1^+$	A new input (“19-01-1981”) is observed after the training phase, thereby increasing the false positive rate	The IT specialist re-train the model for parameter with the new input and the positive signature $\Rightarrow date=1^+/1^+/1^+ \mid 1^+-1^+-1^+$ is automatically generated

5 **Benchmarks**

The quality of the data used in benchmarks (and the way it was collected) greatly influences the number of successfully detected attacks and false alerts: Test data should be representative of the web server(s) to monitor, and the attack test bed should reflect modern attack vectors. Presently the only (large) public data set for testing intrusion detection systems is the DARPA data set [17], dated back to 1999. Although this data set is still widely used (since public data sets are scarce), it presents significant shortcomings that make it unsuitable to test our system: E.g., only four attacks related to web (and most of them target web server’s vulnerabilities) are available and traffic typology is outdated (see [18,19] for detailed explanations about its limitations). So, to carry out our experiments we collected three different data sets from three different sources: Real production web sites that strongly rely on user parameters to perform their normal activity.

The first data set is a deployment of the widely-known PostNuke (a content-management system). The second comes from a (closed-source) user forum web application, and it contains user messages sent to the forum, which present a variable and heterogeneous content. The third data set has been collected from the web server of our department, where PHP and CGI scripts are mainly used. Each data set contains both GET and POST requests: Sphinx’s engine can interpret the body of POST requests as well, since only the request syntax changes from a GET request, but the content, in case of regular parameters, looks similar. In case of encoded content (for instance, a white space is usually encoded as the hexadecimal value “%20”), the content is first decoded by Sphinx’s engine and then processed.

We collected a number of samples sufficient to perform extensive training and testing (never less than two weeks of traffic and in one case a month, see also Table 6). Data used for training have been made attack-free by using Snort to remove well-known attacks and by manually inspecting them to purge remaining noise.

Comparative benchmarks. To test the effectiveness of Sphinx, we compare it to three state-of-the-art systems, which have been either developed specifically to detect web attacks or have been extensively tested with web traffic.

First, WebAnomaly (Kruegel et al. [14]) combines five different detection models, namely attribute length, character distribution, structural inference, attribute presence and order of appearance, to analyze HTTP request parameters. Second, Anagram (Wang et al. [11]) uses a Bloom filter to store any n-gram (i.e., a sequence of bytes of a given length) observed during a training phase,

Table 6. Collected data sets: code name for tests, source and number of samples

Data set	Web Application	# of samples (HTTP requests)
<i>DSA</i>	PostNuke	~460000 (1 month)
<i>DSF</i>	(Private) User forum	~290000 (2 weeks)
<i>DS_C</i>	CS department’s web site (CGI & PHP scripts)	~85000 (2 weeks)

without counting the occurrences of n-grams. During the detection phase, Anagram flags as anomalous a succession of previously unseen n-grams. Although not specifically designed for web applications, Anagram has been extensively tested with logs captured from HTTP servers, achieving excellent results. We set the parameters accordingly to authors’ suggestions to achieve the best detection and false positive rates. Third, our own POSEIDON, the system we adapted to handle raw-text parameters in Sphinx. POSEIDON, during our previous experiments [9], showed a high detection rate combined with a low false positive rate in tests related to web traffic, outperforming the leading competitor.

We divide tests into two phases. We compare the different engines first by considering only the “regular” parameters. Later, we consider full HTTP requests (with both “regular” and “irregular” parameters).

The goal our tests is twofold. Next to the effectiveness of Sphinx, we are also interested in testing its the *learning rate*: Any anomaly-based algorithm needs to be trained with a certain amount of data before it is able to correctly flag attacks without generating a massive flow of false alerts. Intuitively, the longer the training phase is, the better the IDS should perform. But an anomaly detection algorithm that requires a shorter training phase it is certainly easier to deploy than an algorithm that requires a longer training phase.

Testing the regular-expression engine. In this first test, we compare our CREG algorithm to WebAnomaly, Anagram and POSEIDON using training sets of increasing size with “regular” requests only (requests where raw-data parameters have been previously removed). This test aims to demonstrate the effectiveness of our approach over previous methods when analyzing regular parameters. We use training sets of increasing size to measure the learning rate and to simulate a training phase as it could take place in a real environment, when a system is not always trained thoroughly. For the attack test bed, we selected a set of real attacks which truly affected the web application we collected the logs of and whose exploits have been publicly released. Attacks include path traversal, buffer overflow, SQL Injection and Cross-site Scripting payloads. Attack mutations, generated using the Sploit framework [20], have been included too, to reproduce the behaviour of an attacker attempting to evade signature-based systems. The attack test bed contains then 20 attacks in total. Table 7 reports results for tests with “regular” requests.

Our tests show that with a rather small training set (20000 requests, originally collected in less than two days), CREG generates 43 false positives (~0,009%), less than 2 alerts per day. The “sudden” decrease in FPs shown by CREG (and

Table 7. Results for CREG and comparative algorithms on “regular” requests only

#training samples		CREG	WebAnomaly	Anagram	POSEIDON
5000	Attacks	20/20	18/20	20/20	20/20
	FPS	1062	1766	144783	1461
10000	Attacks	20/20	16/20	20/20	20/20
	FPS	1045	1529	133023	1387
20000	Attacks	20/20	16/20	20/20	20/20
	FPS	43	177	121484	1306
50000	Attacks	20/20	14/20	20/20	20/20
	FPS	16	97	100705	1251

Webanomaly) when we train it with at least 20000 requests is due to the fact that, with less than 20000 training samples, some parameters are not analyzed during training (i.e., some URLs have not been accessed), therefore no model is created for them and by default this event is considered malicious. One surprising thing is the high number of false positives shown by Anagram [11,21]. We believe that this is due to the fact that Anagram raises a high number of false positives on specific fields whose content looks pseudo-random, which are common in web applications. Consider for example the following request parameter *sid=0c8026e78ef85806b67a963ce58ba823* (it is a user’s session ID automatically added by PostNuke in each URL link), being this value randomly generated as a new user comes: Such a string probably contains a number of n-grams which were not observed during the training phase therefore, and Anagram is likely to flag any session ID as anomalous. On the other hand, CREG exploits regularities in inputs, by extracting the syntax of the parameter content (e.g., the regular expression for *sid* is $(a^+|d^+)^+$), and easily recognizes similar values in the future. WebAnomaly shows (unexpectedly, at least in theory) a worse detection rate as the training set samples increase. This is due to the fact that the content of new samples is similar to some attack payloads, thus the system is not able to discern malicious traffic.

Testing Sphinx on the complete input. We show the results of the second test which uses the complete input of the web application (and not only the regular parameters). We use the two data sets DS_B and DS_C : DS_B contains 78 regular and 10 irregular parameters; DS_C respectively 334 and 10. We proceed as before, using different training sets with increasing numbers of samples. To test our system, we have used the attack database presented in [21] which has already been used to assess several intrusion-detection systems for web attacks. We adapted the original attack database and added the same attack set used in our previous test session. We found this necessary because [21] contains some attacks to the platforms (e.g., a certain web server vulnerability in parsing inputs) rather than to the web applications themselves (e.g., SQL Injection attacks are missing). Furthermore, we had to exclude some attacks since they target web server vulnerabilities by injecting the attack payload inside the HTTP headers: Although Sphinx could be easily adapted to process header fields, our logs do not always contain a HTTP header information. In total, our attack bed contains

Table 8. Results for Sphinx and comparative algorithms on full requests from DS_B : we report separate false positive rates for Sphinx (RT stands for “regular-text” models and RD for “raw-data” model)

# training samples	Sphinx				WebAnomaly	Anagram	POSEIDON
	Attacks	FPs	RT	FPs RD			
5000	Attacks	80/80		67/80	80/80	80/80	
	FPs	162	1955		2593	90301	3478
10000	Attacks	80/80		67/80	80/80	80/80	
	FPs	59	141		587	80302	643
20000	Attacks	80/80		53/80	80/80	80/80	
	FPs	43	136		451	71029	572
50000	Attacks	80/80		47/80	80/80	80/80	
	FPs	29	127		319	61130	433

Table 9. Results for Sphinx and comparative algorithms on full requests from DS_C : we report detailed false positive rates for Sphinx (RT stands for “regular-text” models and RD for “raw-data” model)

# training samples		Sphinx		WebAnomaly	Anagram	POSEIDON
		FPs	RT FPs RD			
5000	Attacks	80/80		78/80	80/80	80/80
	FPs	36	238	607	16779	998
10000	Attacks	80/80		77/80	80/80	80/80
	FPs	24	109	515	13307	654
20000	Attacks	80/80		49/80	80/80	80/80
	FPs	10	98	459	7417	593
50000	Attacks	80/80		46/80	80/80	80/80
	FPs	3	47	338	4630	404

80 vectors, including the 20 attacks previously used to test DS_A (adapted to target the new data set).

The tests show that the presence of irregular parameters significantly influences the false positive rate of Sphinx. We need an extensive training to achieve a rate of 10 false positives per day: this is not surprising, since we observed a similar behaviour during previous tests (see [7,9]).

6 Related Work

Despite the fact that web applications have been widely developed only in the last half-decade years, the detection of web-based attacks has immediately received considerable attention.

Ingham et al. [22] use a deterministic finite automaton (DFA) to build a profile of legal HTTP requests. It works by tokenizing HTTP request parameters, and storing each token type and (optionally) its value. Pre-defined heuristic functions are used to validate and generalize well-known input values (e.g., dates, file types, IP addresses and session cookies). Each state in the DFA represents an unique token, and the DFA has a transition between any two states that were seen consecutively (from a chronological point of view) in the request. A similarity function determines if a request has to be considered anomalous. It reflects the changes (i.e., for each missed token a new transition would have to be added) that would have to be made to the DFA for it to accept the request.

Despite its effectiveness, this approach relies on predefined functions which can be used to analyse only certain (previously known) input types. Furthermore, for some parameters (e.g., blog messages) it could be difficult to find a good function to validate the content. Sphinx, on the other hand, is able to learn in an automatic way the syntax of most of parameter values and uses a content-based anomaly detector for parameters whose syntax cannot be extracted.

WebAnomaly (Kruegel et al. [10]) analyses HTTP requests and takes significant advantage of the parameter-oriented URL format common in web applications. The system applies up to nine different models at the same time to detect possible attacks, namely: attribute length and character distribution, structural inference, token finder, attribute presence and order, access frequency, inter-request time delay and invocation order. We have compared WebAnomaly to Sphinx in our benchmarks.

Jovanovic et al. [23] present a static-analysis tool (called Pixy) for web applications. The tool detects data flow vulnerabilities by checking how inputs could affect the (intended) behaviour of the web application, leading to an outflow of information. This approach requires the sources code of the web application to be available.

Finally, we should mention that Almgren et al. [24] and Almgren and Lindqvist [25] present similar systems which are based on signature-based techniques and either analyse web server logs ([24]) or are integrated inside the web server itself.

7 Conclusion

Sphinx is conceptually simple, and – as our benchmarks show – to detect attacks to web applications it performs better than competing systems. Here we want to stress that the system we have compared it to are really the best ones now available and that the set of benchmarks we have carried out (with 3 different data sets) is very extensive. Another aspect we want to stress is that Sphinx presents also a better learning curve than competitors (i.e., it needs a lower number of samples to train itself). This is very important in the practical deployment phase, when changes to the underlying application require that every now and then the system be retrained (and retraining the system requires cleaning up the training set from possible attacks, an additional operation which needs to be done – accurately – off-line).

Sphinx, instead of using solely mathematical and statistical models, takes advantage of the *regularities* of HTTP request parameters and is able to automatically generate, for most of the parameters, human-readable regular expressions (we call them “positive signatures”). This also means that the IT specialist, if needed, could easily inspect and modify/customize the signatures generated by Sphinx, thereby modifying the behaviour of the detection engine. This aspect should be seen in the light of the criticisms that is often addressed to anomaly-based systems: That they are as black-boxes which cannot be tuned by the IT specialists in ways other than modifying, e.g., the alert threshold [5]. Sphinx is – to our knowledge – the first anomaly-detection system which relies heavily on signatures which can be seen, interpreted, and customized by the IT specialists.

References

1. The MITRE Corporation: Common Vulnerabilities and Exposures database (2004), <http://cve.mitre.org>
2. Robertson, W., Vigna, G., Kruegel, C., Kemmerer, R.A.: Using generalization and characterization techniques in the anomaly-based detection of web attacks. In: NDSS 2006: Proc. of 17th ISOC Symposium on Network and Distributed Systems Security (2006)
3. Symantec Corporation: Internet Security Threat Report (2006), <http://www.symantec.com/enterprise/threat-report/index.jsp>
4. Web Application Security Consortium: Web Application Firewall Evaluation Criteria (2006), <http://www.webappsec.org/projects/wafec/>

5. Kruegel, C., Toth, T.: Using Decision Trees to Improve Signature-based Intrusion Detection. In: Vigna, G., Kruegel, C., Jonsson, E. (eds.) RAID 2003. LNCS, vol. 2820, pp. 173–191. Springer, Heidelberg (2003)
6. Axelsson, S.: The base-rate fallacy and the difficulty of intrusion detection. *ACM Trans. Inf. Syst. Secur. (TISSEC)* 3(3), 186–205 (2000)
7. Bolzoni, D., Crispo, B., Etalle, S.: ATLANTIDES: An Architecture for Alert Verification in Network Intrusion Detection Systems. In: LISA 2007: Proc. 21th Large Installation System Administration Conference, USENIX Association, pp. 141–152 (2007)
8. Balzarotti, D., Cova, M., Felmetzger, V.V., Vigna, G.: Multi-Module Vulnerability Analysis of Web-based Applications. In: CCS 2007: Proc. 14th ACM Conference on Computer and Communication Security, pp. 25–35. ACM Press, New York (2007)
9. Bolzoni, D., Zambon, E., Etalle, S., Hartel, P.: POSEIDON: a 2-tier Anomaly-based Network Intrusion Detection System. In: IWIA 2006: Proc. 4th IEEE International Workshop on Information Assurance, pp. 144–156. IEEE Computer Society Press, Los Alamitos (2006)
10. Kruegel, C., Vigna, G., Robertson, W.: A multi-model approach to the detection of web-based attacks. *Computer Networks* 48(5), 717–738 (2005)
11. Wang, K., Parekh, J.J., Stolfo, S.J.: Anagram: A Content Anomaly Detector Resistant to Mimicry Attack. In: Zamboni, D., Krügel, C. (eds.) RAID 2006. LNCS, vol. 4219, pp. 226–248. Springer, Heidelberg (2006)
12. Debar, H., Dacier, M., Wespi, A.: A Revised Taxonomy of Intrusion-Detection Systems. *Annales des Télécommunications* 55(7–8), 361–378 (2000)
13. Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., Berners-Lee, T.: Hypertext Transfer Protocol – HTTP/1.1 (1999)
14. Kruegel, C., Vigna, G.: Anomaly Detection of Web-based Attacks. In: CCS 2003: Proc. 10th ACM Conference on Computer and Communications Security, pp. 251–261 (2003)
15. Fernau, H.: Algorithms for Learning Regular Expressions. In: Jain, S., Simon, H.U., Tomita, E. (eds.) ALT 2005. LNCS (LNAI), vol. 3734, pp. 297–311. Springer, Heidelberg (2005)
16. van Trees, H.L.: Detection, Estimation and Modulation Theory. Part I: Detection, Estimation, and Linear Modulation Theory. John Wiley and Sons, Inc., Chichester (1968)
17. Lippmann, R., Haines, J.W., Fried, D.J., Korba, J., Das, K.: The 1999 DARPA off-line intrusion detection evaluation. *Computer Networks: The International Journal of Computer and Telecommunications Networking* 34(4), 579–595 (2000)
18. Mahoney, M.V., Chan, P.K.: An Analysis of the 1999 DARPA/Lincoln Laboratory Evaluation Data for Network Anomaly Detection. In: Vigna, G., Kruegel, C., Jonsson, E. (eds.) RAID 2003. LNCS, vol. 2820, pp. 220–237. Springer, Heidelberg (2003)
19. McHugh, J.: Testing Intrusion Detection Systems: a critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln Laboratory. *ACM Transactions on Information and System Security (TISSEC)* 3(4), 262–294 (2000)
20. Vigna, G., Robertson, W.K., Balzarotti, D.: Testing network-based intrusion detection signatures using mutant exploits. In: CCS 2004: Proc. 11th ACM Conference on Computer and Communications Security, pp. 21–30. ACM Press, New York (2004)

21. Ingham, K.L., Inoue, H.: Comparing Anomaly Detection Techniques for HTTP. In: Kruegel, C., Lippmann, R., Clark, A. (eds.) RAID 2007. LNCS, vol. 4637, pp. 42–62. Springer, Heidelberg (2007)
22. Ingham, K.L., Somayaji, A., Burge, J., Forrest, S.: Learning DFA representations of HTTP for protecting web applications. *Computer Networks: The International Journal of Computer and Telecommunications Networking* 51(5), 1239–1255 (2007)
23. Jovanovic, N., Kruegel, C., Kirda, E.: Pixy: A Static Analysis Tool for Detecting Web Application Vulnerabilities. In: S&P 2006: Proc. 26th IEEE Symposium on Security and Privacy, pp. 258–263. IEEE Computer Society, Los Alamitos (2006)
24. Almgren, M., Debar, H., Dacier, M.: A lightweight tool for detecting web server attacks. In: NDSS 2000: Proc. of 11th ISOC Symposium on Network and Distributed Systems Security (2000)
25. Almgren, M., Lindqvist, U.: Application-integrated data collection for security monitoring. In: Lee, W., Mé, L., Wespi, A. (eds.) RAID 2001. LNCS, vol. 2212, pp. 22–36. Springer, Heidelberg (2001)

Principal Components of Port-Address Matrices in Port-Scan Analysis

Hiroaki Kikuchi¹, Naoya Fukuno¹, Masato Terada², and Norihisa Doi³

¹ School of Information Technology, Tokai University, 1117 Kitakaname, Hiratsuka, Kangawa, 259-1292, Japan

² Hitachi, Ltd. Hitachi Incident Response Team (HIRT), 890 Kashimada, Kawasaki, Kanagawa, 212-8567, Japan

³ Dept. of Info. and System Engineering, Faculty of Science and Engineering, Chuo University, 1-13-27 Kasuga, Bunkyo, Tokyo, 112-8551, Japan

Abstract. There are many studies aiming at using port-scan traffic data for the fast and accurate detection of rapidly spreading worms. This paper proposes two new methods for reducing the traffic data to a simplified form comprising significant components of smaller dimensionality. (1) Dimension reduction via Term Frequency – Inverse Document Frequency (TF-IDF) values, a technique used in information retrieval, is used to choose significant ports and addresses in terms of their “importance” for classification. (2) Dimension reduction via Principal Component Analysis (PCA), widely used as a tool in exploratory data analysis, enables estimation of how uniformly the sensors are distributed over the reduced coordinate system. PCA gives a scatter plot for the sensors, which helps to detect abnormal behavior in both the source address space and the destination port space. In addition to our proposals, we report on experiments that use the Internet Scan Data Acquisition System (ISDAS) distributed observation data from the Japan Computer Emergency Response Team (JPCERT).

1 Introduction

The Internet backbone contains port-scanning packets that are routinely generated by malicious hosts, e.g., worms and botnets, looking for vulnerable targets. These attempts are usually made on a specific destination port for which services with known vulnerable software are available. Ports 135, 138, and 445 are frequently scanned. There is also malicious software that uses particular ports to provide a “back door” to companies. The number of packets targeting the destination port for the back door is not large, but the statistics for these ports are sometimes helpful for detecting a new type of attack, a coordinated attack made by a botnet, or targeted attacks.

Related Works. There have been several attempts to identify attacks via changes in the traffic data observed by sensors distributed across the Internet. A honeypot is a semipassive sensor that pretends to be a vulnerable host in faked communications with intruders or worms [10]. Some sensors are *passive*

in the sense that capture packets are sent to an unused IP address without any interaction. The Network Telescope [8], Internet Storm Center [11], DShield [12], and ISDAS [3] are examples of passive sensors.

There are many studies aiming at using port-scan traffic data for the fast and accurate detection of rapidly spreading worms. Kumar uses the characteristics of the pseudorandom number generation algorithm used in the Witty worm to reconstruct the spread of infected hosts [13]. Ishiguro et al. propose the Wavelet coefficients used as metrics for anomaly detection [14]. Jung et al. present an algorithm to detect malicious packets, called Sequential Hypothesis Testing based on Threshold of Random Walk (TRW) [2]. Dunlop et al. present a simple statistical scheme called the Simple Worm Detection Scheme (SWorD) [15], where the number of connection attempts is tested with threshold values.

The accuracy of detection, however, depends on an assumption that *the set of sensors is distributed uniformly over the address space*. Because the installation of sensors is limited to unused address blocks, it is not easy to ensure uniform sensor distribution. Any distortion of the address distribution could cause false detection and a misdetection, and therefore uniformity of sensor distribution is one of the issues we should consider. Nevertheless, it is not trivial to evaluate a distribution of sensors in terms of its uniformity because the traffic data comprise ports and addresses that are correlated in high-dimensional domains.

Contribution. This paper proposes a new method for reducing the traffic data to a simplified form comprising significant components of smaller dimensionality. Our contribution is twofold:

1. **Dimension reduction via TF-IDF values.** We apply a technique used in information retrieval and text mining, called the *TF-IDF weight*, given that there are similarities between our problem and the information retrieval problem. Both deal with high-dimensional data, defined sets of words (ports or addresses), and documents (sensors). Both sets are discrete. Most elements are empty.
2. **Dimension reduction via PCA.** Our second proposal is based on an orthogonal linear transformation, which is widely used as a tool in exploratory data analysis. PCA enables estimation of how uniformly the sensors are distributed over the reduced coordinate system. The results of PCA give a scatter plot of sensors, which helps to detect abnormal behavior in both the source address space and the destination port space.

We give experimental results for our method using the JPCERT/ISDAS distributed observation data.

2 Proposed Methods

2.1 Preliminary

We give the fundamental definitions necessary for discussion about the characteristics of worms.

Definition 1. A scanner is a host that performs port-scans on other hosts, looking for targets to be attacked.

A sensor is a host that can passively observe all packets sent from scanners. Let S be a set of sensors $\{s_1, s_2, \dots, s_n\}$, where n is the number of sensors.

Typically, a scanner is a host that has some vulnerability and thereby is controlled by malicious code such as a worm or virus. Some scanners may be human operated, but we do not distinguish between malicious codes and malicious operators. Sensors have always-on static IP addresses, i.e., we will omit the dynamic behavior effects of address assignments provided via Dynamic Host Control Protocol (DHCP) or Network Address Translation (NAT).

An IP packet, referred to as a “datagram”, specifies a *source address* and a *destination address*, in conjunction with a *source port number* and a *destination port number*, specified in the TCP header.

Definition 2. Let P be a set of ports $\{p_1, p_2, \dots, p_m\}$, where m is the number of possible port numbers. Let A be a set of addresses $\{a_1, a_2, \dots, a_\ell\}$, where ℓ is the number of all IP addresses.

In IP version 4, possible values for m and ℓ are 2^{16} and 2^{32} , respectively. Because not all address blocks are assigned as yet, the numbers of addresses and ports observed by the set of sensors are typically limited, i.e., $m \ll 2^{16}$, $\ell \ll 2^{32}$. To handle reduced address set sizes, we distinguish addresses with respect to the two highest octets. For example, address $a = 221.10$ contains the range of addresses from 221.10.0.0 through 221.10.255.255.

Let c_{ij} be the number of packets whose destination port is p_j that are captured by sensor s_i in duration T . Let b_{ik} be the number of packets that are observed by sensor s_i and sent from source address a_k . An *observation* of sensor s_i is characterized by two vectors

$$\mathbf{c}_i = \begin{pmatrix} c_{i1} \\ \vdots \\ c_{im} \end{pmatrix} \quad \text{and} \quad \mathbf{b}_i = \begin{pmatrix} b_{i1} \\ \vdots \\ b_{im} \end{pmatrix},$$

which are referred to as the *port vector* and the *address vector*. All packets observed by n independent sensors are characterized by the $n \times m$ matrix $\mathbf{C}$ and $\ell \times n$ matrix $\mathbf{B}$ specified by $\mathbf{C} = (\mathbf{c}_1 \cdots \mathbf{c}_n)$ and $\mathbf{B} = (\mathbf{b}_1 \cdots \mathbf{b}_n)$. Matrices $\mathbf{B}$ and $\mathbf{C}$ will usually contain many unexpected packets caused by possible misconfigurations or by a small number of unusual worms, which we wish to ignore to reduce the quantity of observation data.

2.2 Reduced Matrix Via TF-IDF Values

Observation by a limited number of sensors shows an incomplete and small fragment of the Internet traffic of unauthorized packets. Therefore, the observation matrices P and A are “thinly populated”, i.e., most elements are empty. To

reduce the dimension of the matrices to a subset of the matrix comprising significant elements from the given P and A , we try to apply a technique used in information retrieval and text mining, called the *TF-IDF weight*.

The TF-IDF weight gives the degree of importance of a word in a collection of documents. The importance increases if the word is frequently used in the set of documents (TF) but decreases if it is used by too many documents (IDF). The *term frequency* in the given set of documents is the number of times the term appears in the document sets. In our study, we use the term frequency to evaluate how important a specific destination port p_j is to a given set of packets $C = \{c_1, \dots, c_n\}$ observed by n sensors, and defined as the average number of packets for the port p_j , i.e.,

$$TF(p_j) = \frac{1}{n} \sum_{i=1}^n c_{ij}.$$

The *document frequency* of destination port p_j is defined by

$$DF(p_j) = |\{c_i \in C | c_{ij} > 0, i \in \{1, \dots, n\}\}|,$$

which gives the degree of “uselessness”, because a destination port with the highest $DF(p_j) \approx n$ implies that the port is always specified by any sensor, and therefore we would regard the port p_j as unable to distinguish between sensors. By taking the logarithm of the inverse of the document frequency, we obtain a *TF-IDF* for a given port p_j as

$$TF-IDF(p_j) = TF(p_j) \cdot \log_2\left(\frac{n}{DF(p_j)} + 1\right),$$

where the constant 1 is used to avoid the *TF-IDF* of a port with $DF(p_j) = n$ from being zero.

Similarly to the destination port, we define the *TF-IDF* weight of source address a_k as $TF-IDF(a_k) = TF(a_k) \cdot \log_2\left(\frac{n}{DF(a_k)} + 1\right)$, where

$$TF(a_k) = \frac{1}{n} \sum_{i=1}^n c_{ik},$$

$$DF(a_k) = |\{c_i \in B | b_{ik} > 0, i \in \{1, \dots, n\}\}|.$$

Note that a high value for *TF-IDF* is reached by a high term (port/address) frequency and a low document (sensor) frequency for the port among the whole set of packets, thereby working to filter out common ports. Based on the order of *TF-IDF* values, we can choose the most important destination ports within the 2^{16} possible values, from the perspective of frequencies of sets of packets.

2.3 Reduced Matrix Via PCA

PCA is a well-known technique, which is used to reduce multidimensional data to a smaller set that contributes most to its variance by keeping lower-order principal components and ignoring higher-order components.

Our goal is to transform a given matrix $\mathbf{C} = (\mathbf{c}_1 \cdots \mathbf{c}_m)$ of m dimensions (observations) to an alternative matrix $\mathbf{Y}$ of smaller dimensionality as follows.

Given a matrix of packets

$$\mathbf{C} = \begin{pmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & \ddots & \vdots \\ c_{m1} & \cdots & c_{mn} \end{pmatrix},$$

where c_{ij} is the number of packets such that the destination port is p_j , captured by sensor s_i , we subtract the mean for every port to obtain $\mathbf{C}' = (\mathbf{c}'_1 \cdots \mathbf{c}'_m)$, where

$$\mathbf{c}'_i = \begin{pmatrix} c_{i1} - \overline{c_1} \\ \vdots \\ c_{im} - \overline{c_m} \end{pmatrix}$$

and $\overline{c_j}$ is the average number of packets at the j -th port, i.e., $\overline{c_j} = 1/n \sum_{i=1}^n c_{ij}$.

PCA transforms $\mathbf{C}'$ to $\mathbf{Y} = (\mathbf{y}_1, \dots, \mathbf{y}_m)$ such that, for $i = 1, \dots, n$,

$$\mathbf{c}'_i = U \mathbf{y}_i = y_{i1} \mathbf{u}_1 + \cdots + y_{im} \mathbf{u}_m,$$

where $\mathbf{u}_1, \dots, \mathbf{u}_m$ are m unit vectors, called the *principal component basis*, which minimizes the mean square error of the data approximation. The principal component basis is given by a matrix U comprising the eigenvectors $\mathbf{u}_1, \dots, \mathbf{u}_n$, sorted in order of decreasing eigenvalue $\lambda_1 > \cdots > \lambda_n$, of the *covariance matrix* that is defined by

$$V = \frac{1}{m} \sum_{i=1}^m \mathbf{c}_i \mathbf{c}_i^\top.$$

From a fundamental property of eigenvectors, the elements of the principal component basis are orthogonal, i.e., $\mathbf{u}_i \cdot \mathbf{u}_j = 0$ for any $i \neq j \in \{1, \dots, m\}$. This gives the matrix $\mathbf{Y} = (\mathbf{y}_1 \cdots \mathbf{y}_m)$, where

$$\mathbf{y}_i = U^\top \mathbf{c}'_i = (y_{i1} \cdots y_{im})^\top, \quad (1)$$

which maximizes the variance for each element and gives a zero average, for $i = 1, \dots, m$.

The first principal component, namely y_{i1} , contains the most significant aspect of the observation data, and the second component y_{i2} contributes the second most significant effect on its variance. These “lower-frequency” components give a first impression of the port-scanning pattern, even though the “higher-frequency” ones are ignored.

We apply the PCA transform not only to the matrix $\mathbf{C}$ defined over the sensor and port spaces ($n \times m$) but also to the matrix $\mathbf{B}$ of the sensor and the address spaces ($n \times m$), and to the transposed matrices $\mathbf{C}^\top$ and $\mathbf{B}^\top$. We use the notation $\mathbf{u}(\mathbf{C})$ and $\mathbf{u}(\mathbf{B})$ if we need to distinguish between matrices $\mathbf{C}$ and $\mathbf{B}$.

3 Analysis

We apply the proposed methods to the dataset of packets observed by sensors distributed over the Internet.

3.1 Experimental Data

ISDAS Distributed Sensors. ISDAS is a distributed set of sensors [3], under the operation of the JPCERT Coordination Center (JPCERT/CC), that can estimate the scale of a current malicious event and its performance.

Table 1 shows the statistics for $m = 30$ sensors from April 1, 2006 through March 31, 2007. The most frequently scanned sensor is s_1 with about 451,000 counts, which is 70 times that for the least frequently scanned sensor s_{15} . In this sense, the destination addresses to scan are not uniformly distributed.

Institutional Access Sensors. Table 2 shows a set of sensors installed in institutional LANs and some commercial Internet Service Providers (ISPs). The bandwidth and the method of address assignment are listed for each of the sensors.

3.2 TF-IDF Analysis

We show the results of TF-IDF analysis in Table 3, where the top 20 ports and source addresses (two octets) are listed in order of corresponding TF-IDF values. In the table, destinations 135, 445, ICMP, 139, and 30 are known as frequently scanned ports and are therefore listed at the top, while destination ports 23310 and 631 are listed because of their low DFs, implying their “importance” in classifying sensors. On the other hand, we note that the top 20 source addresses have higher DFs. For example, the third address 203.205 has $DF = 16$, i.e., the address is found by 16 of the 30 sensors.

Table 1. Statistics for ISDAS distributed sensors

	sensor	count	unique $h(x)$	$\Delta h(x)$ [/day]
Average	—	146000	37820	104.9
Standard deviation	—	134900	29310	82.72
Max	s_1	450671	98840	270.79
Min	s_{15}	6475	1539	4.22

Table 2. Specification of sensors from Nov. 30, 2006 through Jan. 12, 2007

	s_{101}	s_{102}	s_{103}	s_{104}	s_{105}	s_{106}	s_{107}	s_{108}
Subnet class	B	C	B				C	C
Bandwidth [Mbps]	100	8	100				12	8
Type	inst. 1	ISP 1	institutional 2				ISP 2	ISP 3
IP assignment	static	dynamic	static				dynamic	dynamic

Table 3. Top 20 ports and addresses, ordered by TF-IDF value (ISDAS)

p_j	TF(p_j)	DF(p_j)	TF-IDF(p_j)	a_k	TF(a_k)	DF(a_k)	TF-IDF(a_k)
135	19499.73	29	20160.80	219.111	4668.60	23	5909.06
445	15326.47	27	16941.27	58.93	4490.57	24	5492.61
ICMP	6537.40	29	6759.03	203.205	2939.13	16	4786.70
139	5778.23	27	6387.03	222.148	3055.33	25	3612.39
80	3865.90	30	3865.90	61.252	2159.63	21	2929.92
1026	3705.97	30	3705.97	61.193	1994.30	21	2705.62
23310	789.57	2	2927.75	61.205	1858.40	21	2521.24
1433	2423.33	30	2423.33	220.221	2035.27	26	2326.52
631	552.17	3	1823.58	61.199	1810.27	25	2140.32
1027	1268.73	30	1268.73	222.13	1504.80	20	2114.94
1434	1130.90	27	1250.05	219.2	561.77	12	1076.51
137	989.53	26	1131.14	218.255	676.33	17	1060.48
4899	1007.90	30	1007.90	222.159	774.90	23	980.79
1025	713.13	29	737.31	220.109	722.17	22	946.15
4795	150.67	1	663.11	221.208	861.07	29	890.26
22	470.47	30	470.47	219.114	750.70	25	887.57
32656	119.17	2	441.88	203.174	408.50	12	782.80
12592	92.47	1	406.96	221.188	600.40	25	709.87
113	174.57	8	405.30	221.16	245.23	6	639.92
1352	108.37	2	401.83	219.165	533.77	25	631.08

Filtering out the less important ports and addresses in terms of TF-IDF values gives reduced matrices of 20 dimensions, which are small enough for the PCA transform to be applied.

3.3 PCA

We have performed PCA for each of the matrices $\mathbf{C}$, $\mathbf{B}$, $\mathbf{C}^\top$, and $\mathbf{B}^\top$, namely the ports-and-sensors, addresses-and-sensors, sensors-and-ports, and sensors-and-ports matrices, respectively.

Principal Component Basis. Table 4 shows the experimental results for the first two orthogonal vectors of principal component basis $\mathbf{u}_1(C), \mathbf{u}_2(C), \dots$ for the ports-and-sensors matrix $\mathbf{C}$ and basis $\mathbf{u}_1(B), \mathbf{u}_2(B), \dots$ for the addresses-and-sensors matrix $\mathbf{B}$. The elements indicated in boldface are the dominant elements of each basis. For example, the ports 445 and 135, having the largest (in absolute value) elements -0.37 and -0.36 in $\mathbf{u}_1(C)$, are the primary elements determining the value of the first principal component y_1 . Informally, we regard the first coordinate as the *degree of well-scanned ports* because 445 and 135 are likely to be vulnerable. In the same way, the second principal component basis $\mathbf{u}_2(C)$ indicates attacks on web servers ($p = 80$) and ICMP, and we may therefore refer to y_2 as the *degree of http attack*. The second principal component has about half the effect of the projected values because eigenvalue λ_1 is almost double λ_2 .

The addresses-and-sensors matrix $\mathbf{B}$ provides the principal component vectors indicating the degree of importance in source address set A , as shown in

Table 4. The first two vectors of principal component basis $\mathbf{u}_1(C), \mathbf{u}_2(C), \dots$ for port matrix $\mathbf{C}$ and basis $\mathbf{u}_1(B), \mathbf{u}_2(B), \dots$ for address matrix $\mathbf{B}$

p_j	$\mathbf{u}_1(C)$	$\mathbf{u}_2(C)$	a_k	$\mathbf{u}_1(B)$	$\mathbf{u}_2(B)$
445	-0.37	0.01	221.188	-0.54	0.20
135	-0.36	0.01	222.148	-0.54	0.20
137	-0.34	-0.07	219.114	-0.53	0.20
1433	-0.33	0.17	219.165	-0.28	-0.52
4899	-0.30	0.27	221.208	-0.17	-0.41
1434	-0.30	0.16	220.221	-0.14	-0.59
1026	-0.28	-0.27	58.93	-0.01	-0.20
1025	-0.28	-0.01	222.13	0.00	-0.09
1027	-0.25	-0.28	222.159	0.01	-0.06
22	-0.23	0.08	61.199	0.03	0.03
32656	-0.13	-0.27	219.111	0.03	0.02
12592	-0.13	-0.27	220.109	0.03	0.03
139	-0.10	0.18	61.205	0.03	0.03
23310	-0.09	-0.03	221.16	0.03	0.03
80	-0.02	0.45	61.252	0.03	0.04
ICMP	-0.02	0.44	203.174	0.03	0.04
113	0.00	0.25	61.193	0.03	0.04
4795	0.00	0.25	203.205	0.04	0.04
631	0.05	-0.04	219.2	0.06	0.14
1352	0.09	-0.08	218.255	0.06	0.14
eigenvalue λ_i	6.19	2.49	eigenvalue λ_i	3.16	2.29

Table 5, as well as in matrix $\mathbf{C}$. In these results, we find that $\mathbf{u}_1(B)$ has dominant addresses that are disjoint from those of $\mathbf{u}_2(B)$.

Scatter Plot for Sensors in Reduced Coordinate System. In Fig. 1, we illustrate how the observed data are projected into the new coordinate system defined by the first two principal components y_1 and y_2 as the X-axis and Y-axis of the scatter plot for the sensors. The sensors $s_{101}, \dots, s_{108}$, specified in Table 2, are indicated at the coordinate (y_{i1}, y_{i2}) , computed by Eq. (1). The plot shows that there are three clusters: (1) sensors in institutional LANs, $\{s_{101}, s_{103}, \dots, s_{106}\}$, (2) commercial ISPs, $\{s_{107}, s_{108}\}$, and (3) ISP 3, $\{s_{102}\}$. ISP 3 uses a cable modem, whereas the access network for ISP 1 and 2 is ADSL. We see that the two-dimensional principal components successfully retain the characteristics of each cluster of sensors. In other words, the 20-dimensional data for the ports are reduced to just two dimensions.

The resulting clusters depend on the given matrix. The same set of sensors are classified differently into the three clusters shown in Fig. 2 if we begin with the matrix $\mathbf{B}$. It is interesting that sensors s_{107} and s_{108} are distributed quite differently, even though they were close in Fig. 1.

Analysis from Several Perspectives. PCA can be applied to arbitrary matrices prepared from different perspectives. If we are interested in the independence of sensors, PCA enables us to show how uniformly the set of sensors is

Table 5. The principal component basis $\mathbf{u}_1(\mathbf{C}^\top), \mathbf{u}_2(\mathbf{C}^\top), \dots$ for sensor-port matrix $\mathbf{C}^\top$ and basis $\mathbf{u}_1(\mathbf{B}^\top), \mathbf{u}_2(\mathbf{B}^\top), \dots$ for sensor-address matrix $\mathbf{B}^\top$

s_i	$\mathbf{u}_1(\mathbf{C}^\top)$	$\mathbf{u}_2(\mathbf{C}^\top)$	s_i	$\mathbf{u}_1(\mathbf{B}^\top)$	$\mathbf{u}_2(\mathbf{B}^\top)$
s_7	-0.04	0.34	s_{12}	-0.34	0.16
s_{20}	-0.03	0.30	s_{18}	-0.34	0.18
s_8	-0.03	0.42	s_6	-0.34	0.18
s_{22}	-0.01	0.42	s_{20}	-0.34	0.02
s_{26}	-0.01	0.25	s_{22}	-0.34	0.18
s_{30}	0.03	-0.12	s_{13}	-0.32	0.21
s_{28}	0.05	-0.19	s_{17}	-0.32	0.01
s_{12}	0.06	0.37	s_{29}	-0.28	-0.20
s_{15}	0.06	-0.16	s_{28}	-0.21	-0.35
s_{29}	0.07	-0.22	s_{27}	-0.20	-0.11
s_{25}	0.17	-0.01	s_4	-0.17	-0.27
s_{23}	0.18	-0.08	s_{23}	-0.10	-0.33
s_6	0.18	0.24	s_1	-0.05	-0.30
s_{24}	0.19	0.04	s_3	-0.05	-0.21
s_5	0.21	0.02	s_5	-0.03	-0.03
s_4	0.22	0.08	s_{11}	-0.01	0.03
s_{17}	0.22	-0.12	s_{10}	0.00	-0.15
s_{16}	0.22	-0.09	s_{14}	0.01	-0.08
s_{21}	0.22	-0.02	s_{26}	0.01	-0.05
s_{27}	0.23	-0.06	s_9	0.01	0.07
s_{13}	0.23	0.03	s_2	0.01	0.06
s_{14}	0.24	-0.02	s_{15}	0.02	-0.11
s_{18}	0.24	0.10	s_{30}	0.02	-0.07
s_{11}	0.24	0.07	s_{16}	0.03	-0.00
s_{19}	0.24	0.01	s_{19}	0.03	0.12
s_3	0.24	0.05	s_{24}	0.04	0.15
s_1	0.24	0.03	s_8	0.04	0.13
s_2	0.24	0.01	s_{25}	0.04	0.32
s_{10}	0.24	-0.02	s_{21}	0.06	0.31
s_9	0.24	0.03	s_7	0.07	0.18
eigenvalue λ_i	16.64	3.73	eigenvalue λ_i	7.81	2.66

distributed over the reduced coordinate system. If we wish to identify abnormal behavior of source addresses, PCA with respect to a sensors-and-address matrix $\mathbf{B}^\top$ gives a scatter plot of addresses in which particular addresses stand out from the cluster of the standard behaviors.

For these purposes, we show the experimental results of ISDAS observation data, in Figs. 3, 4, 5, and 6, corresponding to matrices $\mathbf{C}$, $\mathbf{B}$, $\mathbf{C}^\top$, and $\mathbf{B}^\top$, respectively.

The set of ISDAS sensors is independently distributed in Fig. 3, but the distribution is skewed by some irregular sensors in Fig. 4, where the horizontal axis has more elements with source addresses in class C. As a consequence, the distribution of ISDAS sensors may be distorted in terms of differences between source addresses.

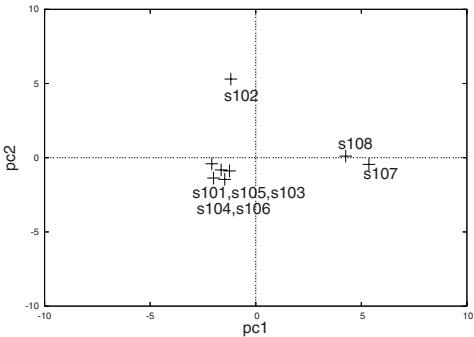


Fig. 1. Scatter plot for institutional sensors of a dataset with $n = 8$, indicating the coefficients of the first (X-axis) and second (Y-axis) principal components, $y_1(C)$ and $y_2(C)$, respectively

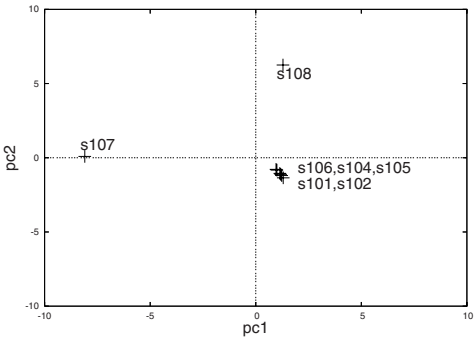


Fig. 2. Scatter plot for institutional sensors of a dataset with $n = 8$, indicating the coefficients of the first (X-axis) and the second (Y-axis) principal components, $y_1(B)$ and $y_2(B)$, respectively

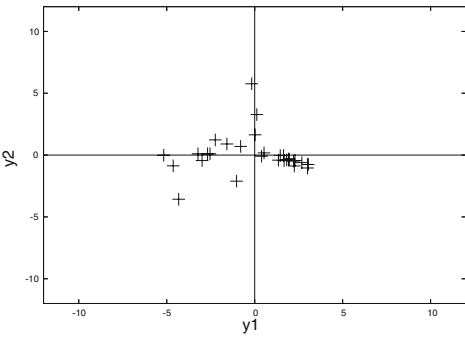


Fig. 3. Scatter plot for *ISDAS* sensors S of a dataset with $n = 30$, displaying the coefficients of the first two principal components in terms of *ports*

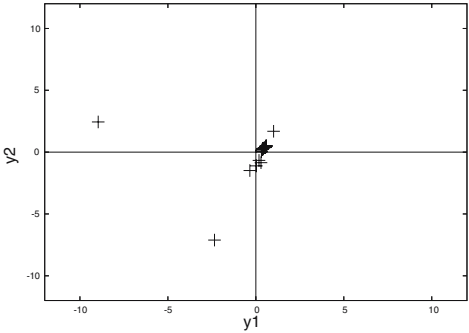


Fig. 4. Scatter plot for *ISDAS sensors S* of a dataset with $n = 30$, displaying the coefficients of the first two principal components in terms of *addresses*

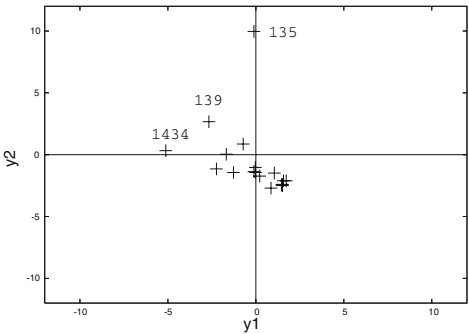


Fig. 5. Scatter plot for destination ports *P* displaying the coefficients of the first two principal components in terms of *ISDAS sensors*

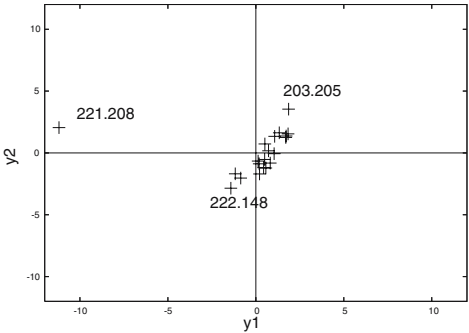


Fig. 6. Scatter plot for source addresses *A* displaying the coefficients of the first two principal components in terms of *ISDAS sensors*

In Fig. 5, the set of ports P is centrally distributed, with exceptions such as ports 135 and 139 at the top of the plot.

In Fig. 6, there are two clusters: a singleton $\{221.208\}$ and the remainder. Any botnet-like behavior can be seen from the clustering in the plot. A scatter plot of the principal components provides a useful viewgraph by which any small change is perceptible by human operators.

4 Conclusion

We have proposed a new analysis method for the distributed observation of packets with high-dimensional attributes such as port numbers (2^{16}) and IP addresses (2^{32}). Our methods are based on the TF-IDF value mainly developed for information retrieval, and on PCA. Experimental results demonstrate that both methods correctly reduce a given high-dimension dataset to smaller dimensionality, by at least a factor of two. The principal components of port numbers, in terms of distinguishable sensors, include 445, 135, 137, 1433, 4899, 1434, 80, and ICMP, which enable any sensors to be classified. The source addresses 221.188, 222.148, 219.114, 219.165, 221.208 and 220.221 are specified as dominant on a principal component basis.

Future studies will include the stability of the basis, an accuracy evaluation for a few components, and an application of the orthogonal basis to intrusion detection.

Acknowledgments

We thank Mr. Taichi Sugiyama of Chuo University for the discussion, and the JPCERT/CC for the ISDAS distributed data.

References

1. Terada, M., Takada, S., Doi, N.: Network Worm Analysis System. IPSJ Journal 46(8), 2014–2024 (2005) (in Japanese)
2. Jung, J., Paxson, V., Berger, A.W., Balakrishnan, H.: Fast Portscan Detection Using Sequential Hypothesis Testing. In: Proc. of the 2004 IEEE Symposium on Security and Privacy (S&P 2004) (2004)
3. JPCERT/CC, ISDAS, <http://www.jpccert.or.jp/isdas>
4. Number of Hosts advertised in the DNS, Internet Domain Survey (July 2005), <http://www.isc.org/ops/reports/2005-07>
5. Moore, D., Paxson, V., Savage, S., Shannon, C., Staniford, S., Weaver, N.: Inside the Slammer Worm. IEEE Security & Privacy, 33–39 (July 2003)
6. Shannon, C., Moore, D.: The Spread of the Witty Worm. IEEE Security & Privacy 2(4), 46–50 (2004)
7. Changchun Zou, C., Gong, W., Towsley, D.: Code Red Worm Propagation Modeling and Analysis. In: ACM CCS 2002 (November 2002)
8. Moore, D., Shannon, C., Voelker, G., Savage, S.: Network Telescopes: Technical Report, Cooperative Association for Internet Data Analysis (CAIDA) (July 2004)

9. Kumar, A., Paxson, V., Weaver, N.: Exploiting Underlying Structure for Detailed Reconstruction of an Internet-scale Event. In: ACM Internet Measurement Conference (IMC 2005), pp. 351–364 (2005)
10. The Distributed Honeypot Project: Tools for Honeynets, <http://www.lucidic.net>
11. SANS Institute: Internet Storm Center, <http://isc.sans.org>
12. DShield.org, Distributed Intrusion Detection System, <http://www.dshield.org>
13. Kumar, A., Paxson, V., Weaver, N.: Exploiting Underlying Structure for Detailed Reconstruction of an Internet-scale Event. In: ACM Internet Measurement Conference (2005)
14. Ishiguro, M., Suzuki, H., Murase, I., Shinoda, Y.: Internet Threat Analysis Methods Based on Spatial and Temporal Features. *IPSJ Journal* 48(9), 3148–3162 (2007)
15. Dunlop, M., Gates, C., Wong, C., Wang, C.: SWorD – A Simple Worm Detection Scheme. In: Meersman, R., Tari, Z. (eds.) OTM 2007, Part II. LNCS, vol. 4804, pp. 1752–1769. Springer, Heidelberg (2007)

A Novel Worm Detection Model Based on Host Packet Behavior Ranking

Fengtao Xiao¹, HuaPing Hu^{1,2}, Bo Liu¹, and Xin Chen¹

¹ School of Computer Science, National University of Defense Technology,
Chang Sha, 410073

myfri2001@126.com,
boliu615@yahoo.com.cn, cx917@21cn.com

² The 61070 Army Fu Zhou, Fu Jian, 350003 China
howardnudt@yahoo.com.cn

Abstract. Traditional behavior-based worm detection can't eliminate the influence of the worm-like P2P traffic effectively, as well as detect slow worms. To try to address these problems, this paper first presents a user habit model to describe the factors which influent the generation of network traffic, then a design of HPBRWD (Host Packet Behavior Ranking Based Worm detection) and some key issues about it are introduced. This paper has three contributions to the worm detection: 1) presenting a hierarchical user habit model; 2) using normal software and time profile to eliminate the worm-like P2P traffic and accelerate the detection of worms; 3) presenting HPBRWD to effectively detect worms. Experiments results show that HPBRWD is effective to detect worms.

Keywords: worm detection, behavior based worm detection, user habit model.

1 Introduction

Computer worms have become a serious threat to the Internet in recent years. Many researches [1] [2] have found that well designed worms can infect the whole Internet in a few minutes. Now, there exist two kinds of technologies on detecting worms: signature-based technology and behavior-based technology. Signature-based worm detection (SBWD) [3-6] can detect worms in real time, but it is a kind of worm-specific and passive technology. Although widely used in commercial AV software, signature-based worm detection cannot deal with the new coming and polymorphic or metamorphic worms effectively. Behavior-based worm detection (BBWD) [7-14] in itself is a kind of statistics method, so it is less effective in real time than SBWD, but on the other hand, BBWD is independent of packet content so that it can exceed signature-based in the ability of detecting unknown worms and polymorphic and metamorphic worms. BBWD has also its drawbacks. One of them is that it is difficult to distinguish between normal traffic and abnormal traffic. For example, in recent years, P2P software has been widely used. Our previous work [17] has showed that the widely used P2P software nowadays has the worm-like traffic behavior more or less.

In our opinion, the difficulties of detecting worms lie in four points: 1) propagation speed of worms is getting much faster so that the detection time left become less; 2) the application of polymorphic or metamorphic technology makes the

signature-based worm detection less effective or completely fail; 3) the emergence of worm-like traffic such as P2P traffic makes the network-behavior based worm detection influenced with high false positives; 4) many work on BBWD is effective on fast spreading worms, but tends to miss the detection of slow worms or has high false positives.

In this paper, we mainly focus on BBWD. To try to address the four points mentioned above, a worm detection system based on host packet behavior ranking (HPBRWD) is presented. Its contributions to the four difficulties above lie in four points: 1) Using behavior based method to try to solve the second problem above; 2) Using normal software and time profile generated by HPBRWD to eliminated the influence of worm-like P2P traffic; 3) Using normal software profile and worm behavior profile to reduce the detection time; 4) detecting slow worms through the cumulative effect of HPBRWD.

The remainder of this paper is organized as follows: section 2 introduces the user habit model on Internet access; section3 overviews the design of HPBRWD; section4 discuss several key design issues and our solutions; section5 describe our prototype implementation and evaluations; section6 reviews related work. Finally, section7 makes concluding remarks with future work.

2 User Habit Model on Internet Access

Work [7] [8] have presented a definition for user habit on Internet access. According to that definition, user habit is influenced by user hobbies, characters and limitations of using internet. The definition is good, but it is lack in further description, so some important information is missed, such as habitual software. In our opinion, user habit is influenced with multi-level factors. Fig.1 lists the factors which influence the user habit.

In this model, user habit is described at three layers: user representation layer, use representation layer and network representation layer. To understand this classification, we can try to answer these questions: 1) what are the motivations for a user to access the internet? 2) Given these motivations, a user must find a way to satisfy these motivations under certain limitations. So how can he accomplish this? 3) Information is located in the internet, so what on earth is the representation of two layers above on the network packets?

We can see later that the three layers are just designed to answer the three questions above. At the same time, we can also see that each layer can separately describe the user habit, but they are in different levels.

User representation layer: this layer tries to answer the first question, that is to say, to describe what information or knowledge a user wants to acquire. User's characters, interests or hobbies, unexpected factors and other factors are included in this layer. Here other factors mean the customary access because of group behavior etc, for example, if you are impressionable, then once your colleagues tell you something interesting in the Internet, you will perhaps be attracted to visit it right now.

Use representation layer: this layer is used to answer the second question, that is to say, to describe what tools we are using and what limitations we will have to get the information. From fig.1 we can see there are two elements we shall pay attention to: Time and Software:

1) Time means that the time limitation to access the Internet. It includes starting time limitation and ending time limitation. Detail to say, this limitation has two meanings:

a) The time slots which can be used to access the internet. For example, a user in a demanding corporation can only use the Internet during non-working time which is the time limitation for this user. In such a case, the time limitation always has fixed start-time or end-time or both;

b) Unexpected changes of time limitation in length or start-time and end-time. This case is usually the result of user representation layer. For example, when a world-cup final is held, football fans will tend to spend more time using the Internet and browsing relative websites for further information about football. In such a case, the total time spent on Internet, the start-time or end-time of using Internet are always unexpected, but on the user representation layer side, the unexpected changes are in fact the result of user's interest on football.

2) Software means that the tools used to acquire the information needed. Different choosing of software will directly affects the network packets generated. For example, a user wants to watch world-cup final through the Internet webcast. There exist two kinds of software at least. One is the direct webcast on the official website or some web portals, such as www.sina.com; the other is using P2P software, such as ppstream or pplive etc. We can find that the two methods are greatly different in the destination IP address selection and the traffic of communication.

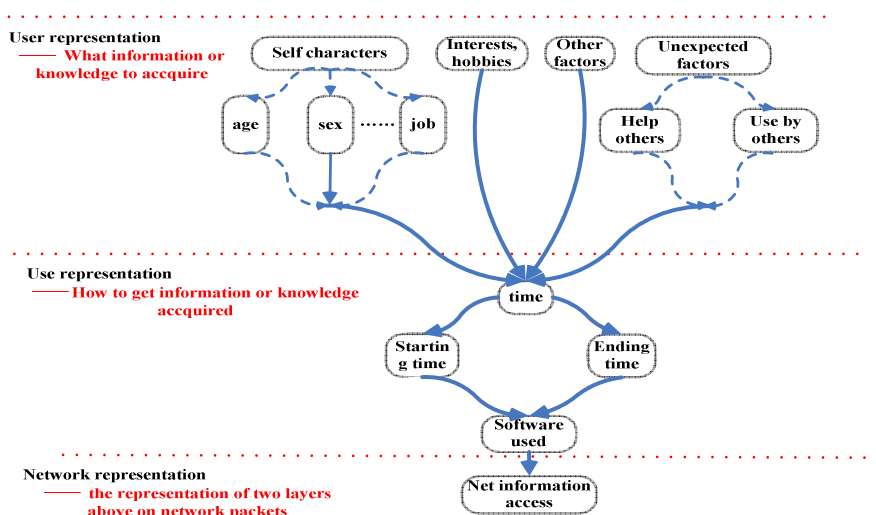


Fig. 1. User habit model

Network representation layer: this layer is used to answer the third question, that is to say, to describe the network packets created during a user accesses internet. These packets can be seen as the results of the two higher layers.

From this model we can see that to distinguish normal user network behavior and worm behavior, we must pay attention to both the use representation layer and network representation layer. But unfortunately, traditional behavior-based worm detection

only values at network representation layer. In fact, network representation layer is the least stable compared with the two other layers. But to tell the truth, network representation is the easiest to describe and use on detecting worms.

In this paper, we will try to present a novel worm detection algorithm through combining the information of user representation layer and use representation layer. Detail to say, we will try to setup time and software profile automatically to improve worm detection.

3 Overview of HPBRWD Design

3.1 Design Goals and Principles

The design goals of HPBRWD include three points: 1) detecting worms as early as possible; 2) eliminating worm-like P2P traffic; 3) detecting both fast worms and slow worms.

To achieve such goals, HPBRWD is designed to be host-based so as to get extra information such as normal software profile and normal time profile easily. Information about normal software and normal time profile will be detailed addressed later.

We monitor the software used on the host and the traffic from and to the host. Based on the information, we dynamically setup an initial profile about normal software and corresponding ports used by the software automatically.

Profile of normal time slots is dynamic created and HPBRWD has a default profile which sets all the time slots usable. After the steps above, HPBRWD begins to monitor in real-time.

The principles underlying the HPBRWD design are as follows:

- 1) Connecting different IPs with the same destination port within a time slot is suspicious. For a fast worm, it needs to propagate as quickly as possible, so the slot will be short. HPBR is designed to not be limited to a short time slot, so it can not only detect fast worms, but also it is effective to detect slow worms.
- 2) Same messages sent to different nodes construct a tree or chain. This behavior is also considered to be suspicious. Up to now, except some P2P software, Normal traffic will not have this behavior. For example, if a host receives a message on port A, then it sends to port A on another host the same or similar message, then we can consider this process is suspicious.
- 3) Software not in the normal software profile and having one or both of the two behaviors above is suspicious.
- 4) Packets' timestamp not in the normal time profile is suspicious.

With the emergence of P2P software, we can find that the first and the second principles are not the unique principles of computer worms. But our previous work [17] has showed that based on HPBR, we can eliminate most or whole worm-like P2P traffic.

3.2 HPBRWD Architecture and Flow of Control

Figure 2 shows the modules of HPBRWD. HPBRWD is host-based, so all the modules are located on just one host. There are six modules in HPBRWD:

- 1) Profile setup module. This module focuses on setting up profiles of normal software and normal time slots.

2) Packet capture module. This module focuses on capturing the packets to and from the host. Different from general packet capture process, the traffic created by software in the normal software profile will not be captured.

3) Packet preprocess module. This module preprocesses the packets to create formatted packets denoted as meta-access information (MI). These formatted packets will be the data source for HPBR detector module.

4) HPBR detector module. This module ranks every formatted packet and the selectively stored historical formatted packets. When the ranking value is greater than the threshold, alert will be triggered.

5) Profile update module. This module includes the update of these profiles: normal software, normal time slots, blacklist and selectively stored packet history (denoted as MI library). Blacklist and MI library are generated in HPBR detector module. When HPBR detector module completes, these two profiles are also updated.

6) Alert process module. This module processes the responses after an alert is triggered. It will look for the program which sent the corresponding packets. In our implementation, this module will take three measures: alert, log and process termination.

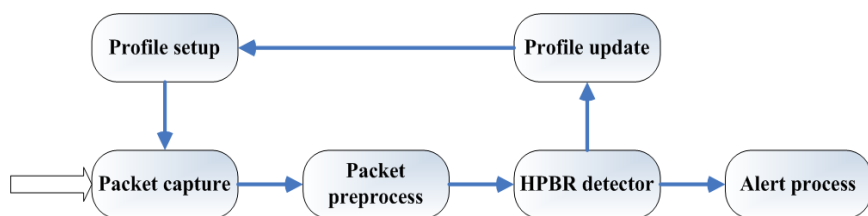


Fig. 2. Architecture of HPBRWD

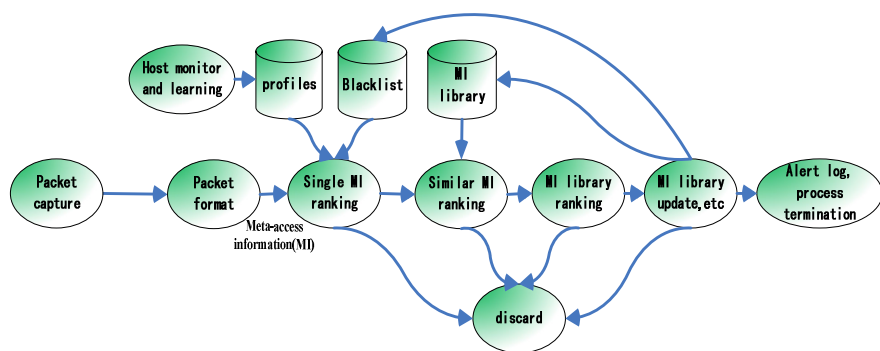


Fig. 3. Flow of HPBRWD

Fig.3 shows the process flow of HPBRWD. The concrete steps are as follows:

- 1) Creating normal software profile and normal time periods profile through host monitor and learning. The profiles will be stored into database.
- 2) Capturing and filtering packets according to normal software profiles. After this step, all the traffic coming from the normal software will be filtered. Only the traffic from the abnormal software and the traffic to the host are left.

- 3) Formatting packets. After this step, packets are formatted to be meta-access information (MI).
- 4) Ranking single MI. According to profiles created in step 1), we will first rank the single MI. If the single MI is in the blacklist, alert will be triggered at once.
- 5) Ranking similar MIs. In this step, the current MI will be ranked with MI library. We will find whether there is any MI in the MI library which is similar with current MI. If existing, we will update the rank value of corresponding MI in the MI library.
- 6) Ranking MI library. The ranking value of whole MI library helps us making decisions on whether to trigger an alert or not. If this ranking value is over the threshold, an alert will be triggered.
- 7) Updating MI library. When MI is ranked, it will be added to MI library according to MI library updating algorithm. The blacklist will also be updated in this step.
- 8) Processing alerts. When a worm is detected, measures such as log, alert or process termination will be taken.

4 Key Issues in the Design of HPBRWD

In this section, we discuss some key design issues and our solutions.

4.1 How Do We Carry on HPBR?

To answer this question, we will first give some denotations on data structure used in HPBR, then the definitions of MI and similar MIs. Based on these two definitions, we will introduce the design of HPBR. At last, we will present the algorithm of updating MI library.

Definitions and data structures. Before explaining HPBR, we first list the denotations, data structures and definitions.

Remark 1. Denotations

Table 1. Denotations

Denotation	Description	Denotation	Description
SW	Habitual used software	T	Habitual time accessing internet
B	Blacklist	W	White list
DP_i	Ports used by software i	DP	The total ports on one host. $DP = \cup DP_i$
$g(MI_i, DP)$	Weight of destination port of MI_i	$h(MI_i, SW)$	Weight of software sending MI_i
$q(MI_{[0..x-1]}, MI_x)$	Number of history MIs which are similar with MI_x	$k(MI_i, T)$	Weight of time-stamp of MI_i
$w(MI_x, W)$	If MI_x is in W	$B(MI_x, B)$	If MI_x is in B
α	Accelerating coefficient of anomalous software	β	Accelerating coefficient of anomalous destination port

Table 1. (continued)

γ	Accelerating coefficient of anomalous similar MIs	θ	Accelerating coefficient of anomalous time-stamp
ε	Max length of MI library	φ	Max number of MIs for a single destination IP
μ	Threshold of triggering alert		

At the same time, we definite some structures corresponding to some of the denotations above.

$$h(MI_i, SW) = \begin{cases} 0, & MI_i \in SW, i \geq 0 \\ 1, & \text{otherwise} \end{cases} \quad (1)$$

$$k(MI_i, T) = \begin{cases} 0, & MI_i.pTime \in T \\ 1, & \text{otherwise} \end{cases} \quad (2)$$

$$g(MI_i, DP) = \begin{cases} 1, & \sim MI_i.dPort \in DP \\ 0.5, & (\sim MI_i.dPort \in DP_i) \cap (MI_i.dPort \in (DP \setminus DP_i)) \\ 0, & MI_i.dPort \in DP_i \end{cases} \quad (3)$$

$$q(A_{[1..y]}, MI_x) = \begin{cases} A_i.num + 1, & (x \geq 1) \cap (\exists i \in [1..y], \text{st } A_i.MI \cong MI_x) \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

$$q(A_{[1..y]}, MI_x)_i = \begin{cases} A_i.num + 1, & (x \geq 1) \cap (A_i.MI \cong MI_x) \\ 0, & \text{therwise} \end{cases} \quad (5)$$

$$w(MI_x, W) = \begin{cases} 0, & (MI_x.dPort, MI_x.progName) \in W \\ 1, & \text{otherwise} \end{cases} \quad (6)$$

$$b(MI_x, B) = \begin{cases} 0, & (MI_x.dPort, MI_x.protocol, MI_x.pSize, MI_x.progName) \in B \\ \delta + 1, & \text{otherwise} \end{cases} \quad (7)$$

Remark 2. Definitions

Definition 1: *MI (Meta-access Information).* MI is in fact the formatted packet. We denote it as:

$MI = \{sIP, sPort, dIP, dPort, protocol, pSize, pTime, pProgName\}$. The elements of MI stands for source IP, source port, destination IP, destination port, protocol, size, timestamp of the a packet and the software which sent the packet.

Definition 2: *Similar MIs.* We call two MIs as similar MIs when they meet with the first two principles in section 3.1. If MI_1 and MI_2 are similar, we denote them as $MI_1 \cong MI_2$, to make the judgment of similar MIs easy, we define such functions:

For MI_1 and MI_2 :

$$f_1(MI_1, MI_2) = \begin{cases} 1, (MI_1.dPort = MI_2.dPort) \cap (MI_1.dIP = MI_2.sIP) \\ \quad \cap (MI_1.pTime < MI_2.pTime) \\ 0, \text{otherwise} \end{cases} \quad (8)$$

Here f_1 means whether two MIs have the same destination port and the latter MI's source IP is the same with the former MI's destination IP. If true, we set $f_1 = 1$. In fact, from the view of host, f_1 tells us how to judge the principle 2).

$$f_2(MI_i) = \begin{cases} 1, MI_i.dPort \neq MI_i.sPort \\ 0, \text{otherwise} \end{cases} \quad (9)$$

Here f_2 means that for one MI, whether the destination port equals its source port. If true, we set $f_2 = 1$. In fact, this judgment is from the results of experiments. We found that P2P software always has such feature while computer worms don't have because their propagation process is always multi-thread based. So we add f_2 to eliminate false negatives created by P2P software.

$$f_3(MI_1, MI_2) = \begin{cases} 1, (MI_1.dPort = MI_2.dPort) \cap \\ \quad \left(\sim((MI_1.dPort \in DP_{MI_1.progName}) \cup (MI_2.dPort \in DP_{MI_2.progName})) \right) \\ 0, \text{otherwise} \end{cases} \quad (10)$$

Here f_3 is to judge whether two MI have the same destination port and this port is not in port list of software sending MI_1 and MI_2 . If true, we set $f_3 = 1$. So In fact, f_3 tells us how to judge the principle 1) and 3).

$$f_4(MI_1, MI_2) = \begin{cases} 1, (MI_1.protocol = MI_2.protocol) \cap (MI_1.pTime \neq MI_2.pTime) \\ \quad \cap (MI_1.dIP \neq MI_2.dIP) \cap (MI_1.dPort \neq MI_2.sPort) \\ \quad \cap (MI_1.pSize = MI_2.pSize) \\ 0, \text{otherwise} \end{cases} \quad (11)$$

Here f_4 lists the base characters two MIs should have to judge whether a worm exists or not.

$$f(MI_1, MI_2) = (f_1(MI_1, MI_2) * f_2(MI_1) * f_2(MI_2) + f_3(MI_1, MI_2)) * f_4(MI_1, MI_2) \quad (12)$$

Here f is to judge whether two MIs are suspicious MIs. It is easy to know $MI_1 \cong MI_2$ when $f(MI_1, MI_2) \geq 1$. At the same time, $MI_1 \cong MI_3$ when $MI_1 \cong MI_2$ and $MI_3 \cong MI_2$.

Definition 3: MI library. We call the set which saves MI history information and statistics information of MI as MI library. MI library is denoted as A and we set $A = \{a | a = (MI_i, Fvalue, Num, IPList)\}$. $|A|$ means the length of A , $Fvalue$ stands for the ranking value of MI, Num means the number of MI which is the same with MI_i , $IPList$ means the IP list of MI which is the same with MI_i .

Host Packet Behavior Ranking. Based on the denotations and definitions above, for a coming MI_x , we use $F_1(A_{[1..y]}, MI_x)$ as ranking function:

$$F_1(A_{[1..y]}, MI_x) = \begin{cases} q(A_{[1..y]}, MI_x) * \gamma * (1 + \alpha * h(MI_x, SW) + \theta * k(MI_x, T)), & y \geq 1 \\ 0, & y = 0 \end{cases} \quad (13)$$

Supposing that there exists $MI_x \cong MI_k$ in formula (6), then for an A_i in MI library A, the corresponding rank function is:

$$A_i.Fvalue = \begin{cases} \sum_{j=1}^{x-1} (1 + \alpha * h(MI_j, SW) + \theta * k(MI_j, T)) * \gamma * q(A_{[1..y]}, MI_j)_i, & x \geq 2 \\ \sum_{j=1}^{x-1} ((1 + \alpha * h(MI_j, SW) + \theta * k(MI_j, T)) * \gamma * q(A_{[1..y]}, MI_j)_i) \\ \quad + F_1(A_{[1..y]}, MI_x), & x \geq 2, i = k \\ 0, & x = 1 \end{cases} \quad (14)$$

So the ranking function for the whole MI is:

$$F(A_{[1..y]}, MI_x) = \max\{A_i.Fvalue\}, x - 1 \geq i \geq 1 \quad (15)$$

When $F(A_{[1..y]}, MI_x)$ is greater than the threshold μ , an alert will be triggered. At the same time, suppose that $F(A_{[1..y]}, MI_x) = A_m.Fvalue$, then when an alert is triggered, we set $A_m.Fvalue = 0$, but for a new coming MI_{x1} , if $q(A_{[1..y]}, MI_{x1})_m > 0$, then an alert will also be triggered.

MI library updating policy

MI library caches the recently coming MI, and at the same time, from section 4.1, we can see the access to the MI library is also very frequent when performing HPBR. Thus, the length of MI library cannot be infinite because it has a great influence on the performance of HPBR. In our work, to effectively use the MI library, we use Modify_A algorithm to update it.

Algorithm Modify_A:

Input: MI library A, new coming MI_x

Output: none

1 compute $q(A_{[1..y]}, MI_x)$

2 if $q(A_{[1..y]}, MI_x) > 0$ and $A_i.MI \cong MI_x$, then add $MI_x.dIP$ to A_j endif;

3 if $q(A_{[1..y]}, MI_x) = 0$ then

3.1 if $|A| < \hat{a}$ then add MI_x to A endif;

3.2 If $|A| = \hat{a}$ then

3.2.1 Traverse A and find the minimal from the entire MI library element's rank values;

3.2.2 If more than one element in MI library has the minimal rank value and the value is 0

Then for every element A_i that has the minimal rank value 0

Compute $\min \{1 + \hat{a} * h(A_i.MI, SW) + \hat{e} * k(A_i.MI, T)\}$

```

        If more than one  $A_i$  meets the condition,
        Then random select one  $A_i$  to delete
        Endif;
    Endif;
3.2.3 If only one element has the minimal rank value
    Then delete this element
    Endif;
3.2.4 If the minimal rank value  $>0$  then
    Set  $|A|=|A| + 1$ ;
    Endif;
Endif;
3.3 Add  $MI_x$  to  $A$ ;

```

Modify_A makes the similar MIs have the biggest chance to stay in the MI library, so that we can effectively rank them and detect worms. From Modify_A we can also find that $|A|$ is not certain in all situations. If the minimal rank value is not equal 0, $|A|$ will increase. This is the core of detecting slow worms and the detail description will be introduced in section 4.3.

4.2 How to Setup Normal Software Profile and Time Profile Automatically

For HPBRWD, normal software profile and time profile can help us in three aspects: 1) filtering unnecessary traffic to improve processing efficiency; 2) accelerating detection of computer worms through corresponding accelerating coefficient; 3) eliminating the influence of P2P software. So how to setup these two profiles is one of the key issues of HPBRWD.

Definition 4: *normal software*. If software doesn't have the features like the principle 1) and 2), then we consider it as normal software.

Definition 5: *suspicious software*. If we can't decide whether software has the features like the principle 1) and 2), we consider it as suspicious software.

Definition 6: *infected software*. If software has the features like 1) or 2) or both, we consider it as infected software.

In HPBRWD, the construction of normal software profile is a dynamic process. You can see it in fig 4. The state changes of three kinds of software are also listed in fig 4.

The software which we are using on a host can be divided into two classes: 1) installed software; 2) green software. So to get the software list, we can focus on such two classes.

For installed software, we can check registry to get detail information. Detail to say, all the installed software save their installation information under the registry item "HKEY_LOCAL_MACHINE\software\Microsoft\windows\uninstall\InstallLocation" tells us the name and path of software.

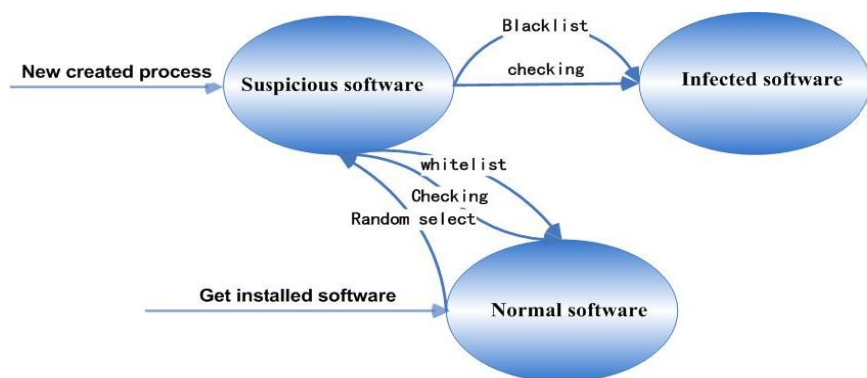


Fig. 4. State changes of the three kinds of software

For Green software, we decide to get its information when it begins to run. We hooked the function *NtResumeThread* to get information of new created process. Thus we can get software which user is running.

For the new created process, we first consider them to be suspicious until we can't find that they have the features like principles 1) and 2) in a given time period. To be suspicious software means all the traffic sending or receiving will be recorded to HPBR.

There are two situations can software A be shifted from suspicious software to normal software: 1) A is in white list; 2) based on HPBR checking, if no principles 1) and 2) are found. When either of these two situations is satisfied, we add this software to Normal software profile.

For those installed software, they will first be classified into normal software, but it doesn't mean that each software in normal software profile is always immune to computer worms, so we randomly select software from normal software profile and change their state to suspicious software every given time slot. So the creation of normal software profile is dynamic.

There are also two situations for software A to change from suspicious software to infected software: 1) A is in blacklist; 2) based on HPBR, an alert is triggered.

For the creation of normal time profile, we first give the definition of normal time:

Definition 7: normal time. Normal time means the time slot when a user always accessing internet. Detail to say; if % of normal software is running during time slot A, we call this time slot A as normal time. is a parameter, but in our implementation now, we decide that a time slot will be consider normal time only if there is one normal software running.

Normal time profile is not stable. To create it, we first look up all the process running now to see if there is any in normal software profile, if true, the time slot between now and 5 minutes later are considered to be in normal time profile. When 5 minutes passed, there will be another checking on normal time.

Of course, we can also manually set the normal software profile and normal time profile.

4.3 How Does HPBRWD Eliminate the Influence of Worm-Like P2P Traffic?

In this paper, to eliminate the influence of worm-like P2P traffic is relatively easy. The reason is as follows: traditional P2P traffic identification methods mainly use communication information, while ours is based on host information. We can easily get the P2P software list used in the host. Through hooking *ntDeviceIoControl*, we can easily acquire the correlation between process and port. Source port is the bridge between the packet's information and P2P software self's information, the whole process can be shown in fig5.

When a P2P application sends or receives packets, it will call *ntDeviceIoControl* function. We have hooked this function and save the corresponding source ports of this application. If this P2P application is in Normal software profile, we will update the source ports information to the normal software profile. At the same time, we use winpcap library to capture the packets of all applications. Source ports are also can be acquired in the packet's information. So according to the source port, we can know whether the packet captured is sent by a normal software or not. So at last, only packets sent by applications which are not in normal software profile will be formatted to MIs.

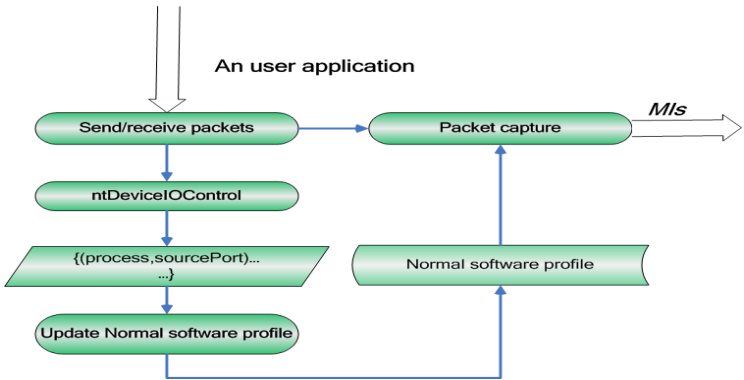


Fig. 5. Process of eliminating P2P traffic

4.4 How Does HPBRWD Detect the Slow Worms?

Slow worm has not become public in the internet, but we think in the near future they will come out. A well-designed slow worm should meet such conditions: 1) they are profit-driven, not only to do damage to the Internet. So they have clear and certain victims; 2) to survive longer, they will not choose fast propagation. They will send exploits in much longer interval. Perhaps they will also learn the communication character before sending exploits. Traditional worm detection methods mainly focus on fast worms and have a bad effect on slow worms because they are based on the characters of fast worms or silent worms.

In our work, we detect slow worms through HPBR. When a slow worm is running on the host, it will first be classified into suspicious software. When a *MI* denoted as MI_x sent by the slow worm is captured, $h(MI_x, SW) > 0$ and $1 + \alpha * h(MI_x, SW) + \theta * k(MI_x, T) > 0$, so according to *Modify_A*, MI_x will stay in the *MI library*. Only

if HPBRWD is running, whenever the next *MI* from the slow worm will come, the slow worm will be detected.

4.5 What Are the Considerations for Performance in the HPBRWD?

In HPBRWD, there are three points needed to consider the performance: 1) capturing packets; 2) computing ranking functions in HPBR; 3) updating *MI library*. We will discuss the three aspects as follows:

1) Packet capture. In our work, packet capture module not only captures the packets, but also filters the traffic from the normal software. Although there is not as much traffic in one host as that in one network, with the emergence of P2P software, video and file traffic have become more and more. To improve the performance, we use winpcap to capture the header of packets and install a driver to get the map of port and software process. At the same time, we use a memory hash table to store the map of source port and process so that we can reduce the time of lookup.

2) Updating of *MI library*. When implementing *Modify_A*, it is time-consuming to look for the minimal ranking value in *MI library*. We use a separate memory structure to record the minimal rank value. After a new *MI* is added to *A*, the minimal rank value and *MI*'s index in the *MI library* are also updated. In this way, we can greatly reduce the compute time cost.

5 Implementation and Evaluations

5.1 Implementation Introduction

To prove the concept of HPBRWD, we have implemented a prototype using Delphi language on windows2003. To capture the packets, we use the famous winpcap (version 4.0) library to monitor the communication to and from the host (with Delphi). To get the relationship between source port and process, we have hooked ntDeviceIoControl, ntResumeThread and other related functions. In our prototype, all the profiles are stored as plain text file, but after the HPBRWD runs, they will be loaded into the memory and modified during runtime. When HPBRWD ends, these profiles data in memory will be saved to text files again. At the same time, in order to carry on experiments easily, we have dumped the filtered traffic for offline analyzing.

To evaluate HPBRWD, we seek to answer the following questions: 1) Is HPBRWD effective in eliminating the worm-like P2P traffic? 2) Is HPBRWD effective in detecting worms?

5.2 Is HPBRWD Effective in Eliminating the Worm-Like P2P Traffic?

Dataset Setting

Table 2 lists the P2P applications used in this experiment. We select three kinds of typical P2P applications which are widely used: P2P instant messenger, P2P file sharing software and P2P video. All the P2P applications are installed before the experiment begins. Table2 also lists the common operations during using these P2P applications. "Idle" means no manual operations; "Send massive message" means sending message

to all the friends in a group at the same time(in this experiment, we sent a message to a group with 10 friends); “send file” means sending files to a friend; in “download file” and “movie playing”, we choose the popular songs and TVs recently.

Table 2. P2P applications used

Type of P2P applications	name	Operations tested
P2P instant messenger	QQ	Send massive message; idle; send file;
P2P file sharing software	Bit torrent, Thund- er version 5	Idle; download file
P2P video software	UUsee, PPstream	Idle; movie playing

Algorithms Used and Parameter Setting

To make the effect obvious, we implement a simple connection-rate based worm detection algorithm called CRWD (Connection-Rate based Worm Detection). CRWD is similar with the work in [7], but it is implemented on host not on the network. We use the parameter ϵ as the detection threshold. In this experiment, we set $\epsilon = 4$.

For HPBRWD, we set $(\alpha, \beta, \gamma, \theta, \epsilon, \phi, \mu) = (1, 1, 5, 256, 20, 80)$. In fact, after computing, this set of parameters means only if two similar MIs exist, an alert will be triggered. We can see that the threshold of HPBRWD is stricter than CRWD. We want to prove whether HPBRWD is effective in eliminating worm-like P2P traffic.

5.2.1 Result

Table 3 list the false positives created by these P2P applications when using HPBRWD and CRWD. From the table, we can see that there exists worm-like P2P traffic which can create false positives and HPBRWD has successfully eliminating the traffic. Table 3 also lists the destination ports of P2P application which create the false positives. We found that UUsee created more of the worm-like P2P traffic. UUsee is a P2P video application which always sends much maintenance information and request information to other peers.

The reasons of why HPBRWD can eliminate these worm-like P2P traffic are: 1) all the P2P applications have been installed before this experiment, so through automatic

Table 3. Result

software	False positives In HPBRWD	False positives In CRWD	Destination ports of P2P application which create False positives
Bit tor- rent	No	Yes	137,53,6969
ppstream	No	Yes	7202,7201,8000,53,33366
QQ	No	Yes	8001,8002,8003
UUsee	No	Yes	80,443,11111,444,8665,53,8242,8001,9638, 7775,17024,8565,9600
thunder	No	yes	80,53,3077,8000

normal software profile setup and packets captured, we successfully eliminate the worm-like P2P traffic. We can understand the detail process in section 4.3; 2) these P2P applications do not have too much source ports when used, which makes a high performance in hooking ntDeviceIoControl and updating normal software profile.

5.3 How Effective Is HPBRWD in Detecting Computer Worms?

Dataset Setting

In this experiment, we will use computer worms in the wild: codegreen.a, Sasser.a and Webdav.a. The three worms have different propagation speed, which we can see later.

To carry out this experiment, we choose the five famous computer worms. We executed them manually on our own host, which is in an intranet of our own. The corresponding process names listed above are considered not in the normal software profile because we can't get them from the registry which stores the information of installed software.

In this experiment, we set $(\alpha, \beta, \gamma, \theta, \varepsilon, \varphi, \mu) = (1, 1, 5, 3, 256, 20, 20)$.

Result

Table 4 lists the result of detection using HPBRWD:

Table 4. Detection result

	Codegreen.a	Sasser.a	Webdav.a
Release time	22:11:21.000916	22:47:53.350332	22:20:23.619331
Stop time	22:12:10.470961	22:50:03.340721	22:44:29.096500
Total attack number	928	1954	183
Propagation speed (average)	18.8/s	15/s	7.6/min
First alert time	22:11:21.111759	22:47:53.354992	22:20:52.002402
Time spent on first detection	0.110843	0.004660s	28.383071s
False positives number	0	0	0
False negatives numbers	18	59	0
False negatives ratio	1.94%	2.99%	0%

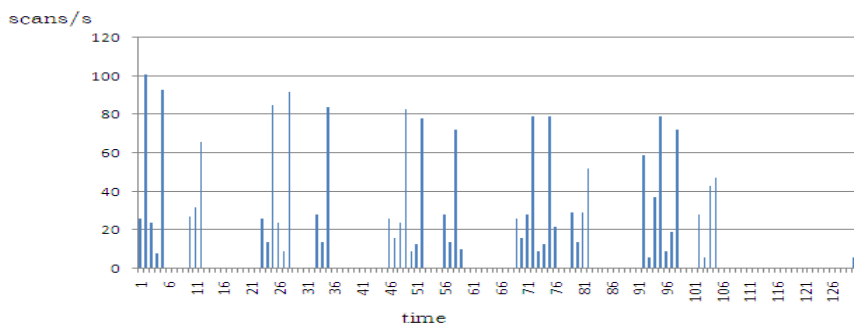


Fig. 6. Speed of sasse.a

We have tested the three worms on win2000 sp4 by manually executing them. Table 4 lists the release time and stop time of them. We can see from the table that the three computer worms have different propagation speed. It seems that codegreen.a spread fastest at 18.8/s. Sasser.a was at 5.35/s. But we will find that Sasser.a was detected earliest which only used 0.004660s, while detecting codegreen.a used 0.110843. The reason is that Sasser.a didn't spread at a stable speed, just as fig. 6.

In fig. 6, time means the timestamp at which sasser.a propagated. The "1" time means 22:47:53.350332, while the last time means 22:50:03.340721. We can see that the propagation of sasser.a has the characteristics of periodicity and instability. It even chose to be silent at some timestamps, which will bring challenges to some worm detection algorithm based on fast propagation. The detection result also tells us that HPBRWD is good enough to detect such kind of propagation.

At the same time, we find that HPBRWD can not only detect fast-propagation worms (such as codegreen.a and Sasser.a), but also slow worms. Webdav.a is just such an example. It spread at a speed of 7.6/min and from the detection log, we can see that its speed is stable, which means that it can evade almost all the fast-propagation based worm detection systems. While our HPBRWD is immune to the slow speed and can detect webdav well.

HPBRWD is good at reducing false positives. In the experiment, we can see that false positives are 0. The reason is that the most challenge for behavior-based worm detection is worm-like traffic created by P2P software. We have successfully eliminated them which you can see from the section 5.2. But with the different speed of computer worms, there exists different number of false negatives. It looks like the faster worm speed is the more false negatives HPBRWD has. In our opinion, the reason is that HPBRWD is designed to be a real-time detector, but with worm speed gets faster, the compute cost will also increase, which created some false negatives. This is a problem of both design and implementation. In the future work, we will improve HPBR and its implementation to get a higher performance so as to try to solve this problem.

6 Related Work

Work [3-5] are all automatic worm signatures generation system, but they can't detect unknown worms or polymorphic worms. Work [6] has a try in detecting polymorphic worms, but it cannot detect complex polymorphic and metamorphic worms and it has been proven that polygraph is vulnerable to noise injection [7].

Behavior-based worm detection has been a research hotspot in recent years. Many famous IDS system has added the support for the malicious behavior detection, such as connection rate-based detection [7] and failed connection number based detection [8]. Detection based on netlike association analysis is presented by [9], it analyzes the data similarity when coming out, invalid destination IP and service requests and the similar propagation behavior between hosts. Work [10] collects behavior data from computer and network, and then detects worm through pattern library. In work [11], malicious network behavior is divided to three classes: traffic, responder and connector, and then worm detection is carried out based on these behaviors.

Cliff ChangChun Zou [12] presents Kallman filtering method to evaluate the infection rate of worm. Work [13] [14] studies the frequency of destination IP scanning and source port changing, and then detects worms based on Bayesian analysis and entropy. Recent work [15] has presented a method based on velocity of the number of new connections an infected host makes and control system theory. Work [16] has used both low- and high-interaction honeypot to detect worms. The detection technologies introduced above have a good effect on fast worm detection without many P2P traffic. These technologies are concentrated at the instant or statistics behavior from the network traffic and good detection effect lies in good distinction between normal traffic and malicious traffic. But the emergence of P2P software makes the distinction difficult for the technologies above.

7 Conclusions and Future Work

In this paper, we have presented a hierarchical user habit model. From this model, we found that software and time period is also the important factors to generate network traffic, while they are not paid attention to by traditional user habit model or network behavior based worm detection. An overview of HPBRWD design is also introduced to make the design goal and principles clear. To make the work easy to understand, we introduced several key issues in design and gave our solutions. In this paper, we also explained that host based gave us the advantage to make the worm detection more effective.

At last, several experiments were carried on to test the effect of HPBRWD. Results of the experiments showed that HPBRWD had done a good job.

We plan to optimize the multi-level ranking function in HPBRWD. In the implementation now, it seems a bit complex; for the profile of normal software, in the future work, we want to make the generation more automatic; in section 3.1, we have presented an assumption, in the future work, we will solve the problem of injecting into other processes to run the worm code.

Acknowledgement

We would like to thank the High Technology Research and Development Program of China (863 Program) for support of this project with grant 2006AA01Z401 and 2008AA01Z414, we also thank National Natural Science Foundation of China for support of this project with grant 60573136.

References

1. Moore, D., Paxson, V., Savage, S., Shannon, C., Staniford, S., Weaver, N.: Inside the slammer worm. *IEEE Security & Privacy* 1(4), 33–39 (2003)
2. Staniford, S., Moore, D., Paxson, V., Weaver, N.: The top speed of flash worms. In: Paxson, V. (ed.) *Proc. of the 2004 ACM Workshop on Rapid Malcode*, pp. 33–42. ACM Press, Washington (2004)

3. Kim, H., Karp, B.: Autograph: Toward automated distributed worm signature detection. In: Proceedings of USENIX Security, San Diego, CA (August 2004)
4. Kreibich, C., Crowcroft, J.: Honeycomn-creating intrusion detection signatures using honeypots. In: Proceedings of HotNets, Boston, MA (November 2003)
5. Singh, S., Estan, C., Varghese, G., Savage, S.: Automated worm fingerprinting. In: Proceedings of OSDI, San Francisco, CA (December 2004)
6. Newsome, J., Karp, B., Song, D.: Polygraph: Automatically generating signatures for polymorphic worms. In: Proceedings of IEEE Symposium on Security and Privacy, Oakland, CA (May 2005)
7. Roesch, M.: Snort: Lightweight intrusion detection for networks. In: Proceedings of Conference on system administration (November 1999)
8. Paxson, V.: Bro: a system for detection network intruders in real time. *Computer Networks* 31 (December 1999)
9. Si-Han, Q., Wei-Ping, W., et al.: A new approach to forecasting Internet worms based on netlike association analysis. *Journal On Communications* 25(7), 62–70 (2004)
10. Stanford-Chen, S., et al.: GrIDS: A Graph-Based Intrusion Detection System for Large Networks. In: Proceedings of the 19th National Information Systems Security Conference, vol. 1, pp. 361–370 (1996)
11. Dubendorfer, T., Plattner, B.: Host Behavior Based Early Detection of Worm Outbreaks in Internet Backbones. In: Proceedings of 14th IEEE WET ICE/STCA security workshop, pp. 166–171 (2005)
12. Zou, C.C., Gong, W., Towsley, D., et al.: Monitoring and early detection of internet worms[A]. In: Proceedings of the 10th ACM Conference on Computer and Communications Security[C], Washington DC, USA, pp. 190–199. ACM Press, New York (2003)
13. Internet Threat Detection System Using Bayesian Estimation. In: 16th Annul FIRST Conference on Computer Security Incident Handling. 20 Sumeet Singh, Cristian Estanm (2004)
14. Wagner, A., Plattner, B.: Entropy based worm and anomaly detection in fast ip networks. In: WET ICE 2005, pp. 172–177 (2005)
15. Dantu, R., Cangussu, J.W., et al.: Fast worm containment using feedback control. *IEEE Transactions On Dependable And Secure Computing* 4(2), 119–136 (2007)
16. Portokalidis, G., Bos, H.: SweetBait: Zero-hour worm detection and containment using low- and high-interaction honeypots. *Computer Networks* 51(5), 1256–1274 (2007)
17. Xiao, F., Hu, H., et al.: ASG - Automated signature generation for worm-like P2P traffic patterns. In: waim 2008 (2008)

Anonymous Resolution of DNS Queries

Sergio Castillo-Perez¹ and Joaquin Garcia-Alfaro^{1,2}

¹ Universitat Autònoma de Barcelona,
Edifici Q, Campus de Bellaterra, 08193, Bellaterra, Spain
scastillo@deic.uab.es

² Universitat Oberta de Catalunya,
Rambla Poble Nou 156, 08018 Barcelona, Spain
joaquin.garcia-alfaro@acm.org

Abstract. The use of the DNS as the underlying technology of new resolution name services can lead to privacy violations. The exchange of data between servers and clients flows without protection. Such an information can be captured by service providers and eventually sold with malicious purposes (i.e., spamming, phishing, etc.). A motivating example is the use of DNS on VoIP services for the translation of traditional telephone numbers into Internet URLs. We analyze in this paper the use of statistical noise for the construction of proper DNS queries. Our objective aims at reducing the risk that sensible data within DNS queries could be inferred by local and remote DNS servers. We evaluate the implementation of a proof-of-concept of our approach. We study the benefits and limitations of our proposal. A first limitation is the possibility of attacks against the integrity and authenticity of our queries by means of, for instance, man-in-the-middle or replay attacks. However, this limitation can be successfully solved combining our proposal together with the use of the DNSSEC (DNS Security extensions). We evaluate the impact of including this complementary countermeasure.

Keywords: IT Security, Privacy, Anonymity, Domain Name System, Privacy Information Retrieval.

1 Introduction

The main motivation of the present work comes from privacy and security concerns regarding the use of the protocol DNS (Domain Name System) as the underlying mechanism of new Internet protocols, such as the ENUM (*telephone Number Mapping*) service. ENUM is indeed a set of service protocols used on VoIP (Voice over IP) applications. One of the main characteristics of ENUM is the mapping of traditional phone numbers associated to the ITU-T (International Telecommunications Union) E.164 recommendation, to URIs (Universal Resource Identifiers) from VoIP providers, as well as to other Internet-based services, such as e-mail, Web pages, etc. We overview in this section some of the features of this service, as well as some security and privacy concerns regarding the use of the DNS protocol in ENUM.

1.1 The ENUM Service

The ENUM service is a suite of protocols used in VoIP applications whose main goal is the unification of the traditional telephone E.164 system with the IP network of the

Internet. Designed and developed by the *Internet Engineering Task Force* (IETF) in late nineties, ENUM allows the mapping of IP services by using an indirect lookup method based on DNS technologies. In this manner, an by simply using existing DNS implementations, ENUM allows retrieving lists of IP based services, such as SIP (Session Initiation Protocol) identifiers for VoIP applications, e-mail addresses, Web pages, etc., associated to the principal of an E.164 telephone number. ENUM uses a particular type of DNS records, called Naming Authority Pointer (NAPTR) [9]. Instead of resolving host or service names into IP addresses, the ENUM service translates E.164 telephone numbers into Uniform Resource Locators (URLs) embedded within NAPTR records. At long term, ENUM is expected to become a decentralized alternative to the E.164 system. For a more detailed introduction to the suite of protocols associated with ENUM, we refer the reader to [6].

As a matter of fact, ENUM is just a simple convention for the translation of E.164 telephone numbers, such as +1-012345678, into URI (*Uniform Resource Identifier*) strings. These strings are associated to the DNS system by using the following convention: (1) special symbols like '+' and '-' are deleted (e.g., +1-012345678 becomes 1012345678); (2) the resulting string of digits is inverted from left to right (e.g., 8765432101); (3) a symbol '.' is inserted between each two digits (e.g., 8.7.6.5.4.3.2.1.0.4.1); (4) the domain name .e164.arpa (registered by the IETF for ENUM resolution) is finally concatenated to the previous string (e.g., 8.7.6.5.4.3.2.1.0.1.e164.arpa). The resulting string of characters and digits is then ready to be used as a normal query towards the DNS system. At the server side, the URI associated to every possible telephone number registered by ENUM is stored together with information about its principal (e.g., owners or users of those telephone numbers). Such an information is stored on DNS records of type NAPTR. The internal structure of these records offers to ENUM enough storage space and flexibility for managing complex information (e.g., use of regular expressions).

Let us show in the following a complete example in which ENUM is used for the translation of the telephone number +1-012345678 associated to a user U_1 . Let us assume that a user U_2 wants to get in contact with user U_1 . First of all, user U_2 translates the previous telephone number into the string 8.7.6.5.4.3.2.1.0.1.e164.arpa. U_2 then uses the obtained URI to construct a DNS query of type NAPTR by using the command line tool *dig*:

```
dig @$NS -t NAPTR 8.7.6.5.4.3.2.1.0.1.e164.arpa
```

As a result, U_2 obtains the following information:

Order	Pref.	Flags	Service	Regexp.	Replacement
100	10	u	sip+E2U	!.*\$!sip:u1@sip.com!	.
101	10	u	mailto+E2U	!.*\$!mailto:u1@mail.com!	.
102	10	u	http+E2U	!.*\$!http://www.u1.com!	.
103	10	u	tel+E2U	!.*\$!tel:+1-01235567!	.

Let us analyze the response returned by *dig*. As we introduced above, NAPTR records support the use of regular expression pattern matching [9]. In case a series of regular expressions from distinct NAPTR records need to be applied consecutively

to an input, the field *Order* is used. The value given in the first line, set to 100, indicates that from the four results of the query, the service *SIP* has the highest priority. In case of having more than one record with the same *order* values, the following field, i.e., *Pref.*, decides which information must be used first. The field *Flag* given for each line, and set to the value *u*, indicates that the field *Regex*, associated with every record contains the URI associated to the requested E.164 telephone number. A field *Replacement* containing the operator '.' indicates to the ENUM client of user U_2 that the final URL is indeed the string placed between the markers '*r.*\$!*' and '*'*' of the expression contained within the field *Regex*. The field *Service* indicates the kind of IP service that can be found in the resulting URL. For example, the field *Service* associated with the first line indicates that the resulting service is based on the SIP protocol [13]. The other three options returned as a result of the query are (1) an e-mail address associated with user U_1 , (2) his personal Web page, and (3) the use of an additional E.164 telephone number.

Let us notice from our example that the ENUM service does not resolve the IP addresses associated to the URLs embedded within the NAPTR records. A DNS query of type 'A' must follow after an ENUM resolution with the objective of resolving the appropriate IP address that will eventually be used to contact the final service. In our example, and given the values of the field *Order* discussed above, user U_2 contacts again the DNS server in order to obtain the IP address associated to the SIP at `sip.u1.com` to request the connection to user U_1 (i.e., `u1@sip.u1.com`).

1.2 Threats to the ENUM Service

The use of the DNS protocol as the underlying mechanism of the ENUM service leads to security and privacy implications. The exploitation of well known vulnerabilities of DNS-based procedures is a clear way of attacking the ENUM service. A recent analysis of critical threats to ENUM may be found in [19,20]. Rossebø et al. present in these works their risk assessment analysis of the ENUM service based on a methodology proposed by the European Telecommunications Standards Institute (ETSI) [5]. Both threats and vulnerabilities reported in these works are indeed an heritage of the vulnerabilities existing in DNS mechanisms. We can find in [2] a complete analysis of threats to DNS technologies. The most important threats to DNS technologies can be grouped as follows: (1) authenticity and integrity threats to the trustworthy communication between resolvers and servers; (2) availability threats by means of already existing denial of service attacks; (3) escalation of privilege due to software vulnerabilities in server implementations. Moreover, the DNS protocol uses clear text operations, which means that either a passive attack, such as eavesdropping, or an active attack, such as man-in-the-middle, can be carried out by unauthorized users to capture queries and responses. Although this can be considered as acceptable for the resolution of host names on Web services, an associated loss of privacy when using DNS for the resolution of ENUM queries is reported in [19,20] as a critical threat.

We consider that the loss of privacy in ENUM queries is an important concern. Beyond the engineering advance that the ENUM service supposes, it is worth considering the consequences that the exposure of people's information may suppose. The achievement of such information by dishonest parties exploiting flaws and weaknesses

in the service itself or its underlying protocols must be avoided. We can consider, for instance, worst case scenarios where dishonest servers associated to unscrupulous service providers start keeping statistics of ENUM queries and building people's profiles based on their communication patterns [23]. These scenarios may lead to further violations, such as spam, scams, untruthful marketing, etc. Consumers must be ensured that these activities are not possible [7]. However, current DNS query methods used by ENUM can be exploited if the whole process is not handled by appropriate countermeasures.

1.3 Privacy Weakness in the DNS Protocol

When the DNS protocol was designed, it was not intended to guarantee privacy to people's queries. This makes sense if we consider that DNS is conceived as a distributed hierarchical database which information must be accessed publicly. In scenarios where the DNS protocol is used for the mapping of host and domain names towards traditional Internet services, the inference of information by observing queries and responses can fairly be seen as acceptable — from the point of view of people's privacy. Nevertheless, the use of the DNS protocol on new lookup services, such as the ENUM suite of protocols, clearly introduces a new dimension. Vulnerabilities on the DNS, allowing the disclosure of data associated with people's information, such as their telephone numbers, is a critical threat [19,20]. Let us summarize these privacy weaknesses from the following three different scopes: (1) DNS local resolvers, (2) communication channel, and (3) remote DNS servers.

On the first hand, Zhao et al. identify in [23] some privacy threats related with local malware targeting the client. Applications such as keyloggers, trojans, rootkits and so on can be considered as a way to obtain the relation between DNS queries and the client who launches them. Let us note that our work does not address the specific case of malware targeting the privacy of the DNS service at the client side. On the second hand, we can identify two main threats targeting the communication channel: (1) passive eavesdropping and (2) active attacks against the network traffic. In the first case, the eavesdropping of plaintext DNS traffic flowing across unprotected wired or wireless LAN networks can be used as a form of anonymity violation. In the second case, traffic injection can also be used to attack the privacy. These attacks can be used to redirect the traffic to a malicious computer, such as ARP spoofing, ICMP redirect, DHCP spoofing, port stealing, etc. Thus, an attacker can redirect every query to a malicious DNS server with the objective of impersonating the correct one and, as a result, to compromise the client privacy. On the third hand, the existence of dishonest or malicious servers can also reduce the level of privacy. Indeed, the DNS cache model allows intermediate servers to maintain a query-response association during a given period of time. The expiration time of every entry in the cache of a server is based on the IP TTL field of a DNS response — as it is defined in [10]. During this period of time, if a client queries a cached entry, the response will be served without any additional resolution. Otherwise, after this time has elapsed, the entry is removed from the cache and, if a client requests it again, the server resolves it, caches it, and sends the response to the client.

Under certain conditions, the observation of the TTL field can be used by attackers to infer the relation between a client and a particular query, reducing the level of anonymity. If attackers suspect that a given client has launched a specific query, they

can resolve the same query on the server used by the client. After the response has been retrieved by the attackers, they can determine the current cache expiration time provided by the server. If the returned value is the maximum expiration time defined by the authoritative server, the attackers can deduce that the query has not been launched by the client in, at least, a period that equals the maximum cache expiration time. However, if the value is less than the TTL value, the attackers can consider, with a certain level of probability, that this query was made by the client at most at *maximum expiration time minus current expiration time*. This strategy can be applied by potential attackers under certain circumstances. First of all, it can only be considered in networks composed by a few number of clients and/or a DNS server that receives few queries by these clients. Otherwise, the probability of a correct correlation between the specific query and a given client must be considered almost zero. Secondly, if the expiration time defined by the authoritative server has a low value, it can lead to a situation where attackers might launch the query after it expires in the DNS cache (previously created by the client).

1.4 Privacy Countermeasures and Security Enhancements

Some initial measures for special DNS-based services, like the ENUM service, have been proposed by the IETF. Some examples are the limitation and the kind of information to be stored by the servers, the necessity of requesting the consent of people and/or institutions, etc. Nonetheless, beyond limiting and granting access to store people's information, no specific mechanisms have been yet proposed in order to preserve the invasion of privacy that a service like ENUM may suppose. The use of anonymity-based infrastructures and anonymizers (e.g., the use of the Tor infrastructure [14], based on *Onion Routing* cryptography [21]) is often seen as a possible solution in order to hide the origin (i.e., the sender) of the queries. However, these infrastructures might not be useful for anonymizing the queries themselves against, for example, insecure channels or dishonest servers [8]. The use of the security extensions for DNS (known as DNSSEC), and proposed by the IETF in the late nineties, cannot address those privacy violations discussed in this section. DNSSEC only addresses at the moment authentication and integrity problems in the DNS. Although it must certainly be seen as an important asset to enhance the security of DNS applications, it requires to be combined with additional measures to cope the kind of violations discussed in this section. Finally, the use of Privacy Information Retrieval (PIR) [18] based approaches can also be seen as a mechanism to handle the private distribution of information on the DNS service [23,24]. Unfortunately, no specific evaluations or practical results are presented in these works. The processing and communication bandwidth requirements of a PIR approach seem to be impractical for low latency services like the DNS/ENUM [22]. We consider however that these approaches head into the right direction in order to address the problematic discussed in this section.

Inspired by the approaches proposed by Zhao et al. in [23,24], we sketch in this paper an alternative model for perturbing DNS queries with random noise. The goal of our model is to prevent privacy violations due to attacks against the communication channel level or the existence of dishonest servers. Our approach addresses and enhances some security deficiencies detected in [23,24], such as the possibility of response manipulation or range intersections. We also present in this work the evaluation of a first

proof-of-concept developed and tested upon GNU/Linux setups. Our implementation combines our approach together with the use of DNSSEC extension to preserve authentication and integrity of queries. Although our experimentations reveal high bandwidth consumption as the main drawback, we consider the results as a prove of the validity of our approach.

1.5 Paper Organization

The remainder of this paper has been organized as follows. Section 2 introduces some related works. Section 3 sketches our proposal. Section 4 overviews the use of the security extensions for DNS (i.e., DNSSEC). Section 5 presents the evaluation results of combining our proposal with the security enhancements offered by DNSSEC. Section 6 closes the paper with some conclusions.

2 Related Work

A first solution to address the privacy concerns discussed in Section 1 is the use of anonymous-based communication infrastructures. The use of strong anonymity infrastructures can suppose however a high increase of the latency of a service like the DNS and the ENUM services. We recall that a communication infrastructure for these services must ensure that the service itself is able to deliver both queries and responses accurately and in a timely fashion. Thus, strong anonymity does not seem to be compatible with this requirement. On the other hand, the use of low latency infrastructures, such as the anonymous infrastructure of the Tor (*The second generation Onion Router*) project [14], based in turn on the *Onion Routing* model [21], is more likely to meet the performance requirements of the DNS/ENUM service. Nevertheless, a solution based on both Tor and *Onion Routing* may only be useful for hiding the origin of the queries. Although by using such proposals senders are indeed able to hide their identities through a network of *proxies*, they do not offer anonymity to the queries themselves. For instance, threats due to the existence of dishonest servers, are not covered by these solutions [8].

The approach presented by Zhao et al. in [23,24] aims at preserving the anonymity of DNS/ENUM queries from the point of view of the channel and/or the service providers. The main objective of these proposals is the achievement of anonymity by using a PIR (Privacy Information Retrieval) model [18]. The authors propose devising the communication protocol involved between DNS clients and servers by considering queries as secrets. Instead of querying the server by a specific host name h , for example, Zhao et al. propose in [23] the construction and accomplishment of random sets of host names $[h_1, h_2, \dots, h_n]$. The resulting protocol aims at avoiding that by listening into the channel or controlling the destination service, an attacker learns nothing about the specific host name h from the random list of names. The main benefit of this proposal is the simplicity of the approach. The main drawback is the increase in communication bandwidth that it may suppose. Zhao et al. extend in [24] this first proposal towards a two-servers PIR model. The objective of the new protocol is to guarantee that DNS clients can resolve a given query, at the same time that they hide it to each one of the servers. Nevertheless, compared with the previous proposal, this approach reduces the bandwidth

consumption. The approach requires, however, significant modifications on traditional DNS implementations. We analyze more in detail these two proposals in Section 3.

The proposals presented in [23,24], as well as Tor, do not offer preservation of authenticity and integrity of DNS responses. Therefore, without other countermeasures, these solutions cannot avoid man-in-the-middle or replay attacks aiming at forging DNS responses. A proper solution for avoiding this problem is to combine the use of anonymity with the integrity and authenticity offered by the security extensions of DNS — often referred in the literature as DNSSEC (cf. Section 4 for more information about DNSSEC). In this manner, we can guarantee the legitimacy of the response while maintaining an acceptable performance. We show in Section 5 that the impact on the latency of the service when using DNSSEC is minimal. We consider that authenticity and integrity threats are hence reduced by combining a proper anonymity model together with DNSSEC. None of these proposals guarantees the confidentiality of the queries. Although the use of alternative techniques such as IPsec [16] could be seen as a complementary solution to protect the exchanges on data between servers and clients of DNS, we consider that they are not appropriate for solving our motivation problem. First of all, the bandwidth and processing time overheads of using IPsec are much higher, and can render the solution impractical [17]. Secondly, IPsec does not offer protection during the caching processes between resolvers and/or intermediate servers. Furthermore, it is quite probable that servers of a global DNS service may not be IPsec capable. We consider that this approach is not an appropriate solution to our problem. Since our motivation is focused on privacy issues rather than confidentiality concerns, we consider that the combination of anonymity preservation together with integrity and authentication aspects offered by DNSSEC are worth enough to conduct our study.

3 Use of Random Ranges to Anonymize DNS Queries

Before going further in this section, let us first recall here the schemes presented by Zhao et al. in [23,24]. The first approach [23] works as follows: a user U , instead of launching just a single query to the DNS server NS , constructs a set of queries $Q\{H_i\}_{i=1}^n$. If we assume DNS queries of type A , the previous range of queries will include up to n different domain names to be resolved. The query $Q\{H_i\}$ will be the only one that includes the domain name desired by U . All the other queries in $Q\{H_1\} \dots Q\{H_{i-1}\}$ and $Q\{H_{i+1}\} \dots Q\{H_n\}$ are chosen at random from a database DB . The authors claim that this very simple model considerably increases the privacy of user U queries. Indeed, the only information disclosed by user U to third parties (e.g., DNS server NS and possible attackers with either active or passive access to the channel between U and NS) is that the real query $Q\{H_i\}$ is within the interval $[1, n]$. Zhao et al. presume that the probability to successfully predict query $Q\{H_i\}$ requested by user U can be expressed as follows: $P_i = \frac{1}{n}$. We refer the reader to [23] for a more accurate description of the whole proposal.

We consider that the probability model presented in [23] is unfortunately very optimistic. In fact, we believe that the degree of privacy offered by the model can clearly

be degraded if we consider active attacks, in which an adversary is capable of interacting with the channel. For example, we consider that the approach does not address possible cases in which the resolution of query $Q\{H_i\}$ fails. If we assume an active attacker manipulating network traffic (e.g., by means of RST attacks, or sending suitable ICMP traffic) to drop query $Q\{H_i\}$ — or its associated response. If so, user U will be forced to restart the process and generate a new range of queries — i.e., requesting once again $Q\{H_i\}$. Depending on how this new range is managed, the degree of privacy estimated by the probabilistic model in [23] will clearly decrease. Let $Q_j\{H_i\}_{i=1}^n$ be the n -th consecutive range exchanged for the resolution of query $Q\{H_i\}$, the probability of success for an attacker trying to guess $Q\{H_i\}$ must then be defined as follows:

$$P_{ij} = \frac{1}{|Q_1\{H_i\}_{i=1}^n \cap Q_2\{H_i\}_{i=1}^n \cap \dots \cap Q_j\{H_i\}_{i=1}^n|}$$

Zhao et al. present in [24] a second approach intended to reduce the bandwidth consumption imposed by the previous model. The new approach gets inspiration from Privacy Information Retrieval (PIR) approaches [3]. It relies indeed on the construction of two ranges $Q_1\{H_i\}_{i=1}^n$ and $Q_2\{H_i\}_{i=1}^{n+1}$, where $H_{n+1} \in Q_2$ is the true query defined by user U . Once defined Q_1 and Q_2 , such ranges are sent to two independent server NS_1 and NS_2 . Assuming the resolution of DNS queries of type A , each server resolves every query associated with its range, obtaining all the associated IP addresses (defined in [24] as X_i) associated to the query H_i . NS_1 computes $R_1 = \sum_{i=1}^n \otimes X_i$ and NS_2 computes $R_2 = \sum_{i=1}^{n+1} \otimes X_i$. Both R_1 and R_2 are sent to user U , who obtains the resolution associated to H_{n+1} using the expression $X_{n+1} = R_1 \otimes R_2$. As we can observe, the bandwidth consumption of this new approach is considerably smaller than the one in [23], since only two responses (instead of n) are exchanged.

The main benefit of this last proposal, beyond the reduction of bandwidth consumption, is its achievement on preserving the privacy of the queries from attacks at the server side. However, it presents an important drawback due to the necessity of modifying DNS protocol and associated tools. Let us note that the proposal modifies the mechanisms for both querying the servers and responding to the clients. Moreover, it still presents security deficiencies that can be violated by means of active attacks against the communication channel between resolvers and servers. Indeed, attackers controlling the channel can still intercept both range Q_1 and Q_2 . If so, they can easily obtain the true query established by user U by simply applying $Q_1 \setminus Q_2 = H_{n+1}$. Similarly, if attackers successfully intercept both R_1 and R_2 coming from servers NS_1 and NS_2 , they can obtain the corresponding mapping address by performing the same computation expected to be used by user U , i.e., by computing $X_{n+1} = R_1 \otimes R_2$. Once obtain such a value, they can simply infer the original query defined by user U by requesting a reverse DNS mapping of X_{n+1} . Analogously, an active control of the channel can lead attackers to forge resolutions. Indeed, without any additional measures, a legitimate user does not have non-existence proofs to corroborate query failures. This especially relevant on UDP-based lookup services, like the DNS, where delivery of messages is not guaranteed. Attacker can satisfactorily apply these kind of attacks by intercepting, at least, one of the server responses. An attacker can for example intercept R_1 , compute $R_2^* = R_1 \otimes R_3$ (where R_3 is a malicious resolution), and finally send as a resulting

response coming from server NS_2 . Then, the resolver associated to user U will resolve the mapping address as follows: $R_1 \otimes R_2^* = R_1 \otimes R_1 \otimes R_3 = R_3$.

As an alternative to the proposals presented in [23,24], we propose to distribute the load of the set of ranges launched by user U among several servers $NS_1 \dots NS_m$. Unlike the previous schemes, our approach aims at constructing different ranges of queries for every server $NS_1 \dots NS_m$. The ranges will be distributed from $Q\{H_1^{NS_1}\} \dots Q\{H_{\frac{n}{m}}^{NS_1}\}$ to $Q\{H_1^{NS_m}\} \dots Q\{H_{\frac{n}{m}}^{NS_m}\}$. When the responses associated to these queries are obtained from the set of servers, user U verifies that the desired query has been successfully processed. If so, the rest of information is simply discarded. On the contrary, if the query is not processed, i.e., user U does not receive the corresponding response, a new set of ranges is generated and proposed to the set of servers. To avoid the inference attack discussed above, ranges are constructed on independent sessions to preserve information leakage of the legitimate query. Let us note that by using this strategy, we preserve privacy of queries from both server and communication channel. In order to guarantee integrity of queries, authenticity of queries, and non-existence proofs, our proposal relies moreover on the use of the DNS security extensions. We survey the use of DNSSEC in the following section. An evaluation of our approach is presented in Section 5.

4 The DNSSEC Specifications

The Domain Name System SECurity (DNSSEC) extension is a set of specifications of the IETF for guaranteeing authenticity and integrity of DNS Resource Records (RRs) such as NAPTR records. DNSSEC is based on the use of asymmetric cryptography and digital signatures. DNSSEC is often criticized for not being yet deployed after more than ten years of discussions and revisions. It is however the best available solution (when used properly) to mitigate active attacks against the DNS, such as man-in-the-middle and cache poisoning. DNSSEC only addresses threats on the authenticity and integrity of the service. Although early DNSSEC proposals presented clear problems of management associated with its key handling schema, the latest established version of DNSSEC overcomes key management issues based on the Delegation Signer (DS) model proposed in RFCs 3658 and 3755. DNSSEC is being currently deployed on experimental zones, such as Sweden, Puerto Rico, Bulgaria, and Mexico (cf. <http://www.xelerance.com/dnssec/>). At the moment of writing this paper, more than ten thousand DNSSEC zones are enabled (cf. <http://secspider.cs.ucla.edu/>). Deployment at the root level of DNS is currently being debated, although the difficulties of deploying DNSSEC at this level seem to be of political nature rather than technical issues.

The main characteristics of the latest version of DNSSEC are described in RFCs 3658, 3755, 4033, 4034, and 4035. An analysis of threats addressed and handled by DNSSEC is also available in RFC 3833. DNSSEC provides to DNS resolvers origin authentication of Resource Records (RRs) (such as A, CNAME, MX, and NAPTR), as well as RR integrity and authenticated denial of existence (e.g., if a NAPTR record is queried in the global DNS service and it does not exist, a signed proof of non-existence is returned to the resolver). As we pointed out above, DNSSEC allows two different strategies to guarantee authenticity and integrity. On the one hand, administrators of a

given domain zone can digitally sign their zones by employing their own private key and making available to resolvers the corresponding public key. On the other hand, administrators can rely on the use of a chain of trust between parent and child zones that enables resolvers to verify when the responses received from a given query are trustworthy. In order to implement these two strategies, DNSSEC relies on the use of four new DNS RR types: (1) Resource Record Signature (RRSIG) RRs that store the signature associated to every RR in a given zone, (2) DNS Public Key (DNSKEY) RR that contains the specific public key that will allow the resolver to validate the digital signatures of each RR, (3) Delegation Signer (DS) RRs that are added in parent zones to allow delegation functions on child zones, and (4) Next Secure (NSEC) RRs that contain information about the next record in the zone, and that allow the mechanism for verifying the nonexistence of RRs on a given zone. DNSSEC includes two bit flags unused on DNS message's headers to indicate (1) that the resolver accepts unauthenticated data from the server and (2) that those RRs included in the response were previously authenticated by the server.

Regarding the set of keys for signing RRs, one or two key pairs must be generated. If administrators decide to sign zones without a chain of trust, the complete set of RRs of each zone are signed by using a single pair of Zone Signing Keys (ZSKs). On the other hand, if the administrators decide to use a chain of trust between parent and child zones, two key pairs must be generated: a pair of Key Signing Keys (KSKs) is generated to sign the top level DNSKEY RRs of each zone; and a pair of ZSKs keys are used to sign all the RRs of each zone. Several algorithms can be used for the generation of key pairs, such as RSA, DSA (Digital Signature Algorithm), and ECC (Elliptic Curve Cryptosystem). These keys are only used for signatures, and not for encryption of the information. Signatures are hashed by using MD5 or SHA1, being the combination RSA/SHA1 the mandatory signature process that must be implemented at servers and resolvers. The type and length of these keys must be chosen carefully since it significantly affects the size of the response packets as well as the computational load on the server and the response latency. Results in [1] pointed out to an overhead of 3% up to 12% for KSK/ZSK keys based on RSA and length of 1200/1024 bits; and 2% up to 6% for ECC based keys of length 144/136 bits.

The validity period associated with KSK/ZSK keys must also be defined carefully in order to avoid problems with key rollovers, since data signed with previous keys may still be alive in intermediary caches. Synchronization parameters are therefore very important in DNSSEC. Another issue, often referred in the literature as *zone enumeration* or *zone walking*, relies on the use of the NSEC RR. As we pointed out above, NSEC allows chaining the complete set of RRs of a zone to guarantee nonexistence of records and so, it also allows retrieving all the information associated to a given zone. Although the DNSSEC working group originally stated that this is not a real problem (since, by definition, DNS data is or should be public) they proposed an alternative method that uses a new RR called NSEC3 which prevents trivial zone enumeration to introduce a signed hash of the following record instead of including directly its name. Secure storage of trust anchors has also been actively discussed in the literature. Unlike PKI solutions, the chain of trust of DNSSEC offers higher benefits compared to the security of X.509 certificates since the number of keys to protect in DNSSEC is much lower.

5 Evaluation of Our Proposal

This section shows the outcome of our evaluation steered towards measuring the latency penalty due to the use of our approach on a real network scenario for the resolution of DNS and DNSSEC queries. The hardware setup of our experimental scenario is the following. A host R , running on an Intel Core 2 Duo 2 GHz and 1 GB of memory, performs queries of type NAPTR to a global resolution service G . The implementation and deployment of our proposal in R is based on the *Python* language. More specifically, we base our implementation on the module *dnspython* [11] for the construction and resolution of DNS queries; and the module *m2crypto* [12] to access the *OpenSSL* library [4] for the verification of digital signatures defined by DNSSEC.

The global resolution service G is in turn implemented by means of three different hosts: S_1 , that runs on an AMD Duron 1 GHz with 256 MB of memory; S_2 , that runs on an Intel PIII 1 GHz with 512 MB of memory; and S_3 , that runs on an Intel Xeon 2.4 GHz with 1 GB of memory. Servers in G are located on different networks and on different countries: server S_1 is located in North America; and servers S_2 and S_3 are located in Europe. DNS and DNSSEC services configured on each one of these hosts are based on BIND 9.4.2 (cf. <http://www.isc.org/products/BIND/>). The configuration of each server in G consists of a database $\mathcal{N}$ that contains more than twenty thousand NAPTR records generated at random. Each one of these records are linked moreover with appropriate DNSSEC signatures. We use for this purpose the *dnssec-keygen* and *dnssec-signzone* tools that come with BIND 9.4.2. The key sizes are 1200 bits for the generation of Key Signing Keys (KSKs) and 1024 bits for Zone Signing Keys (ZSKs). The generation of keys is based on the RSA implementation of *dnssec-keygen*. Although the use of ECC signatures seems to reduce the storage space of signed zones [1], the algorithm we use is RSA instead of ECC since the latter is not yet implemented in BIND 9.4.2.

We measured in our evaluations the time required for resolving queries from R to G with different testbeds, where the size of the query range of each testbed increments from thirty to more than one hundred. Each testbed consists indeed on the generation of three sets of random queries, one for each $S_i \in G$. Each testbed is launched multiple times towards cumulative series of NAPTR queries. Each series is created at random during the execution of the first testbed, but persistently stored. It is then loaded into the rest of testbeds to allow comparison of results. We split our whole evaluation in four different stages. During the first two stages, the transport layer utilized between R and G is based on the TCP protocol. First stage is used for the resolution of DNS queries, while stage two is used to resolve DNSSEC queries. Similarly, stage three and four are based on UDP traffic for the resolution of, respectively, DNS and DNSSEC queries. During these two last experiments based on DNSSEC, R verifies the integrity and the authenticity of the queries received from the different servers in G . The verification procedures have been implemented as defined in DNSSEC RFCs (cf. Section 4). We show in Figure 1 the results that we obtained during the execution of these four experiments.

We can appreciate by looking at Figure 1 that the latency increases linearly with the size of the range of queries. TCP-based experiments show worst performance than UDP-based queries — due to the overhead imposed by the establishment of sessions. UDP protocol is clearly the best choice for the deployment of our proposal. Given an

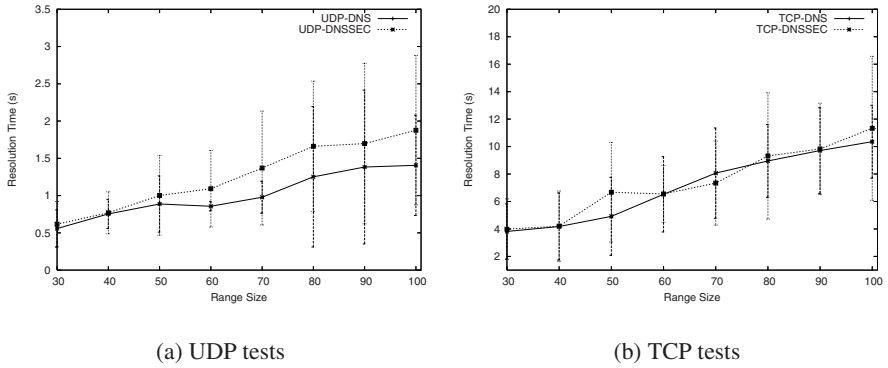


Fig. 1. Evaluation of our proposal

acceptable latency of no more than two seconds, UDP results show that the probability of guessing the true query is $P_i = \frac{1}{3 \cdot 80} = \frac{1}{240} \simeq 0.004167$. We consider this result as satisfactory. In general terms, we should expect that the certainty for obtaining a query i within a range of size n and m different servers is $P_i = \frac{1}{n \cdot m}$.

Besides the difficulties imposed by our model for predicting the original petition, we are conscious of the high bandwidth increase that it represents. This is an important drawback in scenarios where the bandwidth consumption is a critical factor. However, if this is the case, it is possible to reduce the size of the range of queries. Since there is a clear relation between both parameters, i.e., the bandwidth consumption is inversely proportional to the prediction probability, we believe that a proper balance between bandwidth consumption and prediction probability can be enough to enhance the privacy of the service. Let us recall that reducing the size of each range of queries to a fifty per cent, the prediction probability for the attacker is proportionally increased by two. On the other hand, let us observe how the penalty in the response times introduced by DNSSEC is not specially significant, solving the integrity and authenticity problems that appeared in the other approaches. This is the reason why we consider the activation of DNSSEC as a decisive factor for avoiding manipulation network traffic attacks.

6 Conclusion

The use of the DNS (*Domain Name System*) as the underlying technology of new lookup services might have unwanted consequences in their security and privacy. We have analyzed in the first part of this paper privacy issues regarding the use of DNS procedures in the ENUM (*tElephone Number Mapping*) service. The loss of privacy due to the lack of security mechanisms of the DNS data in transit over insecure channels or with dishonest servers is, from our point of view, the main peculiarity of the threat model associated to the ENUM service — compared with the threat model of traditional DNS applications. We have then analyzed in the second part of our work, the use of statistical noise and the construction of range of queries as a possible countermeasure to reduce the risk associated to this threat.

The implementation of our proposal is inspired on a PIR (*Privacy Information Retrieval*) model introducing random noise in the DNS queries. The goal of our model is to reduce privacy threats at both channel (e.g., eavesdroppers trying to infer sensible information from people's queries) and server level (e.g., dishonest servers from silently recording people's queries or habits). The proposal is indeed inspired on two previous works surveyed in Section 3. Security deficiencies detected in both contributions have been addressed, such as response manipulation and range intersections. The combination of our model with the use of DNSSEC allows us to prevent, moreover, from authenticity and integrity threats. The main drawback of our contribution is still a high increase on the bandwidth consumption of the service. We are working on an improvement of our model to address this limitation.

Acknowledgments. The authors graciously acknowledge the financial support received from the following organizations: Spanish Ministry of Science and Education (projects *CONSOLIDER CSD2007-00004 "ARES"* and *TSI2006-03481*).

References

1. Ager, B., Dreger, H., Feldmann, A.: Predicting the DNSSEC Overhead Using DNS Traces. In: 40th Annual Conf. on Information Sciences and Systems, pp. 1484–1489 (2006)
2. Atkins, D., Austein, R.: Threats Analysis of the Domain Name System (DNS). Request for Comments, RFC 3833, IETF (2004)
3. Chor, B., Kushilevitz, E., Goldreich, O., Sudan, M.: Private Information Retrieval. *Journal of the ACM*, 965–981 (1998)
4. Young, E.A., Hudson, T.J.: OpenSSL: The Open Source Toolkit for SSL/TLS, <http://www.openssl.org/>
5. ETSI, Methods and Protocols for Security; part 1: Threat analysis. Technical Specification ETSI TS 102 165-1 V4.1.1 (2003)
6. Faltstrom, P., Mealling, M.: The E.164 to Uniform Resource Identifiers Dynamic Delegation Discovery System Application. Request for Comments, RFC 3761, IETF (2004)
7. Federal Trade Commission. Protecting Consumers from Spam, Spyware, and Fraud. A Legislative Recommendation to Congress (2005)
8. Garcia-Alfaro, J., Barbeau, M., Kranakis, E.: Evaluation of Anonymized DNS Queries. In: 1st Workshop on Security of Autonomous and Spontaneous Networks (SETOP 2008), Locutudy, Brittany, France (October 2008)
9. Mealling, M., Daniel, R.: The Naming Authority Pointer (NAPTR) DNS Resource Record. Request for Comments, RFC 2915, IETF (2000)
10. Mockapetris, P.: Domain Names - Implementation and Specification. Request for Comments, RFC 1035, IETF (1987)
11. Nomium Inc. A DNS Toolkit for Python, <http://www.dnspython.org/>
12. Siong, N.P., Toivonen, H.: Mee Too Crypto, <http://chandlerproject.org/bin/view/Projects/MeTooCrypto>
13. Rosenberg, J., et al.: Session Initiation Protocol. Request for Comments, RFC 3261 (2002)
14. Ingledine, R., Mathewson, N., Syverson, P.F.: Tor: The second-generation Onion Router. In: 13th conference on USENIX Security Symposium (2004)
15. DNSSEC Deployment Initiative, <http://dnssec-deployment.org/>
16. IETF IPsec, <http://www.ietf.org/ids.by.wg/ipsec.html>

17. Meenakshi, S.P., Raghavan, S.V.: Impact of IPSec Overhead on Web Application Servers. In: Advanced Computing and Communications (ADCOM 2006), pp. 652–657 (2006)
18. Ostrovsky, R., Skeith, W.E.: A Survey of Single Database PIR: Techniques and Applications. In: Okamoto, T., Wang, X. (eds.) PKC 2007. LNCS, vol. 4450, pp. 393–411. Springer, Heidelberg (2007)
19. Rossebø, J., Cadzow, S., Sijben, P.: eTVRA, a Threat, Vulnerability and Risk Assessment Method and Tool for eEurope. In: 2nd Int'l Conf. on Availability, Reliability and Security, ARES 2007, Vienna, Austria, pp. 925–933 (2007)
20. Rossebø, J., Cadzow, S., Sijben, P.: eTVRA, a Threat, Vulnerability and Risk Assessment Tool for eEurope. In: Stølen, K., Winsborough, W.H., Martinelli, F., Massacci, F. (eds.) iTrust 2006. LNCS, vol. 3986, pp. 467–471. Springer, Heidelberg (2006)
21. Reed, M.G., Syverson, P.F., Goldschlag, D.M.: Anonymous Connections and Onion Routing. *IEEE Journal on Selected Areas in Communications* 16(4), 482–494 (1998)
22. Sion, R., Carbunar, B.: On the Computational Practicality of Private Information Retrieval. In: Network and Distributed Systems Security Symposium (NDSS) (2007)
23. Zhao, F., Hori, Y., Sakurai, K.: Analysis of Privacy Disclosure in DNS Query. In: IEEE Int'l Conf. on Multimedia and Ubiquitous Engineering, pp. 952–957 (2007)
24. Zhao, F., Hori, Y., Sakurai, K.: Two-Servers PIR Based DNS Query Scheme with Privacy-Preserving. In: IEEE Int'l Conf. on Intelligent Pervasive Computing, pp. 299–302 (2007)

Steganography of VoIP Streams

Wojciech Mazurczyk and Krzysztof Szczypiorski

Warsaw University of Technology,
Faculty of Electronics and Information Technology,
Institute of Telecommunications,
15/19 Nowowiejska Str., 00-665 Warsaw, Poland
{W.Mazurczyk, K.Szczypiorski}@tele.pw.edu.pl

Abstract. The paper concerns available steganographic techniques that can be used for creating covert channels for VoIP (Voice over Internet Protocol) streams. Apart from characterizing existing steganographic methods we provide new insights by presenting two new techniques. The first one is network steganography solution which exploits free/unused protocols' fields and is known for IP, UDP or TCP protocols but has never been applied to RTP (Real-Time Transport Protocol) and RTCP (Real-Time Control Protocol) which are characteristic for VoIP. The second method, called LACK (Lost Audio Packets Steganography), provides hybrid storage-timing covert channel by utilizing delayed audio packets. The results of the experiment, that was performed to estimate a total amount of data that can be covertly transferred during typical VoIP conversation phase, regardless of steganalysis, are also included in this paper.

Keywords: VoIP, information hiding, steganography.

1 Introduction

VoIP is one of the most popular services in IP networks and it stormed into the telecom market and changed it entirely. As it is used worldwide more and more willingly, the traffic volume that it generates is still increasing. That is why VoIP is suitable to enable hidden communication throughout IP networks. Applications of the VoIP covert channels differ as they can pose a threat to the network communication or may be used to improve the functioning of VoIP (e.g. security like in [12] or quality of service like in [13]). The first application of the covert channel is more dangerous as it may lead to the confidential information leakage. It is hard to assess what bandwidth of covert channel poses a serious threat – it depends on the security policy that is implemented in the network. For example, US Department of Defense specifies in [24] that any covert channel with bandwidth higher than 100 bps must be considered insecure for average security requirements. Moreover for high security requirements it should not exceed 1 bps.

In this paper we present available covert channels that may be applied for VoIP during conversation phase. A detailed review of steganographic methods that may be applied during signalling phase of the call can be found in [14]. Here, we introduce two new steganographic methods that, to our best knowledge, were not described earlier.

Next, for each of these methods we estimate potential bandwidth to evaluate experimentally how much information may be transferred in the typical IP telephony call.

The paper is organized as follows. In Section 2 we circumscribe the VoIP traffic and the communication flow. In Section 3, we describe available steganographic methods that can be utilized to create covert channels in VoIP streams. Then in Section 4 we present results of the experiment that was performed. Finally, Section 5 concludes our work.

2 VoIP Communication Flow

VoIP is a real-time service that enables voice conversations through IP networks. It is possible to offer IP telephony due to four main groups of protocols:

- a. *Signalling protocols* that allow to create, modify, and terminate connections between the calling parties – currently the most popular are SIP [18], H.323 [8], and H.248/Megaco [4],
- b. *Transport protocols* – the most important is RTP [19], which provides end-to-end network transport functions suitable for applications transmitting real-time audio. RTP is usually used in conjunction with UDP (or rarely TCP) for transport of digital voice stream,
- c. *Speech codecs* e.g. G.711, G.729, G.723.1 that allow to compress/decompress digitalized human voice and prepare it for transmitting in IP networks.
- d. Other *supplementary protocols* like RTCP [19], SDP, or RSVP etc. that complete VoIP functionality. For purposes of this paper we explain the role of RTCP protocol: RTCP is a control protocol for RTP and it is designed to monitor the Quality of Service parameters and to convey information about the participants in an ongoing session.

Generally, IP telephony connection consists of two phases: a *signalling phase* and a *conversation phase*. In both phases certain types of traffic are exchanged between calling parties. In this paper we present a scenario with SIP as a signalling protocol

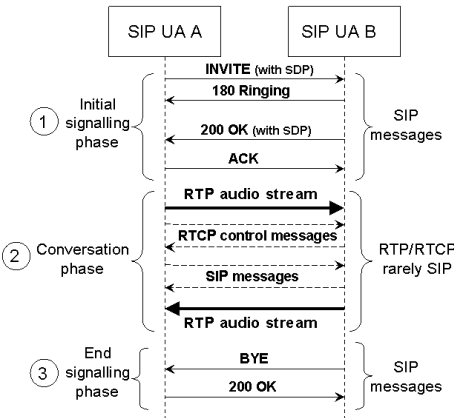


Fig. 1. VoIP call setup based on SIP/SDP/RTP/RTCP protocols (based on [9])

and RTP (with RTCP as control protocol) for audio stream transport. That means that during the signalling phase of the call certain SIP messages are exchanged between SIP endpoints (called: SIP User Agents). SIP messages usually traverse through SIP network servers: proxies or redirects that allow end-users to locate and reach each other. After this phase, the conversation phase begins, where audio (RTP) streams flow bi-directly between a caller and a callee. VoIP traffic flow described above and distinguished phases of the call are presented in Fig. 1. For more clarity we omitted the SIP network server in this diagram. Also potential security mechanisms in traffic exchanges were ignored.

3 Covert Channels in VoIP Streams Overview and New Insights

Besides characterizing IP telephony traffic flow Fig. 1 also illustrates steganographic model used in this paper for VoIP steganography evaluation. The proposed model is as follows. Two users *A* and *B* are performing VoIP conversation while simultaneously utilizing it to send steganograms by means of all possible steganographic methods that can be applied to IP telephony protocols. We assume that both users control their end-points (transmitting and receiving equipment) thus they are able to modify and inspect the packets that are generated and received. After modifications at calling endpoint, packets are transmitted through communication channel which may introduce negative effects e.g. delays, packet losses or jitter. Moreover, while traveling through network packets can be inspected and modified by an active warden [5]. Active wardens act like a semantic and syntax proxy between communication sides. They are able to modify and normalize exchanged traffic in such a way that it does not break, disrupt or limit any legal network communication or its functionality. Thus, active wardens can inspect all the packets sent and modify them slightly during the VoIP call. It must be emphasized however that they may not erase or alter data that can be potentially useful for VoIP non-steganographic (overt) users. This assumption forms important active wardens' rule although sometimes elimination of the covert channel due to this rule may be difficult.

To later, in section 4, practically evaluate covert channels that can be used for VoIP transmission we must first define three important measures that characterizes them and which must be taken into consideration during VoIP streams covert channels analysis. These measures are:

- *Bandwidth* that may be characterized with *RBR* (Raw Bit Rate) that describes how many bits may be sent during one time unit [bps] with the use of all steganographic techniques applied to VoIP stream (with no overheads included) or *PRBR* (Packet Raw Bit Rate) that circumscribe how much information may be covertly sent in one packet [bits/packet],
- *Total amount of covert data* [bits] transferred during the call that may be sent in one direction with the use of all applied covert channels methods for typical VoIP call. It means that, regardless of steganalysis, we want to know how much covert information can be sent during typical VoIP call,
- *Covert data flow distribution during the call* – how much data has been transferred in a certain moment of the call.

We will be referencing to abovementioned measures during the following sections while presenting available steganographic methods for VoIP communication and later during the experiment description and results characterization.

In this section we will provide an overview of existing steganographic techniques used for creation of covert channels in VoIP streams and present new solutions. As described earlier during the conversation phase audio (RTP) streams are exchanged in both directions and additionally, RTCP messages may be sent. That is why the available steganographic techniques for this phase of the call include:

- *IP/UDP/TCP/RTCP* protocols steganography in network and transport layer of TCP/IP stack,
- *RTCP* protocol steganography in application layer of TCP/IP stack,
- *Audio watermarking* (e.g. LSB, QIM, DSSS, FHSS, Echo hiding) in application layer of TCP/IP stack,
- *Codec SID frames* steganography in application layer of TCP/IP stack,
- *Intentionally delayed audio packets* steganography in application layer of TCP/IP stack,
- *Medium dependent* steganographic techniques like HICCUPS [22] for VoWLAN (Voice over Wireless LAN) specific environment in data link layer of TCP/IP stack.

Our contribution in the field of VoIP steganography includes the following:

- Describing RTP/RTCP protocols' fields that can be potentially utilized for hidden communication,
- Proposing *security mechanisms fields steganography* for RTP/RTCP protocols,
- Proposing intentionally delayed audio packets steganographic method called LACK (Lost Audio Packets Steganographic Method).

3.1 IP/TCP/UDP Protocols Steganography

TCP/UDP/IP protocols steganography utilizes the fact that only few fields of headers in the packet are changed during the communication process ([15], [1], [17]). Covert data is usually inserted into redundant fields (provided, but often unneeded) for abovementioned protocols and then transferred to the receiving side. In TCP/IP stack, there is a number of methods available, whereby covert channels can be established and data can be exchanged between communication parties secretly. An analysis of the headers of TCP/IP protocols e.g. IP, UDP, TCP results in fields that are either unused or optional [15], [25]. This reveals many possibilities where data may be hidden and transmitted. As described in [15] the IP header possesses fields that are available to be used as covert channels. Notice, that this steganographic method plays an important role for VoIP communication because protocols mentioned above are present in every packet (regardless, if it is a signalling message, audio packet, or control message). For this type of steganographic method as well as for other protocols in this paper (RTP and RTCP steganography) achieved steganographic bandwidth can be expressed as follows:

$$PRBR_{NS} = \frac{\left(SB_0 + \sum_{j=1}^l SB_j \right)}{l+1} \text{ [bits / packet]} \quad (1)$$

where:

$PRBR_{NS}$ (Packet Raw Bit Rate) denotes bandwidth of the covert channel created by IP/TCP/UDP steganography [bits/packet],
 SB_0 is total amount of bits for IP/TCP/UDP protocols that can be covertly send in the fields of the first packet. This value differs from the value achieved for the following packets because in the first packet initial values of certain fields can be used (e.g. sequence number for TCP protocol),
 SB_j denotes total amount of bits for IP/TCP/UDP protocols that can be covertly sent in the fields of the following packets,
 l is number of packets send besides first packet.

3.2 RTP Protocols Steganography

3.2.1 RTP Free/Unused Fields Steganography

In conversation phase of the call when the voice stream is transmitted, besides protocols presented in section 3.1 also the fields of RTP protocol may be used as a covert channel. Fig. 2 presents the RTP header.

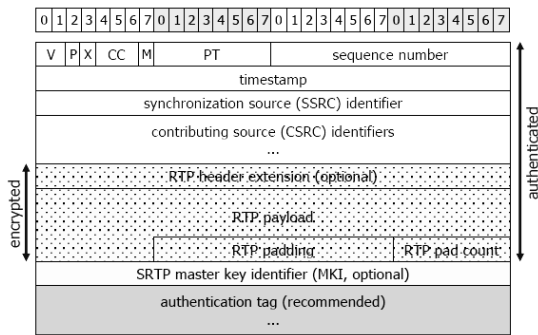


Fig. 2. RTP header with marked sections that are encrypted and authenticated

RTP provides the following opportunities for covert communication:

- *Padding* field may be needed by some encryption algorithms. If the padding bit (P) is set, the packet contains one or more additional padding octets at the end of header which are not a part of the payload. The number of the data that can be added after the header is defined in the last octet of the padding as it contains a count of how many padding octets should be ignored, including itself,
- *Extension header* (when X bit is set) – similar situation as with the padding mechanism, a variable-length header extension may be used,
- Initial values of the *Sequence Number* and *Timestamp* fields – because both initial values of these fields must be random, the first RTP packet of the audio stream may be utilized for covert communication,
- Least significant bits of the *Timestamp* field can be utilized in a similar way as proposed in [6].

It must be emphasized however that steganography based on free/unused/optional fields for RTP protocol (as well as for protocols mentioned in section 3.1) may be potentially eliminated or greatly limited by active wardens. Normalization of RTP headers' fields values (e.g. applied to *Timestamps*) or small modifications applied may be enough to limit covert bandwidth. On the other hand it is worth noting that so far no documented active warden implementation exists.

3.2.2 RTP Security Mechanisms Steganography

There is also another way to create high-bandwidth covert channel for RTP protocol. In Fig. 5 one can see what parts of RTP packet is secured by using encryption (payload and optionally header extension if used) and authentication (*authentication tag*). For steganographic purposes we may utilize security mechanisms' fields. The main idea is to use *authentication tag* to transfer data in a covert manner. In SRTP (Secure RTP) standard [2] it is recommended that this field should be 80 bits long but lower values are also acceptable (e.g. 32 bits). Similar steganographic method that utilizes security mechanism fields was proposed for e.g. IPv6 in [11]. By altering content of fields like *authentication tag* with steganographic data it is possible to create covert channel because data in these fields is almost random due to the cryptographic mechanism operations. That is why it is hard to detect whether they carry real security data or hidden information. Only receiving calling party, as he is in possession of pre-shared key (*auth_key*) is able to determine that. For overt users wrong authentication data in packet will mean dropping it. But because receiving user is controlling its VoIP equipment, when *authentication tag* fields are utilized as covert channel, he is still able to extract steganograms in spite of invalid authentication result.

Thus, most of steganalysis methods will fail to uncover this type of secret communication. The only solution is to strip off/erase such fields from the packets but this is a serious limitation for providing security services for overt users. Moreover it will be violation of the active warden rule (that no protocol's semantic or syntax will be disrupted).

Because the number of RTP packets per one second is rather high (depends on the voice frame generation interval) exploiting this tag provides a covert channel that bandwidth can be expressed as follows:

$$RBR_{SRTP} = SB_{AT} \cdot \frac{1000}{I_p} \quad [bits/s] \quad (2)$$

where:

RBR_{SRTP} (Raw Bit Rate) denotes bandwidth of the covert channel created by RTP security mechanism steganography (in bits/s),

SB_{AT} is total amount of bits in *authentication tag* for SRTP protocol (typically 80 or 32 bits),

I_p describes voice packet generation interval, in milliseconds (typically from 10 to 60 ms).

For example, consider a scenario in which *authentication tag* is 32 bits long and audio packet is generated each 20 ms. Based on equation 2 we can calculate that

$RBR_{SRTT} = 1.6$ kbit/s which is considerably high result for bandwidths of covert channel presented in this paper.

3.3 RTCP Protocol Steganography

3.3.1 RCTP Free/Unused Fields Steganography

To our best knowledge this is the first proposal to use RTCP protocol messages as a covert channel. RTCP exchange is based on the periodic transmission of control packets to all participants in the session. Generally it operates on two types of packets (reports) called: Receiver Report (RR) and Sender Report (SR). Certain parameters that are enclosed inside those reports may be used to estimate network status. Moreover all RTCP messages must be sent in compound packet that consists of at least two individual types of RTCP reports. Fig. 3 presents headers of SR and RR reports of the RTCP protocol.

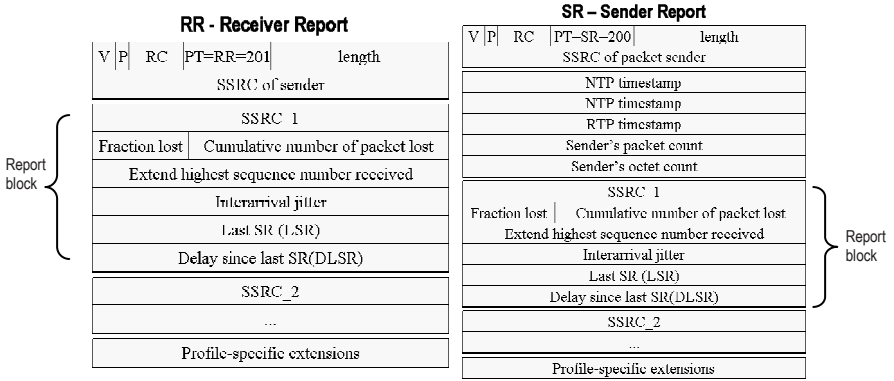


Fig. 3. RTCP Receiver Report (RR) and Sender Report (SR)

For sessions with small number of the participants the interval between the RTCP messages is 5 seconds and moreover sending RTCP communication (with overhead) should not exceed 5% of the session's available bandwidth. For creating covert channels report blocks in SR and RR reports (marked in Fig. 6) may be utilized. Values of the parameters transferred inside those reports (besides SSRC_1 which is the source ID) may be altered, so the amount of information that may be transferred in each packet is 160 bits. It is clear, that if we use this type of steganographic technique, we lose some (or all) of RTCP functionality (it is a cost to use this solution). Other free/unused fields in these reports may be also used in the similar way. For example *NTP Timestamp* may be utilized in a similar way as proposed in [6].

Other RTCP packet types include: SDES, APP or BYE. They can also be used in the same way as SR and RR reports. So the total PRBR for this steganographic technique is as follows:

$$PRBR_{RTCP} = S_{CP} \cdot N_{RB} \cdot S_{RB} \text{ [bits / packet]} \quad (3)$$

where:

$PRBR_{RTCP}$ (Packet Raw Bit Rate) denotes bandwidth of the covert channel created with RCTP Free/Unused Fields Steganography (in bits/packet),
 S_{CP} denotes size of the compound RTCP packet (the number of RTCP packet types inside the compound one),
 N_{RB} is number of report blocks inside each RTCP packet type,
 S_{RB} is the number of bits that can be covertly send in one RTCP report block.

It is also worth noting that RTCP messages are based on IP/UDP protocols, so additionally, for one RTCP packet, both protocols can be used for covert transmission.

To improve capacity of this covert channel one may send RTCP packets more frequently than each 5 seconds (which is default value proposed in standard) although it will be easier to uncover. Steganalysis of this method is not so straightforward as in case of security mechanism fields steganography. Active warden can be used to eliminate or greatly limit the fields in which hidden communication can take place although it will be serious limitation of RTCP functionality for overt users.

3.3.2 RTCP Security Mechanisms Steganography

Analogously as for RTP protocol the same steganographic method that uses SRTP security mechanism may be utilized for RTCP and the achieved RBR_{RTCP} rate is as follows:

$$RBR_{SRTP} = \frac{SB_{AT} \cdot l}{T} \quad [bits/s] \quad (4)$$

where:

RBR_{SRTP} (Raw Bit Rate) denotes bandwidth of the covert channel created with SRTP security mechanism steganography [in bps],
 SB_{AT} is total amount of bits in *authentication tag* for SRTP protocol,
 T denotes duration of the call (in seconds),
 l is number of RTCP messages exchanged during the call of length T .

3.4 Audio Watermarking

The primary application of audio watermarking was to preserve copyrights and/or intellectual properties called DRM (Digital Right Management). However, this technique can be also used to create effective covert channel inside a digital content. Currently there is a number of audio watermarking algorithms available. The most popular methods that can be utilized in real-time communication for VoIP service, include: *LSB* (Least Significant Bit), *QIM* (Quantization Index Modulation), *Echo Hiding*, *DSSS* (Direct Sequence Spread Spectrum), and *FHSS* (Frequency Hopping Spread Spectrum) [3]. For these algorithms the bandwidth of available covert channels depends mainly on the sampling rate and the type of audio material being encoded. Moreover, if covert data rate is too high it may cause voice quality deterioration and increased risk of detection. In Table 1 examples of digital watermarking data rates are presented under conditions that they do not excessively affect quality of the conversation and limit probability of disclosure. Based on those results one can clearly see that, besides *LSB* watermarking, other audio watermarking algorithms covert channels' bandwidth range from few to tens bits per second.

Table 1. Audio watermarking algorithms and their experimentally calculated RBRs

Audio watermarking algorithm	Covert bandwidth RBR (based on [21])	Covert bandwidth RBR (based on [1])
LSB	1 kbps / 1 kHz (of sampling rate)	4 kbps
DSSS	4 bps	22.5 bps
FHSS	-	20.2 bps
Echo Hiding	16 bps	22.3 bps

Thus, we must consider that each audio watermarking algorithm affects perceived quality of the call. That means that there is a necessary tradeoff between the amount of data to be embedded and the degradation in users' conversation. On the other hand by using audio watermarking techniques we gain an effective steganographic method: because of the audio stream flow the achieved bandwidth of the covert channel is constant. Thus, although the bit rate of audio watermarking algorithms is usually not very high, it still may play important role for VoIP streams covert channels.

Steganalysis of audio watermarking methods (besides for LSB algorithm which is easy to eliminate) is rather difficult and must be adjusted to watermarking algorithm used. It must be emphasized however that if hidden data embedding rate is chosen reasonably then detecting of the audio watermarking is hard but possible and in most cases erasing steganogram means great deterioration of voice quality.

3.5 Speech Codec Silence Insertion Description (SID) Frames Steganography

Speech codecs may have built-in or implement mechanisms like Discontinuous Transmission (DTX)/VAD (Voice Activity Detection)/CNG (Comfort Noise Generation) for network resources (e.g. bandwidth) savings. Such mechanisms are able to determine if voice is present in the input signal. If it is present, voice would be coded with the speech codec in other case, only a special frame called Silence Insertion Description (SID) is sent. If there is a silence, instead of sending large voice packets that do not contain conversation data only small amount of bits are transmitted. Moreover, during silence periods, SID frames may not be transferred periodically, but only when the background noise level changes. The size of this frame depends on the speech codec used e.g. for G.729AB it is 10 bits per frame while for G.723.1 it is 24 bits per frame. Thus, when DTX/VAD/CNG is utilized, during the silence periods SID frames can be used to covertly transfer data by altering information of background noise with steganogram. In this case no new packets are generated and the covert bandwidth depends on the speech codec used. It is also possible to provide higher bandwidth of the covert channel by influencing rate at which SID frames are issued. In general, the more of these frames are sent the higher the bandwidth of the covert channel. It must be however noted that the covert bandwidth for this steganographic is rather low. What is important, for this steganographic method steganalysis is simple to perform. Active warden that is able to modify some of the bits in SID frames (e.g. least significant) can eliminate or greatly reduce the bandwidth of this method.

3.6 LACK: Intentionally Delayed Audio Packets Steganography

To our best knowledge this is the first proposal of using intentionally delayed (and in consequence lost) packets payloads as a covert channel for VoIP service. Although

there was an attempt how to use channel erasures at the sender side for covert communication [20] but this solution characterizes low bandwidth especially if we use it for VoIP connection (where the packet loss value must be limited). It is natural for IP networks that some packets can be lost due to e.g. congestion. For IP telephony, we consider a packet lost when:

- It does not reach the destination point,
- It is delayed excessive amount of time (so it is no longer valid), and that is why it may not be used for current voice reconstruction in the receiver at the arrival time.

Thus, for VoIP service when highly delayed packet reaches the receiver it is recognized as lost and then discarded. We can use this feature to create new steganographic technique. We called this method LACK (Lost Audio Packets Steganographic Method). In general, the method is intended for a broad class of multimedia, real-time applications. The proposed method utilizes the fact that for usual multimedia communication protocols like RTP excessively delayed packets are not used for reconstruction of transmitted data at the receiver (the packets are considered useless and discarded). The main idea of LACK is as follows: at the transmitter, some selected audio packets are intentionally delayed before transmitting. If the delay of such packets at the receiver is considered excessive, the packets are discarded by a receiver not aware of the steganographic procedure. The payload of the intentionally delayed packets is used to transmit secret information to receivers aware of the procedure. For unaware receivers the hidden data is “invisible”.

Thus, if we are able to add enough delay to the certain packets at the transmitter side they will not be used for conversation reconstruction. Because we are using legitimate VoIP packets we must realize that in this way we may influence conversation quality. That is why we must consider the accepted level of packet loss for IP telephony and do not exceed it. This parameter is different for various speech codecs as researched in [16] e.g. 1% for G.723.1, 2% for G.729A, 3% for G.711 (if no additional mechanism is used to cope with this problem) or even up to 5% if mechanisms like PLC (Packet Loss Concealment) is used. So the number of packets that can be utilized for proposed steganographic method is limited. If we exceed packet loss threshold for chosen codec then there will be significant decrease in voice quality.

Let us consider RTP (audio) stream (S) that consists of n packets (a_n):

$$S = (a_1, a_2, a_3, \dots, a_n) \text{ and } n = T / I_f \quad (5)$$

where:

S denotes RTP (audio) stream,

a_n is n -th packet in the audio stream S ,

n a number of packets in audio stream.

For every packet (a_n) at the transmitter output total delay (d_T) is equal to:

$$d_T(a_n) = \sum_{m=1}^3 d_m \quad (6)$$

where:

d_1 is speech codec processing delay,

d_2 is codec algorithm delay,

d_3 is packetization delay.

Now, from stream S we choose i -th packet a_i with a probability (p_i):

$$p_i < p_{Lmax} \quad (7)$$

where:

$p_{Lmax} \in \{1\%, 5\%\}$ where 1% packet loss ratio is for VoIP without PLC mechanism and 5% packet loss ratio is for VoIP with PLC mechanism.

To be sure that the RTP packet will be recognized as lost at the receiver, as mentioned earlier, we have to delay it by certain value. For the proposed steganographic method two important parameters must be considered and set to the right value: amount of time by which the chosen packet is delayed (d_d), to ensure that it will be considered as lost at the receiver side and the packet loss probability (p_i) for this steganographic method, to ensure that in combination with p_{Lmax} probability will not degrade perceived quality of the conversation. To properly choose a delay value, we must consider capacity of the receiver's de-jitter buffer. The de-jitter buffer is used to alleviate the jitter effect (variations in packets arrival time caused by queuing, contention and serialization in the network). Its value (usually between 30-70 ms) is important for the end-to-end delay budget (which should not exceed 150 ms). That is why we must add d_d delay (de-jitter buffer delay) to the d_T value for the chosen packet (a_i). If we ensure that d_d value is equal or greater than de-jitter buffer delay at the receiver side the packet will be considered lost. So the total delay (d_T) for a_i packets with additional d_d delay looks as follows (8):

$$d_T(a_i) = \sum_{m=1}^4 d_m \quad (8)$$

where d_d is de-jitter buffer delay.

Now that we are certain that the chosen packet (a_i) is considered lost at the receiver, we can use this packet's payload as a covert channel.

As mentioned earlier, the second important measure for proposed steganographic method is a probability p_i . To properly calculate its value we must consider the following simplified packet loss model:

$$p_T = 1 - (1 - p_N)(1 - p_i) \quad (9)$$

where:

p_T denotes total packet loss probability in the IP network that offers VoIP service with the utilizing of delayed audio packets,

p_N is a probability of packet loss in the IP network that offers VoIP service without the utilizing delayed audio packets (network packet loss probability),

p_i denotes a maximum probability of the packet loss for delayed audio packets.

When we transform (9) to calculate p_i we obtain:

$$p_i \leq \frac{p_T - p_N}{1 - p_N} \quad (10)$$

From (10) one can see that probability p_i must be adjusted to the network conditions. Information about network packet loss probability may be gained e.g. from the

RTCP reports during the transmission. So, based on earlier description, we gain a covert channel with *PRBR* (Packet Raw Bit Rate) that can be expressed as follows:

$$PRBR = r \cdot \frac{I_f}{1000} \cdot p_i \quad [bits / packet] \quad (11)$$

where r is the speech codec rate.

And available bandwidth expressed in *RBR* (Raw Bit Rate) can be described with the following equation (12):

$$RBR = r \cdot p_i \quad [bits / s] \quad (12)$$

For example, consider a scenario with G.711 speech codec where: speech codec rate: $r = 64$ kbit/s and $p_i = 0.5\%$ and $I_f = 20$ ms. For these exemplary values *RBR* is 320 b/s and *PRBR* is 6.4 bits/packet. One can see that the available bandwidth of this covert channel is proportional to the speech codec frame rate, the higher the rate, the higher the bandwidth. So the total amount of information (I_T) that can be covertly transmitted during the call of length d (in seconds) is:

$$I_T = d \cdot RBR = d \cdot r \cdot p_i \quad [bits] \quad (13)$$

Proposed steganographic method has certain advantages. Most of all, although it is an application layer steganography technique, it is less complex than e.g. most audio steganography algorithms and the achieved bandwidth is comparable or even higher.

Steganalysis of LACK is harder than in case of other steganographic methods that are presented in this paper. This is mainly because it is common for IP networks to introduce losses. If the amount of the lost packets used for LACK is kept reasonable then it may be difficult to uncover hidden communication. Potential steganalysis methods include:

- Statistical analysis of the lost packets for calls in certain network. This may be done by passive warden (or other network node) e.g. based on RTCP reports (Cumulative number of packets lost field) or by observing RTP streams flow (packets' sequence numbers). If for some of the observed calls the number of lost packets is higher than it can indicate potential usage of the LACK method,
- Active warden which analyses all RTP streams in the network. Based on the SSRC identifier and fields: *Sequence number* and *Timestamp* from RTP header it can identify packets that are already too late to be used for voice reconstruction. Then active warden may erase their payloads fields or simply drop them. One problem with this steganalysis method is how greatly the packets' identifying numbers must differ from other packets in the stream to be discarded without eliminating really delayed packets that may be still used for conversation. The size of jitter buffer at the receiver is not fixed (and may be not constant) and its size is unknown to active warden. If active warden drops all delayed packets then it could remove packets that still will be usable for voice reconstruction. In effect, due to active warden operations quality of conversation may deteriorate.

Further in-depth steganalysis for LACK is surely required and is considered as future work.

3.7 Medium Dependent Steganography

Medium dependent steganography typically uses layer 1 or layer 2 of ISO OSI RM. For VoIP e.g. in homogenous WLAN environment data link layer methods that depend on available medium like HICCUPS [22] system can be utilized. Exemplary, the data rate for this system is 216 kbit/s (IEEE 802.11g 54 Mbit/s, changing of frame error rate from 1.5% into 2.5%, bandwidth usage 40%).

It must be emphasized however that this steganographic method is difficult to implement as it requires modification to network cards. Moreover, steganalysis for HICCUPS is difficult too as it is necessary to analyze frames in physical layer of OSI RM model.

4 Experimental Evaluation of VoIP Streams Covert Channels Bandwidth

Total achieved covert channel bandwidth (B_T) for the whole VoIP transmission is a sum of each, particular bandwidth of each steganographic methods that are used during voice transmission (each steganographic subchannel). It can be expressed as follows:

$$B_T = \sum_{j=1}^k B_j \quad (14)$$

where:

B_T denotes a total bandwidth for the whole VoIP voice transmission (may be expressed in RBR or PRBR),

B_j describes a bandwidth of the covert channel created by each steganographic method used during VoIP call (may be expressed in RBR or PRBR),

k is a number of steganographic techniques used for VoIP call.

The value of B_T is not constant and depends on the following factors:

- The number of *steganographic techniques* applied to the VoIP call,
- The *choice of the speech codec* used. Three important aspects must be considered here: *compression rate* (e.g. G.711 achieves 64 kbit/s while G.729AB only 8 kbit/s), *size of the voice frame* that is inserted into each packet and *voice packet generation interval*. Compression rate influences the available bandwidth of the steganographic methods that rely on it. The size of the voice frame (typically from 10 to 30 ms) and voice packet generation interval influence the number of packets in audio stream.
- If the mechanisms like *VAD/CNG/DTX* are used. Some of the speech codecs have those mechanisms built-in, for some of them they must be additionally implemented. These solutions influence the number of packets that are generated during VoIP call. The lower number of packets are transmitted the lower total covert channel bandwidth B_T value.
- The *probability value of the packet loss* in IP network. Firstly, if this value is high we lose certain number of packets that are sent into the network, so the information covertly transferred within them is also lost. Secondly, while using delayed audio packets steganography we must adjust the probability of the intentionally lost

packets to the level that exists inside the network to be sure that the perceived quality of the call is not degenerated.

- Less important steganographic methods specific conditions like: how often are RTCP reports are sent to the receiving party or if security mechanisms for communication are used.

To evaluate measures presented at the beginning of Section 3 the following test scenario, as depicted in Fig. 4, has been setup. Two SIP User Agents were used to make a call – the signalling messages were exchanged with SIP proxy and the audio streams flowed directly between endpoints. Moreover RTCP protocol was used to convey information about network performance. Audio was coded with ITU-T G.711 A-law PCM codec (20 ms of voice per packet, 160 bytes of payload). The ACD (Average Call Duration) for this experiment was chosen based on duration of the call for Skype service [21] and for other VoIP providers. In [7] results obtained that ACD for Skype is about 13 minutes, while VoIP providers typically uses a value between 7 and 11 minutes. That is why we chose ACD for the experiment at 9 minutes. There were 30 calls performed and all diagrams show average results.

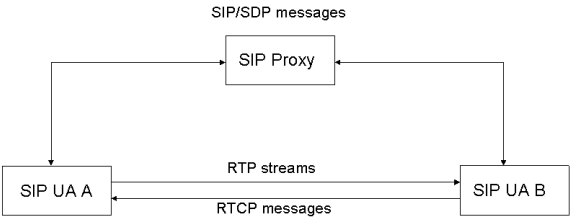


Fig. 4. VoIP steganography experimental test setup

The calls were initiated by *SIP UA A* and the incoming traffic was sniffed at *SIP UA B*. This way we were able to measure covert channel behavior for only one direction traffic flow. Based on the analysis of the available steganographic methods in section 3 the following steganographic techniques were used during the test (and the amount of data that were covertly transferred) as presented in Table 2.

Table 2. Steganographic methods used for experiment and their PRBR

Steganographic method	Chosen PRBR
IP/UDP protocol steg.	32 bits/packet
RTP protocol steg.	16 bits/packet
RTCP steg.	192 bits/packet
LACK	1280 bits/packet (used 0.1% of all RTP packets)
QIM (audio watermarking)	0.6 bits/packet

We chose these steganographic methods for the experiment because they are easy to implement and/or they are our contribution. Besides they are the most natural choice for VoIP communication (based on the analysis' results from section 3) and,

additionally, they represent different layers steganography. It is also important to note that assumed PRBR values for these methods were chosen to be reasonable in steganalysis context. We are interested however only in estimating a total amount of data that can be covertly transferred during the typical conversation phase of the VoIP call, and not how hard is to perform steganalysis. We want to see if the threat posed by steganography applied to VoIP is serious or not.

Achieved results of the experiment are presented below. First in Table 3 traffic flow characteristics, that were captured during performed VoIP calls are presented.

Table 3. Types of traffic distribution average results

Type of traffic	Percent [%]
SIP messages	0.016
RTP packets	99.899
RTCP reports	0.085

From Table 3 can be concluded that the steganographic methods that that utilizes RTP packets have the most impact on VoIP steganography as they cover 99.9% of the whole VoIP traffic. Next in Fig. 5 and Fig. 6 averaged results of the covert data flow distribution (RBR and PRBR respectively) during the average call are presented.

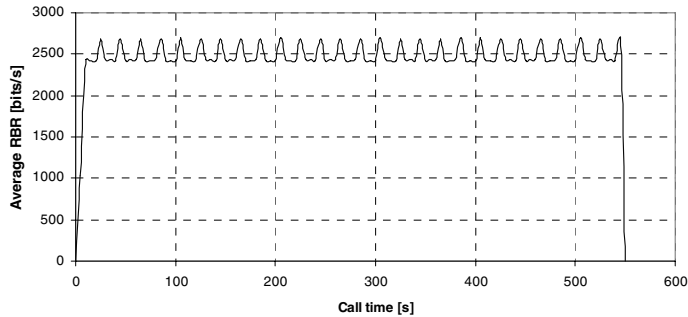


Fig. 5. Covert transmission data flow distribution for the experimental setup

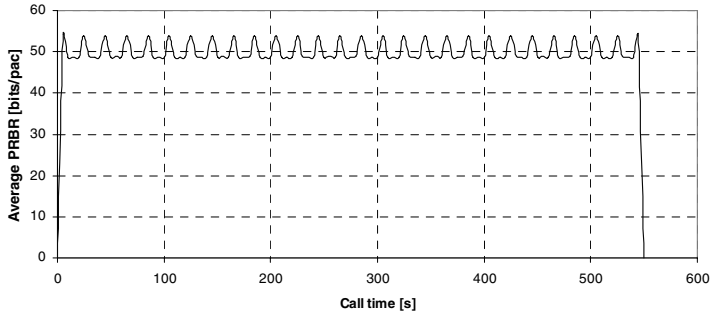


Fig. 6. PRBR during the average call

As one can see VoIP covert channels bandwidth expressed in RBR and PRBR changes in rather constant range during the call (between 2450 and 2600 bits/s for RBR and between 48 and 53 bits/packet for PRBR). The periodic peaks for curves presented in both figures are caused by steganographic bandwidth provided by LACK method. In every certain period of time packets are selected to be intentionally delayed and their payloads carry steganograms. For instants when these packets reach receiver the steganographic bandwidth increases. For this experiment the following average values were obtained and were presented in Table 4:

Table 4. Experimental results for typical call (for one direction flow only)

Measure	Value	Standard Deviation
Average total amount of covert data	1364170 [bits]	4018.711
Average RBR	2487,80 [bits/s]	4.025
Average PRBR	50,04 [bits/packet]	2.258

From the Table 4 we see that during the typical call one can transfer more than 1.3 Mbits (170 KB) of data in one direction with RBR value at about 2.5 kbit/s (50 bits/packet for PRBR).

Table 5. Types of traffic and theirs covert bandwidth fraction

Type of traffic	Bandwidth fraction [%]	Bandwidth fraction [%] per steganographic method	
RTP packets	99.646	IP/UDP	64.11
		RTP	32.055
		Delayed audio packets	2.633
		Audio watermarking	1.202
RTCP reports	0.354	-	

As results from Table 5 show vast part of covert channels' bandwidth for VoIP is provided by network steganography (for protocols IP/UDP it is about 64% and for RTP 32%). Next steganographic method is delayed audio packets steganography (about 2.6%) and audio watermarking (about 1.2%). RTCP steganography provides only minor bandwidth if we compare it with other methods.

5 Conclusions

In this paper we have introduced two new steganographic methods: one of them is RTP and RTCP protocols steganography and the second is intentionally delayed audio packets steganography (LACK). We also briefly described other existing steganographic

methods for VoIP streams. Next, for chosen steganographic method the experiment was performed. Obtained results showed that during typical VoIP call we are able to send covertly more than *1.3 Mbits* of data in one direction.

Moreover, the next conclusion is that the most important steganographic method in VoIP communication experiment is IP/UDP/RTP protocols steganography, while it provides over 96% of achieved covert bandwidth value. Other methods that contribute significantly are delayed audio packets steganography (about 2.6%) and audio watermarking techniques (about 1.2%).

Based on the achieved results we can conclude that total covert bandwidth for typical VoIP call is high and it is worth noting that not all steganographic methods were chosen to the experiment. Steganalysis may limit achieved bandwidth of the covert channels to some extent. But two things must be emphasized. Firstly, currently there is no documented active warden implementation thus there are no real counter measurements applied in IP networks so all the steganographic methods can be used for this moment. Secondly, analyzing each VoIP packet in active warden for every type of steganography described here can potentially lead to loss in quality due to additional delays – this would require further study in future. So, whether we treat VoIP covert channels as a potential threat to network security or as a mean to improve VoIP functionality we must accept the fact that the number of information that we can covertly transfer is significant.

References

1. Ahsan, K., Kundur, D.: Practical Data Hiding in TCP/IP. In: Proc. of: Workshop on Multimedia Security at ACM Multimedia 2002, Juan-les-Pins, France (2002)
2. Baugher, M., McGrew, D., Naslund, M., Carrara, E., Norrman, K.: The Secure Real-time Transport Protocol (SRTP), IETF, RFC 3711 (2004)
3. Bender, W., Gruhl, D., Morimoto, N., Lu, A.: Techniques for Data Hiding. IBM. System Journal. 35(3,4), 313–336 (1996)
4. Cuervo, F., Greene, N., Rayhan, A., Huitema, C., Rosen, B., Segers, J.: Megaco Protocol Version 1.0. IETF, RFC 3015 (2000)
5. Fisk, G., Fisk, M., Papadopoulos, C., Neil, J.: Eliminating Steganography in Internet Traffic with Active Wardens. In: Petitcolas, F.A.P. (ed.) IH 2002. LNCS, vol. 2578, pp. 18–35. Springer, Heidelberg (2003)
6. Giffin, J., Greenstadt, R., Litwack, P.: Covert Messaging Through TCP Timestamps. In: Proc. of: Privacy Enhancing Technologies Workshop (PET), pp. 194–208 (2002)
7. Guha, S., Daswani, N., Jain, R.: An Experimental Study of the Skype Peer-to-Peer VoIP System. In: Proc. of: IPTPS – Sixth International Workshop on Peer-to-Peer Systems (2006)
8. ITU-T Recommendation H.323: Infrastructure of Audiovisual Services – Packet-Based Multimedia Communications Systems Version 6, ITU-T (2006)
9. Johnston, A., Donovan, S., Sparks, R., Cunningham, C., Summers, K.: Session Initiation Protocol (SIP) Basic Call Flow Examples. IETF, RFC 3665 (2003)
10. Korjik, V., Morales-Luna, G.: Information Hiding through Noisy Channels. In: Proc. of: 4th International Information Hiding Workshop, Pittsburgh, PA, USA, pp. 42–50 (2001)
11. Lucena, N., Lewandowski, G., Chapin, S.: Covert Channels in IPv6. In: Danezis, G., Martin, D. (eds.) PET 2005. LNCS, vol. 3856, pp. 147–166. Springer, Heidelberg (2006)

12. Mazurczyk, W., Kotulski, Z.: New Security and Control Protocol for VoIP Based on Steganography and Digital Watermarking. In: Proc. of: IBIZA 2006, Kazimierz Dolny, Poland (2006)
13. Mazurczyk, W., Kotulski, Z.: New VoIP Traffic Security Scheme with Digital Watermarking. In: Górski, J. (ed.) SAFECOMP 2006. LNCS, vol. 4166, pp. 170–181. Springer, Heidelberg (2006)
14. Mazurczyk, W., Szczypiorski, K.: Covert Channels in SIP for VoIP Signalling. In: Jahankhani, H., Revett, K., Palmer-Brown, D. (eds.) ICGes 2008. CCIS, vol. 12, pp. 65–72. Springer, Heidelberg (2008)
15. Murdoch, S., Lewis, S.: Embedding Covert Channels into TCP/IP. Information Hiding, 247–266 (2005)
16. Na, S., Yoo, S.: Allowable Propagation Delay for VoIP Calls of Acceptable Quality. In: Chang, W. (ed.) AISA 2002. LNCS, vol. 2402, pp. 469–480. Springer, Heidelberg (2002)
17. Petitcolas, F., Anderson, R., Kuhn, M.: Information Hiding – A Survey. IEEE Special Issue on Protection of Multimedia Content (1999)
18. Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A.: SIP: Session Initiation Protocol. IETF, RFC 3261 (2002)
19. Schulzrinne, H., Casner, S., Frederick, R., Jacobson, V.: RTP: A Transport Protocol for Real-Time Applications, IETF, RFC 3550 (2003)
20. Servetto, S.D., Vetterli, M.: Communication Using Phantoms: Covert Channels in the Internet. In: Proc. of IEEE International Symposium on Information Theory (2001)
21. Skype, <http://www.skype.com>
22. Szczypiorski, K.: HICCUPS: Hidden Communication System for Corrupted Networks. In: Proc. of ACS 2003, Międzyzdroje, Poland, October 22–24, 2003, pp. 31–40 (2003)
23. Takahashi, T., Lee, W.: An Assessment of VoIP Covert Channel Threats. In: Proc. of 3rd International Conference on Security and Privacy in Communication Networks (SecureComm 2007), Nice, France (2007)
24. US Department of Defense – Department of Defense Trusted Computer System Evaluation Criteria, DOD 5200.28-STD (The Orange Book) (1985)
25. Zander, S., Armitage, G., Branch, P.: A Survey of Covert Channels and Countermeasures in Computer Network Protocols. IEEE Communications Surveys & Tutorials, 3rd Quarter 2007 9(3), 44–57 (2007)

TrustMAS: Trusted Communication Platform for Multi-Agent Systems

Krzysztof Szczypiorski, Igor Margasiński, Wojciech Mazurczyk,
Krzysztof Cabaj, and Paweł Radziszewski

Warsaw University of Technology, Faculty of Electronics and Information
Technology, 15/19 Nowowiejska Str., 00-665 Warsaw, Poland
{K.Szczypiorski, I.Margasinski, W.Mazurczyk,
K.Cabaj, P.Radziszewski}@elka.pw.edu.pl

Abstract. The paper presents TrustMAS – Trusted Communication Platform for Multi-Agent Systems, which provides trust and anonymity for mobile agents. The platform includes anonymous technique based on random-walk algorithm for providing general purpose anonymous communication for agents. All agents, which take part in the proposed platform, benefit from trust and anonymity that is provided for their interactions. Moreover, in TrustMAS there are StegAgents (SA) that are able to perform various steganographic communication. To achieve that goal, SAs may use methods in different layers of TCP/IP model or specialized middleware enabling steganography that allows hidden communication through all layers of mentioned model. In TrustMAS steganographic channels are used to exchange routing tables between StegAgents. Thus all StegAgents in TrustMAS with their ability to exchange information by using hidden channels form distributed steganographic router (Steg-router).

Keywords: multi agents systems, information hiding, steganography.

1 Introduction

In this paper, we present and evaluate a concept of *TrustMAS* - Trusted Communication Platform for Multi-Agent Systems which was initially introduced in [21]. For this purpose we have developed a *distributed steganographic router* and *steganographic routing protocol*. To evaluate the proposed concept we have analyzed: security, scalability, convergence time, and traffic overheads imposed by TrustMAS. Presented in this paper simulation results proved that proposed system is efficient.

TrustMAS is based on agents and their operations; an agent can be generally classified as stationary or mobile. The main difference between both types is that stationary agent resides only on a single platform (host that agent operates on) and mobile one is able to migrate from one host to another while preserving its data and state.

Generally, systems that utilize agents benefit from improved: fault tolerance (it is harder for intruder to interrupt communication when it is distributed), scalability and flexibility, performance, lightweight design and ability to be assigned to different tasks to perform. Moreover, systems that consist of many agents interacting with each other form MAS (Multi-Agent System). The common applications of MAS include:

- *network monitoring* (IDS/IPS systems like in [13]),
- *network management*,
- *information filtering and gathering* (e.g. Google),
- *building self-healing, high scalable networks or protection systems* (like proposed in [20]),
- *transportation, logistics* and others (e.g. graphics computer games development [9]).

Multi-Agent Systems are usually implemented on platforms which are the tools that simplify implementation of specific systems. The most popular examples of such platforms are: JADE [12], AgentBuilder [1], JACK [11], MadKit [15] or Zeus [24].

Mobile agents, which are used in TrustMAS, create dynamic environment and are able to establish ad-hoc trust relations to perform intended tasks collectively and efficiently. Particularly, challenging goals are authentication process where an identity of agent may be unknown and authorization decisions where a policy should accommodate to distributed and changing structure. Trusted cooperation in heterogeneous MAS environment requires not only trust establishment but also monitoring and adjusting existing relations. Currently, two main concepts of the trust establishment for distributed environment exist:

- *reputation-based* trust management (TM) ([5], [14]), which utilizes information aggregated by system entities to evaluate reputation of chosen entity; basically, decisions are made according to recommendations from other entities where some of them can be better than others; the most popular example of such trust management is PageRank implemented in Google,
- *credential- (or rule-) based* trust management ([3], [2]) that uses secure (e.g. cryptographically signed) statements about a chosen entity; decisions based on this TM are more reliable but require better defined semantics than reputation-based TM.

For MAS environment we propose a *distributed steganographic router* which will provide ability to create the covert channels between chosen agents (StegAgents). Paths between agents may be created with the use of any of the steganographic methods in any OSI RM (Open System Interconnection Reference Model) layer and be adjusted to the heterogeneous characteristics of a given network. This concept of a steganographic router, as stated earlier, is new in the steganography state of the art and also MAS technology seems to be very accurate to implement such router in this environment.

To develop safe and a far-reaching agent communication platform it is required to enhance routing process with anonymity. The first concept of network anonymity was Mixnet proposed by Chaum in [4]. It has become a foundation of modern anonymity systems. The concept of Mixnet *chaining with encryption* has been used in a wide range of applications such as E-mail ([6]), Web browsing [10] and general IP traffic anonymization (e.g. Tor [8]). Other solutions like e.g. Crowds [17], may be considered as simplifications of Mixnet. By means of forwarding traffic for others it is possible to provide every agents' untraceability. The origin of collaboration intent in this manner can be hidden from untrusted agents and eavesdroppers.

2 TrustMAS Concept and Main Components

This section is based on paper [21], where initial concept of TrustMAS was introduced in greater details. Examples of Steg-router operations and other key TrustMAS components may be also found in [21]. The following section only briefly describes proposed solution to focus later mainly on security and performance analysis.

2.1 Trust and Anonymity in TrustMAS

MAS gives an opportunity to build an agents' community. In such environments, like in human society, trust and anonymity become important issues as they enable agents to build and manage their relationships. Taking this into consideration we assume that there are no typical behaviors of the agents involved in the particular MAS community, all agents may exist and live their lives in their own way (we do not define agents' interests and there is no information about characteristics of exchanged messages). Additionally, because TrustMAS is focused on information hiding in MAS, we don't assume that any background traffic exists. Abovementioned assumptions are generic and allow to theoretically describe TrustMAS. In real environment, such as IP networks, a background traffic exists and will aggravate detection of the system.

Moreover, in order to minimize the uncertainty of the interactions each agent in TrustMAS must possess a certain level of trust for other agents. Agents interactions often happen in uncertain, dynamically changing and distributed environment. Trust supports agents in right decisions making, and is usually described as reliability or trustworthiness of the other communication sides. When the trust value is high, the party with which agent is operating gives more chances to succeed e.g. agents need less time to find and achieve their goals. On the contrary, when the trust value is low, the choice of the operating party is more difficult, time-consuming and provides less chances for success. In the proposed TrustMAS platform we provide trust and anonymity for each agent wishing to join it. Main trust model of TrustMAS platform is based on a specific behavior of agents – waiting for expected scenario and following a dialog process means that agents are trusted. Other trust models, not included in this work, depend mainly on application of TrustMAS and can be changed accordingly.

One of the important components in TrustMAS is an anonymous technique based on the random-walk algorithm [18]. It is used to provide anonymous communication for every agent in the MAS platform. The idea of this algorithm is as follows. If the agent wants to send a message anonymously, it sends a message (which contains a destination address) to a randomly chosen agent, which is selected based on the result of the flipping of an asymmetric coin (whether to forward the message to the next random agent or not). The coin asymmetry is described by a probability p_f . The proxy agent forwards the message to the next random proxy agent with the probability of p_f and skips forwarding with a probability of $1-p_f$. This probabilistic forwarding assures anonymity because any agent cannot conclude if messages received in this manner are originated from their direct sender.

TrustMAS benefits from the large number of agents that are operating within it, because if many agents join TrustMAS it will be easier to hide covert communication (exchanged between secretly collaborating agents). Agents are likely to join the proposed MAS platform because they want to use both trust and anonymity services that

are provided for their interactions. To benefit from these features each agent has to follow one rule: if it wants to participate in TrustMAS it is obligated to forward discovery steganographic messages according to the random-walk algorithm (which is described in Section 3.1). This may be viewed as the “cost” that agents have to “pay” in order to benefit from the trusted environment.

2.2 Agents in TrustMAS

In the TrustMAS we distinguish two groups of agents. One of them consists of *Ordinary Agents* (OAs) which use proposed platform to benefit from two security services it provides (trust and anonymity). Members of the second group are *Steganographic Agents* (StegAgents, SAs), that besides OAs functionality, use TrustMAS to perform a covert communication.

The following are features of agents in TrustMAS:

- OAs are not aware of the presence of SAs,
- OAs use TrustMAS to perform overt communication e.g. for anonymous web surfing, secure instant messaging or anonymous file-sharing,
- SAs possess the same basic functionality as OAs but they are capable of exchanging steganograms through covert channels,
- each StegAgent is characterized by its address and steg-capabilities (which describe the steganographic techniques that SA can use to create a hidden channel to communicate with other SAs).
- StegAgents that are localized in TrustMAS platform act as a distributed steganographic router, by exchanging hidden data (steganograms) through covert channels but also if they rely on the end-to-end path between two SAs they are able to convert hidden data from one steganographic method to another,
- if in proposed platform malicious agents exist trying to uncover SAs (and their communication exchange), certain mechanisms are available (described in later sections) to limit potential risk of disclosure,
- StegAgents perform steganographic communication in various ways, especially by utilizing methods in different layers of the TCP/IP model. In particular, SAs may exploit other than application layer steganographic techniques by using specialized middleware enabling steganography through all layers in this model. In some cases there is a possibility to use only application layer steganography i.e. image or audio hiding methods. Hidden communication via middleware in different layers gives opportunity for SAs to establish links outside the MAS platform. Examples of techniques in different layers of the TCP/IP model that enable covert channels include: audio, video, still images, text hiding steganography, protocol (network) steganography or methods that depend on available medium e.g. on WLAN links a HICCUPS [22].

In TrustMAS possibility of utilizing cross-layer steganography has certain advantages. It provides more possibilities of exchanging hidden data and it is harder to uncover. However, building the communication paths with many different steganographic methods may introduce additional delays. Therefore, some state of the art

information hiding techniques may be not sufficient to carry network traffic (reminding that in some steganographic applications delay is not the best measure, because the best one is just to be hidden).

2.3 TrustMAS Three-Plane Architecture

The proposed architecture of the TrustMAS may be described on three planes (Fig. 1). In the *MAS PLATFORMS* plane, the gray areas represent homogenous MAS platforms, black dots represent StegAgents and white ones - Ordinary Agents involved in TrustMAS. StegAgents act as a distributed steganographic router (Steg-Router) as shown on *STEG ROUTING* plane. Connections are possible between StegAgents with the use of hidden channels, located in different network layers (*NETWORK* plane), and at the platform level. As mentioned earlier, the choice of steganographic methods used to communicate between each StegAgents, depends on their steg-capabilities.

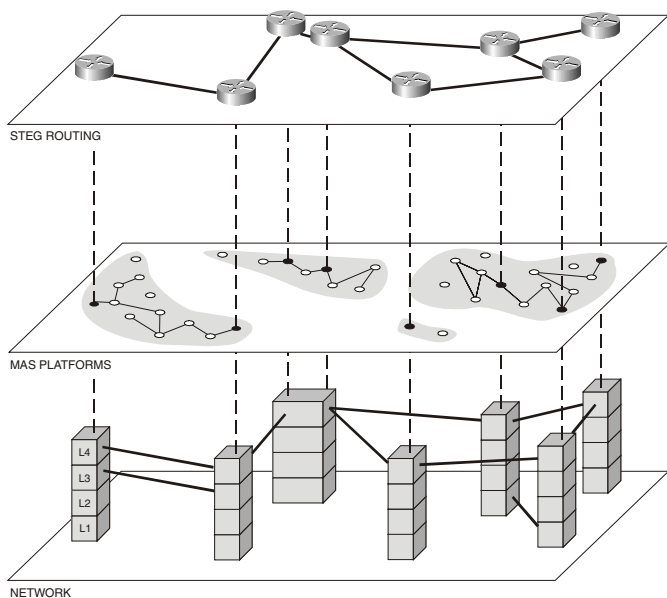


Fig. 1. Architecture of TrustMAS

3 Steg-Router: Distributed Steganographic Router

As mentioned earlier, all StegAgents in TrustMAS with their ability to exchange information by using hidden channels form a distributed steganographic router (Steg-router). Proposed Steg-router is a new concept of building a distributed router to carry/convert hidden data through different types of covert channels, where typically a covert channel utilizes only one steganographic method and is bounded to end-to-end connection. Moreover, it is responsible for creating and maintaining the covert channels (steg-paths) between chosen SAs. Conversion of hidden channels is performed in

heterogeneous environment (e.g. a hidden information in an image converted into the hidden information in WLAN) and the MAS platform is used here as the environment to implement this concept. This gives opportunity to evaluate a new communication method and explore new potential threats in the MAS environment.

The most important part of the proposed Steg-router is a *steganographic routing protocol* (Steg-routing protocol) which is described in next sections. The effective routing protocol is vital for agents' communication and their performance. The routing protocol that will be developed for TrustMAS must take into account all specific features that cannot be found in any other routing environment. That includes providing anonymity with the random walk algorithm (and to perform discovery of new SAs) and usage of steganographic methods. Both these aspects affect performance of the routing convergence. The first one influences updates: in order to provide anonymity service they must be periodic. The second one affects available bandwidth of the links. Due to these characteristic features the steganographic routing protocol for TrustMAS must be designed carefully. For abovementioned reasons none of the existing routing protocols for MANETs (Mobile Ad-hoc Networks) is appropriate. In agents environment, for security reasons, as well as for memory and computation power requirements, provided routing protocol is kept as simple as possible that is why should it belong to a distance vector routing protocols group. We chose a distance vector routing protocol without triggered updates for security reasons – mainly to avoid potential attacks connected with monitoring agents behavior. We can imagine a situation in which the aim of the malicious attacks is to observe agents behavior after removing a random agent from the TrustMAS. If the removed agent was a StegAgent and if the Steg-routing protocol used triggered updates then suddenly there will be a vast activity in the TrustMAS, because triggered updates will be sent to announce changes in the network topology. From the same reason a distance vector protocol was chosen over the link state or a hybrid one.

Proposed ste-routing protocol will be characterized by describing: discovery and maintenance of the neighbors (section 3.1), exchanging the routing tables (section 3.2) and creating steg-links and steg-paths (section 3.3).

3.1 Discovery of New SAs and Neighbors Table Maintenance

As mentioned earlier, all the agents involved in the TrustMAS (both OAs and SAs) perform anonymous exchange based on random-walk algorithm. Thus StegAgents may also utilize this procedure to send anonymous messages with embedded stegmessage (covert data), which consist of StegAgents' addresses and steg-capabilities (available steganographic methods to be used for hidden communication). Such mechanism is analogous to sending hello packets to the neighbors in classical distance vector protocols, where it is responsible for discovery and maintenance of the neighbors table. In the proposed routing protocol random walk algorithm performs only discovery role. The maintenance phase is performed by all SAs that are already involved in TrustMAS and by new SAs that want to join it.

Moreover, each StegAgent maintains in its memory two tables: neighbors and routing table. The neighbors table is created based on information obtained from random-walk algorithm operations. The neighbor relation is formed between two StegAgents if there is a steg-link (exists a covert channel – a connection using steganographic

method that two SAs share) that connects them. Maintenance of the actual information in the neighbors table is achieved by sending, periodically, hello packets through formed steg-links. Such solution allows to identify the situation when one of the StegAgents becomes unavailable.

Based on the information collected from the neighbors the routing table for each SA is formed. Analogously like in standard routing protocols, the routing table possesses best steg-paths (collections of steg-link between each two SAs).

3.2 Routing Tables Exchange

To exchange routing tables between StegAgents steganographic channels are used. In TrustMAS routing updates are sent at regular intervals to finally achieve proactive hidden routing. Routing proactivity provides unlinkability of the steganographic connections and discovery processes. This procedure as well as further hidden communication is cryptographically independent. After the discovery phase, when the new SA's neighbors table has the actual information, it receives entire routing tables from its neighboring StegAgents. Then the routing information is exchanged periodically between SAs. When a new SA receives the routing tables from its neighbors, it learns about other distant SAs and how to reach them. Based on this information formation of new steg-links with other SAs is possible.

If one of the SAs becomes unavailable, the change is detected with the hello mechanism. Then the routing table is updated and the change is sent to all the neighbors in the neighbors table (when there is periodic time to send the entire routing table). Each routing entry in the routing table represents the best available steg-path to distance StegAgent with its metric. The metric is based on three factors: *available capacity* of the steg-links along the end-to-end steg-path, *delays introduced* along the steg-path and *available steganographic methods*. For security reasons some steganographic methods may be preferred over others (e.g. because they are more immune to steganalysis or less affect the content that is used to send covert data).

3.3 Forming Steg-Links and Steg-Paths

Steg-path is an end-to-end connection between two distant StegAgents. Every steg-path is created based on available steg-links between SAs that form the steg-path. The algorithm of forming a steg-path uses metrics that are set for each steg-link. Routing metrics in TrustMAS are calculated as described in section 3.2.

In case there are two equal hops to one destination available, the chosen steg-link is the one that has higher capacity value, introduces less delay and uses more preferred steganographic method. It is also possible that on the one steg-link two or more steganographic methods may be available. In this case metrics are calculated for each steganographic method and the best is chosen to the steg-path. Each SA is also responsible, if it is necessary, for converting steganographic channels according to the next hop SA steg-capabilities. In this way a steganographic router functionality is provided in TrustMAS.

If the routing table is created and up to date then StegAgent is able to send data via hidden channels, where metrics are calculated based on the available steganographic methods.

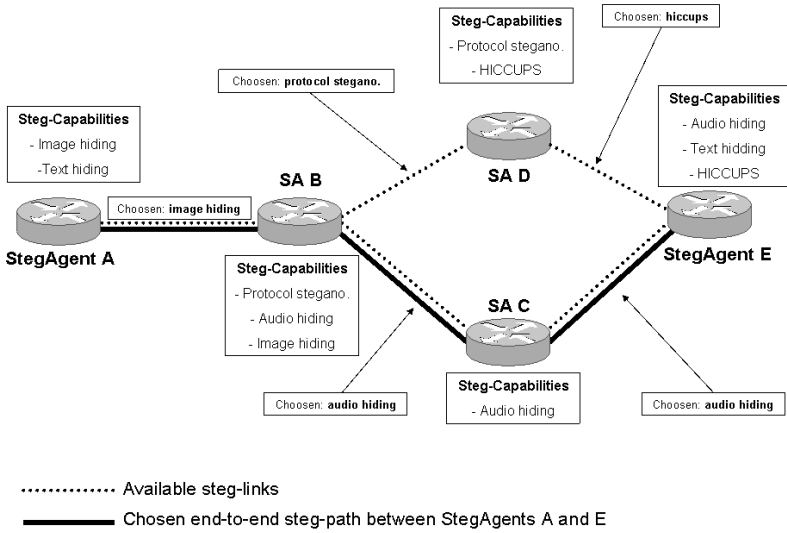


Fig. 2. Forming a steg-path based on available steg-links between SAs

Fig. 2 illustrates a simple example for six StegAgents between which five exemplary steg-links are created based on their steganographic capabilities. The discovery phase of the the StegAgents is omitted. Based on all available steg-links an end-to-end steg-path is formed between StegAgent A and E (through proxy SAs B and C). Created steg-path consists of three steg-links. For each steg-link there is a steganographic method selected which will be used between neighboring StegAgents for hidden communication. Every proxy StegAgent that relays covert exchange is responsible for conversion of hidden data between steganographic methods that it supports (e.g. in Fig. 2 if hidden data is sent through an end-to-end steg-path SA B is obligated to convert a steganogram from image to audio steganography).

4 TrustMAS Security Analysis

Security analysis of TrustMAS will cover an analytical study of protocol based on an entropy measurement model. In 2002 Diaz et al. [7] and Serjantov et al. [18], simultaneously and independently, introduced a new methodology for anonymity measurement based on Shannon's information theory [19]. The information entropy proposed by Shannon can be applied to the anonymity quantification by assignment of probability of being an initiator of a specified action in the system to its particular users, nodes or agents.

The adversary who foists colluding agents on the network can assign probabilities of being the initiator to particular agents. Based on [7] and [18] we can assign such a probability to the predecessor of the first colluding agent from the forwarding path

$$p_{c+1} = 1 - p_f \frac{N - C - 1}{N}. \quad (1)$$

The rest of the agents will be assigned equal probabilities as the adversary has no additional information about them. All colluding agents should not be considered.

$$p_i = \frac{p_f}{N}, \quad (2)$$

then

$$H_{paTM} = \frac{N - p_f(N - C - 1)}{N} \log_2 \left(\frac{N}{N - p_f(N - C - 1)} \right) + \frac{p_f}{N} (N - C - 1) \log_2 \left(\frac{N}{p_f} \right). \quad (3)$$

In the analyzed scenario it is assumed that the adversary has yet colluding agents among nodes which actively anonymize specified request. Practically, the scenario may be different, and what is more, a probability that the adversary can find this group of agents (referred to as an “active set”) also determines the efficiency of the system anonymization [16]. The scenario described above should be called adaptive attack as it is assumed that the adversary has possibilities to adapt an area of his observation to the scope of activity of system users. Though it is important to consider also more general case where the adversary cannot be certain of successive collaboration of proper active agents. This attack will be referred to as a static attack as in this scenario the adversary “injects” colluding agents in a static manner and cannot dynamically predict which random agents will actively anonymize the specified request. A probability that none of the collaborating agents can become a member of the random-walk forwarding path is

$$p_r = \frac{N - C}{N} (1 - p_f) \sum_{i=0}^{\infty} \left(\frac{N - C}{N} p_f \right)^i = 1 - \frac{C}{N - p_f(N - C)}, \quad (4)$$

then an entropy for passive-static attacks equals

$$H_{psTM} = - \frac{C}{N - pf(N - C)} \frac{N - p_f(N - C - 1)}{N} \log_2 \left(\frac{N - p_f(N - C - 1)}{N} \right) + \left(1 - \frac{C}{N - pf(N - C)} \right) p_f \frac{N - C - 1}{N} \log_2 \left(\frac{p_f}{N} \left(1 - \frac{C}{N - pf(N - C)} \right) \right). \quad (5)$$

Figure 3 shows entropy of TrustMAS as a function of the parameter number of colluding agents C for both adaptive and static attacks.

As one can expect, the entropy highly depends on the number of colluding agents. What is more, we can observe a significant impact of the static observation for the anonymity of TrustMAS system. TrustMAS entropy is significantly lower for static attacks than for adaptive scenarios.

Next we will analyze how exactly p_f configuration impacts the entropy of TrustMAS system for both attack scenarios. Figures 4 and 5 show entropy of TrustMAS in the full spectrum of available p_f configuration.

In the adaptive scenario, a low entropy (close to zero) is obtained for low p_f values and a high (close to the maximum) entropy is achieved for large p_f . In the static scenario, the dependency is quite different and the best results are achieved for the lowest

p_f values. As p_f grows, the entropy grows slower logarithmically. This decrease (the static attack) of entropy is slightly faster in the small network, contrary to the adaptive scenario, where, in the small network, the decrease of entropy is slower than for large agent platform. Longer cascades can impose not only larger traffic overheads but can also make it easier for the adversary to become a member of this set and effectively compromise the security of particular systems, especially when we consider small networks. In a small network, agents from the forwarding path constitute a significant part of all network nodes.

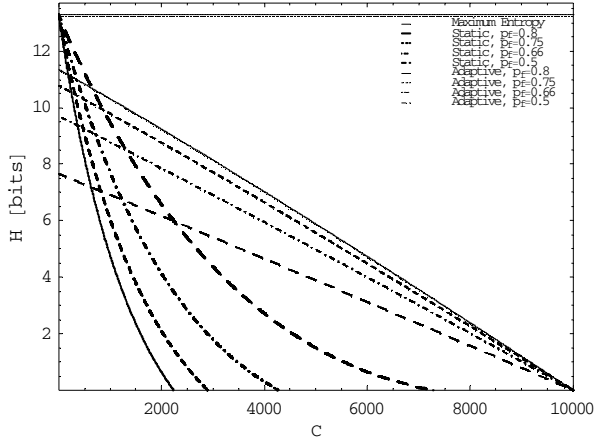


Fig. 3. Impact of the number of collaborating agents C on Entropy of TrustMAS, Static and Adaptive Attacks, $N = 10 \times 10^3$

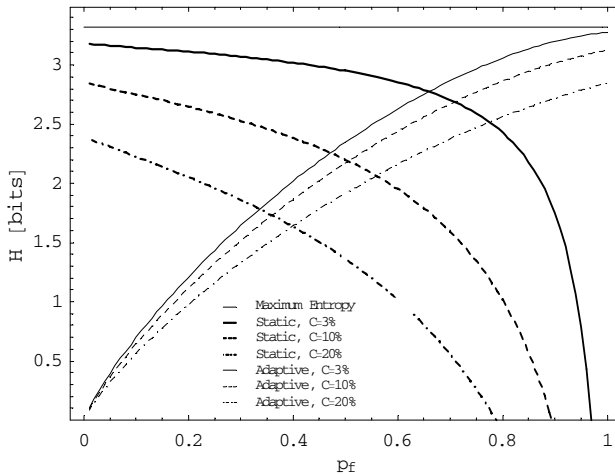


Fig. 4. Entropy of TrustMAS, Static and Adaptive Attacks, $N=10$

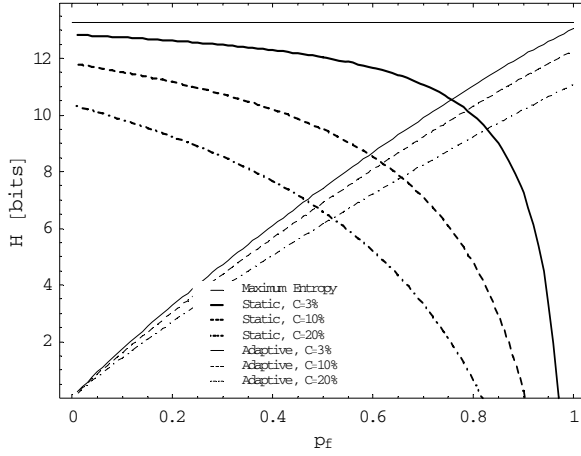


Fig. 5. Entropy of TrustMAS, Static and Adaptive Attacks, $N = 10 \times 10^3$

The results show that p_f configuration of TrustMAS should be in the range of $[0.66 .. 0.8]$. Values lower than 0.66 expose the originator against the adaptive adversary and values higher than 0.8 compromise him by the static attacker. Mean random-walk path length is

$$P = \sum_{i=2}^{\infty} i p_f^{i-2} (1 - p_f) = \frac{p_f - 2}{p_f - 1}, \quad (6)$$

then we can stress that acceptable TrustMAS mean path lengths are:

- minimum: $P_{\min TM} = 4$ for $p_f = 0.66$,
- maximum: $P_{\max TM} = 6$ for $p_f = 0.8$.

Forwarding paths shorter than $P_{\min TM}$ cannot provide sufficient “crowd” of agents which actively anonymize the initiator. If the adversary is yet among this set of agents there should be additional 3 other honest agents. On the other hand, the forwarding paths longer than 6 agents ($P_{\max TM}$) become too easy to enter as the quantity of “crowd” provided by agents that passively anonymize the active set becomes insufficient.

5 Traffic Performance

Based on the results achieved during the security evaluation we have analyzed TrustMAS traffic performance. Our goal was to measure convergence efficiency of proposed routing protocol and its overheads. We have designed and developed own MAS simulation environment (written in C++) that allowed us to evaluate: level of known routes among SAs, traffic generated by routing protocol for SAs, usage of platform’s capacity, and SAs’ links saturation levels. Presented results have been achieved under the following assumptions:

1. Simulation time $T = 30$ min. – after this period we have observed the stable operation of the system.
2. Number of agents $N \in \{250, 500, 1000, 5000, 10000\}$ – includes small and large sizes of agent community.
3. StegAgents percentage $N_{SA} = 10\%$ – typical for open and distributed network environments top limit of agents level controlled by one entity.
4. Probability $p_f \in \{0.66, 0.75, 0.8\}$ – obtained during the security analysis of the TrustMAS.
5. Migration rate $M \in \{0, 120^{-1}, 60^{-1}\} [s^{-1}]$ – during traffic performance analysis we have observed that from $M = 60^{-1}$ TrustMAS operates unstable.
6. We selected six generic steganographic methods:
 - Network (Internet), bandwidth: 300000 delay: 0, probability: 0.90
 - Image, bandwidth: 100, delay: 0, probability: 0.10
 - Video, bandwidth: 100, delay: 0, probability: 0.10
 - Audio, bandwidth: 80, delay: 0, probability: 0.10
 - Text, bandwidth: 80, delay: 0, probability: 0.05
 - Network (HICCUPS), bandwidth: 225000, delay: 0, probability: 0.05
7. Routing timers were chosen based on EIGRP routing protocol defaults. Default values from EIGRP were chosen because it is one of the most efficient distance vector routing protocols.

First steganographic group describes all techniques that involves protocol steganography for Internet network. That includes e.g. IP, UDP/TCP, HTTP, ICMP etc. steganography. Because of these protocols popularity and the amount of traffic they generate, we assumed covert bandwidth's value for this steganographic group at 300 kbit/s and probability of occurrence for StegAgents in TrustMAS platform at 0.9. Next, there are four steganographic groups that correspond to techniques for data hiding in the digital content that may be sent through the network (voice, image, video and text respectively). We assumed a covert bandwidth for these steganographic methods from 80 to 100 bits/s and probability of occurrence for a StegAgent between 0.05 and 0.1. The last steganographic group characterizes more rarely used steganographic methods, e.g. medium-dependant solutions like HICCUPS. As stated earlier, the achieved steganographic bandwidth for this method may be, in certain conditions, about 225 kbit/s and this value was used during simulations.

5.1 Convergence Analysis

We consider the mean convergence level characteristics under dynamically changing network traffic conditions. First we analyze system behavior for no-migration scenario and then for scenarios with migration rates $M = 120^{-1} \text{ min}^{-1}$ and $M = 60^{-1} \text{ min}^{-1}$ respectively. Simulation results show 95% confidence intervals and from 25% to 75% quantiles surrounding the mean levels of known routes.

The full convergence is achieved after about 9 minutes in no-migration scenario, under dynamically changing conditions, for $M=60^{-1}$ the TrustMAS platform is not fully converged.

In the first analyzed scenario the convergence is always achieved – confidence intervals equal the mean value (100% after about 9 minutes). In the second scenario 100% convergence level is possible, however we have observed that mean value does

not reach the optimum. In the last scenario the 100% level of convergence is rarely observed.

As one expects, the higher p_f values are in favor of increasing the convergence time. However all analyzed configurations provide similar results.

5.2 Traffic Overheads Analysis

We have observed that traffic overheads imposed by the steg-routing protocol are between 10 and 12 kbps. Lower values have been obtained for higher migration rate, as when agents leave the platform, the number of exchanged large routing tables diminishes. Similarly to the convergence analysis, we have found that p_f configuration has no significant impact.

The analysis of the TrustMAS capacity usage shows that the steg-routing protocol consumes less than 0.01% of the whole platform bandwidth of steg-links.

The fraction of saturated steg-links in the observed system configuration is negligible. Even for pessimistic high migration rate the saturation of the system is close to zero.

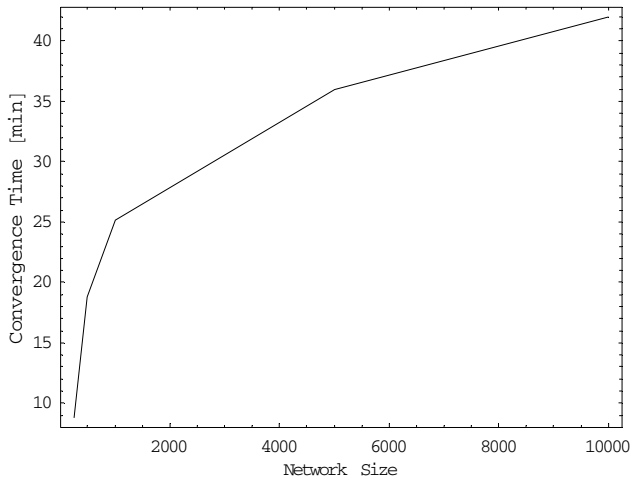


Fig. 6. Convergence Time of TrustMAS

5.3 Scalability Analysis

We have repeated the traffic performance analysis for different sizes of simulated platform. We have found that the network size extends time of convergence process (Fig. 6).

Moreover, we have observed that under the stable operation of large networks ($N \in \{5000, 10000\}$), some insignificant number of undiscovered routes remains. Table 1 contains a summary of the results obtained throughout the analyzed network sizes. There, we can observe how long it takes for TrustMAS to reach a stable operation and its conditions.

Table 1. Convergence of TrustMAS

Network size	250	500	1 000	5 000	10 000
Convergence time [min]	8.8	18.8	25.2	36	42
Undiscovered routes [%]	0	0	0	0,52	0,8

When we consider small networks, with 25 StegAgents collaborating among other 225 agent, less than 10 minutes is required for the proposed StegRouting protocol to provide 100% of routes between SAs. Considering very large platforms, when we have 1000 StegAgents among other 99000 agents, it would take about 40 minutes for TrustMAS to reach the stable operation. However, about 0.8% of routes would remain undiscovered.

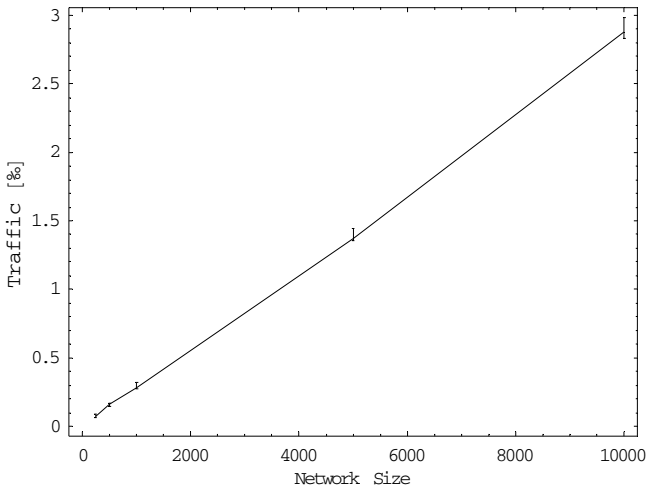


Fig. 7. Capacity Usage of TrustMAS

The impact of the network size on TrustMAS overheads has been shown on Fig. 7. We have found that dependency between scale and the usage of the network’s available bandwidth is linear. In the whole analyzed spectrum of network sizes the level of StegRouting protocol is very low, as even in the very large platform ($N = 10000$) routing management communication consumes less than 3 % of all available system capacity.

The traffic performance analysis has also covered observations of the number of saturated links. In the analyzed scenario the level of links saturated by TrustMAS routing is insignificant. The highest values have been observed for the network size of $N = 1000$ agents (Fig. 8). Further extension of network size is in favor of TrustMAS communication.

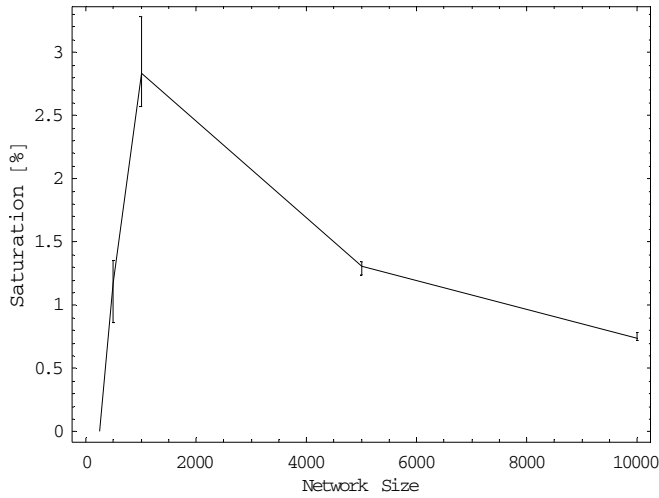


Fig. 8. Mean Level of Saturated Links for TrustMAS

6 Conclusions

We have evaluated efficiency of TrustMAS both for its security and routing performance. Moreover, we have measured overheads imposed by TrustMAS platform required to assure the proper level of agents anonymity and their full connectivity. We have found that the protocol is efficient and the impact of its overheads is not significant.

Steganographic agents in the TrustMAS platform can communicate anonymously in configuration of random-walk algorithm limited to $p_f \in [0.66 .. 0.8]$. This range corresponds to a mean length of forwarding paths from $P \in [4 .. 6]$. Then, each StegAgent should involve about 3 to 5 other agents into the process of the discovery message forwarding to effectively hide an association between its identity and the sent content. Basically, at least 3 additional TrustMAS agents should relay discovery communication to hide the information that the agent is a StegAgent. On the other hand, involving more than 5 agents into the random-walk forwarding process also significantly reduces anonymity of TrustMAS. When we consider platform containing 10-20% colluding agents longer forwarding paths finally facilitate dishonest agents to penetrate the platform area where the StegAgent is hidden.

Using obtained results we have simulated the TrustMAS routing in the configuration of random-walk algorithm with $p_f \in \{0.66, 0.75, 0.8\}$ and the routing timers configuration typical for the popular EIGRP routing protocol. We have found that proposed steganographic routing is efficient and in less than 10 minutes the platform becomes fully converged. Moreover, we have observed that the protocol is robust against fast agents migration ($M = 120^{-1} \text{ s}^{-1}$). A borderline case was observed for high migration rate $M = 60^{-1} \text{ s}^{-1}$ where agents lose all discovered routing information at the average rate of one per minute.

To evaluate practical usefulness of TrustMAS we have measured the traffic overheads imposed by the proposed routing protocol. We have found that it requires about

11 kbps per link which corresponds to less than 0.01% of the system capacity. The fraction of saturated links is also negligible.

The traffic performance analysis confirmed our expectation that the impact of the random-walk p_f configuration is not significant for platform overheads as the short discovery messages generate low traffic. However, higher values of p_f are in favor of the routing efficiency as including 1 more agent into the discovery forwarding process provides the convergence faster by about 1 minute.

The foregoing results have been obtained for platform of hundreds of agents ($N = 250$). Taking into account a possible global and large scale environment of TrustMAS operation we have analyzed behavior of proposed protocol with simultaneous $N \in \{500, 1000, 5000, 10000\}$ communicating agents. We have found that the large scale of the network does not significantly reduce the system performance. However, in very large platforms with $N = 5000..10000$, some imperfection of the proposed routing protocol has been exposed. We should bear in mind that in such a scale the proposed distance vector protocol will discover about 99.5..99.2% of all the available routes. Still, the proposed solution scales well and can operate vastly in large scale networks.

We have proven that the proposed system is secure, fast-convergent and scalable. It can efficiently hide collaboration of designated agents (i.e. StegAgents) in various scale networks (up to ten of thousands agents). Moreover, we have proven that TrustMAS quickly enables connectivity among SAs. The convergence for small and medium size networks is fully achieved and for very large scale networks the proposed distance vector routing protocol does not discover insignificant number of routes. The overheads imposed by routing protocol are negligible.

Future work will include routing protocol improvements to support large platforms (more than 5000 agents) to eliminate negative routing effects (e.g. routing loops) and to gain faster convergence time than currently achieved. Moreover, different analyses of various scenarios for other steganographic profiles may be performed and a concept of TrustMAS may be adopted to the other environments than MAS. Additionally, a prototype of the proposed system for proof-of-concept purposes will be created and analyzed.

Acknowledgment

This material is based upon work supported by the European Research Office of the US Army under Contract No. N62558-07-P-0042. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the European Research Office of the US Army.

References

1. AgentBuilder, <http://www.agentbuilder.com>
2. Blaze, M., Feigenbaum, J., Keromytis, A.: KeyNote: Trust Management for Public-Key Infrastructures. In: Christianson, B., Crispo, B., Harbison, W.S., Roe, M. (eds.) Security Protocols 1998. LNCS, vol. 1550, pp. 59–63. Springer, Heidelberg (1999)

3. Blaze, M., Feigenbaum, J., Lacy, J.: Decentralized Trust Management. In: Proc. of: IEEE 17th Symposium on Research in Security and Privacy, pp. 164–173 (1996)
4. Chaum, D.: Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms. *Communications of the ACM* 24(2), 84–88 (1981)
5. Damiani, E., Vimercati, D., Paraboschi, S., Samarati, P., Violante, F.: A Reputation-based Approach for Choosing Reliable Resources in Peer-to-peer Networks. In: Proc. of: The 9th ACM Conference on Computer and Communications Security CCS 2002, pp. 207–216. ACM Press, Washington (2002)
6. Danezis, G., Dingledine, R., Mathewson, N.: Mixminion: Design of a Type III Anonymous Remailer Protocol. In: Proc. of the IEEE Symposium on Security and Privacy (2003)
7. Diaz, C., Seys, S., Claessens, J., Preneel, B.: Towards Measuring Anonymity. In: Dingledine, R., Syverson, P.F. (eds.) PET 2002. LNCS, vol. 2482, pp. 54–68. Springer, Heidelberg (2003)
8. Dingledine, R., Mathewson, D., Syverson, P.: Tor: The Second Generation Onion Router. In: Proceedings of the 13th USENIX Security Symposium (2004)
9. Doyle, P.: Believability through Context: Using Knowledge in the World to Create Intelligent Characters. In: Proc. of: the International Joint Conference on Autonomous Agents and Multi-Agent Systems (2002)
10. Handel, T., Sandford, M.: Hiding Data in the OSI Network Model. In: Anderson, R. (ed.) IH 1996. LNCS, vol. 1174, pp. 23–38. Springer, Heidelberg (1996)
11. JACK, <http://www.agent-software.com.au>
12. JADE, <http://jade.tilab.com>
13. Jansen, W., Karygiannis, T.: NIST Special Publication 800-19 – Mobile Agent Security (2000)
14. Lee, S., Sherwood, R., Bhattacharjee, B.: Cooperative peer groups in nice. In: Proc.: INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies, vol. 2, pp. 1272–1282. IEEE, Los Alamitos (2003)
15. MADKIT, <http://www.madkit.org>
16. Margasiński, I., Pióro, M.: A Concept of an Anonymous Direct P2P Distribution Overlay System. In: Proceedings of the 22nd IEEE International Conference on Advanced Information Networking and Applications (AINA 2008), Okinawa, Japan (2008)
17. Reiter, M., Rubin, A.: Crowds: Anonymity for Web Transactions. *ACM Transactions on Information and System Security (TISSEC)* 1(1), 66–92 (1998)
18. Serjantov, A., Danezis, G.: Towards an Information Theoretic Metric for Anonymity. In: Dingledine, R., Syverson, P.F. (eds.) PET 2002. LNCS, vol. 2482, pp. 41–53. Springer, Heidelberg (2003)
19. Shannon, C.: A Mathematical Theory of Communication. *The Bell System Technical Journal* 27, 379–423; 623–656 (1948)
20. Sheng, S., Li, K.K., Chan, W., Xiangjun, Z., Xianzhong, D.: Agent-based Self-healing Protection System. *IEEE Transactions* 21(2), 610–618 (2006)
21. Szczypiorski, K., Margasiński, I., Mazurczyk, W.: Steganographic Routing in Multi Agent System Environment - *Journal of Information Assurance and Security (JIAS)*. Dynamic Publishers Inc., Atlanta, GA 30362, USA 2(3), 235–243 (2007)
22. Szczypiorski, K.: HICCUPS: Hidden Communication System for Corrupted Networks. In: Proc. of the Tenth International Multi-Conference on Advanced Computer Systems ACS 2003, Międzyzdroje, Poland, pp. 31–40 (2003)
23. Weiss, G. (ed.): *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, ch. 12, pp. 505–534. MIT Press, Cambridge (1999)
24. Zeus, <http://www.labs.bt.com/projects/agents/zeus>

Computing Exact Outcomes of Multi-parameter Attack Trees

Aivo Jürgenson^{1,2} and Jan Willemson³

¹ Tallinn University of Technology, Raja 15, 12618 Tallinn, Estonia
aivo.jurgenson@eesti.ee

² Elion Enterprises Ltd, Endla 16, 15033 Tallinn, Estonia

³ Cybernetica, Aleksandri 8a, Tartu, Estonia
jan.willemson@gmail.com

Abstract. In this paper we introduce a set of computation rules to determine the attacker's exact expected outcome based on a multi-parameter attack tree. We compare these rules to a previously proposed computational semantics by Buldas *et al.* and prove that our new semantics always provides at least the same outcome. A serious drawback of our proposed computations is the exponential complexity. Hence, implementation becomes an important issue. We propose several possible optimisations and evaluate the result experimentally. Finally, we also prove the consistency of our computations in the framework of Mauw and Oostdijk and discuss the need to extend the framework.

1 Introduction

Attack tree (also called threat tree) approach to security evaluation is several decades old. It has been used for tasks like fault assessment of critical systems [1] or software vulnerability analysis [2,3]. The approach was first applied in the context of information systems (so-called *threat logic trees*) by Weiss [4] and later more widely adapted to information security by Bruce Schneier [5]. We refer to [6,7] for good overviews on the development and applications of the methodology.

Even though already Weiss [4] realised that nodes of attack trees have many parameters in practise, several subsequent works in this field considered attack trees using only one estimated parameter like the cost or feasibility of the attack, skill level required, etc. [3,5,8]. Opel [9] considered also multi-parameter attack trees, but the actual tree computations in his model still used only one parameter at a time. Even though single-parameter attack trees can capture some aspects of threats reasonably well, they still lack the ability to describe the full complexity of the attacker's decision-making process.

A substantial step towards better understanding the motivation of the attacker was made in 2006 by Buldas *et al.* [10]. Besides considering just the cost of the attack, they also used success probability together with probabilities and amount of penalties in the case of success or failure of the attack in their analysis. As a result, a more accurate model of the attack game was obtained and it was later used to analyse the security of several e-voting schemes by Buldas and

Mägi [11]. The model was developed further by Jürgenson and Willemson [12] extending the parameter domain from point values to interval estimations.

However, it is known that the computational semantics given in [10] is both imprecise and inconsistent with the general framework introduced by Mauw and Oostdijk [8] (see Section 2). The motivation of the current paper is to develop a better semantics in terms of precision and consistency. For that we will first review the tree computations of [10] in Section 2 and then propose an improved semantics in Section 3. However, it turns out that the corresponding computational routines are inherently exponential, so optimisation issues of the implementation become important; these are discussed in Section 4. In Section 5 we prove that the new semantics always provides at least the same expected outcome for an attacker as the tree computations of [10]. We also argue that the new semantics is consistent with the framework of Mauw and Oostdijk. Finally, in Section 6 we draw some conclusions and set directions for further work.

2 Background

In order to better assess the security level of a complex and heterogeneous system, a gradual refinement method called *threat tree* or *attack tree method* can be used. The basic idea of the approach is simple — the analysis begins by identifying one or more *primary threats* and continues by splitting the threat into subattacks, either all or some of them being necessary to materialise the primary threat. The subattacks can be divided further etc., until we reach the state where it does not make sense to split the resulting attacks any more; these kinds of non-splittable attacks are called *elementary* or *atomic attacks* and the security analyst will have to evaluate them somehow. During the splitting process, a tree is formed having the primary threat in its root and elementary attacks in its leaves. Using the structure of the tree and the estimations of the leaves, it is then (hopefully) possible to give some estimations of the root node as well. In practise, it mostly turns out to be sufficient to consider only two kinds of splits in the internal nodes of the tree, giving rise to AND- and OR-nodes. As a result, an AND-OR-tree is obtained, forming the basis of the subsequent analysis. An example attack tree originally given by Weiss [4] and adopted from [6] is presented in Figure 1.

We will use the basic multi-parameter attack tree model introduced in [10]. Let us have the AND-OR-tree describing the attacks and assume all the elementary attacks being pairwise independent. Let each leaf X_i have the following parameters:

- Cost_i – the cost of the elementary attack
- p_i – success probability of the attack
- π_i^- – the expected penalty in case the attack was unsuccessful
- π_i^+ – the expected penalty in case the attack was successful.

Besides these parameters, the tree has a global parameter **Gains** showing the benefit of the attacker in the case he is able to mount the root attack. For practical

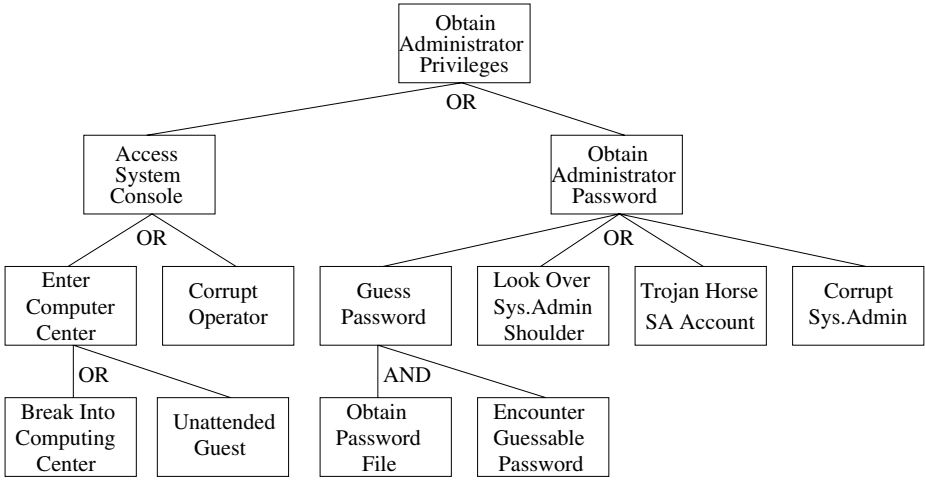


Fig. 1. Example of an attack tree

examples on how to evaluate those parameters for real-life attacks, please refer to [11] and [13].

The paper [10] gives a simple computational semantics to the attack trees, which has further been extended to interval estimates in [12]. After the above-mentioned parameters have been estimated for the leaf nodes, a step-by-step propagation algorithm begins computing the same parameters for all the internal nodes as well, until the root node has been reached. The computational routines defined in [10] are the following:

- For an OR-node with child nodes with parameters $(\text{Cost}_i, p_i, \pi_i^+, \pi_i^-)$ ($i = 1, 2$) the parameters $(\text{Cost}, p, \pi^+, \pi^-)$ are computed as:

$$(\text{Cost}, p, \pi^+, \pi^-) = \begin{cases} (\text{Cost}_1, p_1, \pi_1^+, \pi_1^-), & \text{if } \text{Outcome}_1 > \text{Outcome}_2 \\ (\text{Cost}_2, p_2, \pi_2^+, \pi_2^-), & \text{if } \text{Outcome}_1 \leq \text{Outcome}_2 \end{cases},$$

$$\text{Outcome}_i = p_i \cdot \text{Gains} - \text{Cost}_i - p_i \cdot \pi_i^+ - (1 - p_i) \cdot \pi_i^-.$$

- For an AND-node with child nodes with parameters $(\text{Cost}_i, p_i, \pi_i^+, \pi_i^-)$ ($i = 1, 2$) the parameters $(\text{Cost}, p, \pi^+, \pi^-)$ are computed as follows:

$$\begin{aligned} \text{Costs} &= \text{Costs}_1 + \text{Costs}_2, \quad p = p_1 \cdot p_2, \quad \pi^+ = \pi_1^+ + \pi_2^+, \\ \pi^- &= \frac{p_1(1 - p_2)(\pi_1^+ + \pi_2^-) + (1 - p_1)p_2(\pi_1^- + \pi_2^+)}{1 - p_1p_2} + \\ &\quad + \frac{(1 - p_1)(1 - p_2)(\pi_1^- + \pi_2^-)}{1 - p_1p_2}. \end{aligned}$$

The formula for π^- represents the average penalty of an attacker, assuming that at least one of the two child-attacks was not successful. For later computations, it will be convenient to denote expected expenses associated with

the node i as $\text{Expenses}_i = \text{Cost}_i + p_i \cdot \pi_i^+ + (1 - p_i) \cdot \pi_i^-$. Then it is easy to see that in an AND-node the equality $\text{Expenses} = \text{Expenses}_1 + \text{Expenses}_2$ holds. Note that the formulae above have obvious generalisations for non-binary trees.

At the root node, its **Outcome** is taken to be the final outcome of the attack and the whole tree is considered to be beneficial for a rational attacker if $\text{Outcome} > 0$. Following the computation process it is possible to collect the corresponding set of leaves which, when carried out, allow the attacker to mount the root attack and get the predicted outcome. Such leaf sets will subsequently be called *attack suites*.¹

However, while being very fast to compute, this semantics has several drawbacks:

1. In order to take a decision in an OR-node, the computational model of [10] needs to compare outcomes of the child nodes and for that some local estimate of the obtained benefit is required. Since it is very difficult to break the total root gain into smaller benefits, the model of [10] gives the total amount of **Gains** to the attacker for each subattack. This is clearly an overestimation of the attacker's outcome.
2. In an OR-node, the model of [10] assumes that the attacker picks exactly one descendant. However, it is clear that in practise, it may make sense for an attacker to actually carry out several alternatives if the associated risks and penalties are low and the success probability is high.
3. There is a general result by Mauw and Oostdijk [8] stating which attack tree computation semantics are inherently consistent. More precisely, they require that the semantics of the tree should remain unchanged when the underlying Boolean formula is transformed to an equivalent one (e.g. to a disjunctive normal form). Semantics given in the [10] are not consistent in this sense. For example, let's take two attack trees, $T_1 = A \vee (B \& C)$ and $T_2 = (A \vee B) \& (A \vee C)$, both having same parameters $\text{Gains} = 10000$, $p_A = 0.1$, $p_B = 0.5$, $p_C = 0.4$, $\text{Expenses}_A = 1000$, $\text{Expenses}_B = 1500$, $\text{Expenses}_C = 1000$. Following the computation rules of [10], we get $\text{Outcome}_{T_1} = 8000$ and $\text{Outcome}_{T_2} = 6100$, even though the underlying Boolean formulae are equivalent.

The aim of this paper is to present an exact and consistent semantics for attack trees. The improved semantics fixes all the three abovementioned shortcomings. However, a major drawback of the new approach is the increase of the computational complexity from linear to exponential (depending on the number of elementary attacks). Thus finding efficient and good approximations becomes a vital task. In this paper, we will evaluate suitability of the model of [10] as an approximation; the question of better efficient approximations remains an open problem for future research.

¹ Note that our terminology differs here from the one used by Mauw and Oostdijk [8]. Our attack suite would be just attack in their terms and their attack suite would be the set of all possible attack suites for us.

3 Exact Semantics for the Attack Trees

3.1 The Model

In our model, the attacker behaves as follows.

- First, the attacker constructs an attack tree and evaluates the parameters of its leaves.
- Second, he considers all the potential attack suites, i.e. subsets $\sigma \subseteq \mathcal{X} = \{X_i : i = 1, \dots, n\}$. Some of these materialise the root attack, some of them do not. For the suites that do materialise the root attack, the attacker evaluates their outcome for him.
- Last, the attacker decides to mount the attack suite with the highest outcome (or he may decide not to attack at all if all the outcomes are negative).

Note that in this model the attacker tries all the elementary attacks independently. In practise, this is not always true. For example, if the attacker has already failed some critical subset of the suite, it may make more sense for him not to try the rest of the suite. However, the current model is much more realistic compared to the one described in [10], since now we allow the attacker to plan its actions with redundancy, i.e. try alternative approaches to achieve some (sub)goal.

3.2 Formalisation

The attack tree can be viewed as a Boolean formula $\mathcal{F}$ composed of the set of variables $\mathcal{X} = \{X_i : i = 1, \dots, n\}$ (corresponding to the elementary attacks) and conjunctives $\vee$ and $\&$. Satisfying assignments $\sigma \subseteq \mathcal{X}$ of this formula correspond to the attack suites sufficient for materialising the root attack.

The exact outcome of the attacker can be computed as

$$\text{Outcome} = \max\{\text{Outcome}_\sigma : \sigma \subseteq \mathcal{X}, \mathcal{F}(\sigma := \text{true}) = \text{true}\}. \quad (1)$$

Here Outcome_σ denotes the expected outcome of the attacker if he decides to try the attack suite σ and $\mathcal{F}(\sigma := \text{true})$ denotes evaluation of the formula $\mathcal{F}$, when all of the variables of σ are assigned the value **true** and all others the value **false**. The expected outcome Outcome_σ of the suite σ is computed as follows:

$$\text{Outcome}_\sigma = p_\sigma \cdot \text{Gains} - \sum_{X_i \in \sigma} \text{Expenses}_i, \quad (2)$$

where p_σ is the success probability of the attack suite σ .

When computing the success probability p_σ of the attack suite σ we must take into account that the suite may contain redundancy and there may be (proper) subsets $\rho \subseteq \sigma$ sufficient for materialising the root attack. Because we are using the full suite of σ to mount an attack, those elementary attacks in the $\sigma \setminus \rho$ will

contribute to the success probability of p_ρ with $(1 - p_j)$. Thus, the total success probability can be computed as

$$p_\sigma = \sum_{\rho \subseteq \sigma} \prod_{X_i \in \rho} p_i \prod_{X_j \in \sigma \setminus \rho} (1 - p_j). \quad (3)$$

$\mathcal{F}(\rho := \text{true}) = \text{true}$

Note that the formulae (1), (2) and (3) do not really depend on the actual form of the underlying formula $\mathcal{F}$, but use it only as a Boolean function. As a consequence, our framework is not limited to just AND-OR trees, but can in principle accommodate other connectives as well. Independence of the concrete form will also be the key observation when proving the consistency of our computation routines in the framework of Mauw and Oostdijk (see Proposition 1 in Section 5).

3.3 Example

To explain the exact semantics model of the attack trees, we give the following simple example. Lets consider the attacktree with the Boolean formula $T = (A \vee B) \& C$ with all elementary attacks (A, B, C) having equal parameters $p = 0.8$, $\text{Cost} = 100$, $\pi^+ = 1000$, $\pi^- = 1000$ and $\text{Gain} = 10000$. That makes $\text{Expenses} = 1100$ for all elementary attacks. When we follow the approximate computation rules in the [10], we get the $\text{Outcome}_T = 4200$.

By following the computation rules in this article, we have the attack suites $\sigma_1 = \{A, C\}$, $\sigma_2 = \{B, C\}$, $\sigma_3 = \{A, B, C\}$, which satisfy the original attack tree T . The outcome computation for attack suites σ_1 and σ_2 is straightforward and $\text{Outcome}_{\sigma_1} = \text{Outcome}_{\sigma_2} = 4200$. The $\text{Outcome}_{\sigma_3}$ is a bit more complicated as there are three subsets $\rho_1 = \{A, C\}$, $\rho_2 = \{B, C\}$, $\rho_3 = \{A, B, C\}$ for the suite σ_3 , which also satisfy the attack tree T . Therefore we get the $p_{\sigma_3} = p_{APBP_C} + p_{AP_C}(1 - p_B) + p_{BP_C}(1 - p_A) = 0.768$ and $\text{Outcome}_{\sigma_3} = 4380$. By taking the maximum of the three outcomes, we get $\text{Outcome}_T = 4380$.

As the Cost parameters in this example for elementary attacks A and B were chosen quite low and the success probability p_A and p_B of these attacks were quite high, it made sense for an attacker to mount both of these subattacks and get bigger expected outcome, even though the attack tree would have been satisfied as well by only one of them.

4 Implementation

The most time-consuming computational routine among the computations given in Section 3.2 is the generation of all the satisfiable assignments of a Boolean formula $\mathcal{F}$ in order to find the maximal outcome by (1). Even though the computation routine (3) for finding p_σ formally also goes through (potentially all) subsets of σ , it can be evaluated in linear time in the number of variables n . To do so we can set $p_i = 0$ for all $X_i \notin \sigma$ and leave all the p_i for $X_i \in \sigma$ untouched.

Then for each internal node of the tree with probabilities of the child nodes being $p_{i_1}, p_{i_2}, \dots, p_{i_k}$ we can compute the probability of the parent node to be

$$\prod_{j=1}^k p_{i_j} \quad \text{or} \quad 1 - \prod_{j=1}^k (1 - p_{i_j})$$

depending on whether it is an AND or an OR node. Propagating throughout the tree, this computation gives exactly the success probability p_σ of the suite σ at the root node.

The routine (1) can be optimised as well by cutting off hopeless cases (see Theorem 1), but it still remains worst-case exponential-time. Thus for performance reasons it is crucial to have an efficient implementation of this routine. We are using a modified version of DPLL algorithm [14] to achieve this goal. The original form of the DPLL algorithm is only concerned about satisfiability, but it can easily be upgraded to produce all the satisfying assignments as well. Note that all the assignments are not needed at the same time to compute (1), but rather one at a time. Hence we can prevent the exponential memory consumption by building a serialised version, obtaining Algorithm 1.

Algorithm 1 works recursively and besides the current Boolean formula $\mathcal{F}$ it has two additional parameters. The set S contains the variables of which the satisfying assignments should be composed from. The set A on the other hand contains the variables already chosen to the assignments on previous rounds of recursion. As a technical detail note that the satisfying assignments are identified by the subset of variables they set to **true**.

The computation starts by calling `process_satisfying_assignments`($\mathcal{F}$, $\mathcal{X}$, $\emptyset$). Note that Algorithm 1 does not really produce any output, a processing subroutine is called on step 1 instead. This subroutine computes Outcome_σ for the given assignment σ and compares it with the previous maximal outcome.

4.1 Optimisations

Even with the help of a DPLL-based algorithm, the computations of (1) remain worst-case exponential time. In order to cut off hopeless branches, we can make some useful observations.

When we consider a potential attack suite σ and want to know, whether it is sufficient to materialise the root attack, we will set all the elements of σ to **true**, all the others to **false** and evaluate the formula $\mathcal{F}$ corresponding to the attack tree. In the process, all the internal nodes of the tree get evaluated as well (including the root node, showing whether the suite is sufficient). In Section 3, we allowed the suites σ to have more elements than absolutely necessary for materialising the root node, because in OR-nodes it often makes a lot of sense to try different alternatives. In AND nodes, at the same time, no choice is actually needed and achieving some children of an AND node without achieving some others is just a waste of resources.

Thus, intuitively we can say that it makes no sense to have AND-nodes with some children evaluating to **true** and some children to **false**. Formally, we can state and prove the following theorem.

Algorithm 1. Processing all the satisfying assignments of a formula**Procedure** process_satisfying_assignments($\mathcal{F}, S, A$)**Input:** Boolean CNF-formula $\mathcal{F}$, a subsets S of its variables and a subset $A \subseteq \mathcal{X} \setminus S$

1. **If** $\mathcal{F}$ contains **true** in every clause **then**
 - Process the assignment $A \cup T$ for every $T \subseteq S$; **return**
2. **If** $\mathcal{F}$ contains an empty clause **or** $S = \emptyset$ **then return** #no output in this branch
3. **If** $\mathcal{F}$ contains a unit clause $\{X\}$, where $X \in S$ **then**
 - **Let** $\mathcal{F}'$ be the formula obtained by setting $X = \text{true}$ in $\mathcal{F}$
 - process_satisfying_assignments($\mathcal{F}', S \setminus \{X\}, A \cup \{X\}$)
 - **Return**
4. Select a variable $X \in S$
5. **Let** $\mathcal{F}'$ be the formula obtained by setting $X = \text{true}$ in $\mathcal{F}$
6. process_satisfying_assignments($\mathcal{F}', S \setminus \{X\}, A \cup \{X\}$)
7. **Let** $\mathcal{F}''$ be the formula obtained by deleting X from $\mathcal{F}$
8. process_satisfying_assignments($\mathcal{F}'', S \setminus \{X\}, A$)
9. **Return**

Theorem 1. Let $\mathcal{F}$ be a Boolean formula corresponding to the attack tree T (i.e. AND-OR-tree, where all variables occur only once) and let σ be its satisfying assignment (i.e. an attack suite). Set all the variables of σ to **true** and all others to **false** and evaluate all the internal nodes of T . If some AND-node has children evaluating to **true** as well as children evaluating to **false**, then there exists a satisfying assignment $\sigma' \subset \sigma$ ($\sigma' \neq \sigma$) such that $\text{Outcome}_{\sigma'} \geq \text{Outcome}_{\sigma}$.

Proof. Consider an AND-node Y having some children evaluating to **true** and some evaluating to **false**. Then the node Y itself also evaluates to **false**, but the set of variables of the subformula corresponding to Y has a non-empty intersection with σ ; let this intersection be τ . We claim that we can take $\sigma' = \sigma \setminus \tau$. First it is clear that $\sigma' \subset \sigma$ and $\sigma' \neq \sigma$. Note also that σ' is a satisfying assignment and hence $\sigma' \neq \emptyset$. Now consider the corresponding outcomes:

$$\begin{aligned} \text{Outcome}_{\sigma} &= p_{\sigma} \cdot \text{Gains} - \sum_{X_i \in \sigma} \text{Expenses}_i, \\ \text{Outcome}_{\sigma'} &= p_{\sigma'} \cdot \text{Gains} - \sum_{X_i \in \sigma'} \text{Expenses}_i. \end{aligned}$$

Since $\sigma' \subset \sigma$, we have

$$\sum_{X_i \in \sigma} \text{Expenses}_i \geq \sum_{X_i \in \sigma'} \text{Expenses}_i,$$

as all the added terms are non-negative.

Now we claim that the equality $p_{\sigma} = p_{\sigma'}$ holds, which implies the claim of the theorem. Let

$$R_{\sigma} = \{\rho \subseteq \sigma : \mathcal{F}(\rho := \text{true}) = \text{true}\}$$

and define $R_{\sigma'}$ in a similar way. Then by (3) we have

$$p_{\sigma} = \sum_{\rho \in R_{\sigma}} \prod_{X_i \in \rho} p_i \prod_{X_j \in \sigma \setminus \rho} (1 - p_j),$$

$$p_{\sigma'} = \sum_{\rho' \in R_{\sigma'}} \prod_{X_i \in \rho'} p_i \prod_{X_j \in \sigma' \setminus \rho'} (1 - p_j).$$

We claim that $R_{\sigma} = \{\rho' \cup \tau' : \rho' \in R_{\sigma'}, \tau' \subseteq \tau\}$, i.e. that all the satisfying subassignments of σ can be found by adding all the subsets of τ to all the satisfying subassignments of σ' . Indeed, the node Y evaluates to **false** even if all the variables of τ are **true**, hence the same holds for every subset of τ due to monotonicity of AND and OR. Thus, if a subassignment of σ satisfies the formula $\mathcal{F}$, the variables of τ are of no help and can have arbitrary values. The evaluation **true** for the root node can only come from the variables of σ' , proving the claim.

Now we can compute:

$$\begin{aligned} p_{\sigma} &= \sum_{\rho \in R_{\sigma}} \prod_{X_i \in \rho} p_i \prod_{X_j \in \sigma \setminus \rho} (1 - p_j) = \sum_{\substack{\rho = \rho' \cup \tau' \\ \rho' \in R_{\sigma'}, \tau' \subseteq \tau}} \prod_{X_i \in \rho} p_i \prod_{X_j \in \sigma \setminus \rho} (1 - p_j) = \\ &= \sum_{\rho' \in R_{\sigma'}} \sum_{\tau' \subseteq \tau} \prod_{X_i \in \rho' \cup \tau'} p_i \prod_{X_j \in \sigma \setminus (\rho' \cup \tau')} (1 - p_j) = \\ &= \sum_{\rho' \in R_{\sigma'}} \sum_{\tau' \subseteq \tau} \prod_{X_i \in \rho'} p_i \prod_{X_i \in \tau'} p_i \prod_{X_j \in \sigma' \setminus \rho'} (1 - p_j) \prod_{X_j \in \tau \setminus \tau'} (1 - p_j) = \\ &= \sum_{\rho' \in R_{\sigma'}} \prod_{X_i \in \rho'} p_i \prod_{X_j \in \sigma' \setminus \rho'} (1 - p_j) \sum_{\tau' \subseteq \tau} \prod_{X_i \in \tau'} p_i \prod_{X_j \in \tau \setminus \tau'} (1 - p_j) = \\ &= \sum_{\rho' \in R_{\sigma'}} \prod_{X_i \in \rho'} p_i \prod_{X_j \in \sigma' \setminus \rho'} (1 - p_j) \prod_{X_i \in \tau} [p_i + (1 - p_i)] = \\ &= \sum_{\rho' \in R_{\sigma'}} \prod_{X_i \in \rho'} p_i \prod_{X_j \in \sigma' \setminus \rho'} (1 - p_j) = p_{\sigma'}, \end{aligned}$$

since $\sigma \setminus (\rho' \cup \tau') = (\sigma' \setminus \rho') \dot{\cup} (\tau \setminus \tau')$. The claim of the theorem now follows easily. $\square$

Note that Theorem 1 really depends on the assumption that $\mathcal{F}$ is an AND-OR-tree and that all variables occur only once. Formulae (1) and (3) together with Algorithm 1 can still be applied if the structure of the formula $\mathcal{F}$ is more complicated (say, a general DAG with other connectives in internal nodes), but the optimisation of Theorem 1 does not necessarily work.

This theorem allows us to leave many potential attack suites out of consideration by simply verifying if they evaluate children of some AND-node in a different way.

4.2 Performance

We implemented Algorithm 1 in Perl programming language and ran it on 500 randomly generated trees. The tests were ran on a computer having 3GHz dual-core Intel processor, 1GB of RAM and Arch Linux operating system.

The tree generation procedure was the following:

1. Generate the root node.
2. With probability 50% let this node have 2 children and with probability 50% let it have 3 children.
3. For every child, let it be an AND-node, an OR-node or a leaf with probability 40%, 40% and 20%, respectively.
4. Repeat the steps number 2 and 3 for every non-leaf node until the tree of depth up to 3 has been generated and let all the nodes on the third level be leaves.
5. To all the leaf nodes, generate the values of **Cost**, π^+ and π^- as integers chosen uniformly from the interval $[0, 1000)$, and the value of p chosen uniformly from the interval $[0, 1)$.
6. Generate the value of **Gains** as an integer chosen uniformly from the interval $[0, 1000000)$.

Thus, the generated trees may in theory have up to 27 leaves. That particular size limit for the trees was chosen because the running time for larger trees was already too long for significant amount of tests.

Performance test results showing the average running times and the standard deviation of the running times of the algorithm depending on the number of leaves are displayed in Figure 2. Note that the time scale is logarithmic. The times are measured together with the conversion of the attack tree formula to the conjunctive normal form. In Figure 2 we have included the trees with only up to 19 leaves, since the number of larger trees generated was not sufficient to produce statistically meaningful results. The number of the generated trees by the number of leaves is given later in Figure 3.

5 Analysis

In this Section we provide some evaluation of our tree computations compared to the ones given by Buldas *et al.* [10] and within the framework of Mauw and Oostdijk [8].

5.1 Comparison with the Semantics of Buldas *et al.*

Our main result can be shortly formulated as the following theorem.

Theorem 2. *Let us have an attack tree T . Let the best attack suites found by the routines of the current paper and the paper [10] be σ and σ' respectively. Let the corresponding outcomes (computed using the respective routines) be Outcome_σ and $\text{Outcome}_{\sigma'}$. The following claims hold:*

1. *If $\sigma = \sigma'$ then $\text{Outcome}_\sigma = \text{Outcome}_{\sigma'}$.*
2. *$\text{Outcome}_\sigma \geq \text{Outcome}_{\sigma'}$.*

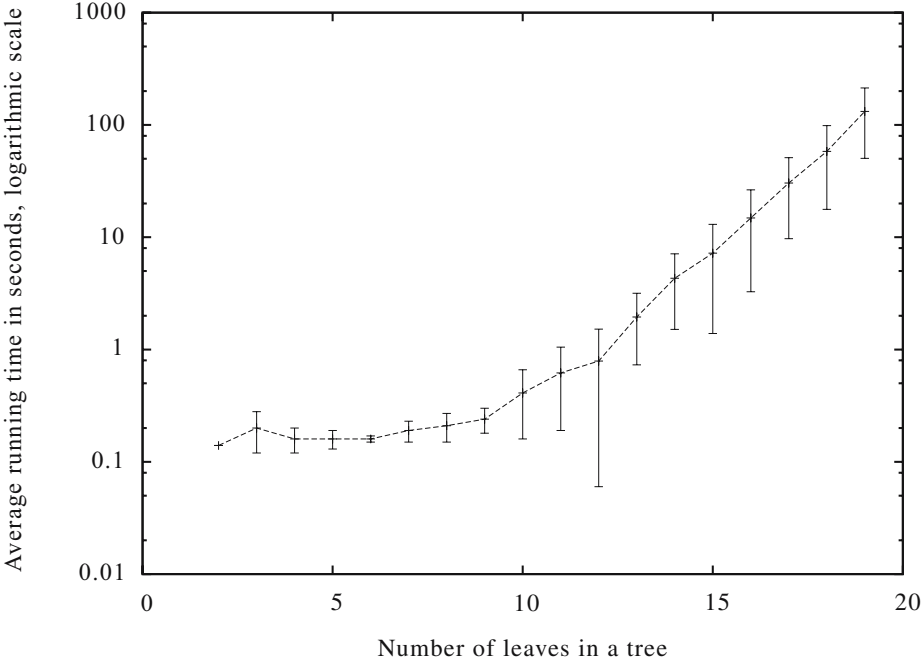


Fig. 2. Performance test results

Proof.

1. We need to prove that if $\sigma = \sigma'$ then

$$\text{Outcome}_{\sigma'} = p_{\sigma} \cdot \text{Gains} - \sum_{X_i \in \sigma} \text{Expenses}_i .$$

First note that the attack suite output by the routine of [10] is minimal in the sense that none of its proper subsets materialises the root node, because only one child is chosen in every OR-node. Hence, $p_{\sigma} = \prod_{X_i \in \sigma} p_i$. Now consider how $\text{Outcome}_{\sigma'}$ of the root node is computed in [10]. Let the required parameters of the root node be p' , Gains' and $\text{Expenses}'$. Obviously, $\text{Gains}' = \text{Gains}$. By looking at how the values of the attack success probability and the expected expenses are propagated throughout the tree, we can also conclude that

$$p' = \prod_{X_i \in \sigma} p_i = p_{\sigma} \quad \text{and} \quad \text{Expenses}' = \sum_{X_i \in \sigma} \text{Expenses}_i ,$$

finishing the first part of the proof.

2. Since σ' is a satisfying assignment of the Boolean formula underlying the tree T , we can conclude that σ' is considered as one of the attack suite candidates in (1). The conclusion now follows directly from the first part of the proof. □

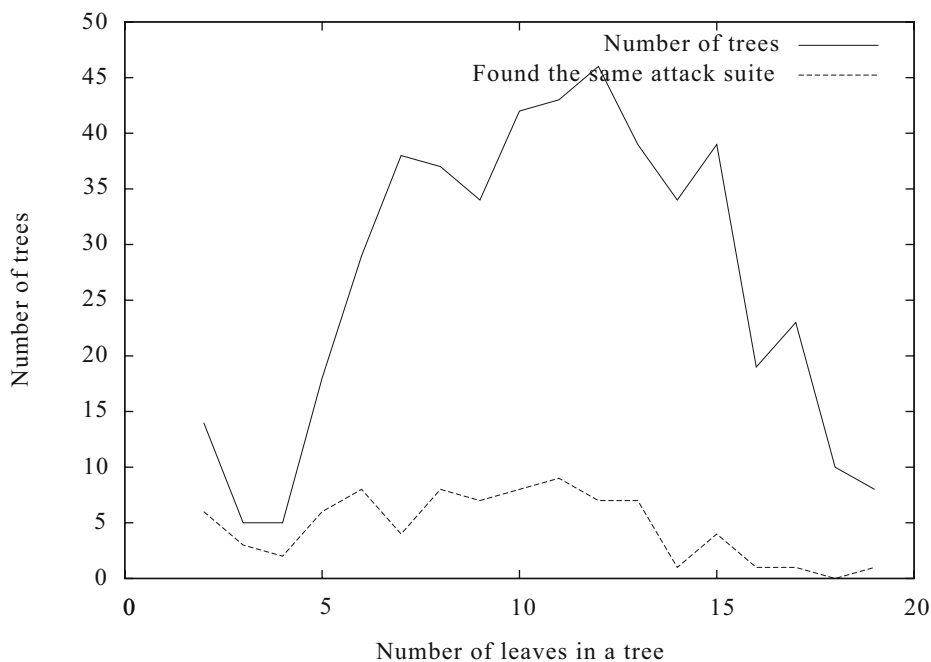


Fig. 3. Precision of the computational routine of Buldas *et al.* [10]

Theorem 2 implies that the exact attack tree computations introduced in the current paper always yield at least the same outcome compared to [10]. Thus, the potential use of the routine of [10] is rather limited, because it only allows us to get a lower estimate of the attacker’s expected outcome, whereas the upper limit would be of much higher interest. We can still say that if the tree computations of [10] show that the system is insufficiently protected (i.e. $\text{Outcome}_{\sigma'} > 0$) then the exact computations would yield a similar result ($\text{Outcome}_{\sigma} > 0$).

Following the proof of Theorem 2, we can also see that the semantics of [10] is actually not too special. Any routine that selects just one child of every OR-node when analysing the tree would essentially give a similar under-estimation of the attacker’s expected outcome.

Together with the performance experiments described in Section 4.2 we also compared the outcome attack suites produced by the routines of the current paper and [10] (the implementation of the computations of [10] was kindly provided by Alexander Andrusenko [15]). The results are depicted in Figure 3.

The graphs in Figure 3 show the number of the generated trees by the number of leaves and the number of such trees among them, for which the routine of [10] was able to find the same attack suite that the exact computations introduced in the current paper. Over all the tests we can say that this was the case with 17.4% of the trees.

5.2 Consistency with the Framework of Mauw and Oostdijk

Working in a single parameter model, Mauw and Oostdijk [8] first define a set V of attribute values and then consider an attribute function $\alpha : \mathbb{C} \rightarrow V$, where $\mathbb{C}$ is the set of elementary attacks (called *attack components* in [8]). In order to extend this attribution to the whole tree, they essentially consider the tree corresponding to the disjunctive normal form of the underlying Boolean formula. To obtain the attribute values of the conjunctive clauses (corresponding to our attack suites), they require a conjunctive combinator $\Delta : V \times V \rightarrow V$, and in order to get the value for the whole DNF-tree based on clause values they require a disjunctive combinator $\nabla : V \times V \rightarrow V$. Mauw and Oostdijk prove that if these combinators are commutative, associative and distributive, all the nodes of the tree in the original form can also be given attribute values and that the value of the (root node of the) tree does not change if the tree is transformed into an equivalent form. This equivalence is denoted as $\equiv$ and it is defined by the set of legal transformations retaining logical equivalence of the underlying Boolean formulae (see [8]). The structure (α, ∇, Δ) satisfying all the given conditions is called *distributive attribute domain*.

Even though the semantics used in [10,12] formally require four different parameters, they still fit into a single parameter ideology, since based on the quadruples of the child nodes, similar quadruples are computed for parents when processing the trees. However, it is easy to construct simple counterexamples showing that the computation rules of [10,12] are not distributive, one is given in the Section 2.

The computation rules presented in the current paper follow the framework of Mauw and Oostdijk quite well at the first sight. Formula (1) essentially goes through all the clauses in the complete disjunctive normal form of the underlying formula $\mathcal{F}$ and finds the one with the maximal outcome. So we can take $V = \mathbb{R}$ and $\nabla = \max$ in the Mauw and Oostdijk framework. However, there is no reasonable way to define a conjunctive combinator $\Delta : V \times V \rightarrow V$, since the outcome of an attack suite can not be computed from the outcomes of the elementary attacks; the phrase “outcome of an elementary attack” does not even have a meaning.

Another possible approach is to take $V = [0, 1] \times \mathbb{R}^+$ and to interpret the first element of $\alpha(X)$ as the success probability p and the second element as **Expenses** for an attack X . Then the disjunctive combinator can be defined as outputting the pair which maximises the expression $p \cdot \text{Gains} - \text{Expenses}$. This combinator has a meaning in the binary case and as such, it is both associative and commutative, giving rise to an obvious n -ary generalisation. For the conjunctive combinator to work as expected in the n -ary case, we would need to achieve

$$\Delta_{X_i \in \sigma} \alpha(X_i) = (p_\sigma, \Sigma_{X_i \in \sigma} \text{Expenses}_i).$$

However, it is easy to construct a formula and a satisfying assignment σ such that constructing p_σ from success probabilities of the descendant instances using a conjunctive combinator is not possible. For example, we can take the formula

$\mathcal{F} = X_1 \vee X_2 \& X_3$, where X_1, X_2, X_3 are elementary attacks with success probabilities p_1, p_2, p_3 , respectively. Let α_1 denote the first element of the output of α and let Δ_1 denote the combinator Δ restricted to the first element of the pair (so $\Delta_1(X_i) = p_i$, $i = 1, 2, 3$). Then for $\sigma = \{X_1, X_2, X_3\}$ we would need to obtain

$$(p_1 \Delta_1 p_2) \Delta_1 p_3 = (\alpha_1(X_1) \Delta_1 \alpha_1(X_2)) \Delta_1 \alpha_1(X_3) = p_\sigma = p_1 + p_2 p_3 - p_1 p_2 p_3$$

for any $p_1, p_2, p_3 \in [0, 1]$, which is not possible. Indeed, taking $p_3 = 0$ we have $(p_1 \Delta_1 p_2) \Delta_1 0 = p_1$. In the same way we can show that $(p_2 \Delta_1 p_1) \Delta_1 0 = p_2$, which is impossible due to commutativity of Δ_1 when $p_1 \neq p_2$.

All of the above is not a formal proof that our computations do not form a distributive attribute domain, but we can argue that there is no obvious way to interpret them as such. Additionally, if we had a distributive attribute domain then Theorem 3 with Corollary 2 of [8] would allow us to build a linear-time value-propagating tree computation algorithm, but this is rather unlikely.

However, we can still state and prove the following proposition.

Proposition 1. *Let T_1 and T_2 be two attack trees. If $T_1 \equiv T_2$, we have $\text{Outcome}(T_1) = \text{Outcome}(T_2)$.*

Proof. It is easy to see that the formulae (1), (2) and (3) do not depend on the particular form of the formula, but use it only as a Boolean function. Since the tree transformations defined in [8] keep the underlying Boolean formula logically equivalent, the result follows directly. $\square$

In the context of [8], this is a somewhat surprising result. Even though the attribute domain defined in the current paper is not distributive (and it can not be easily turned into such), the main goal of Mauw and Oostdijk is still achieved. This means that the requirement for the attribute domain to be distributive in the sense of Mauw and Oostdijk is sufficient to have semantically consistent tree computations, but it is not really necessary. It would be interesting to study, whether the framework of Mauw and Oostdijk can be generalised to cover non-propagating tree computations (like the one presented in the current paper) as well.

6 Conclusions and Further Work

In this paper we introduced a computational routine capable of finding the maximal possible expected outcome of an attacker based on a given attack tree. We showed that when compared to rough computations given in [10], the new routine always gives at least the same outcome and mostly it is also strictly larger. This means that the tree computations of [10] are not very useful in practise, since they strongly tend to under-estimate attacker's capabilities. We also proved that unlike [10], our new semantics of the attack tree is consistent with the general ideology of the framework of Mauw and Oostdijk, even though our attribute domain is not distributive. This is a good motivation to start looking for further generalisations of the framework.

On the other hand, the routines of the current paper are computationally very expensive and do not allow practical analysis of trees with the number of leaves substantially larger than 20. Thus, future research needs to address at least two issues. First, there are some optimisations possible in the implementation (e.g. precomputation of frequently needed values), they need to be programmed and compared to the existing implementations. Still, any optimisation will very probably not decrease the time complexity of the algorithm to a subexponential class. Thus the second direction of further research is finding computationally cheap approximations, which would over-estimate the attacker's exact outcome.

As a further development of the attack tree approach, more general and realistic models can be introduced. For example, the model presented in the current paper does not take into account the possibility that the attacker may drop attempting an attack suite after a critical subset of it has already failed. Studying such models will remain the subject for future research as well.

Acknowledgments

This research has been supported by the Estonian Science Foundation grant no. 7081. Also, we would like to thank Ahto Buldas, Peeter Laud and Sven Laur for helpful discussions.

References

1. Vesely, W.E., Goldberg, F.F., Roberts, N.H., Haasl, D.F.: *Fault Tree Handbook*. US Government Printing Office, Systems and Reliability Research, Office of Nuclear Regulatory Research, U.S. Nuclear Regulatory Commission (January 1981)
2. Viega, J., McGraw, G.: *Building Secure Software: How to Avoid Security Problems the Right Way*. Addison Wesley Professional, Reading (2001)
3. Moore, A.P., Ellison, R.J., Linger, R.C.: *Attack modeling for information security and survivability*. Technical Report CMU/SEI-2001-TN-001, Software Engineering Institute (2001)
4. Weiss, J.D.: A system security engineering process. In: *Proceedings of the 14th National Computer Security Conference*, pp. 572–581 (1991)
5. Schneier, B.: *Attack trees: Modeling security threats*. *Dr. Dobbs's Journal* 24(12), 21–29 (1999)
6. Edge, K.S.: *A Framework for Analyzing and Mitigating the Vulnerabilities of Complex Systems via Attack and Protection Trees*. Ph.D thesis, Air Force Institute of Technology, Ohio (2007)
7. Espedahlen, J.H.: *Attack trees describing security in distributed internet-enabled metrology*. Master's thesis, Department of Computer Science and Media Technology, Gjøvik University College (2007)
8. Mauw, S., Oostdijk, M.: Foundations of attack trees. In: Won, D., Kim, S. (eds.) *ICISC 2005*. LNCS, vol. 3935, pp. 186–198. Springer, Heidelberg (2006)
9. Opel, A.: *Design and implementation of a support tool for attack trees*. Technical report, Otto-von-Guericke University, Internship Thesis (March 2005)
10. Buldas, A., Laud, P., Priisalu, J., Saarepera, M., Willemson, J.: Rational Choice of Security Measures via Multi-Parameter Attack Trees. In: López, J. (ed.) *CRITIS 2006*. LNCS, vol. 4347, pp. 235–248. Springer, Heidelberg (2006)

11. Buldas, A., Mägi, T.: Practical security analysis of e-voting systems. In: Miyaji, A., Kikuchi, H., Rannenberg, K. (eds.) *Advances in Information and Computer Security, Second International Workshop on Security, IWSEC*. LNCS, vol. 4752, pp. 320–335. Springer, Heidelberg (2007)
12. Jürgenson, A., Willemson, J.: Processing multi-parameter attacktrees with estimated parameter values. In: Miyaji, A., Kikuchi, H., Rannenberg, K. (eds.) *IWSEC 2007*. LNCS, vol. 4752, pp. 308–319. Springer, Heidelberg (2007)
13. Rätsep, L.: The influence and measurability of the parameters of the security analysis of the Estonian e-voting system. M.Sc thesis, Tartu University (2008) (in Estonian)
14. Davis, M., Logemann, G., Loveland, D.: A machine program for theorem proving. *Communications of the ACM* 5(7), 394–397 (1962)
15. Andrusenko, A.: Multiparameter attack tree analysis software. B.Sc thesis, Tartu University (2008) (in Estonian)

Automatic Generation of Secure Multidimensional Code for Data Warehouses: An MDA Approach

Carlos Blanco¹, Ignacio García-Rodríguez de Guzmán¹,
Eduardo Fernández-Medina¹, Juan Trujillo², and Mario Piattini¹

¹ Dep. of Information Technologies and Systems. Escuela Superior de Informática
Alarcos Research Group – Institute of Information Technologies and Systems
Univ. of Castilla-La Mancha. Paseo de la Universidad, 4. 13071. Ciudad Real. Spain
{Carlos.Blanco, Ignacio.GRodriguez, Eduardo.Fdezmedina,
Mario.Piattini}@uclm.es

² Department of Information Languages and Systems. Facultad de Informática
University of Alicante. San Vicente s/n. 03690. Alicante. Spain
jtrujillo@dlsi.ua.es

Abstract. Data Warehouses (DW) manage enterprise information for the decision making process, and the establishment of security measures at all stages of the DW development process is also highly important as unauthorized users may discover vital business information. Model Driven Architecture (MDA) based approaches allow us to define models at different abstraction levels, along with the automatic transformations between them. This has thus led to the definition of an MDA architecture for the development of secure DWs. This paper uses an example of a hospital to show the benefits of applying the MDA approach to the development of secure DWs. The paper is focused on transforming secure multidimensional Platform Independent Models (PIM) at the conceptual level into Platform Specific Models (PSM) at the logical level by defining the necessary set of Query/Views/Transformations (QVT) rules. This PSM model is therefore used to obtain the corresponding secure multidimensional code for a specific On-Line Analytical Processing (OLAP) platform such as SQL Server Analysis Services (SSAS).

Keywords: Data Warehouses, Security, MDA, QVT, OLAP, SQL Server Analysis Services.

1 Introduction

The survival of organizations depends on the correct management of information security and confidentiality [1], and DWs manage enterprises' historical information which is used to support the decision making process and must be ensured by establishing security measures from the early stages of the development lifecycle [2]. Therefore, it is necessary to consider security constraints in models at all abstraction levels and to ultimately take these security issues into account in the final tools in order to avoid the situation of users being able to access unauthorized information by using operations.

Furthermore, MDA [3] is the Object Management Group (OMG) standard approach for model driven software development based on the separation of the specification of the system functionality and its implementation. MDA allows us to define models at different abstraction levels: computer-independent models (CIM) at business level and platform-independent models (PIM) at conceptual level which do not include information about specific platforms and technologies, and platform-specific models at logical level (PSM) with information about the specific technology used. Moreover, MDA proposes the use of *model transformations* as a mechanism with which to move from one level of abstraction to another, by transforming input models into new models or searching for matchings, among the other models involved. Many languages for model transformations exist. Nonetheless, the OMG proposes Query / Views / Transformations (QVT) [4] as a new standard for model transformation based on the Meta-Object Facility (MOF) standard [5] through which to define model transformation in an intuitive manner. Supporting the development of DWs with an MDA approach provides many advantages such a better separation of models including security requirements from the first stages of the DWs lifecycle and automatic translations through which to obtain other models and final code for different target platforms.

An architecture for developing secure DWs by using MDA and QVT transformations has been proposed in [6]. This architecture supports the modeling of secure DWs at different abstraction levels: CIM (specifying goals and subgoals), PIM (a multidimensional model), PSM (a relational model) and the final implementation in a database management system (DBMS). However, these aforementioned works are focused on a relational approach and the greatest part of DW is managed by OLAP tools over a multidimensional approach. We have therefore deployed a specialization of this architecture which defines a secure multidimensional PSM and implements secure DWs in SQL Server Analysis Services (SSAS) as a specific OLAP platform. In this architecture, we have decided to specify those QVT rules which directly transform our secure multidimensional PIM into a secure multidimensional PSM, and to then use this PSM to obtain secure multidimensional code for this OLAP platform (SSAS). In this paper we show the benefits of our approach through its application to an example. We show the steps involved in a conceptual model (PIM), a logical model (PSM), multidimensional secure code and the application of a set of QVT rules which transform the concepts of our secure multidimensional model at the conceptual level (PIM) into a logical model (PSM) which is used to obtain pieces of the code of our target platform (both the structural and security aspects of the final DW).

The remainder of the paper is organised as follows: Section 2 will present related work. Section 3 will introduce our MDA architecture for the development of secure DWs, and will be focused both on our source and target metamodels (PIM, PSM and code), and on the transformations which are necessary to obtain PSM from PIM and code from PSM. Section 4 will introduce our example regarding the admission system of a hospital. We will present models at the conceptual (PIM), logical (PSM) and code levels and will show how the proposed QVT transformations have been applied to obtain PSM from PIM. We will then demonstrate how to obtain the final code from PSM. Finally, Section 5 will present our conclusions and future work.

2 Related Work

OLAP systems are mechanisms with which to discover business information and use a multidimensional analysis of data to make strategic decisions. This information is organized according to the business parameters, and users can discover unauthorized data by applying a set of OLAP operations to the multidimensional view. Therefore, it is of vital importance for the organization to protect its data from unauthorized accesses. Several works attempting to include security issues in OLAP tools by implementing the previously defined security rules at a conceptual level have been proposed, but these works focus solely upon Discretionary Access Control (DAC) policy and use a simplified role concept implemented as a subject. For instance, Katic et al. [7] proposed a DWs security model based on metamodels which provides us with views for each user group and uses DAC with classification and access rules for security objects and subjects. However, this model does not allow us to define complex confidentiality constraints. Kirkgöze et al. [8] defined a role-based security concept for OLAP by using a “constraints list” for each role, and this concept is implemented through the use of a discretionary system in which roles are defined as subjects.

Priebe and Pernul later proposed a security design methodology, analyzed security requirements, classifying them into basic and advanced, and dealt with their implementation in commercial tools. Firstly, in [9] they used ADAPTEd UML to define a DAC system with roles defined as subjects at a conceptual level. They then went on to implement this in SQL Server Analysis Services 2000 by using Multidimensional Expressions (MDX). They created a Multidimensional Security Constraint Language (MDSCL) based on MDX and put forward HIDE statements with which to represent negative authorization constraints on certain multidimensional elements: cube, measure, slice and level.

Our proposal uses an access control and audit model specifically designed for DWs to define security constraints in early stages of the development lifecycle. By using an MDA approach we consider security issues in all stages of the development process and automatically transform models at upper abstraction level towards logical models over a relational or multidimensional approach and finally obtain from these models secure code for DBMS or OLAP tools.

3 An MDA Approach for Developing Secure DWs

Our MDA architecture [6] is an adaptation of an MDA architecture for developing DWs [10] which has been improved with security capabilities. Our approach is made up of several models which allow us to model the DW at different abstraction levels (see Figure 1): at the business level (CIM) with a UML profile [11] based on the i* framework [12], which is an agent orientated approach towards requirement engineering centering on the intentional characteristics of the agent; at the conceptual level (PIM) with a UML profile called SECDW [13]; and at the logical level (PSM) with an extension of the relational package of Common Warehouse Metamodel (CWM) called SECRDW [14]. As has previously been mentioned, this paper considers a specialization of this architecture (represented in grey in Figure 1) focused on defining a PSM metamodel over a multidimensional approach and transforming structural and

security issues from PIM into this multidimensional PSM which allows us to obtain final multidimensional code for OLAP tools. The transformation from PSM models into secure multidimensional code for a specific OLAP platform (SSAS) is also treated in this paper. The source (PIM) and target (PSM) metamodels are briefly described in the following subsections, and an overview of the QVT rules defined to support this transformation will be presented.

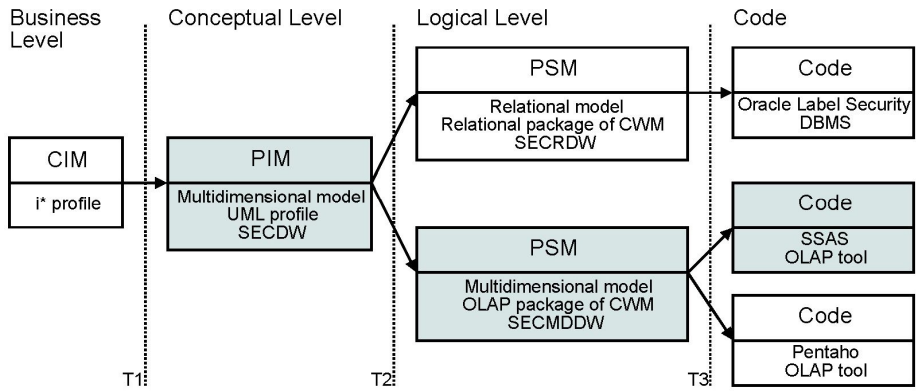


Fig. 1. MDA architecture for developing secure DWs

3.1 Secure Multidimensional PIM

A secure multidimensional conceptual metamodel called SECDW, has been defined in [13] by using a UML profile. This metamodel is shown in Figure 2 and is based on a UML profile for the conceptual design of DWs [15] which allows us to define fact, dimension and base classes, and considers specific aspects of DWs such as many-to-many relations, degenerated dimensions, multiple classifications or alternative paths of hierarchies. SECDW is enriched with security capabilities through the use of an access control and audit model (ACA) [16], which was specifically designed to consider security in DWs.

ACA allows us to define secure classes (SecureClass) and properties (SecureProperty), to classify authorization subjects and objects into security roles (SecurityRole) which organize users into a hierarchical role structure according to the responsibilities of each type of work, levels (SecurityLevel) which indicate the clearance level of the user, and compartments (SecurityCompartment) which classify users into a set of horizontal compartments or groups.

ACA also considers the definition of security rules over multidimensional elements of DWs by using stereotypes and Object Constraints Language (OCL) notes (Constraint). Three kinds of security rules are permitted: sensitive information assignment rules (SIAR) which specify multilevel security policies and allow us to define sensitivity information for each element in the multidimensional model; authorization rules (AUR) which permit or deny access to certain objects by defining the subject that the rule applies to, the object that the authorization refers to, the action that the rule refers to and the sign describing whether the rule permits or denies access; and audit rules (AR) to ensure that authorized users do not misuse their privileges.

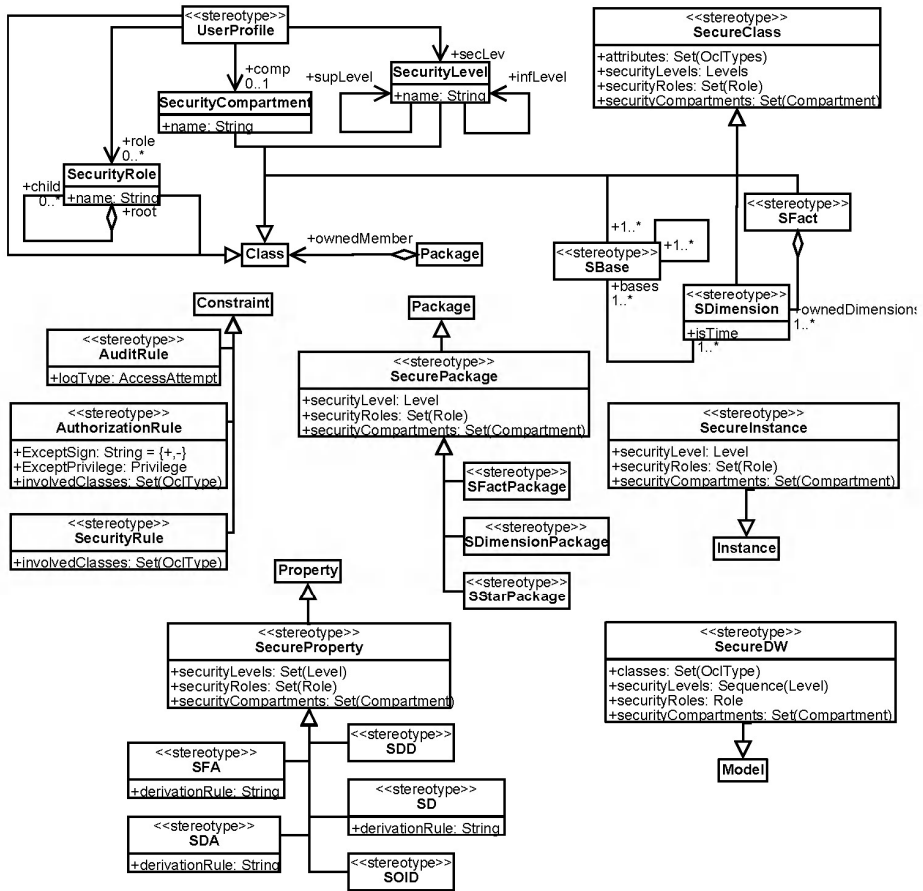


Fig. 2. Secure multidimensional PIM (SECDDW)

3.2 Secure Multidimensional PSM

A secure multidimensional metamodel at the logical level (PSM), called SECDDW, has been defined in this work by extending the OLAP package of CWM. This metamodel represents the intermediate step between conceptual and code levels, that is, our multidimensional PSM is obtained from PIM and can be used to obtain code towards different OLAP tools. Our PSM uses a multidimensional approach and considers the *security configuration* of the system, *structural* elements of the DW's and *security* constraints defined at class (fact, dimension or base) or attribute levels. Figure 3 shows the *security configuration* metamodel obtained. Our logical metamodel (PSM) only considers a role-based access control policy (RBAC) because the vast majority of OLAP tools use this policy and the PSM metamodel is closer to the final platform than the PIM metamodel. However, at the conceptual level we have considered security roles, levels and compartments defined by using our ACA model which have to be translated into roles. To accomplish this process we will follow the

methodology to implement secure DWs into OLAP tools presented in [17]. Furthermore, we have defined two metamodels to support the definition of the *structural* and *security* issues of cubes and dimensions.

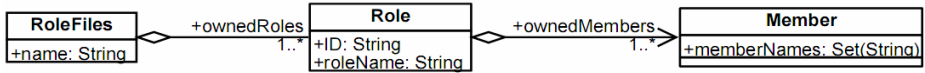


Fig. 3. Secure multidimensional PSM (SECMDDW): security configuration

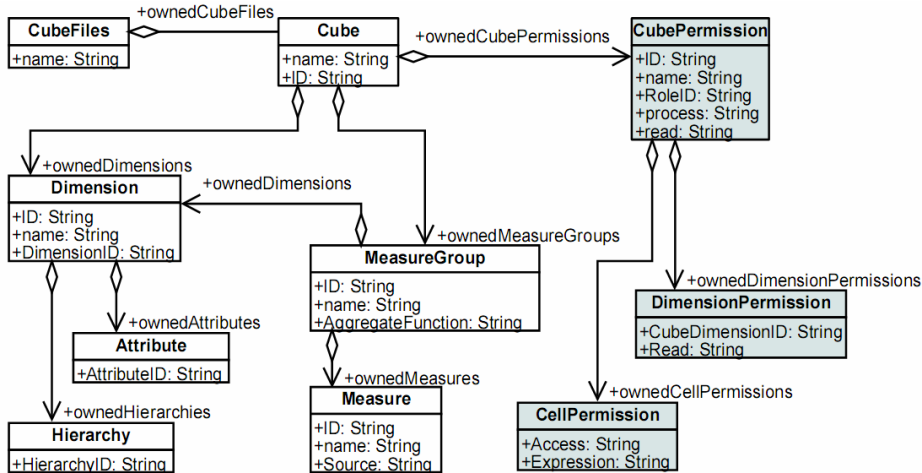


Fig. 4. Secure multidimensional PSM (SECMDDW): cubes

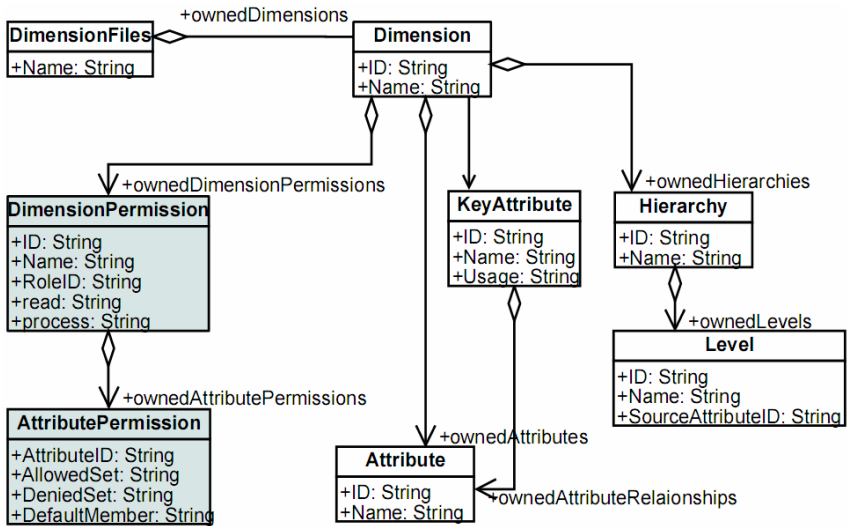


Fig. 5. Secure multidimensional PSM (SECMDDW): dimensions

Figures 4 and 5 show metamodels for cubes and dimensions which allow us to define facts classes (Cube), measures (MeasureGroup, Measure), dimension classes (Dimension), attributes (Attribute), hierarchies (Hierarchy) and base classes as attributes of the related dimension, and which also allow us to define security constraints over these multidimensional elements by using permissions over cubes (CubePermission), dimensions (DimensionPermission), cells (CellPermission) or attributes (AttributePermission). In these figures, the security-related aspects are represented in grey.

PIM to PSM transformation. A set of QVT transformations has been developed to automatically obtain secure multidimensional logical models, defined according to our PSM metamodel (SECMDDW), from conceptual models defined according to our PIM metamodel (SECDW). In order to develop these rules we have followed a methodology to implement multidimensional security in OLAP tools presented in [17]. These proposed transformations are made up of three main transformations which obtain our various kinds of target models from source conceptual models: *SECDW2Role*, *SECDW2Cube* and *SECDW2Dimension*.

SECDW2Role deals with the security configuration of the system. As our ACA model is richer than our PSM, which only considers roles, this transformation generates a new role for each security role (SR), level (SL) and compartment (SC) defined in our source model (PIM); *SECDW2Cube* generates the cube files that represent cubes, measures, dimensions, and cube permissions from the SECDW model; and *SECDW2Dimension* generates the dimension files that represent dimensions, bases, attributes, hierarchies and security permissions defined over dimensions and attributes. The security measures defined at the conceptual level (PIM) over classes or attributes by using SC, SR and SL, are translated into a set of permissions for involved roles (SC, SR and SL are translated into roles). As the transformations are quite verbose and the available space is limited, Section 4 will show some examples of complete rules and this section only presents the signatures of the developed rules.

SECDW2Role is composed of a top relation “*Package2RoleFiles {...}*” and relations “*SCompartment2Role {...}*”, “*SRole2Role {...}*” and “*SLevel2Role {...}*”. The signatures for the remainder of the developed rules are shown in Table 1.

Table 1. PIM to PSM transformations (signatures)

SECDW2Cube	SECDW2Dimension
top relation <i>Package2CubeFiles {...}</i>	top relation <i>Package2DimensionFiles {...}</i>
relation <i>SFact2Cube {...}</i>	relation <i>SDimension2Dimension {...}</i>
relation <i>CreateMeasureGroups {...}</i>	relation <i>KeyProperty2KeyAttribute {...}</i>
relation <i>SProperty2Measure {...}</i>	relation <i>NonKeyProperty2Attribute {...}</i>
relation <i>SDimension2Dimension {...}</i>	relation <i>SBase2Attribute {...}</i>
relation <i>ProcessSBase {...}</i>	relation <i>createDimensionSIARForSCompartment {...}</i>
relation <i>CreateOwnedHierarchies {...}</i>	relation <i>createDimensionSIARForSRole {...}</i>
relation <i>SProperty2Property {...}</i>	relation <i>createDimensionSIARForSLevel {...}</i>
relation <i>SCompartmentClass2CubePermission {...}</i>	relation <i>authorizeSCompartment {...}</i>
relation <i>SRoleClass2CubePermission {...}</i>	relation <i>authorizeSRole {...}</i>
relation <i>SLevelClass2CubePermission {...}</i>	relation <i>authorizeSLevel {...}</i>
relation <i>SCompartmentAtt2CellPermission {...}</i>	relation <i>processSecureProperty {...}</i>
relation <i>SRoleAtt2CellPermission {...}</i>	relation <i>createPositiveAttributePermission {...}</i>
relation <i>SLevelAtt2CellPermission {...}</i>	relation <i>createNegativeAttributePermission {...}</i>

3.3 Secure Multidimensional Code

As a target OLAP platform we have selected SQL Server Analysis Services (SSAS), which deals with multidimensional elements and allows us to establish security measures over them. Furthermore, SSAS uses several kinds of XML files to manage this information, the most important of which are role, cubes and dimension files. We have analyzed this OLAP tool by studying how structural and security information could be implemented in this platform, in order to obtain secure multidimensional code from PSM.

PSM to Code Transformation. Obtaining secure multidimensional code from our secure multidimensional PSM is a simple task since both consider structural and security issues by using a multidimensional approach and the vast majority of the destination concepts are defined in our source metamodel. This paper is focused on obtaining PSM from PIM, but also deals with the transformation of PSM into a specific OLAP platform, SSAS. In order to obtain code for the security measures defined in the conceptual models we have followed the methodology to implement multidimensional security in SSAS which is presented in [17]. The presentation of our example will show a portion of code in SSAS and screenshots of the final implementation in SSAS.

4 Applying MDA for Developing Secure DWs

This section presents the application of our MDA architecture to an example of a hospital that wishes to automate its admission process and requires confidentiality for the information involved. This example will be used to show the application of the transformations to obtain a logical model (PSM) according to our target metamodel and to obtain secure multidimensional code in SSAS from PSM.

4.1 Secure Multidimensional PIM

Figure 6 shows the conceptual model (defined as an instance of the SECDW model) for our hospital which is required to resolve the aforementioned problem. The security configuration of the hospital uses a classification of users and objects in security roles (SR) and security levels (SL). Security compartments have not been defined since they depend on organization policies. The user roles (SR) might be “HospitalEmployee”, “Health” (including “Doctor” and “Nurse” roles) and “NonHealth” (including “Admin” and “Maintenance” roles). The levels of security (SL) used are top secret (TS), secret (S), confidential (C) and undefined (U). The secure fact class “Admission” contains two secure dimension classes (“Diagnosis” and “Patient”). The “UserProfile” metaclass contains information about all the users who will have access to this secure multidimensional model. This information can also define characteristics of users such as age, citizenship, etc. which can be used to establish complex security constraints. We have also defined a set of sensitive information assignment rules (SIAR) over classes and attributes: instances of “Admission” fact class or “Patient” dimension can be accessed by the “Admin” or “Health” roles and the Secret (or upper) security level; the “Diagnosis” dimension can be accessed by the “Health” role and the Secret (or upper) security level; the bases “City” and “DiagnosisGroup” can be accessed by the Confidential (or upper) security level; and attributes “Admission.cost” and “Patient.address” can be accessed by the “Admin” role.

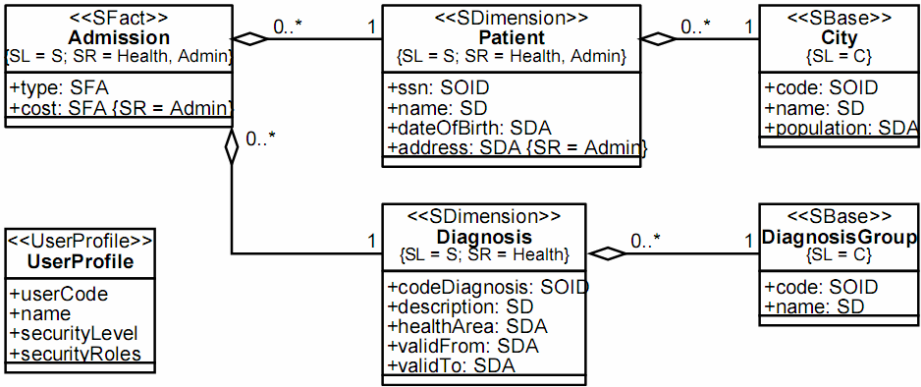


Fig. 6. Secure multidimensional PIM for hospital

4.2 Secure Multidimensional PSM

In this section we obtain a logical model (PSM) from the conceptual model defined above for a hospital according to the SECDW metamodel by using a set of QVT transformations (see Table 1). We have applied our three main defined transformations (SECDW2Role, SECDW2Cube and SECDW2Dimension) and we present the resulting logical models according to our PSM metamodel. These were obtained from conceptual models according to our PIM metamodel.

SECDW2Role. Table 2 shows the application of the *SECDW2Role* transformation to the hospital. The *SRole2Role* rule creates for each security role "r" detected in the source model a new role called "SRr". The QVT code for the rule *SLevel2Role* is shown in Table 3 and also creates a new role called "SLn" for each security level "n" defined at conceptual level, that is, in this example creates "SLTS", "SLS", "SLC" and "SLU" roles. *SCompartment2Role* has not been shown because security compartments have not been defined in the hospital.

Table 2. SECDW2Role transformation for hospital

top relation Package2RoleFiles: Hospital
relation SRole2Role: HospitalEmployee, Health, Doctor, Nurse, NonHealth, Admin, Maintenance
relation SLevel2Role: TS, S, C, U
relation SCompartment2Role: not thrown

Table 3. Relation SLevel2Role

relation SLevel2Role
checkonly domain psm sl:SRole{ name = n; } enforce domain pim r:Role{ fileName = "SL"+n+"role"; ID = "SL"+n; roleName = "SL"+n; ownedMembers = OWNEDMEMBS:Set(Member); }

The target model with the security configuration for the hospital has been represented in Figure 7 according to PSM metamodel (SECMDDW). This model defines roles at logical level for each security role and security level detected at conceptual level.

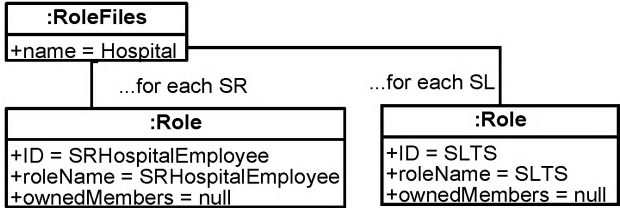


Fig. 7. Secure multidimensional PSM for hospital: security configuration

SECDW2Cube. Next, the *SECDW2Cube* transformation obtains the structure and security of cubes defined in the hospital. Table 4 shows the process: *SFact2Cube* rule creates the "Admission" cube; then the *CreateMeasuresGroups* and *SProperty2Measure* rules create measures and the remainder of the structural rules create dimensions, attributes and hierarchies for dimensions and bases related to the "Admission" cube. Finally, security rules analyze the security constraints defined over the fact class and its attributes, and define cube and cell permissions for involved security roles (which represent the SR, SL and SC of the source model).

Table 4. SECDW2Cube transformation for hospital

top relation Package2RoleFiles: Hospital
relation SFact2Cube: Admission
relation CreateMeasureGroups: Admission
relation SProperty2Measure: type, cost
relation SDimension2Dimension: Patient, Diagnosis
relation ProcessSBase: City, DiagnosisGroup
relation CreateOwnedHierarchies: City-Patient, DiagnosisGroup-Diagnosis
relation SProperty2Attribute: (for Patient) ssn, name, dateOfBirth, address (for Diagnosis) codeDiagnosis, description, healthArea, validFrom, validTo (for City) code, name, population (for DiagnosisGroup) code, name
relation SCompartmentClass2CubePermission: Not thrown
relation SRoleClass2CubePermission: (for Admission) Health, Admin
relation SLevelClass2CubePermission: (for Admission) S
relation SCompartmentAtt2CellPermission: Not thrown
relation SRoleAtt2CellPermission: (for Admission.address) Admin
relation SLevelAtt2CellPermission: Not thrown

Table 5 shows the code for "SLevelClass2CubePermission" rule which permits accesses to cube by creating cube permissions for each authorized role that represent allowed security levels. In this example, cube permissions allowing access to security level secret "S" (SLS role) and its upper security levels (SLTS role) are defined in "Admission" class.

Figure 8 shows the model obtained from SECDW2Cube transformation in which has been created an "Admission" cube with its measure groups (including "type" and "cost" measures), related dimensions ("Patient" and "Diagnosis"), cube permissions

Table 5. Relation SLevelClass2CubePermission

relation <i>SLevelClass2CubePermission</i>
checkonly domain psm sl:SLevel { name = n; } enforce domain pim c:Cube { name = cubeName; ID = cubeName; ownedCubePermissions = OWNCUBEPERMS:Set(CubePermission); } enforce domain pim cp:CubePermission { ID = "CubePermission"+n; name = "CubePermission"+n; RoleID = n; Process = "true"; Read = "Allowed"; } where{ OWNCUBEPERMS->including(cp); }

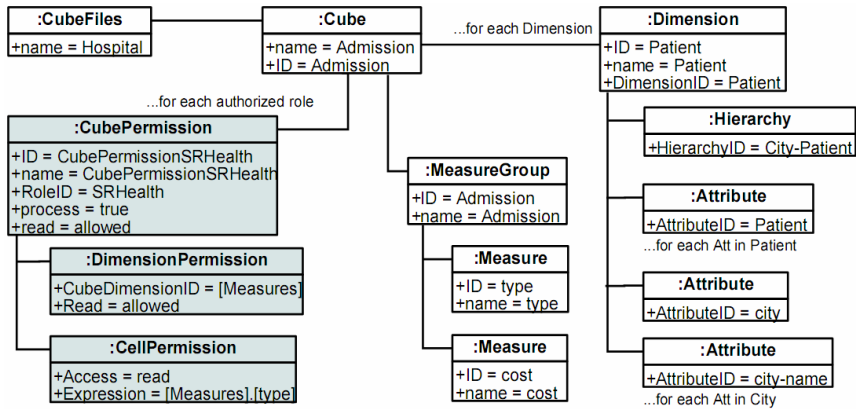


Fig. 8. Secure multidimensional PSM for hospital: cube

over “Admission” cube allowing accesses to authorized roles (“SRHealth”, “SRAdmin” and its descendants, and “SLS” and upper levels) and cell permissions allowing accesses to each allowed measure (“SRAdmin” cannot access “cost” measure) .

SECDW2Dimension. Finally, the *SECDW2Dimension* transformation obtains the structure and security of dimensions and bases by following the process shown in Table 6. Firstly, structural rules obtain dimensions, hierarchies and attributes for the dimensions and bases defined in our SECDW model. Each attribute of the base classes is added as an attribute of its related dimension in our target model. Next, the security rules are applied. These are composed of several rules which obtain security constraints defined over classes (dimension or base classes) and their attributes, and they then analyze the security roles which are involved to obtain dimension and attribute permissions. Security constraints detected at the class level generate dimension permissions for each authorized role, allowing access to this class. Attribute

Table 6. SECDW2Dimension transformation for hospital

top relation Package2Dimension: Hospital
relation SDimension2Dimension: Patient, Diagnosis
relation KeyProperty2KeyAttribute: (for Patient) ssn (for Diagnosis) codediagnosis
relation NonKeyProperty2Attribute: (for Patient) name, dateOfBirth, address (for Diagnosis) description, healthArea, validFrom, validTo
relation SBase2AAttributes: City, DiagnosisGroup
relation createDimensionSIARForSCompartment: Not thrown
relation createDimensionSIARForSRole: (for Patient) Health, Admin (for Diagnosis) Health
relation createDimensionSIARForLevel: (for Patient) S (for Diagnosis) S (for City) C (for DiagnosisGroup) C
relation authorizeSCompartment: Not thrown
relation authorizeSRole: (for Patient) Health, Admin and their descendants (for Diagnosis) Health and its descendants
relation authorizeSLevel: (for Patient) S, TS (for Diagnosis) S, TS (for City) C, S, TS (for DiagnosisGroup) C, S, TS
relation processSecureProperty: (for Patient) address
relation createPositiveAttributePermission: allowed roles (Admin and its descendants)
relation createNegativeAttributePermission: denied roles (distinct to allowed roles)

permissions are also created for the security constraints defined at the attribute level, defining positive attribute permissions for each authorized role and negative attribute permissions to avoid access to unauthorized users.

Table 7 shows a piece of source code for one rule of our developed set of QVT transformation which obtains security constraints defined at the conceptual level over attributes and establishes this constraint by using an attribute permission with an explicit denial over this attribute for the involved roles. In this example, a security constraint over “address” attribute of “Patient” dimension allowing access to security role “Admin” was defined. This rule creates attribute permissions for each unauthorized roles (roles distinct to “SRAdmin” and its descendants) with a denied set over “address” attribute of “Patient” dimension.

Table 7. Relation createNegativeAttributePermissions

relation createNegativeAttributePermissions
checkonly domain pim sp:SecureProperty { name = spName; } enforce domain psm dp:DimensionPermission { ID = "DimensionPermission"+ID; Name = "DimensionPermission"+ID; ownedAttributePermissions= OWNATTPERMS:Set(AttributePermission);} enforce domain psm at:AttributePermission { AttributeID = spName; DeniedSet = "["+sp.class.name+"].["+sp.name+"]"; }

Figure 9 shows the model obtained from SECDW2Dimension transformation in which are defined each dimension (“Patient” and “Diagnosis”), attributes (key attributes, non key attributes and attributes derived from its related bases), hierarchies and security permissions over dimensions and attributes (positive and negative permissions).

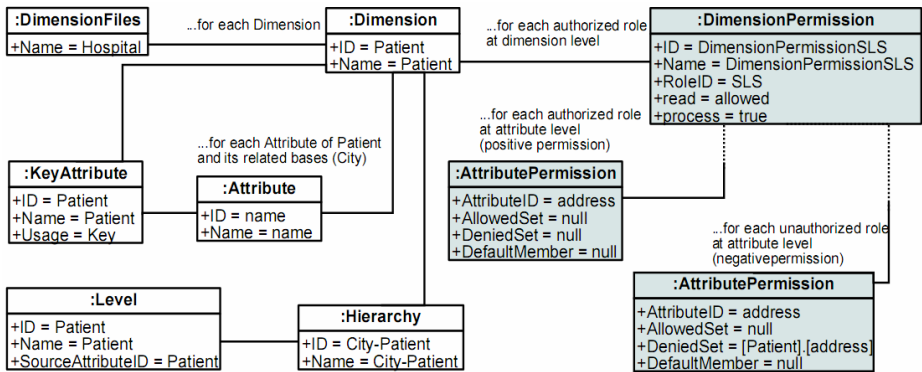


Fig. 9. Secure multidimensional PSM for hospital: dimensions

4.3 Secure Multidimensional Code

Finally, the security multidimensional code for SSAS is obtained from the logical model (PSM). Although SSAS has some particularities, this model, with which to text transformation, is easy to obtain. SSAS manages multidimensional elements (such as cubes, dimensions, hierarchies or measures) and also considers the establishment of security measures over these multidimensional elements with security permissions over cubes, dimensions, cells and attributes.

Table 8. Secure multidimensional Code for hospital: Admission cube

```
<Cube>
  <ID>Admission</ID>
  <Name>Admission</Name>
  <Dimensions>...</Dimensions>
  <MeasureGroups>...</MeasureGroups>
  <CubePermissions>
    <CubePermission>
      <ID>CubePermissionSLS</ID>
      <Name>CubePermissionSLS</Name>
      <RoleID>SLS</RoleID>
      <Process>true</Process>
      <Read>Allowed</Read>
      <CellPermissions>
        <CellPermission>
          <Access>Read</Access>
          <Expression>[Measures].[type]</Expression>
        </CellPermission>
      </CellPermissions>
    </CubePermission>
    ...(cube permissions for each authorized role)
  </CubePermissions>
</Cube>
```

The first example correspond to a security rule defined at conceptual level that allows accesses to “Admission” measures for security level secret or upper and security roles “Health”, “Admin” and their descendants. Table 8 shows a piece of the final code for SSAS with a cell permission over attribute “type” that allows access to security level secret (role “SLS”). In this example we have used a positive permission to allow access to “type” and we have thus denied access to the remaining cube measures (attributes of the “Measures” dimension).

At conceptual level we have defined a rule that hides the “Diagnosis” dimension from users with a security level which is lower than “Secret” (“SLC” and “SLU” at the logical level) and with a security role which is not “Health” or “Admin” or their descendants (“SRMaintenance” at the logical level). Table 9 shows secure multidimensional code obtained from PSM for the “Diagnosis” dimension which hides all its attributes from unauthorized roles. Due to space constraints, this table only shows a piece of the code in which the “DiagnosisGroup” attribute is hidden from users with the “Confidential” security level (“SLC” role). The rest of the code similarly defines attribute permissions for each attribute of the “Diagnosis” dimension and dimension permission for each unauthorized role.

Table 9. Secure multidimensional Code for hospital: Diagnosis dimension

```

<Dimension>
  <ID>Diagnosis</ID>
  <Name>Diagnosis</Name>
  <Attributes>...</Attributes>
  <DimensionPermissions>
    <DimensionPermission>
      <ID>DimensionPermissionSLC</ID>
      <Name>DimensionPermissionSLC</Name>
      <RoleID>SLC</RoleID>
      <Read>Allowed</Read>
      <AttributePermissions>
        <AttributePermission>
          <AttributeID>DiagnosisGroup</AttributeID>
          <DeniedSet>[Diagnosis].[DiagnosisGroup]</DeniedSet>
        </AttributePermission>
        ...(attribute permissions for each attribute of Diagnosis)
      </AttributePermissions>
    </DimensionPermission>
    ...(dimension permissions for each unauthorized role)
  </DimensionPermissions>
</Dimension>

```

Figures 10 and 11 show screenshots of the code generated for this example by working with SSAS in which we can see the result of executing two queries that check a security rule defined at conceptual level that has been automatically translated at logical level by using the set of QVT rules defined and has been finally implemented in SSAS by obtaining the corresponding secure multidimensional code from this logical model. At conceptual level, in our PIM model (Figure 6), we have defined a security constraint over “Patient” dimension that permits accesses to security level secret and upper and security roles “Health”, “Admin” and their descendants. When

logical models are obtained from this conceptual model by applying our set of QVT rules (see Figure 9), this security constraint is transformed into dimension permissions that deny accesses to unauthorized roles (security levels “SLC” and “SLU”, and security roles distinct to authorized roles).

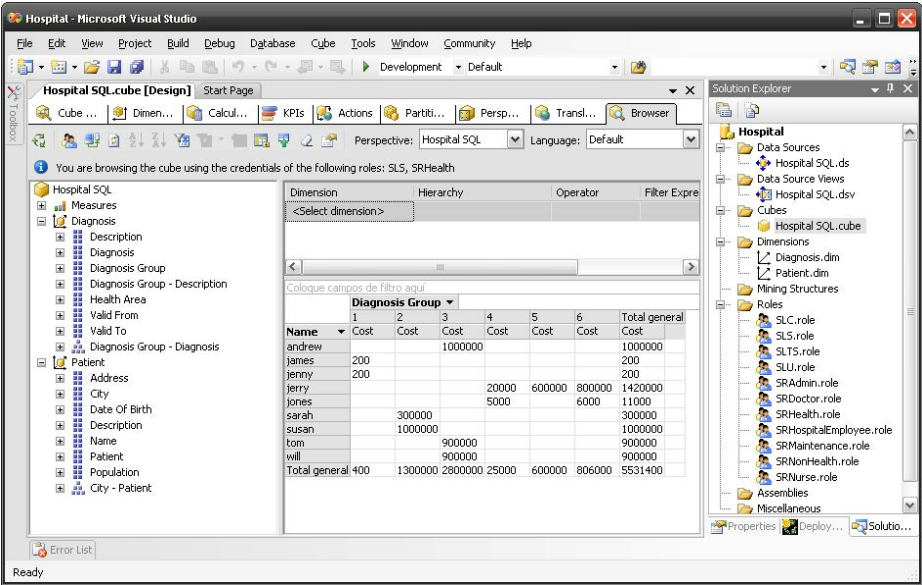


Fig. 10. SSAS implementation for hospital: authorized query

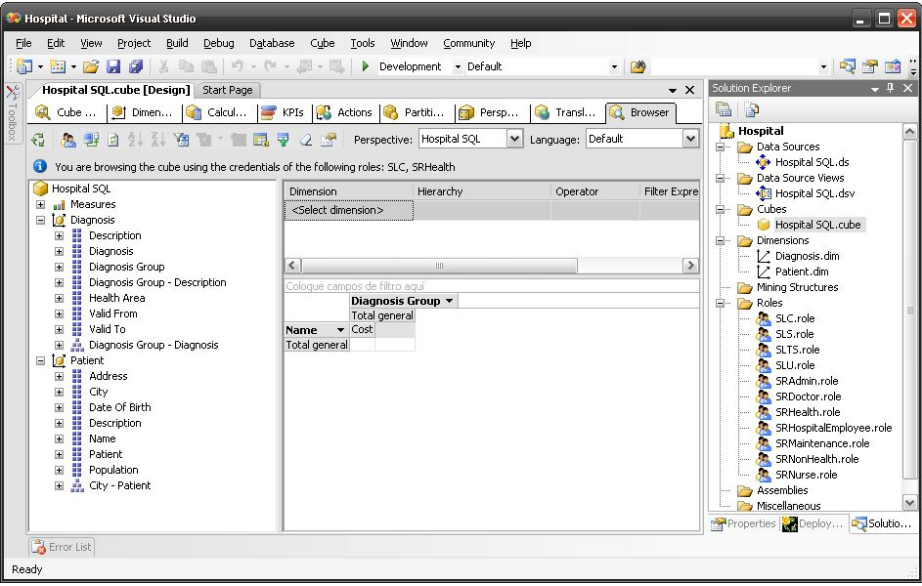


Fig. 11. SSAS implementation for hospital: unauthorized query

Firstly, an authorized user with the security level “Secret” (“SLS” role at the logical level) and the security role “Health” (“SRHealth” role at the logical level) makes a query involving attributes from the “Diagnosis” and “Patient” dimensions. Figure 10 shows the result of this query. Next, an unauthorized user with a lower security level, “Confidential” (“SLC” role at the logical level) and the same security role, “Health”, makes the same query, but in this case it is an unauthorized query and the requested information is hidden. Figure 11 shows the result of this unauthorized query.

5 Conclusions

This work shows the advantages of applying an MDA approach to the development of DWs by analyzing PIM to PSM and PSM to code transformations, which is then applied to an example. This approach allows us to automatically develop DWs, thus saving time and money and obtaining better quality and security by translating the requirements identified at early stages of development into the final implementation.

We have defined the necessary metamodels at the logical level (PSM), multidimensional secure code for a specific OLAP platform (SSAS) and the transformations to obtain PSM from conceptual models defined according to our SECDW metamodel and secure multidimensional code in SSAS from PSM. Furthermore, we have analyzed an example in which we have obtained secure multidimensional PSM and code from a conceptual model of a hospital.

In future works, we intend to improve our MDA architecture for the development of secure DWs in several ways. New security rules and constraints will be included in our ACA model in order to consider the security threats related to specific OLAP operations such as navigations or inferences. These transformations from PIM with which to include the advanced security rules defined in SECDW will be extended by using OCL notes, and we shall also define the transformation from PSM to code for other OLAP tools such as Pentaho and Oracle, and the inverse transformations from code to PSM and PIM.

Acknowledgments. This research is part of the ESFINGE (TIN2006-15175-C05-05) and METASIGN (TIN2004-00779) Projects financed by the Spanish Ministry of Education and Science, and of the MISTICO (PBC-06-0082) Project financed by the FEDER and the Regional Science and Technology Ministry of Castilla-La Mancha (Spain).

References

1. Dhillon, G., Backhouse, y.J.: Information system security management in the new millennium. *Communications of the ACM* 43(7), 125–128 (2000)
2. Mouratidis, H., Giorgini, y.P.: An Introduction. In: *Integrating Security and Software Engineering: Advances and Future Visions*. Idea Group Publishing (2006)
3. MDA, O.M.G., *Model Driven Architecture Guide* (2003)
4. OMG, MOF QVT final adopted specification (2005)
5. OMG, Meta Object Facility (MOF) specification (2002)
6. Fernández-Medina, E., Trujillo, J., Piattini, y.M.: Model Driven Multidimensional Modeling of Secure Data Warehouses. *European Journal of Information Systems* 16, 374–389 (2007)

7. Katic, N., Quirchmayr, G., Schiefer, J., Stolba, M., Tjoa, y.A.: A Prototype Model for DW Security Based on Metadata. In: en 9th Int. Workshop on DB and Expert Systems Applications, Vienna, Austria (1998)
8. Kirkgöze, R., Katic, N., Stolda, M., Tjoa, y.A.: A Security Concept for OLAP. In: en 8th Int. Workshop on Database and Expert System Applications, Toulouse, France (1997)
9. Priebe, T., Pernul, y.G.: A Pragmatic Approach to Conceptual Modeling of OLAP Security. In: en 20th Int. Conference on Conceptual Modeling, Yokohama, Japan (2001)
10. Mazón, J.-N., Trujillo, y.J.: An MDA approach for the development of data warehouses. *Decision Support Systems* 45(1), 41–58 (2008)
11. Soler, E., Stefanov, V., Mazón, J.-N., Trujillo, J., Fernández-Medina, E., Piattini, y.M.: Towards Comprehensive Requirement Analysis for Data Warehouses: Considering Security Requirements. In: en Proceedings of The Third International Conference on Availability, Reliability and Security (ARES). IEEE Computer Society, Barcelona (2008)
12. Yu, E.: Towards modelling and reasoning support for early-phase requirements engineering. In: en 3rd IEEE International Symposium on Requirements Engineering (RE 1997), Washington, DC (1997)
13. Fernández-Medina, E., Trujillo, J., Villarroel, R., Piattini, y.M.: Developing secure data warehouses with a UML extension. *Information Systems* 32(6), 826–856 (2007)
14. Soler, E., Trujillo, J., Fernández-Medina, E., Piattini, y.M.: SECRDW: An Extension of the Relational Package from CWM for Representing Secure Data Warehouses at the Logical Level. In: en International Workshop on Security in Information Systems, Funchal, Madeira, Portugal (2007)
15. Luján-Mora, S., Trujillo, J., Song, y.I.-Y.: A UML profile for multidimensional modeling in data warehouses. *Data & Knowledge Engineering* 59(3), 725–769 (2006)
16. Fernández-Medina, E., Trujillo, J., Villarroel, R., Piattini, y.M.: Access control and audit model for the multidimensional modeling of data warehouses. *Decision Support Systems* 42(3), 1270–1289 (2006)
17. Blanco, C., Fernández-Medina, E., Trujillo, J., Piattini, y.M.: Implementing Multidimensional Security into OLAP Tools. In: en Third International Workshop Dependability Aspects on Data Warehousing and Mining applications (DAWAM 2008). IEEE Computer Society, Barcelona (2008)

Appendix A: Acronyms

ACA: Access Control and Audit model
 AR: Audit Rule
 AUR: Authorization Rule
 C: Confidential
 CIM: Computer Independent Model
 CWM: Common Warehouse Metamodel
 DAC: Discretionary Access Control
 DBMS: Database Management System
 DW: Data Warehouse
 MDA: Model Driven Architecture
 MDSC: Multidimensional Security
 Constraint Language
 MDX: Multidimensional Expressions
 MOF: Meta-Object Facility
 OCL: Object Constraints Language

OLAP: On-Line Analytical Processing
 OMG: Object Management Group
 PIM: Platform Independent Model
 PSM: Platform Specific Model
 QVT: Query / Views / Transformations
 RBAC: Role-Based Access Control
 S: Secret
 SC: Security Compartment
 SIAR: Sensitive Information Assignment Rule
 SL: Security Level
 SR: Security Role
 SSAS: SQL Server Analysis Services
 TS: Top Secret
 U: Undefined

Trusted Reputation Management Service for Peer-to-Peer Collaboration*

Lingli Deng, Yeping He, and Ziyao Xu

Institute of Software, Chinese Academy of Sciences
No.4 NanSi Street, ZhongGuanCun, Beijing, 100190, P.R. China
{denglingli, yphe, ccxu}@ercist.iscas.ac.cn

Abstract. The open and autonomous nature of peer-to-peer (P2P) systems invites the phenomenon of widespread decoys and free-riding. Reputation systems are constructed to ensure file authenticity and stimulate collaboration. We identify the authenticity, availability and privacy issues concerning the previous reputation management schemes. We propose to add integrity control for the reputation storage/computation processing in order to enhance the authenticity of the resultant reputation values; and present an integrity model to articulate necessary mechanisms and rules for integrity protection in a P2P reputation system. We design a fully-distributed and secure reputation management scheme, Trusted Reputation Management Service (TRMS). Employing Trusted Computing and Virtual Machine Technologies, a peer's reputation values and specific transaction records can be stored, accessed and updated in a tamper-proof way by the Trusted Reputation Agent (TRA) on the same platform, which guarantees the authenticity of reputation values. Transaction partners exchange directly with each other for reputation values, services and transaction comments with no reliance on a remote third party, ensuring the availability of reputation and peers' privacy.

Keywords: P2P, reputation management, data integrity, trusted computing, virtual machine.

1 Introduction

The open and autonomous nature of peer-to-peer systems invites the phenomenon of inauthentic files and free-riding. Since anyone can freely join and leave the system, it is easy to inject undesirable data, ranging from decoy files (that are tampered with or do not work)[1] to malware[2], without the fear of being punished. The prevalence of free-riders, peers who attempt to use the resources of others without sharing with them their own resources, has been reported to degrade system performance in popular P2P networks [3][4][5][6]. Reputation systems are constructed in P2P systems to prevent the spread of malicious data and to stimulate collaboration of selfish peers. A reputation system collects, distributes, and

* Supported by the National High Technology Research and Development Program ("863" Program) of China under grants No.2007AA010601 and No.2006AA010201.

aggregates feedback about users' past behavior, encouraging reciprocal behavior and deterring dishonest participation[7].

Two major processes take place in a reputation-based P2P system: the query and response process and the service and comment exchange process. In a query and response process, a requestor sends a resource query request to locate potential providers.¹ Upon receiving a query request which hits with a local resource, a provider may query the requestor's reputation and base its decision whether to respond or not on the requestor's reputation value. Upon receiving responses, the requestor chooses several provider candidates, issues reputation query requests for their reputation values, and chooses the most reputable one as the provider. In a typical service and comment exchange process, the provider maintains a list of current requestors in the order of descending reputation, and serves them in the same order. After consuming the service, the requestor submits a comment, to be used in the provider's reputation calculation.

By dividing a reputation system into a computation model and a reputation management scheme, we identify the authenticity, availability and privacy issues concerning the design of a secure management scheme. Reputation servers in centralized management schemes are prone to Denial of Service (DoS) attacks and have no guarantee for reputation availability, while distributed schemes lack effective evaluation and control over reputation agents to guarantee reputation authenticity. We propose Trusted Reputation Management Service (TRMS), a distributed scheme for reputation aggregation and distribution, to provide security guarantees while maintaining the system's overall efficiency and scalability.

Given a calculation model, the authenticity of a peer's reputation value depends on the authenticity of the involved data and calculating processes, which are prone to malicious modification in a distributed scheme. Therefore, a variation of Clark-Wilson model[8], Rep-CW, is proposed to address the integrity requirements and to guide the mechanism design of TRMS so that reputation authenticity in an open-natured P2P reputation system is assured. In particular, based on the available Trusted Computing (TC) and Virtual Machine (VM) technologies, each participating platform is equipped with a Trusted Reputation Agent (TRA), so that peers' reputation values and related transaction records are stored, accessed and updated by the local TRA in a trusted manner.

The availability of reputation is determined by the availability of the reputation server or the agent peers, and the efficiency of reputation aggregation and distribution processes. By combining a peer with its unique reputation agent into a single platform, TRMS ensures the agent's availability to an honest peer in the following way: first, the traffic overhead for reputation aggregation/distribution is minimized; second, the unfair impact on a peer's reputation due to its agent's resource limit or peer dynamics is eliminated since they represent the same platform owner. Moreover, a peer's private transaction record is kept within its own

¹ According to the resource routing mechanism used in the system, the requestor submits its request to (1) an index server (in a centralized unstructured P2P system); or (2) the whole system (in a distributed unstructured P2P system); or (3) specific index peer (in a structured P2P system).

platform. Finally, separation of running environments and access constraints are introduced to protect the agent against selfish or malicious local peers.

Our contributions include: (i) a discussion of the authenticity, availability and privacy issues in P2P reputation management; (ii) the proposal of the Rep-CW integrity model to enhance reputation authenticity; (iii) the design of TRMS, a distributed implementation of Rep-CW to solve these issues.

The paper is constructed as follows: Section 2 presents our motivation. Section 3 reviews related work. Section 4 describes Rep-CW. TRMS's design and analysis appear in Section 5 and 6. Section 7 concludes.

2 Motivation

A P2P reputation system addresses two concerns: (1) how to calculate a peer's reputation value based on its past transaction history; and (2) where to store and how to distribute peers' reputation values. Hence it can be divided into two layers accordingly: a management scheme on the bottom handling reputation storage and distribution and a calculation model on the top that aggregates the information, provided by the bottom layer, into a meaningful reputation value. We focus on the security and efficiency issues of a management scheme.

As summarized in [9], there are six key issues to be addressed by a cost-effective P2P reputation system: (1) *High accuracy*: the system should calculate the reputation value for a peer as close to its real trustworthiness as possible. (2) *Fast convergence speed*: the reputation aggregation should converge fast enough to reflect the true changes of peer behaviors. (3) *Low overhead*: the system should only consume limited resources for peer reputation monitoring and evaluation. (4) *Adaptiveness to peer dynamics*: since peers come and go in an ad hoc way, the system should adapt to peer dynamics instead of relying on predetermined peers. (5) *Robustness to malicious peers*: the system should be robust to various attacks by both independent and collective malicious peers. (6) *Scalability*: the system should scale to serve a large number of peers. These requirements fall into three groups. High accuracy and fast convergence speed are proposed for the calculation model, but also restrained by the quality of the reputation data provided by the management scheme, while adaptiveness and robustness are meant primarily for the management scheme. Low overhead and scalability are issues to be addressed by both layers. Unfortunately, most previous work make no clear distinction between the calculation model and the management scheme, and some tend to tackle security and efficiency issues on the model layer alone. Security assurance in reputation management has not gained enough attention.

Based on the architecture, existing management schemes are either centralized [10] or distributed [11][12][9]. By having a reputation server manage all peers' reputation, centralized schemes contradict the open and decentralized nature of P2P networking and provide poor scalability. Distributed management schemes aggregate peer transaction comments in a fully distributed manner. Neither of these schemes delivers satisfactory assurance for reputation authenticity, availability or privacy. The critical reputation server(s) in a centralized scheme

are prone to DoS attacks targeting reputation availability. Existing distributed schemes delegate the reputation management of a peer to another randomly selected peer (agent), with no mechanism to regulate the latter's behavior to ensure reputation authenticity and privacy. The attack model below summarizes various attacks on a P2P reputation system by exploiting vulnerabilities of these schemes, and highlights our motivation for TRMS.

2.1 Attack Model

We assume attackers are motivated either by selfish or malicious intent. A selfish attacker (free-rider) seeks to acquire unfair gainings of its own, while malicious rivals try to damage the utility of others or the whole system.

Fraud Attacks. Attackers targeting reputation authenticity may subvert the reputation system with fake transactions or identities through fraud attacks, including: *Collusion*, of a malicious collective extolling each other in a large number of fake transactions, seeking for high reputation values; *Imputation*, of a malicious peer or collective unfairly degrading a victim's reputation by unfounded complaints against a large number of fake transactions; *Sybil*, of a single malicious attacker launching a collusion or imputation collective by assuming multiple fake peer identities; *Faker*, of a peer with low reputation seeking unfair gainings by impersonating another highly reputable peer, or of an unauthorized agent manipulating reputation data by impersonating an authorized agent; and *Tamper*, of a malicious agent distributing tampered reputation data.

Availability Attacks. We consider three potential attacks, including: *Denial*, by malicious or selfish reputation agents which refuse to provide proper reputation service; *Agent-DoS*, denial of service attack by blocking certain reputation agents from proper functioning; and *Network-DoS*, with repeated requests for network-intensive reputation aggregation to congest the whole network.

Privacy Attacks. Two kinds of attacks are considered in this paper: *Census*, where malicious peers collect private records of the victim by simply querying for its reputation; and *Leakage*, of private records by compromised agents.

3 Related Work

Early reputation systems[10] deploy centralized management schemes. However, centralized schemes are not scalable to accommodate large-scale P2P networks, and are therefore used almost exclusively by centralized unstructured P2P networks (e.g. Maze[5]), where servers are used also for service location.

Distributed schemes are proposed to enhance reputation availability and system scalability through fully distributed reputation aggregation and distribution, such as P2PREP[11], EigenTrust[12], and hiRep[13]. In P2PREP, a peer's reputation is locally computed by and stored in its transaction partners. A reputation querying peer submits its request by flooding the entire system, and randomly

audits some of the votes received by sending a vote confirmation message to the voter to verify the vote, resulting in prohibitive traffic overhead. The other two distributed schemes delegate a peer's reputation management responsibility to another peer (agent). By directing the transaction comments and reputation query messages for a specific peer to its agent peer, they yield greater availability and scalability than P2PREP. However, another concern arises in these agent-based schemes: how to choose an agent from the rest of population for a given peer. EigenTrust[12] uses a distributed hash table (e.g. Chord[14]) in agent assignment for a peer by hashing its unique ID. All peers in the system aware of the peer's ID can thus locate its reputation agent. It relies on the robustness of a well-designed DHT to cope with the network dynamics, and assigns multiple agents for a peer to provide resilience against malicious agents. On the other hand, hiRep[13] is proposed for managing reputation in unstructured P2P systems, where no DHT is present. Any peer can volunteer to function as a reputation agent. A peer maintains a list of acquainted agents, keeps updating their expertise values after every transaction, and chooses those with highest expertise as its trusted agents. A peer reports transaction comments only to its trusted agents, and checks only with them to fetch the reputation values of other peers.

Despite of better scalability and availability, the reputation authenticity in a distributed scheme is not as good as a centralized one, for it lacks an effective mechanism to ensure agents' proper behavior. Reputation accuracy is also degraded for incomplete reputation aggregation due to peer dynamics. As a distributed implementation of the Rep-CW model, TRMS ensures reputation authenticity by enforcing integrity, and yields high availability through distributed deployed TRAs. With neither reliance on central servers nor distributed storage structures, TRMS applies to both structured and unstructured P2P networks. Having its local TRA as a peer's only agent, TRMS improves reputation accuracy despite of peer dynamics, and blocks privacy leakage.

To protect reputation authenticity against imputation, TrustGuard[15] employs an electronic fair-exchange protocol to ensure that transaction proofs are exchanged before the actual transaction begins. But it still can not filter out in-authentic transactions between two collusive peers, who give good ratings with exchanged transaction proofs. Instead, collusion is handled at the calculation model layer by maintaining the submitter's credibility and using it as its comment's weight in reputation aggregation. However, without enhancement at the management layer, there is always unfair gaining for collusion, while TRMS is effective in identifying fake transactions coined by either collusion or imputation.

To encourage victims to report misbehavior honestly without fear of retaliation in a polling-based scheme, [16] proposes to provide anonymity for honest claims (i.e. negative comments) while preventing imputation by discarding repeated claims. Although a comment is not anonymous in TRMS, its content is encrypted with the recipient TRA's public key, hence a claim submitter's identity is hidden from its referenced peer; and as an authentic transfer is strictly tied to a single valid comment, imputation by replaying claims is also prevented.

A TC-based agent, protected by its local Trusted Platform (TP)[17] against unauthorized modification, is independent and may be trusted by remote entities as well as the owner of the TP, and has been used by some security proposals for P2P systems [18][19][20], as a virtual trusted third party in establishing mutual trust across platforms. A privacy-enhancing P2P reputation system is developed in [18] for B2C e-commerce, where a trusted agent within the recommender's TP is introduced for forming and collecting sensitive recommendations. [19] proposes a TC-based architecture to enforce access control policies in P2P environments. A trusted reference monitor (TRM) is introduced to monitor and verify the integrity and properties of running applications in a platform, and enforce policies on behalf of object owners. Moreover, a TRM can monitor and verify the information a peer provides to ensure data authenticity (e.g. check the response message to ensure that the responder's peer ID contained is authentic)[20].

4 P2P Reputation Integrity Model

We propose to improve reputation authenticity through effective fraud control at the reputation management layer by enforcing integrity. We introduce Rep-CW, a specialized Clark-Wilson model, to address the integrity policy in a P2P reputation system, demonstrate its effectiveness in preventing fraud attacks, and use it to analyze the vulnerabilities of previous management schemes.

4.1 Rep-CW Integrity Model

A security model characterizes a particular policy in a simple, abstract and unambiguous way and provides guidance in mechanism design for policy implementation. The Clark-Wilson (CW) integrity model [8] is celebrated as the origin for the goals, policies and mechanisms for integrity protection within computer systems. It is based on the well-established commercial practices, which have long served the goal to control error and fraud by enforcing integrity (regulating authorized data modifications as well as preventing unauthorized ones).

There are two kinds of data items in the CW model: Constrained Data Items (*CDIs*), to which the integrity model must be applied; and Unconstrained Data Items (*UDIs*), not covered by the integrity policy and may be manipulated arbitrarily. *UDIs* represent the way new information is fed into the system. A *CDI* is *valid*, if it meets the systems's integrity requirements; and the system is *in a valid state*, if all the *CDIs* are valid. The particular integrity policy desired is defined by two classes of procedures: Integrity Verification Procedures (*IVPs*), and Transformation Procedures (*TPs*). The purpose of an *IVP* is to confirm that all of the *CDIs* in the system conform to the integrity specification at the time the *IVP* is executed, while *TPs* are used to change the set of *CDIs* from one valid state to another. Data integrity assurance is achieved through the well-formed transaction, and separation of duty mechanisms. The former is meant to ensure *internal consistency* of data, so that a user should not manipulate data arbitrarily, but only in constrained ways that preserve or ensure the integrity of

the data, while the latter attempts to ensure the *external consistency* of the data objects, i.e. the correspondence between the data object and the real world object it represents, by dictating that at least two people are involved to cause a critical change. Assume that at some time in the past an *IVP* was executed to verify the system was in a valid state. By requiring the subsequent state transitions (*CDIs* change) are performed only by *TPs*, and each *TP* is certified to preserve the validity of system state, it is ensured that at any point after a sequence of *TPs*, the system is still valid. By enforcing 5 certification rules and 4 enforcement rules, CW assures data integrity in a two-part process: certification (C1-C5), which is done by the security officer, system owner, and system custodian with respect to an integrity policy; and enforcement (E1-E4) by the system.

Dealing with P2P networks of a highly dynamic and open nature, Rep-CW encounters several new issues that are not addressed by the CW model, which assumes a closed system environment. First, with the ever changing user (peer) group, it is not feasible to enforce certification rules by traditional enterprise regulations. Hence, in Rep-CW these rules are enforced by the program logic of related procedures (*TPs* and *IVPs*), and verified by integrity verifications based on programs' hash fingers. Second, as network transfers are needed for reputation aggregation and distribution, their integrity of both origin and content must be ensured by verification. Consequently, Rep-CW uses a digital certificate to bind a value to its origin, and employs integrity verification on a software entity to establish the trust on the integrity of its output data.

Table 1 presents the elements in Rep-CW: peers are the users of the reputation system; unverified transaction comments submitted by peers are new data

Table 1. Rep-CW Integrity Model for P2P Reputation

Element	Description
User	peer (representing the interest of its owner)
UDI	Transaction Comments (<i>TCs</i>) submitted by peers.
CDI	Valid transaction comments, in the form of Transaction Records (<i>TRs</i>); and valid reputation values as Reputation Certificates (<i>RCs</i>).
TP	Transaction Logging Agent (<i>TLA</i>): verifies the validity of received <i>TCs</i> , records valid ones into <i>TRs</i> ; Reputation Calculation Agent (<i>RCA</i>): calculates and updates peers' <i>RCs</i> using <i>TRs</i> .
IVP	Reputation Verification Agent (<i>RVA</i>): verifies the validity of <i>RCs</i> . Transaction Verification Agent (<i>TVA</i>): verifies the validity of <i>TRs</i> .
Rule	Content
C1	A peer's <i>RC</i> (<i>TR</i>) is accredited only after verification by <i>RVA</i> (<i>TVA</i>).
C2	All <i>TLAs</i> and <i>RCAs</i> must be certified by attestation to preserve validity.
E1	A peer's <i>TR</i> is updated only by its authorized <i>TLA</i> and <i>RCA</i> ; and its <i>RC</i> is calculated and issued only by the authorized <i>RCA</i> , accordingly.
E2	Reputation is updated only by valid <i>TC</i> submissions to authorized <i>TLA</i> .
C3	A valid transaction involves at least 2 authentic peers.
C5	All <i>TLAs</i> must be certified by integrity attestation to accept valid <i>TCs</i> into <i>TRs</i> and discard invalid ones.

fed to the system as *UDIs* and are accepted into transaction records (*CDIs*) after successful validity verification; peers' reputation certificates are also *CDIs* subject to system's integrity protection. The data validity is defined as follows.

Definition 1 (External Consistency). *A comment $tc = \langle \text{description}, \text{comment} \rangle$ submitted by peer P is valid, if $tc.\text{description}$ corresponds to a unique cross-platform file transfer from another peer Q .*

The validity of a comment contains two meanings: (1) *authenticity*, that there exists a cross-platform transfer from Q to P , which coheres with tc 's description; and (2) *uniqueness*, that tc is the first valid comment submitted for the transfer. In all, given an authentic file transfer, there is but one valid comment.

Definition 2 (Internal Consistency). *P 's reputation certificate $rc = \langle P, \text{value}, \text{valid period} \rangle$ is valid, if rc has not expired and is issued by a certified *RCA* authorized for managing P 's reputation.*

In a P2P reputation system, reputation value updates indicate transitions of system state, and correspond to the transformation procedure *RCA* in Rep-CW. To ensure external consistency, another transformation procedure *TLA* verifies the validity of each received transaction comment according to Definition 1, records only valid ones into transaction records. To ensure internal consistency, the integrity verification procedure *RVA* verifies the validity of reputation certificates according to Definition 2; while integrity verification procedure *TVA* verifies the validity of transaction records maintained by *TLA*.

Figure 1 shows how the rules (Table 1) of Rep-CW (C-Certification rules; E-Enforcement rules) control the system operation. An *TLA* checks newly submitted *TCs* and accepts valid ones into the system by updating *TRs*. An *RCA* takes *TRs* and *RCs* as input and produces new versions as output. These two sets of both *TRs* and *RCs* represent two successive valid states of the system, while an *RVA* (or *TVA*) verifies the validity of *RCs* (or *TRs*). Associated with each system part is the rule that governs it to ensure integrity.

Rep-CW prevents unauthorized modifications and regulates authorized modifications to reputation data in the following way: First, execution of *RVA* verifies the *RCs*' external and internal consistency (rule C1). Second, for any subsequent state transition (i.e. reputation update), the related *RCA* is certified by means of integrity verification based on program hash finger to ensure internal consistency of the changed *TRs* and *RCs* (rules C2 and C5). Third, the authorization lists specified in rules E1 and E2 are used to prevent unauthorized modifications on *TRs* and *RCs*. Fourth, by dictating that any valid transaction involve at least two authentic peers (rule C3), Rep-CW prevents a single attacker from manipulating reputation data. We preserve the numbering for rules in CW to clarify the correspondence between the two models.²

² Rep-CW contains no counterparts for the rules E3, E4, and C4 of the CW model, because: (1) in an open P2P collaboration environment, no prior authentication and authorization are imposed on participating peers' identities (E3 and E4); and (2) Rep-CW can not protect reputation certificates from manipulation, instead it blocks manipulated ones from being used in the system, so no log *CDI* is used (C4).

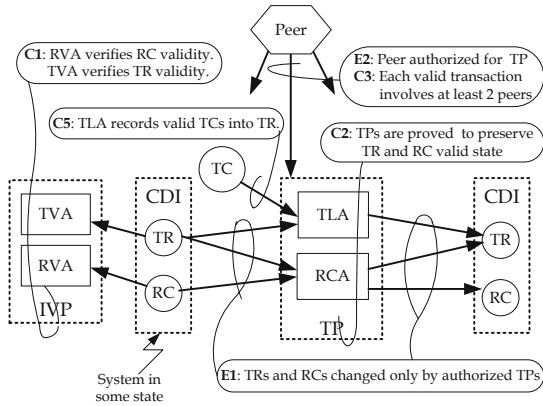


Fig. 1. Rep-CW Integrity Model

4.2 Integrity Analysis Based on Rep-CW

We demonstrate the effectiveness of Rep-CW in enhancing reputation authenticity by preventing fraud attacks. For each fraud attack in Section 2.1, Table 2 summarizes the corresponding Rep-CW integrity rules the attacker must break to launch a successful assault. First of all, to perform a collusion or imputation, comments on inauthentic transactions from attackers must be accepted by the system, which breaks rule C5. Second, by assuming multiple fake identities for a given physical platform and coining inauthentic transactions among them, a Sybil attacker virtually violates rules C3 and C5. Third, the act of a faker, who tries to use another peer’s reputation, is against rules C1 and E2. Finally, a malicious or compromised agent, exploited by a tamper attack, cannot pass the integrity verification to be an authorized *RCA*, as required by C1, C2 and E1.

Table 2. Fraud Attacks v.s. Integrity Rules

Attack	C1	C2	E1	E2	C3	C5
Collusion						×
Imputation						×
Sybil					×	×
Faker	×			×		
Tamper	×	×	×			

Table 3 presents a comparison of typical P2P reputation management schemes from the Rep-CW’s point of view. We make the following observations.

Previous distributed schemes are vulnerable to fraud attacks, because: first, they are prone to collusion and imputation attacks, as reputation agents cannot identify inauthentic transaction comments for the lack of *TLA* (in EigenTrust[12]

Table 3. A Comparison of Management Schemes

Scheme	TLA	RCA	TVA	RVA	C1	C2	E1	E2	C3	C5
eBay	×	✓	×	✓	✓	✓	✓	✓	×	×
P2PREP	✓	✓	×	×	×	×	×	×	×	×
EigenTrust	×	✓	×	✓	×	×	✓	✓	×	×
TrustGuard	✓	✓	×	✓	×	×	✓	✓	×	×
hiRep	×	✓	×	✓	×	×	✓	✓	×	×

and hiRep[13]) or the strict binding of accepted comments to authentic file transfer events (in P2PREP[11] and TrustGuard[15]); second, the (PKI-based) *RVA* procedure provided is not capable of filtering out tampered or inauthentic reputations issued by compromised agents; third, no mechanism for separation of duty is provided to suppress Sybil attacks.

In a centralized scheme (e.g. eBay[10]), the globally trusted reputation server acts as the unique authorized *RCA* in issuing reputation certificates for all peers. A reputation certificate is verified using the server’s public key by the querying peer (corresponding to *RVA*). All comments must be submitted to the server to be used in reputation update, while being immune to faker and tamper attacks, the system with a centralized scheme is still vulnerable to attacks using inauthentic transactions (e.g. collusion, imputation and Sybil), as no mechanisms for *TLA* or separation of duty is implemented. Moreover, the reputation server is prone to DoS attacks, and becomes the bottleneck for system scalability.

In summary, previous distributed schemes hardly provide any integrity protection against fraud attacks, while centralized schemes deliver limited integrity protection at the cost of system scalability and performance. To enhance reputation authenticity while maintaining system scalability and availability, we propose TRMS, a distributed implementation of Rep-CW, whose protection mechanisms are deployed on participating platforms, certified through TCG integrity attestation [21], and protected by Xen virtual machine environment [22].

5 TRMS: Trusted Reputation Management Service

5.1 Background: Trusted Computing and Virtual Machines

A Trusted Platform (TP) is a normal open computer platform equipped with a tamper-resistant hardware Trusted Platform Module (TPM) as the root of trust, providing three basic functionalities to propagate trust to application software and/or across platforms [21]. (1) Through *Integrity Attestation* mechanism, the integrity of a TP, including the integrity of many components of the platform, is recorded by the Platform Configuration Registers (PCRs) in TPM and can be checked by both local users and remote entities to deduce their trust in the platform [17]. (2) *Sealed storage* provides protection against theft and misuse of sensitive data held on the platform so that it is available only when the platform is in a particular integrity state, i.e. when the correct programs are running. (3)

An *Attestation Identification Key* (AIK) is created by the TPM and used in an attestation protocol to provide a signature over PCRs to prove authenticity.

The VM technology [23] allows multiple operating systems to simultaneously run on one machine. A Virtual Machine Monitor (VMM) is a software layer underneath the operating system that provides (1) a VM abstraction that models and emulates a physical machine and (2) isolation between VMs such that each VM runs in its own isolated sandbox. We use Xen [22], an open-source VMM. In Xen-speak, each VM is referred to as a *domain*, and Xen itself the *hypervisor*. The hypervisor remains in full control over the resources given to a domain. Domain0 (*Dom0*) is the first instance of an OS that is started during system boot as a management system for starting further domains. All other domains are user domains that receive access to the hardware under *Dom0*'s mediation.

5.2 Overview

TRMS equips each participating platform with a Trusted Reputation Agent (TRA) to manage local peers' reputation. Peers' reputation values and related transaction records are stored, accessed, and updated by the TRA on the local trusted platform. TRMS uses the TC mechanism for integrity measurement, storage and reporting to verify that a remote TRA is running in an expected manner. For a given transaction, the reputation values, service and comment are exchanged between the two directly involved platforms with no reliance on a remote third party, which guarantees reputation availability and eliminates the traffic overhead for network-wide reputation aggregation and distribution. To protect locally stored reputation data against manipulation by its selfish owners, TRA runs in a protected VM separated from peers to avoid run-time manipulation, and stores reputation data in sealed storage (encrypted and protected by TPM) against unauthorized access other than TRA.

In terms of integrity protection, TRMS realizes the *TPs* and *IVP* of the Rep-CW model through the TRA, by requiring that: (1) A peer's reputation certificate should be verified to be valid by the querying peer's TRA according to Definition 2 (corresponding to *RVA*). (2) The transaction comment, submitted by the consuming peer, is verified according to Definition 1 by the serving peer's local TRA (acting as *TLA*) before accepted into transaction records. (3) The TRA (functioning like *RCA*) on a participating platform is responsible for maintaining the verified transaction records and updating local peers' reputation certificates, according to the calculation model.

5.3 Architecture

There are two kinds of symmetric collaboration in a P2P network under TRMS: a Trusted Agent Community (TAC) formed by TRAs from participating platforms and the network of regular peers. A peer's reputation related information is managed by the TRA on its local platform, ensuring rule E2 in Rep-CW. In TRMS, for a peer to join a P2P network and interact with another peer, their local TRAs have to join TAC and establish mutual trust first. Through TRA

attestation (described later), a querying peer trusts another peer's reputation certificate only if the integrity of the latter's platform and TRA is verified. Rule C2 is ensured by denying reputation from unverified TRAs. Consider the illustrative scenario in Figure 2(a). $TRA_1 - TRA_4$ on $TP_1 - TP_4$ join TAC, and the peers on $TP_1 - TP_4$ join corresponding P2P networks. E.g., both $Peer_{11}$ on TP_1 and $Peer_{21}$ on TP_2 join a P2P network *yellow*, while TP_1 's another peer $Peer_{12}$ collaborates with $Peer_{41}$ on TP_4 in another P2P network *brown*.

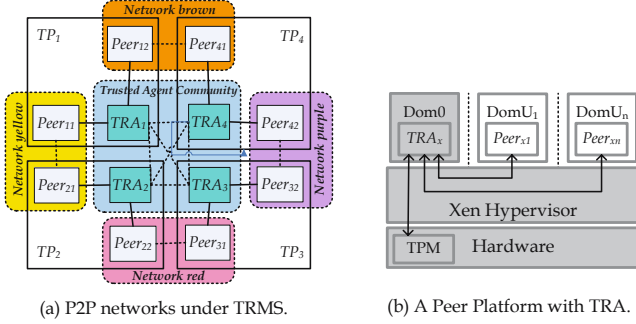


Fig. 2. TRMS Architecture

As the local reputation agent, TRA should be verified and trusted by remote querying peers. Selfish or malicious local peers (or even the platform's owner) have the motive to subvert, replace, or manipulate the TRA and/or its data. Therefore, Xen virtual machines are used to separate TRA from local peers. Figure 2(b) depicts the architecture of a participating platform with TRA, based on Xen. The trusted components (the shaded areas) includes the trusted hardware (TPM), the security kernel (Xen Hypervisor and Dom0), and TRA running in Dom0. The hardware and security kernel provide TRA with necessary security services, including basic cryptographic functions, platform and program attestation, sealed storage, and protected running environments. TRA's sensitive data includes local peers' reputation values and transaction records. The isolation of VMs protects data and processing integrity from being tampered locally.

The following constraints are enforced by each participating platform's trusted components (the first by secure kernel; and the other two by TRA):

1. There is always a unique TRA running on the local platform.
2. There is always at most one local peer joined in a given P2P network.
3. Transactions between local peers are excluded for reputation calculation.

Constraint 1 is intended to prevent malicious local peers or platform owners from subverting TRA's monitoring (rules E1 and E2). Constraints 2 and 3 enforce the Separation of Duty principle (rule C3) by dictating a valid transaction involve two physical platforms. The certification for their proper enforcement is achieved by the integrity attestation of TRA and its platform.

5.4 TRA: Trusted Reputation Agent

Our design is based on three basic assumptions: (1) TC hardware is tamper-resistant; (2) the isolation of VMs is flawless; (3) each participating platform is a trusted platform, with necessary TC hardware and VMM software.

By local/remote attestation, local/remote peers verify the integrity of the issuing TRA and its running environment (both hardware TPM and security kernel software) before accrediting a reputation certificate. We assume the following credentials are bestowed on a trusted platform when performing attestation.

- TPM's AIK, (PK_{AIK}, SK_{AIK}) , is produced by TPM to be used to prove the authenticity of PCR values or programs' public key certificates to a remote challenger. Its private part is protected by TPM, and its public part is issued by a private CA or through Direct Anonymous Authentication protocol.
- TRA's Asymmetric Key, (PK_{TRA}, SK_{TRA}) , used to sign or encrypt data exchanged between TRAs. Its private part is protected by the TPM, and the public key certificate is issued by TPM using AIK.

Employing TPM, TRA has the following primitives:

- $TRA.Seal(H(TRA), x)$, used by TRA to seal x with its own integrity measurement $H(TRA)$. The sealed x is unsealed by TPM only to the same TRA whose integrity measurement equals $H(TRA)$.
- $TRA.Unseal(H(TRA), x)$, unseals x , if $H(TRA)$ was used in sealing x .
- $TRA.GenNonce(n)$, generates a random number n .
- $TRA.Attes(H(TRA), PK_{TRA})$, responds to a remote attestation challenge, by returning local TRA's public key certificate, bound to $H(TRA)$ and signed by local TPM's AIK: $(H(TRA), PK_{TRA})_{SK_{AIK}}$.

5.5 Cross-Platform Interactions in TRMS

According to Rep-CW, TRMS provides three cross-platform interactions: (1) the mutual attestation between TRAs to ensure they (the corresponding *RCAs*) preserve the validity of reputation data; (2) the reputation query process, in which the querying peer verifies (by calling *RVA* implemented by TRA) the reputation certificate's validity according to Definition 2; and (3) the modified service and comment exchange process, where the serving peer's TRA (the corresponding *TLA*) verifies received transaction comment's validity according to Definition 1.

Mutual Attestation between TRAs. Once two peers are to establish mutual trust for a reputation query process, they use TC-based remote attestation to verify the integrity of the other's TRA and platform, before accrediting the other's reputation certificate. Suppose TRA_A on platform TP_A wishes to verify the integrity of TRA_B on platform TP_B . They interact as follows: (Figure 3(a))

1. TRA_A sends TRA_B a *Attestation Request* (*AReq*) via $Peer_A$ and $Peer_B$.
2. TRA_B calls $TRA_B.Attest$ to get its integrity certificate, and wrap it and AIK's certificate with a *Attestation Response* (*ARes*): $(PK_{AIK_B})_{SK_{CA}} \parallel (H(TRA_B), PK_{TRA_B})_{SK_{AIK_B}}$.

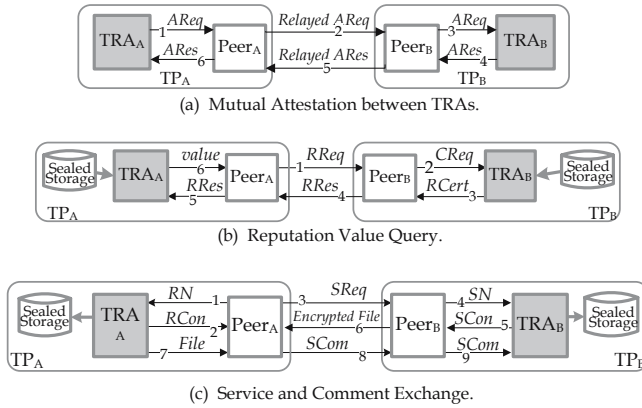


Fig. 3. Cross-Platform Interactions in TRMS

3. *ARes* sent by TRA_B is handed over to TRA_A via $Peer_B$ and $Peer_A$.
4. TRA_A verifies the validity and integrity of the certificates in *ARes*.³

Reputation Value Query. Suppose peer $Peer_A$ wishes to query the reputation of peer $Peer_B$, and the involved TRAs have exchanged public keys (PK_{TRA_A} and PK_{TRA_B}) during a prior successful mutual attestation. According to Rep-CW's rule C1, $Peer_A$ queries for $Peer_B$'s reputation value and verifies its validity as follows: (Figure 3(b))

1. $Peer_A$ sends a *Reputation Request* (*RReq*) to $Peer_B$.
2. $Peer_B$ checks its *Reputation Certificate* (*RCert*) issued by TRA_B on TP_B . If it is still valid, $Peer_B$ skips to Step 4; otherwise, it sends a *Certification Request* (*CRReq*) message to TRA_B , asking for TRA_B a new one.
3. To respond to *CRReq*, TRA_B first calls $TRA_B.Unseal$ to read $Peer_B$'s reputation data from sealed storage, recalculates $Peer_B$'s reputation value; then calls $TRA_B.Seal$ to seal the new reputation data, and issues a new $RCert(Peer_B) = (Peer_B, value, T_{issue}, T_{expiry}, H(TRA_B))_{SK_{TRA_B}}$.
4. $Peer_B$ wraps its valid *RCert* with a *Reputation Response* (*RRes*) to $Peer_A$.
5. TRA_A verifies *RCert*'s validity, and returns the reputation value to $Peer_A$.

Service and Comment Exchange. To enable validity verification for transaction comments (Rep-CW's rule C5), TRMS makes several modifications to the regular service and comment exchange process: First, to filter out fake transactions used by collusion or imputation attackers, the two peer's TRAs are actively involved as witnesses for each cross-platform file transfer event through the request notification/registration. Second, random nonce is used to uniquely identify each authentic transfer, preventing a dishonest peer from replaying comment

³ To realize mutual attestation in one process, we can modify the above protocol in the following way: TRA_A generates its own integrity certificate, and sends it to TRA_B along with TP_A 's certificate, via the *AReq* message in Step 1; and TRA_B verifies their validity before executing Step 2.

duplicates. Third, to deal with potential conceal of negative comments by dishonest local peers, on one hand, the remote reporting peer encrypts its comment with the public key of the serving peer's TRA, hence the serving peer cannot perform content-based filtration against negative comments; on the other hand, a timeout mechanism is included so that a negative comment is automatically generated for the serving peer by its TRA if no valid comment is received in time, hence the serving peer cannot perform source-based filtration either.

Suppose $Peer_A$ on TP_A wishes to download a file from $Peer_B$ on TP_B . It is assumed that the exchange for public keys of TRA_A and TRA_B has been exchanged by a prior mutual attestation. The process is depicted in Figure 3(c).

1. $Peer_A$ sends a *Request Notification* (RN) to TRA_A , including the service description, ($Peer_B$, agent TRA_B , and the requested file's description⁴) expecting a *Request Confirmation* ($RCon$) in reply.
2. TRA_A registers the received RN and generates a $RCon$ message, containing the description from RN and a $nonce_A$ returned by $TRA_A.GenNonce$. $RCon(Peer_A, desc) = ((nonce_A)_{PK_{TRA_B}}, desc, Peer_A)_{SK_{TRA_A}}$.
3. $Peer_A$ wraps the $RCon$ with a *Service Request* ($SReq$) to $Peer_B$.
4. If $Peer_B$ consents to serve $Peer_A$ with the described file, it sends a *Service Notification* (SN) to TRA_B , carrying the $RCon$ from TRA_A .
5. TRA_B registers the file description, the $nonce_A$ in SN and another $nonce_B$ returned by $TRA_B.GenNonce$, and encrypts them with TRA_A 's public key as a *Service Confirmation* ($SCon$) in response to SN from $Peer_B$. $SCon(Peer_B, SN) = (desc, nonce_A, nonce_B)_{PK_{TRA_A}}$.
6. $Peer_B$ encrypts the file with TRA_A 's public key PK_{TRA_A} , and sends it along with the $SCon$ from TRA_B , to TRA_A via $Peer_A$.
7. TRA_A verifies the received $SCon$ and the $SReq$ contained, makes sure that the $nonce_A$ in $SReq$ is used in Step 1, decrypts the file and $nonce_B$, checks the file against $SReq$'s description,⁵ and gives file and $nonce_B$ to $Peer_A$.
8. $Peer_A$ generates a *Service Comment* ($SCom$), $SCom(nonce_B) = (desc, comment, nonce_B)_{PK_{TRA_B}}$, and sends it to $Peer_B$.
9. $Peer_B$ forwards received $SCom$ to TRA_B , who updates $Peer_B$'s transaction record after a successful verification of $SCom$'s $desc$ and $nonce_B$ against its registration records. For each item registered in Step 5, a negative comment is generated automatically by TRA_B , if no valid $SCom$ is received in time.

Summary. In a reputation-based P2P network using TRMS, the query and response process is similar to the one described in Section 1, and a reputation query process is also contained: a service request also serves as a reputation request, and a reputation response is combined with a service response. The service querying peer selects the most reputable responder to submit its service request, triggering a service and comment exchange described above.

⁴ E.g., the file's hash finger, whose authenticity is verifiable by a third party.

⁵ The objective description (such as file's hash finger) is verified by TRA; while the subjective satisfaction is evaluated by the consuming peer later in a comment.

6 Analysis of TRMS

Traffic Overhead. As a peer's TRA resides on its local platform, the network traffic in TRMS occurs exclusively between actual or potential transaction partners. All the network messages can be piggybacked back to the regular service query and exchange messages, except the *SCom* message carrying the transaction comment. As each valid *SCom* is sent from the consumer to the provider for an authentic cross-platform transfer, the traffic overhead is minimal, compared with other distributed schemes performing costly network-scale aggregation periodically.

Table 4. Rep-CW Rules as Implemented by TRMS

Rule	Implementation in TRMS
C1	Before accrediting a reputation certificate, the querying peer calls its TRA to verify the integrity of both the certificate and the issuing TRA. Since transaction records are the private data of TRA and are protected by sealed storage, TRMS provides no further verification of the records.
C2	All TRAs are certified by integrity attestation to preserve reputation validity, otherwise reputation certificates issued by them would be discarded.
E1	Only the certificates issued by certified TRAs are used by honest peers, and a peer's reputation certificate is issued only by its local TRA.
E2	A peer must submit valid transaction comments to its local TRA to have its reputation updated, since: TRA's signature on a certificate ensures the identification and exclusion of tampered reputation; and transaction records, protected by sealed storage, are only accessible to the local TRA.
C3	Since there is at most one local peer participating in a P2P network at any time on the same platform, the generation (by <i>TLA</i> and <i>RCA</i>), verification (by <i>RVA</i>) and usage of a given peer's reputation must involve two platforms.
C5	TRA verifies received comments' validity and discards invalid ones.

Reputation Authenticity. As demonstrated by Table 4, all the integrity rules in Rep-CW are effectively implemented by TRMS. By implementing these rules, TRMS eliminates the weaknesses exploited by various fraud attackers (Table 2). Specifically, in TRMS: (1) invalid comments about fake transactions minted by collusion or imputation attackers are discarded; (2) Sybil attacker's cost for a fake identity is enhanced greatly since each authentic peer has to be on a single physical platform to form an effective collective; (3) the trust chain from a TP via TRA to a peer's reputation certificate must be in place for its value to be accredited, which guarantees the identification and exclusion of tamper attacks; (4) since TRA is the only authorized *RCA* for local peers and there is at most one local peer in a given P2P network, a faker can use neither a remote peer's reputation certificate (not issued by the local TRA) nor another local peer's certificate (not in the same P2P network).

Reputation Availability. It is clear that there is no motive for Denial attacks, since an agent (TRA) and the peers it serves reside on a single platform and represent the same owner. TRMS also provides resilience against DoS attacks:

On one hand, interacting through local peers, TRAs do not handle network traffic directly, therefore are immune to remote Agent-DoS attackers; on the other hand, DoS attacks, which block TRA from functioning by repeated requests, are not in the interest of local peers. Moreover, each platform has its own dedicated TRA, so Agent-DoS attacks targeting one or a few TRAs can hardly affect the whole system. Finally, as any reputation-related process involves at most two platforms, with minimal traffic overhead, there is little chance for a Network-DoS attack.

Transaction Privacy. TRMS confines the exposure of a peer's transaction record to the verified local TRA. First, TRMS does not support transaction history query for a specific peer and the *RRes* message contains only the reputation value not record, which prevents census attacks. Second, a peer's transaction record is protected by sealed storage and accessible only to its local TRA.

Summary. Compared with previous P2P reputation management schemes, TRMS enhances reputation authenticity by the trusted and verifiable data management and a strong binding of a transaction comment with an authentic cross-platform file transfer. Given a calculation model, the system yields higher reputation accuracy as a peer's transaction history is exclusively and more completely collected by its local TRA in the presence of peer dynamics. As a general reputation management scheme, it can be used to support various reputation calculation models. It provides stronger robustness against reputation attacks and delivers great network scalability as a totally distributed scheme, with minimal traffic overhead.

7 Conclusion

We identify the authenticity, availability and privacy issues in P2P reputation management; propose Rep-CW integrity model to address the reputation authenticity requirements; and design TRMS, a fully-distributed reputation management scheme, using available TC and VM technologies, with enhanced reputation authenticity, availability and privacy guarantees.

References

1. Liang, J., Kumar, R., Ross, K.W.: Pollution in P2P File sharing systems. In: 24th Annual Joint Conference of the IEEE Computer and Communications Societies, pp. 1174–1185. IEEE Press, New York (2005)
2. Kalafut, A., Acharya, A., Gupta, M.: A Study of Malware in Peer-to-Peer Networks. In: 6th ACM SIGCOMM conference on Internet measurement, pp. 327–332. ACM Press, New York (2006)
3. Adar, E., Huberman, B.A.: Free riding on Gnutella. *First Monday* 5(10), 2 (2000)
4. Saroiu, S., Gummadi, P.K., Gribble, S.D.: A Measurement Study of Peer-to-Peer File Sharing Systems. In: Kienzle, M.G., Shenoy, P.J. (eds.) MMCN 2002. SPIE Press (2002)
5. Yang, M., Zhang, Z., Li, X., Dai, Y.: An Empirical Study of Free-Riding Behavior in the Maze P2P File-Sharing System. In: Castro, M., van Renesse, R. (eds.) IPTPS 2005. LNCS, vol. 3640, pp. 182–192. Springer, Heidelberg (2005)

6. Pouwelse, J.A., Garbacki, P., Epema, D.H.J., Sips, H.J.: The Bittorrent P2P File-Sharing System: Measurements and Analysis. In: Castro, M., van Renesse, R. (eds.) IPTPS 2005. LNCS, vol. 3640, pp. 205–216. Springer, Heidelberg (2005)
7. Resnick, P., Kuwabara, K., Zeckhauser, R., Friedman, E.: Reputation Systems. *Commun. ACM* 43(12), 45–48 (2000)
8. Clark, D.D., Wilson, D.R.: A Comparison of Commercial and Military Computer Security Policies. In: IEEE Symposium on Security and Privacy, pp. 184–194. IEEE Press, New York (1987)
9. Zhou, R., Hwang, K.: PowerTrust: A Robust and Scalable Reputation System for Trusted Peer-to-Peer Computing. *IEEE Trans. Parallel Distrib. Syst.* 18(4), 460–473 (2007)
10. Houser, D., Wooders, J.: Reputation in Auctions: Theory and Evidence from eBay. *Journal of Economics & Management Strategy* 15(2), 353–369 (2006)
11. Damiani, E., di Vimercati, D.C., Paraboschi, S., Samarati, P., Violante, F.: A Reputation-Based Approach for Choosing Reliable Resources in Peer-to-Peer Networks. In: 9th ACM conference on Computer and Communications Security, pp. 207–216. ACM Press, New York (2002)
12. Kamvar, S.D., Schlosser, M.T., Garcia-Molina, H.: The Eigentrust Algorithm for Reputation Management in P2P Networks. In: 12th International Conference on World Wide Web, pp. 640–651. ACM Press, New York (2003)
13. Liu, X., Xiao, L.: hiREP: Hierarchical Reputation Management for Peer-to-Peer Systems. In: International Conference on Parallel Processing, pp. 289–296. IEEE Press, Washington (2006)
14. Stoica, I., Morris, R., Karger, D., Kaashoek, M., Balakrishnan, H.: Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications. In: Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, pp. 149–160. ACM Press, New York (2001)
15. Srivatsa, M., Xiong, L., Liu, L.: TrustGuard: Countering Vulnerabilities in Reputation Management for Decentralized Overlay Networks. In: 14th International Conference on World Wide Web, pp. 422–431. ACM Press, New York (2005)
16. Zhu, B., Setia, S., Jajodia, S.: Providing Witness Anonymity in Peer-to-Peer Systems. In: 13th ACM Conference on Computer and Communications Security, pp. 6–16. ACM Press, New York (2006)
17. Maruyama, H., Seliger, F., Nagaratnam, N.: Trusted Platform on Demand. IBM Research Report, RT0564 (2004)
18. Kinateder, M., Pearson, S.: A Privacy-Enhanced Peer-to-Peer Reputation System. In: Bauknecht, K., Tjoa, A.M., Quirchmayr, G. (eds.) EC-Web 2003. LNCS, vol. 2738, pp. 206–215. Springer, Heidelberg (2003)
19. Sandhu, R., Zhang, X.: Peer-to-Peer Access Control Architecture Using Trusted Computing Technology. In: 10th ACM Symposium on Access Control Models and Technologies, pp. 147–158. ACM Press, New York (2005)
20. Zhang, X., Chen, S., Sandhu, R.: Enhancing Data Authenticity and Integrity in P2P Systems. *IEEE Internet Computing* 9(6), 42–49 (2005)
21. Trusted Computing Group, <http://www.trustedcomputinggroup.org>
22. Barham, P., Dragovic, B., Fraser, K., Hand, S., Harris, T., Ho, A., Neugebauer, R., Pratt, I., Warfield, A.: Xen and the Art of Virtualization. In: ACM Symposium on Operating Systems Principles, pp. 164–177. ACM Press, New York (2003)
23. Goldberg, R.P.: Survey of Virtual Machine Research. *IEEE Computer* 7(6), 34–45 (1974)

A Model-Driven Approach for the Specification and Analysis of Access Control Policies*

Fabio Massacci¹ and Nicola Zannone²

¹ Department of Information and Communication Technology
University of Trento - Italy
`fabio.massacci@unitn.it`

² Department of Computer Science
University of Toronto - Canada
`zannone@cs.toronto.edu`

Abstract. The last years have seen the definition of many languages, models and standards tailored to specify and enforce access control policies, but such frameworks do not provide methodological support during the policy specification process. In particular, they do not provide facilities for the analysis of the social context where the system operates.

In this paper we propose a model-driven approach for the specification and analysis of access control policies. We build this framework on top of SI*, a modeling language tailored to capture and analyze functional and security requirements of socio-technical systems. The framework also provides formal mechanisms to assist policy writers and system administrators in the verification of access control policies and of the actual user-permission assignment.

Keywords: Security Requirements Engineering, Access Control, Policy Specification.

1 Introduction

Access Control is a critical step in securing IT systems as it aims to prevent unauthorized access to sensitive information. An access control system is typically described in three ways: access control policies, models, and mechanisms [38]. Access control policies (the focus of this paper) are sets of rules that specify what users are allowed or not allowed to do in the application domain.

The last years have seen the emergence of languages, models, and standards intended to support policy writers and system administrators in the specification and enforcement of access control policies [4,8,14,24,33,35]. Those frameworks however do not provide any methodological support to assist policy writers in capturing the organizational context where the policy will be enforced.

* This work has been partially funded by the EU-IST-IP SERENITY and SENSORIA projects, and by the Canada's NSERC Hyperion project.

The understanding of the social context plays a key role as access control policies must be consistent with the actual organization's practices [32]. Policy writers have to guarantee that the (IT mediated) access control policies in place within the system should protect the system without affecting business continuity. At the same time, they have to prevent the assignment of unnecessary authorizations (the so called *least privilege principle* [37]). Last but not least, they have to ensure that users cannot abuse their position within the organization to gain personal advantages. Thus, several questions arise during the definition of access control policies: "Why does a user need a certain access right?", "Does a user have all permissions he needs to achieve the duties assigned by the organization?", "Does a user have permissions he does not need?", "Can a user abuse his access privileges?", etc.

These issues are critical especially when dealing with sensitive personal information: many countries have issued data protection regulations establishing that the collection and processing of personal data shall be limited to the minimum necessary to achieve the stated purpose [36]. Some proposals [7,20,25,28] have partially answered these issues. For instance, Bertino et al. [7] ensure the least privilege principle by deriving access control policies from functional requirements: users are assigned with the access rights necessary to perform their duties. However, this approach leaves little room for verifying the consistency between security and functional requirements. Even though most policy languages, e.g. XACML [33], are coupled with enforcement mechanisms, very few provide frameworks and tools tailored to analyze the consistency of policies with organizational requirements and to verify the actual assignment of permissions to users.

In this paper, we present a model-driven approach that intends to assist policy writers in the specification and analysis of access control policies and system administrators in making decisions about the assignments of permissions to users. In the development of such a methodology we have taken advantages from both Requirements Engineering (RE) (e.g., [2,13]) and Trust Management (TM). (e.g., [5,27]). From RE we get the machinery to model and analyze functional requirements of IT systems and their operational environment. However, RE proposals unlikely address security aspects of organizations. In other words, they focus on what actors should do rather than what actors are authorized to do. TM is orthogonal. It addresses the authorization problem in distributed systems solely. For our purpose, we have chosen the SI* modeling language [30] that integrates concepts from TM, such as permission and its transfer (between actors), into a RE framework. In particular, this language allows the capture and modeling of functional and security aspects of socio-technical systems at the same time.

The first contribution of this paper is a methodological approach for the specifications of access control policies from organizational requirements and their analysis. The consistency of access control policies with functional and security requirements is ensured by verifying the compliance of the requirements models that have generated them with a number of properties of design.

The analysis of security incidents and frauds [3,21,34] has revealed that security breaches are often not apparent in policies specified at organizational level, that is, in terms of roles within an organization. To address this issue, we propose to capture security bugs that may be introduced by only modeling organizational requirements by means of a mechanism for instantiating requirements specified at organizational level. This also allows security and system administrators to discard system configurations that may be harmful to the system or to one of the stakeholders of the system through domain-specific constraints (e.g., separation of duties constraints, cardinality of roles, etc.).

Together with a modeling framework, we present a formal framework based on Answer Set Programming (ASP) with value invention [9] to assist policy writers in the specification and analysis of access control policies and system administrators in the user-permission assignment decision making.

In the rest of the paper we provide at first a primer of the SI* modeling language. We then present the process for the specification of access control policies (§3). We propose an approach for the analysis of access control policy at organizational level (§4) and at user level (§5). Access control policies are also analyzed with respect to specificity of the application domain (§6). Next, we propose a formal framework to assist policy writers and system administrators in their task (§7). Finally, we discuss related work (§8) and conclude with some directions for future work (§9).

2 Capturing Organizational Requirements

The SI* modeling language [30] has been proposed to capture security and functional requirements of socio-technical systems. Its main advantage is that it allows the analysis of the organizational environment where the system-to-be will operate and, consequently, it permits to capture not only the *what* and the *how*, but also the *why* security mechanisms have to be introduced in the system.

SI* employs the concepts of agent, role, goal, and task. An *agent* is an active entity with concrete manifestations and is used to model humans as well as software agents and organizations. A *role* is the abstract characterization of the behavior of an active entity within some context. They are graphically represented as circles. Assignments of agents to roles are described by the *play* relation. For the sake of simplicity, in the remainder of the paper we use the term “actor” to indicate agents and roles when it is not necessary to distinguish them.

A *goal* is a state of affairs whose realization is desired by some actor (*objective*), can be realized by some (possibly different) actor (*capability*), or should be authorized by some (possibly different) actor (*entitlement*). Entitlements, capabilities and objectives of actors are modeled through relations between an actor and a goal: *own* indicates that an actor has full authority concerning access and disposition over his entitlement; *provide* indicates that an actor has the capabilities to achieve the goal; and *request* indicates that an actor intends to achieve the goal. A *task* specifies the procedure used to achieve goals. In the

graphical representation, goals and tasks are respectively represented as ovals and hexagons. Own, provide, and request are represented with edges between an actor and a goal labeled by **O**, **P**, and **R**, respectively.

Goals and tasks of the same actor or of different actors are often related to one another in many ways. *AND/OR decomposition* combines AND and OR refinements of a root goal into subgoals, modeling a finer goal structure. Since subgoals are parts of the whole, objectives, entitlements, and capabilities are propagated from a root goal to its subgoals. *Need* relations identify the goals to be achieved in order to achieve another goal. However, neither such goals might be under the control of the actor nor the actor may have the capabilities to achieve them. Therefore, need relations propagate objectives, but not entitlements and capabilities. *Contribution* relations are used when the relation between goals is not the consequence of a deliberative planning but rather results from side-effects. Therefore, contribution relations propagate neither objectives, capabilities, nor entitlements. The impact can be positive or negative and is graphically represented as edges labeled with “+” and “-”, respectively. Finally, tasks are linked to the goals that they intend to achieve using *means-end* relations.

The relations between actors within the system are captured by the notions of delegation and trust. Assignment of responsibilities among actors can be made by *execution dependency* (when an actor depends on another actor for the achievement of a goal) or *permission delegation* (when an actor authorizes another actor to achieve the goal). Usually, an actor prefers to appoint actors that are expected to achieve assigned duties and not misuse granted permissions. SI* adopts the notions of *trust of execution* and *trust of permission* to model such expectations. In the graphical representation, permission delegations are represented with edges labeled by **Dp** and execution dependencies with edges labeled by **De**. Finally, trust of permission relations are represented with edges labeled by **Tp** and trust of execution relations with edges labeled by **Te**.

To illustrate what SI* models are, let us look at a health care scenario (Figure 1). This example is an excerpt of a case study analyzed in the EU SERENITY project¹ and we will use it through the paper to demonstrate our approach.

Example 1. Patients depend on the Health Care Centre (HCC) for receiving medical services, such as assistance because of faintness alert and home delivery of medicines. When a patient feels giddy, he can send a request for assistance to the Monitoring and Emergency Response Centre (MERC), a department of the HCC. The MERC starts a doctor discovery process that consists in sending a message to a group of doctors. The first doctor that answers the request is appointed to provide medical care to the patient. The selected doctor sets a diagnosis and defines the necessary treatments in terms of medical prescriptions or requests for specialist visits. A patient can also require the MERC to get medicines from the pharmacy. In this case, a social worker is contacted by the MERC to go to the pharmacy and get the medicine to be delivered to the patient.

¹ <http://www.serenity-project.org>

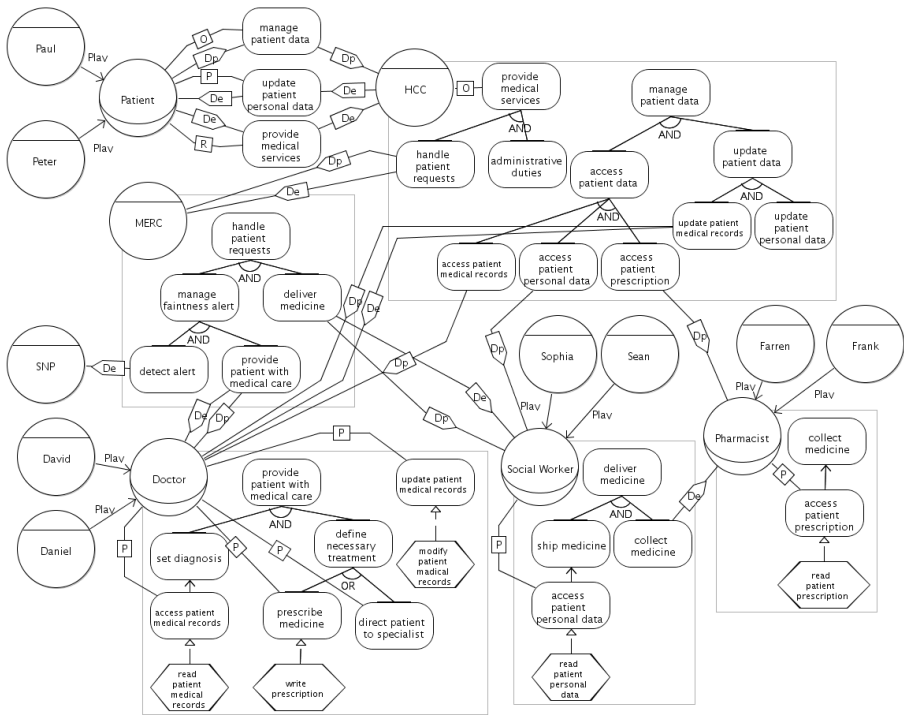


Fig. 1. A SI* model for the health care scenario

3 Access Control Policy Specification

Access control policies shall be compliant with organizational and system requirements. The key idea to the modeling of access rights to data objects is that actors have some kind of permission with respect to the activities they have to perform. Here, we propose a methodological framework for supporting policy writers in the specification of access control policies from the functional and security requirements of the system. The framework intends to analyze the organizational context in terms of the actors who comprise it, their goals (i.e., entitlements, objectives, and capabilities), and the interrelations among them. Goals are then operationalized into specifications of operations to achieve them. The access control policy is defined as the set of permission associated with operations. In the remainder of this section, the phases of the access control specification process (Figure 2) are presented in detail.

Actor modeling (step 1) aims to identify and model the roles (e.g., Doctor, Social Worker, etc.) and agents (e.g., HCC, MERC, David, etc.) within the socio-technical system. Identified actors are described along with their objectives, entitlements, and capabilities. In this phase, agents are also described along the roles they play (e.g., David *plays* role Doctor). As the analysis proceeds (steps 3,

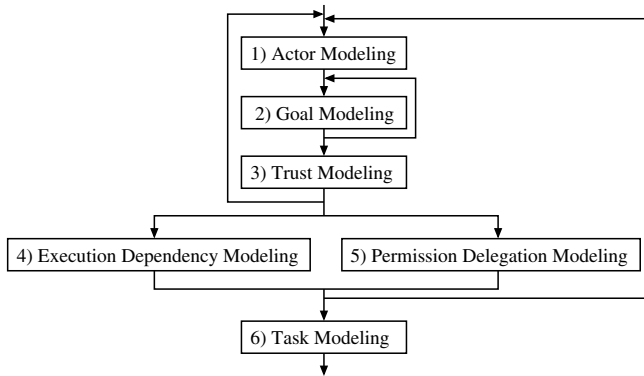


Fig. 2. Access Control Policy Specification Process with SI*

4, and 5), more actors might be identified, leading to new iterations of the actor modeling phase.

Goal modeling (step 2) aims to analyze objectives, entitlements and capabilities of actors. These are analyzed from the perspective of their respective actor using the various forms of goal analysis described earlier. Initially, goal modeling is conducted for all root goals associated with actors. Later on, more goals are created as goals are delegated to existing or new actors (steps 4 and 5). The refinement of goals are considered to reach an adequate level once objectives of every actor have been assigned to actors that are capable to achieve them and permission are specified at an appropriate level of granularity.

The assignment of responsibilities is driven by the expected behavior of other actors. *Trust modeling* (step 3) enriches the requirements model by identifying trust relationships (of both permission and execution) between actors. *Execution dependency modeling* (step 4) aims to discover and establish dependencies between actors (e.g., the *Patient depends* on the HCC for achieving goal *provide medical services*). Entitlements are also analyzed from the perspective of each actor. *Permission delegation modeling* (step 5) aims to identify and model transfers of authority between actors (e.g., the *Patient delegates* the *permission* to *manage patient data* to the HCC).

Once all objectives have been dealt with to the satisfaction² of the actors who want them, *task modeling* (step 6) identifies the actions to be executed in order to achieve goals. For the sake of simplicity, in this work we assume that tasks are atomic actions (e.g., *read*, *write*, *modify*, etc.) and cannot be further refined. The identified tasks are linked to goals via means-end relations that propagate properties of goals (i.e., objectives, entitlements, and capabilities) to tasks. The access control policy is defined as the set of permissions associated with tasks.

Example 2. Figure 3 shows the access control policy derived from the health care scenario of Figure 1. The Doctor is appointed by the HCC to provide patient

² An actor *A* can satisfy a goal *G* if *G* is an objectives of *A* and *A* has the capabilities to achieve *G* or *A* depends on an actor who can satisfy *G* [18].

(Doctor,read patient medical data)
(Doctor,modify patient medical data)
(Doctor,write prescription)
(Social Worker,read patient personal data)
(Pharmacist,read patient prescription)

Fig. 3. Access Control Policy

with medical care and update patient medical records. The requirements analysis process shows that the Doctor shall be authorized to read and modify patient medical records as well as write prescription to achieve assigned duties. Similarly, the Social Worker shall be authorized to read patient personal data for shipping medicine and the Pharmacist shall be authorized to read patient prescription for collecting medicine. It is worth noting that the Social Worker has not the permission to read the prescriptions; only the Pharmacist is authorized to do it. As consequences, the system designer needs to employ some mechanism to protect prescriptions, for instance, by enclosing them into closed and sealed envelopes that only the pharmacist is authorized to open³ (if prescriptions are in paper form) or by encrypting them (if prescriptions are in digital form).

In the above example we showed a RBAC policy as we have only considered the access rights to be associated with roles. The framework is, however, flexible enough to specify policies in other access control models. This might be useful, for instance, when access rights are specified with respect to agents instead of to roles.

4 Policy Verification at Organizational Level

The most frequent question during modeling is whether the policy is consistent with functional requirements and compliant with security requirements. The first issue we tackle concerns the analysis of functional requirements from the perspective of actors who want goals achieved (hereafter requesters). We verify whether actors' objectives are satisfied by the system design.

Pro1. Every requester has assigned (possibly indirectly) the achievement of his objectives to actors that have the capabilities and the necessary access right to achieve them.

Pro2. Every requester has assigned (possibly indirectly) the achievement of his objectives to actors that he trusts.

The first property verifies that actors' objectives are assigned to actors that can actually take charge of their satisfaction. The latter aims to provide additional guarantee about the satisfaction of goals by employing the notion of trust. The satisfaction of these properties ensures requirements engineers that the system design leads to the satisfaction of the objectives of each actor.

³ The pharmacist can report the occurrence of a misuse to the HCC if he receives an envelope without the seal or with a broken seal from the social worker.

From the perspective of actors who control the achievement of goals (hereafter owners), the system design should guarantee that entitlements of actors are not misused. To this purpose, we employ the following properties:

Pro3. Entitlements have been assigned only to actors trusted by their owners.

Pro4. Actors granting permissions to achieve a goal, have the right to do so.

Pro5. Entitlements have been assigned to actors who actually need them to perform their duties.

Pro3 provides owners with assurance that their entitlements are used properly. The system design has also to ensure that actors do not grant privileges they do not have to other actors. This is verified by Pro4. Pro5 verifies the compliance of the model with the least privilege principle.

Finally, designers have to analyze their model from the perspective of actors that are actually in charge of achieving goals (hereafter providers).

Pro6. Every provider has the permissions necessary to accomplish assigned duties.

A failure of Pro6 can be due to the lack of assignments of permission to legitimate users. In this case, designers should revise the model by identifying the permission path that at the end will authorize the provider to achieve the goal. The failure, however, can be also due to the fact that the achievement of objectives has been assigned to actors that should not be authorized to accomplish such tasks. This requires designers to revise the model by identifying other actors that has the capabilities to achieve the goals. It is worth noting that only by modeling security and functional requirements separately one is able to capture both these aspects. Indeed, by deriving access control policies from the functional requirements one always ends up that access rights have not been assigned to users.

5 Policy Verification at User Level

The analysis of industry case studies (e.g., [31,40]) has revealed that security breaches are often not apparent in policies specified at organizational level. They only appear at the instance level, once we see what happens when individuals are mapped into roles and delegation paths are concretely followed. Requirements engineers, however, do not usually want to design the system at the instance level even if they need to reason at that level to detect many security breaches. Our objectives is thus to offer them tools for instantiating organizational requirements and analyzing the instantiated requirements.

The SI* modeling framework allows for a clear distinction between organizational and instance levels as it only employs the notions of agent and role. The organizational level focuses on roles by associating with each role the objectives, entitlements, and capabilities related to the activities that such a role has to perform within the organization and the relationships among them. The instance level focuses on single agents by identifying their personal objectives,

entitlements, and capabilities and the relationships among them as well as the roles they play.

Many access control frameworks determine the permissions assigned to users (possibly via roles assignment) by adopting an inheritance method, that is, a user inherits all privileges associated with the roles he plays. This approach, however, requires access rights to be specified on concrete objects, making difficult policy specification, analysis, and management. In this section, we propose an instantiation procedure that automatically relates goal instances to users with respect to the responsibilities and permissions that have been assigned to them. The basic idea is that when a designer draws a goal, he specifies an “abstract goal”, rather than a “goal instance”. Then, it is matter of the instantiation procedure to automatically generate the instantiated requirements model.

Different SI* concepts instantiate goals and social relations differently. Though it might seem that the instantiation of objectives depends on the responsibilities of agents, an agent will unlikely desire the fulfillment of all instances of a goal. Rather, he is interested in the achievement of a particular instance. For example, a patient (e.g., Peter in Figure 2) cares about medical services provided to him rather than to services provided to other patients. There might be situations where agents want to satisfy more than one instance of the same goal. This is, for instance, the case of the HCC that is in charge of providing medical treatments to those agents, who requested them. However, they are delegated responsibilities. Conversely, an agent to whom capabilities are prescribed (possibly via role assignment), has the capabilities to achieve all instances of a goal. For instance, a doctor (e.g., David) is capable of prescribing medicines to all patients rather than only to one particular patient.

The instantiation of entitlements is on case-by-case basis. The designer may assign permission to a role with the intended meaning that the agents that play that role are entitled to control only a particular instance of the goal. On the contrary, there are situations where an actor is entitled to control all instances of the goal. The difference in the meaning of entitlement is evident by looking at the relationship between the patient and his data and between the HCC and medical services (Figure 2). When a designer says that “a patient is entitled to control the use of patient’s medical records”, he means that every patient is entitled to control only his own medical records. Conversely, the HCC has full authority over all instances of the provisioning of medical services. To distinguish these situations, we have refined the concept of ownership into existential ownership and universal ownership. *Existential ownership* indicates that the actor is the legitimate owner of one instance of the goal. *Universal ownership* indicates that the actor is the legitimate owner of all instances of the goal.

Execution dependencies (permission delegations, resp.) propagate the responsibility (authority, resp.) to achieve the goal instances generated by objective (existential ownership, resp.) instantiation to the agents playing the role of dependee (delegatee, resp.). However, only one instance of the dependee is appointed to perform the assigned duties. This intuition is actually closer to reality than one may think: when the MERC appoints a doctor to provide medical care to a

patient, only one doctor will perform this task. Similarly, permission delegations grant permission only to one of the agents playing the role of delegatee. Trust relations are used to model the expectation of an actor. This expectation is not related to a particular goal instance but refers to the general behavior of the trustee, that is, to all instances of the goal. Accordingly, trust relations are instantiated for all instances of the goal. Moreover, trust relations are instantiated between every agent playing the role of trustor and every agent playing the role of trustee.

This instantiation model can also assist system and security administrators in the configuration decision making process. For instance, the simple SI* model in Figure 2 generates a huge number of possible configurations (i.e., combinations of assignments of objectives and entitlements).⁴ However, not all of them may guarantee a fair and lawful behavior of the socio-technical system. For instance, in many configurations access rights are assigned to agents that are not actually in charge to achieve the goals. The key idea is to apply the properties presented in Section 4 to the instantiated model in order to discard those configurations that are not compliant with security and organizational requirements.

6 Domain-Specific Verification

The analysis at the instance level allows us to capture other situations that might result harmful to the system or one of the stakeholders of the system.

Example 3. The first doctor who answer Peter's request is David. Thereby, David is appointed by the MERC to provide medical care to Peter. David is also a consultant in the health insurance company with whom Peter has stipulated an insurance policy. This situation is clearly to be avoided. The MERC needs to find another doctor to provide medical care to Peter. This demands a revision of the doctor discovery procedure: the first-answer first-appointed policy (see Example 1) should be modified by introducing a check for possible conflicts.

If we look at ways to handle situations like the above example, starting from the landmark paper by Saltzer and Schroeder [37] to other classical papers [1,15,16,19,41], we found that Separation of Duty (SoD) is invariably offered as “the” solution to prevent the violation of business rules. SoD aims to reduce the risk of security breaches by not allowing any individual to have sufficient authority within the system to compromise it on his own [7]. Our framework supports the specification of SoD constraints at three levels of granularity. The basic type of constraint simply denies agents to play conflicting roles.

Example 4. A doctor in the HCC shall not work as a consultant in an insurance company.

A second type focuses on incompatible activities. They prevent users from performing activities whose combination can compromise the system integrity.

⁴ The number of configuration is exponential in the number of OR-decompositions, permission delegations, execution dependencies, and agent-role assignments.

Example 5. A doctor shall not provide both health care in behalf of the HCC and consulting to the insurance company. This constraint, however, does not deny doctors to perform other duties within the HCC and the insurance company.

Above types of constraint can be classified as static SoD constraints [41]. In some cases they impose too strict limits on requirements. To this end, the framework allows for the specification of dynamic SoD constraints [41] by focusing on particular instances of activities.

Example 6. David shall not provide assistance to patients who have stipulated an insurance policy with the insurance company were he works. This constraint, however, does not deny David to provide medical care to patients who do not have any relation with the insurance company.

Other constraints imposed by the application domain can be defined, for instance, to specify the cardinality of roles, that is, the number of agents that can play a role at the same time, or the number of tasks assigned to single agents.

7 Automated Reasoning Support

Looking at the process in Figure 2, it is evident the need of tools to assist policy writers in determining (1) which actors' goals are satisfied and (2) the permission on tasks. These activities can be cumbersome to be manually performed especially when the requirements model is huge. Tool support is also necessary for model instantiation and policy verification.

For our purpose, we have chosen the ASP paradigm with value invention [9]. In [18] the authors have defined the semantics of SI* in the ASP paradigm [26]. Roughly speaking, ASP is a variant of Datalog with negation as failure and disjunction. This paradigm supports specifications expressed in terms of facts and Horn clauses, which are evaluated using the stable model semantics. Here, graphical models are encoded as sets of facts (see [29] for details on the transformation of graphical models into formal specifications). Rules (or axioms) are Horn clauses that define the semantics of SI* concepts. Specifically, axioms are used to propagate objectives, entitlements, and capabilities across the requirements model via goal analysis, execution dependencies, and permission delegations. As an example, we report the axioms for entitlements and permission delegations (Ax1-3) in Table 1 [18]. As described earlier in the paper, the access control policy is defined as the set of permission associated to tasks. Ax4 is used to determine such permission. Axioms are also used to determine the goals that can be satisfied and to propagate satisfaction evidence backward to the requester [18].

Properties of design (Section 4) and domain-specific constraints (Section 6) are encoded as ASP constraints. Constraints are Horn clauses without positive literals and are used to specify conditions which must not be true in the model. In other words, constraints are formulations of possible inconsistencies. Table 2 presents some examples of constraints. Pro3 verifies if an owner is confident that there is no likely misuse of his entitlements. Specifically, an owner is confident that permissions on his entitlements have been assigned only to trusted actors.

Table 1. Axiomatization of Entitlements and Permission Delegations

Ax1	$\text{have\_perm}(X, G) \leftarrow \text{own}(X, G)$
Ax2	$\text{have\_perm}(X, G) \leftarrow \text{delegate}(Y, X, G) \wedge \text{have\_perm}(Y, G)$
Ax3	$\text{have\_perm}(X, G_1) \leftarrow \text{subgoal}(G_1, G) \wedge \text{have\_perm}(X, G)$
Ax4	$\text{access\_control}(X, T) \leftarrow \text{means\_end}(T, G) \wedge \text{have\_perm}(X, G)$

Table 2. Properties of Design and Domain-Specific Constraints

Pro3	$\leftarrow \text{own}(X, G) \wedge \text{not confident\_owner}(X, G)$
Pro4	$\leftarrow \text{delegate}(X, Y, G) \wedge \text{not have\_perm}(X, G)$
Pro5	$\leftarrow \text{have\_perm}(X, G) \wedge \text{not need\_to\_have\_perm}(X, G)$
Pro6	$\leftarrow \text{need\_to\_have\_perm}(X, G) \wedge \text{not have\_perm}(X, G)$
SoD	$\leftarrow \text{play}(A, r_1) \wedge \text{play}(A, r_2)$
RC	$\leftarrow \#count\{X : \text{play}(A, r)\} > n$

Here, literal $\text{confident\_owner}(x, g)$ holds if actor x is confident that permissions on goal g are given only to trusted actors. Pro4 verifies that actors, who delegate the permission to achieve a goal, are entitled to do it, that is, it checks that the permission are well rooted. Pro5 verifies that actors, who have the permission to achieve a goal, actually need such permission. Pro6 is opposite to Pro5. It verifies that actors, who need to have the permission to achieve their duties, have such permission. Thereby, the combination of Pro5 and Pro6 guarantees that actors have access right if and only if they need them. SoD verifies that there are no agents that play both roles r_1 and r_2 . RC verifies that there are not more than n agents that play role r .⁵

Facts, axioms and constraints compound the program that is executed by an ASP inference engine. As result, the engine returns all answer sets (i.e., sets of atoms) satisfying all Horn clauses. These answer sets represent the system configurations in which all properties of design are satisfied. Answer sets include the access control policy, that is, the sets of facts in the form $\text{access\_control}(x, t)$.

The ASP paradigm, however, is not sufficient for implementing the instantiation procedure presented in Section 5. ASP with value invention improves ASP by introducing function symbols. Essentially, functions are treated as external predicates that implement the mechanism of value invention by taking in input a set of values and returning a new value. Accordingly, instances of goals are represented using function $g_i(g, a, r)$, where g is a goal, a is the agent who has generated the instance, and r is the role from which the agent has taken the goal.⁶ The choice of implementing the instantiation procedure in ASP with value invention instead of in other formalisms allows us to reuse the framework proposed in [18] (with some minor changes). Actually, the rules for instantiation are simply added to the ASP program used for the analysis of organizational

⁵ $\#count\{\dots\}$ is a built-in aggregate function that is supported by several ASP solvers.

⁶ We use the constant `null` when the goal is directly associated to an agent.

Table 3. Instantiation of Entitlements and Permission Delegations

I1	$\text{own}_i(A, g_i(G, A, R)) \leftarrow \text{own\_existential}(R, G) \wedge \text{play}(A, R)$
I2	$\text{own}_i(A, g_i(G, B, R)) \leftarrow \begin{cases} \text{own\_universal}(P, G) \wedge \text{play}(A, P) \wedge \\ \text{agent}(B) \wedge \text{role}(R) \wedge \text{goal}(G) \end{cases}$
I3	$\text{delegate}_i(A, B, g_i(G, C, R)) \leftarrow \begin{cases} \text{delegate}(P, Q, G) \wedge \text{play}(A, P) \wedge \text{play}(B, Q) \wedge \\ \text{have\_perm}(A, g_i(G, C, R)) \wedge \\ \text{not other\_agent}(B, Q, g_i(G, C, R)) \end{cases}$
I4	$\text{other\_agent}(A, R, G) \leftarrow \text{play}(A, R) \wedge \text{play}(B, R) \wedge \text{delegate}_i(D, B, G) \wedge A \neq B$

requirements. Table 3 presents the rules for instantiating entitlements and permission delegations.⁷

One can observe the different instantiation for existential and universal ownership. Specifically, the rule for existential ownership (I1) introduces new values, that is, it creates new instances of the goal. On the other hand, the rule for universal ownership (I2) considers all the instances of the goal. Rule I3 implements the instantiation of permission delegations. We introduce predicate `other_agent` to verify if the permission has already been assigned to another agent. Thus, axiom I3 (in combination with I4) will not yield one model but multiple models in which the permission is granted only to one agent playing the role of the delegatee. Another observation concerns the goal instance: an actor can delegate only the permission on the instances that are in his scope, that is, instances which the actor is already entitled to achieve. The proposed approach has been implemented in the DLV system [26] – a state-of-the art implementation of ASP.

8 Related Work

Several languages and models intended to support policy writers and system administrators in the specification and enforcement of access control policies has been proposed [6,23,33,39]. Our work is complementary to those proposals. Indeed, we have not proposed a new access control language. Rather, our objective is to support policy writers in defining access control policies, which can be specified using existing languages.

Several efforts have been spent to close the gap between security requirements analysis and policy specification. Basin et al. [4] propose SecureUML, an UML-based modeling language for modeling access control policies and integrating them into a model-driven software development process. Similar approaches have been proposed by Doan et al. [14], who incorporate Mandatory Access Control (MAC) into UML, and by Ray et al. [35], who model RBAC as a pattern using UML diagram template. Breu et al. [8] propose an approach for the specification of user rights in the context of an object oriented use case driven development process. However, these frameworks do not provide facilities for the analysis of the social context where the system operates.

⁷ For the sake of simplicity, Table 3 reports the rules used when `own` and `delegation` are specified for roles. Similar rules are used when relations are specified for agents.

The problem of specifying access control policies has been partially addressed in workflow management systems. For instance, Bertino et al. [7] formally express constraints on the assignment of roles to tasks in a workflow in order to automatically assign users to roles according to such constraints. Kang et al. [25] propose a fine-grained and context-based access control mechanism for inter-organizational workflows. The idea underlying these proposals is to grant access rights to users on the basis of the duties assigned to the roles they play. This approach, however, does not allow the analysis of the functional requirements of the system to be protected.

Moving towards early requirements, He et al. [20] propose a goal-driven framework for modeling RBAC policies based on role engineering [10]. This framework includes a context-based data model, in which policy elements are represented as attributes of roles, permissions, and objects, and a goal-driven role engineering process, which addresses how the security contexts in the data model can be elicited and modeled. However, roles are derived from task and are not analyzed with respect to the organizational context. Liu et al. [28] propose an access control analysis in *i**. The main difference with our approach lies in the degree of automation. Liu et al. provide a systematic way to specify access control policies, but leave all work to humans. Indeed, they do not provide any tool support for assisting policy writers in their work. Moreover, similarly to workflow access control proposals, the authors propose to grant a permission to an actor every time he needs such a permission. Crook et al. [11] enhance *i** to derive role definition from the organizational context. However, instantiation is still manual. Moreover, as in [7,25] permissions are simply derived by the tasks assigned to users, leaving a little room for verifying the consistency between security and functional requirements. Finally, we mention the work by Fontaine [17], who proposes a mapping of goal models based on KAOS [13], a goal-based requirements engineering methodology, onto Ponder [12], a language for specifying management and security policies for distributed systems. The key point of this work is the transformation of operationalized goals into access control policies. However, KAOS is inadequate to model and analyze policies because it lacks the necessary features necessary for the modeling and analysis of the organization structure.

In the area of policy verification, Sohr et al. [42] propose a framework for the verification and validation of RBAC policies and authorization constraints. Policies and constraints are specified as sentences in first-order LTL and verified using theorem provers. Although the use of theorem provers allows analysts to give proofs that are independent from the number of users and objects, it makes the verification process not completely automated. The authors also propose a validation approach based on UML and OCL: RBAC policies are modeled as class diagrams and authorization constraints are specified in OCL. The UML-based Specification Framework is then used to generate system states and to check those states against specified constraints. Hu et al. [22] propose a framework for verification and conformance testing for secure system development. Verification is intended to ensure that access control policies comply with

security properties, and conformance testing is used to validate the compliance of system implementation with access control policies. However, these proposals mainly focus on the implementation and enforcement of access control policies and do not provide any methodological support for the analysis of the organizational context and the definition of access control policies on the basis of elicited organizational and system requirements.

9 Conclusive Remarks and Future Work

In this paper we have proposed a model driven approach to assist policy writers in the specification and analysis of access control policies with respect to organization and security requirements and system administrators in the user-permission assignment decision making. To support a more accurate analysis, we have defined an approach for instantiating organizational requirements. Readers familiar with RBAC and other access control models will easily find out some differences in the way permissions are assigned to agents. In RBAC a user inherits all permissions associated with the roles he plays. If permission is specified for classes of objects, the user is entitled to access all objects in those classes [23]. We observe that this assumption is not always true especially with regards to the least privilege principle. For example, just because the doctor role can access a patient record does not mean that a doctor can access all patient records. A doctor can only access the records of those patients currently assigned to that doctor.

The instantiation procedure together with policy analysis facilities have been implemented in ASP with value invention. One may claim that the approach suffers from exponential complexity. We argue that this is *the cost of security*: policy writers shall explore the entire space of solutions to identify vulnerabilities in their access control policies. Scalability problems, however, occur at design time where the policy writer can add and remove agents for a more accurate analysis. They disappear at run time when administrators verify whether or not the actual system configuration is secure. Indeed, the problem of verifying if a certain configuration satisfies properties of design and domain-specific constraints is polynomial. The last observation concerns the verification of Pro4 against the instantiated requirements. Rule I3 instantiates permission delegations only for the instances in the scope of the agent. This approach makes every permission well rooted by construction. The verification of Pro4 at the instance level, however, can be done by creating a “fake” instance of the goal, for instance, when the agent is supposed to delegate the permission on a goal but he has not it on any instance of that goal.

The research presented here is still in progress. Much remains to be done to further refine the proposed approach to support the specification of access control policies comparable to the ones that can be expressed, for instance, in XACML [33]. Future work plans include the support for the specification of negative authorizations and obligations. Another direction under investigation involves the capture of behavioral aspects by means of revocation policies.

References

1. Ahn, G.-J., Sandhu, R.: The RSL99 language for role-based separation of duty constraints. In: Proc. of RBAC 1999, pp. 43–54. ACM Press, New York (1999)
2. Antón, A.I., Potts, C.: The use of goals to surface requirements for evolving systems. In: Proc. of ICSE 1998, pp. 157–166. IEEE Press, Los Alamitos (1998)
3. Association of Certified Fraud Examiners. The 2006 report to the nation (2006)
4. Basin, D., Doser, J., Lodderstedt, T.: Model Driven Security: from UML Models to Access Control Infrastructures. TOSEM 15(1), 39–91 (2006)
5. Becker, M.Y., Sewell, P.: Cassandra: flexible trust management, applied to electronic health records. In: Proc. of CSFW 2004, pp. 139–154. IEEE Press, Los Alamitos (2004)
6. Bell, D.E., LaPadula, L.J.: Secure Computer System: Unified Exposition and MULTICS Interpretation. Technical Report MTR-2997 Rev. 1, The MITRE Corporation, Bedford, MA (1976)
7. Bertino, E., Ferrari, E., Atluri, V.: The specification and enforcement of authorization constraints in workflow management systems. TISSEC 2(1), 65–104 (1999)
8. Breu, R., Popp, G., Alam, M.: Model based development of access policies. STTT 9, 457–470 (2007)
9. Calimeri, F., Ianni, G.: External Sources of Computation for Answer Set Solvers. In: Baral, C., Greco, G., Leone, N., Terracina, G. (eds.) LPNMR 2005. LNCS (LNAI), vol. 3662, pp. 105–118. Springer, Heidelberg (2005)
10. Coyne, E.J.: Role engineering. In: Proc. of RBAC 1995, pp. 15–16. ACM Press, New York (1995)
11. Crook, R., Ince, D., Nuseibeh, B.: On Modelling Access Policies: Relating Roles to their Organisational Context. In: Proc. of RE 2005, pp. 157–166 (2005)
12. Damianou, N., Dulay, N., Lupu, E., Sloman, M.: The Ponder Policy Specification Language. In: Sloman, M., Lobo, J., Lupu, E.C. (eds.) POLICY 2001. LNCS, vol. 1995, pp. 18–39. Springer, Heidelberg (2001)
13. Dardenne, A., van Lamsweerde, A., Fickas, S.: Goal-directed Requirements Acquisition. Sci. of Comp. Prog. 20, 3–50 (1993)
14. Doan, T., Demurjian, S., Ting, T.C., Ketterl, A.: MAC and UML for secure software design. In: Proc. of FMSE 2004, pp. 75–85. ACM Press, New York (2004)
15. Dobson, J.E., McDermid, J.A.: A framework for expressing models of security policy. In: Proc. of Symp. on Sec. and Privacy, pp. 229–239. IEEE Press, Los Alamitos (1989)
16. Ferraiolo, D.F., Barkley, J.F., Kuhn, D.R.: A role-based access control model and reference implementation within a corporate intranet. TISSEC 2(1), 34–64 (1999)
17. Fontaine, P.-J.: Goal-Oriented Elaboration of Security Requirements. Ph.D thesis, Université Catholique de Louvain (2001)
18. Giorgini, P., Massacci, F., Zannone, N.: Security and Trust Requirements Engineering. In: Aldini, A., Gorrieri, R., Martinelli, F. (eds.) FOSAD 2005. LNCS, vol. 3655, pp. 237–272. Springer, Heidelberg (2005)
19. Gligor, V.D., Gavrila, S.I., Ferraiolo, D.: On the formal definition of separation-of-duty policies and their composition. In: Proc. of Symp. on Sec. and Privacy, pp. 172–183. IEEE Press, Los Alamitos (1998)
20. He, Q., Antón, A.I.: A Framework for Modeling Privacy Requirements in Role Engineering. In: Proc. of REFSQ 2003, pp. 137–146 (2003)
21. House of Lords. Prince Jefri Bolkiah vs KPMG. 1 All ER 517 (1999)

22. Hu, H., Ahn, G.: Enabling verification and conformance testing for access control model. In: Proc. of SACMAT 2008, pp. 195–204. ACM Press, New York (2008)
23. Jajodia, S., Samarati, P., Sapino, M.L., Subrahmanian, V.S.: Flexible support for multiple access control policies. *TODS* 26(2), 214–260 (2001)
24. Jürjens, J.: *Secure Systems Development with UML*. Springer, Heidelberg (2004)
25. Kang, M.H., Park, J.S., Froscher, J.N.: Access control mechanisms for inter-organizational workflow. In: Proc. of SACMAT 2001, pp. 66–74. ACM Press, New York (2001)
26. Leone, N., Pfeifer, G., Faber, W., Eiter, T., Gottlob, G., Perri, S., Scarcello, F.: The DLV System for Knowledge Representation and Reasoning. *TOCL* 7(3), 499–562 (2006)
27. Li, N., Mitchell, J.C.: RT: A Role-based Trust-management Framework. In: Proc. of DISCEX 2003, vol. 1, pp. 201–212. IEEE Press, Los Alamitos (2003)
28. Liu, L., Yu, E.S.K., Mylopoulos, J.: Security and Privacy Requirements Analysis within a Social Setting. In: Proc. of RE 2003, pp. 151–161. IEEE Press, Los Alamitos (2003)
29. Massacci, F., Mylopoulos, J., Zannone, N.: Computer-Aided Support for Secure Tropos. *ASE* 14(3), 341–364 (2007)
30. Massacci, F., Mylopoulos, J., Zannone, N.: An Ontology for Secure Socio-Technical Systems. In: *Handbook of Ontologies for Business Interaction*, ch. XI, p. 188. The IDEA Group (2008)
31. Massacci, F., Zannone, N.: Detecting Conflicts between Functional and Security Requirements with Secure Tropos: John Rusnak and the Allied Irish Bank. In: *Social Modeling for Requirements Engineering*. MIT Press, Cambridge (to appear, 2008)
32. Mellado, D., Fernández-Medina, E., Piattini, M.: Applying a Security Requirements Engineering Process. In: Gollmann, D., Meier, J., Sabelfeld, A. (eds.) *ES-ORICS 2006*. LNCS, vol. 4189, pp. 192–206. Springer, Heidelberg (2006)
33. OASIS. eXtensible Access Control Markup Language (XACML) Version 2.0. OASIS Standard (2005)
34. Promontory Financial Group, Wachtell, Lipton, Rosen, and Katz. Report to the Board and Directors of Allied Irish Bank P.L.C., Allfirst Financial Inc., and Allfirst Bank Concerning Currency Trading Losses (March 12, 2003)
35. Ray, I., Li, N., France, R., Kim, D.-K.: Using UML to visualize role-based access control constraints. In: Proc. of SACMAT 2004, pp. 115–124. ACM Press, New York (2004)
36. Room, S.: *Data Protection & Compliance in Context*. BCS (2007)
37. Saltzer, J.H., Schroeder, M.D.: The Protection of Information in Computer Systems. *Proceedings of the IEEE* 63(9), 1278–1308 (1975)
38. Samarati, P., di Vimercati, S.D.C.: Access Control: Policies, Models, and Mechanisms. In: Focardi, R., Gorrieri, R. (eds.) *FOSAD 2001*. LNCS, vol. 2946, pp. 137–196. Springer, Heidelberg (2004)
39. Sandhu, R.S., Coyne, E.J., Feinstein, H.L., Youman, C.E.: Role-Based Access Control Models. *IEEE Comp.* 29(2), 38–47 (1996)
40. Schaad, A., Lotz, V., Sohr, K.: A model-checking approach to analysing organisational controls in a loan origination process. In: Proc. of SACMAT 2006, pp. 139–149. ACM Press, New York (2006)
41. Simon, R., Zurko, M.E.: Separation of duty in role-based environments. In: Proc. of CSFW 1997, pp. 183–194. IEEE Press, Los Alamitos (1997)
42. Sohr, K., Drouineaud, M., Ahn, G.-J., Gogolla, M.: Analyzing and managing role-based access control policies. *TKDE* 20(7), 924–939 (2008)

PuRBAC: Purpose-Aware Role-Based Access Control

Amirreza Masoumzadeh and James B.D. Joshi

School of Information Sciences, University of Pittsburgh
{amirreza,jjoshi}@sis.pitt.edu

Abstract. Several researches in recent years have pointed out that for the proper enforcement of privacy policies within enterprise data handling practices the privacy requirements should be captured in access control systems. In this paper, we extend the role-based access control (RBAC) model to capture privacy requirements of an organization. The proposed purpose-aware RBAC extension treats purpose as a central entity in RBAC. The model assigns permissions to roles based on purpose related to privacy policies. Furthermore, the use of purpose as a separate entity reduces the complexity of policy administration by avoiding complex rules and applying entity assignments, coherent with the idea followed by RBAC. Our model also supports conditions (constraints and obligations) with clear semantics for enforcement, and leverages hybrid hierarchies for roles and purposes for enforcing fine grained purpose and role based access control to ensure privacy protection.

1 Introduction

Privacy can be defined as the right of individuals and organizations to control the collection, storage, and dissemination of information about themselves. Nowadays companies and enterprises gather more and more data about their users in order to provide more competitive services. This is especially true about applications on the Web that monitor the behavior of their users, resulting in heightened concern about potential disclosure and misuse of private information. Fortunately, the trend is such that organizations and enterprises are also becoming more serious about respecting the privacy of their customers. They not only are required to comply with existing privacy regulations, but also can take advantage of their privacy practices as an important capital to increase (or at least retain) their market share.

As described in a recently proposed road-map for web privacy by Antón et al. [1], there still remain vital research problems to be addressed. One major challenge is actual enforcement of privacy policies once the data has been collected. A big step towards enforcing privacy policies in an organization is considering them when making decisions over access to private data in information systems. With that vision, Powers et al. suggest privacy policy rules [2], comprising of *data type*, *operation* on the data, *data user*, *purpose* of data access, *condition* that restricts the accesses, and *obligations* that need to be carried out by the organization after the user is allowed to access.

The well-known role-based access control model (RBAC) is a typical choice for organizational access control [3]. Therefore, enabling privacy policy specification within this model can be quite useful. Recently, Ni et al. propose P-RBAC [4], a privacy-aware role-based access control model, which incorporates notions of privacy policies defined in [2] into RBAC model. P-RBAC encapsulate data, action, purpose, condition, and obligation as privacy data permission. Although quite powerful, we argue that P-RBAC is moving away from the spirit of RBAC, that is simplicity of policy administration. With the use of roles as the intermediary entities between users and permissions, RBAC shifts from simply-represented but hard-to-manage paradigm that only consists of authorization rules, to a more manageable user-role and permission-role assignment scheme. However, privacy data permissions in P-RBAC do not consider that characteristic, and in presence of data and purpose hierarchies, the policy administration is as complex as authorization rule approaches such as [5].

We propose a slightly different extension to RBAC, called purpose-aware role-based access control (PuRBAC) model. In this model, we consider *purpose* as an intermediary entity between role and permission entities. The proposed PuRBAC model also includes constraints and obligations defined as conditions on assignment of permissions to purposes. We summarize the reasons for considering purpose as a separate entity in our model as follows:

- The core part of privacy policies usually state purposes for and circumstances under which collected data would be used, and the extent of use of personal information may differ based on the purpose of use. For example, health record of a job applicant may be used for the purpose of approval of qualification for a special job requiring certain degree of health, without disclosing details. However, the details may be used for the purpose of treatment of that person as a patient.
- There is a close relation between the notion of purpose in privacy policies and the notion of role in RBAC. A role in RBAC can be defined as a set of actions and responsibilities associated with a particular activity [6]. Purpose in privacy policies is defined as a reason for data collection and use [7], and business purposes can be identified through task definition within an organization's IT systems and applications [2]. Here, a close relation can be observed between what responsibility a user has (can be modeled as role) and its associated tasks for fulfilling the responsibilities (can be modeled as purpose). On the other hand, fulfilling the tasks requires access to data, which is represented by permissions in RBAC. Thus, considering purpose as intermediate entity between role and permission entities is intuitive.
- The other rational is to follow RBAC trend of breaking policy definition into different entities and relations between them (e.g. assignment relation between role and permission in RBAC), making management of different part of the policy as independent as possible. Unlike P-RBAC [4], treating purpose as a separate entity between role and permission makes our model coherent with that approach.

There is also a difference between our model and traditional access control models in that a subject should specifically assert the purpose of accessing data in its request to access a piece of data. In terms of RBAC, the access request can be treated as as a tuple containing a session identifier, purpose of access, and permission requested. Authorizing such a request ensures that a private information is accessed only for the valid purposes according to the privacy policy, without any ambiguities (in compliance with the use limitation principle as mentioned in [8]). Note that in practice the purpose assertion may be provided without user intervention by the application.

The rest of the paper is organized as follows. In Section 2, we introduce PuRBAC model; a base model is first defined formally (Section 2.1), followed by its hierarchy-extended (Section 2.2) and hybrid hierarchy-extended (Section 2.3) versions. In Section 3, our condition model is described separately from our access control model to simplify presentation. In Section 4, we discuss the strength of our approach and justify the our modeling. We review the previous major research in Section 5, and conclude the paper with a discussion of future directions in Section 6.

2 Purpose-Aware RBAC Model (PuRBAC)

We define a hierarchy of Purpose-aware RBAC (PuRBAC) models to clearly distinguish different features, following classic scheme of RBAC models [3]. Figure 1 shows the PuRBAC family of models. Figure 1.a illustrates the relationship between different models, and Figure 1.b shows different components and relations in the models. PuRBAC_B is the base model that specifies minimum required characteristics PuRBAC. PuRBAC_H extends PuRBAC_B with the notion of hierarchies of roles, purposes, and permissions. Although standard hierarchies are

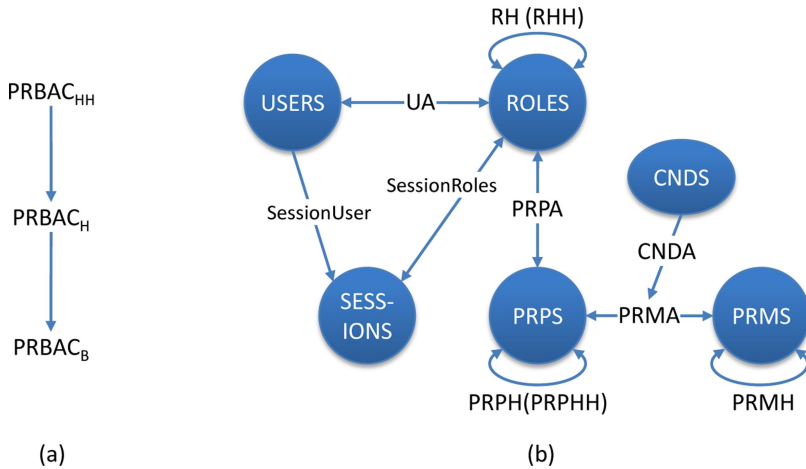


Fig. 1. Proposed Purpose-Aware Role-Based Access Control (PuRBAC) Model

considered a powerful property of RBAC models, they may introduce some risks because of conditional assignments that exist in the proposed PuRBAC model. To preclude such risks, PuRBAC_{HH} extends PuRBAC_H with the notion of hybrid hierarchies [9] for roles and purposes.

In the following, we provide definition and semantics for each proposed model in the family.

2.1 Base Model: PuRBAC_B

Figure 1.b shows the overall structure of the proposed model. USERS, ROLES, and SESSIONS components and the relations among them are inherited from the standard RBAC [10]: individual users (USERS) are assigned to roles (ROLES), who can create sessions (SESSIONS) at runtime and activate assigned roles in the created sessions. The extension in PuRBAC is based on how permissions (PRMS) are exercised. We argue that permissions are exercised for a particular purpose; therefore, permissions are assigned to purposes (PRPS) for which they can be exercised, and then purposes are assigned to proper roles.

We adopt a similar approach to the proposed NIST standard RBAC for permissions, in which permission represents a data type and corresponding action on that data. To enforce privacy policies in an enterprise data access system, such a model would be close to actual implementations. Whenever possible, we try to use the general notion of a permission.

As described before, privacy policies require flexible conditions for privilege management. To provide such flexibility, we enable conditions (CNDS) defined on permission assignments to purposes, that limit the assignment to particular cases or impose obligations. Here, we deliberately do not bring in an accurate definition for conditions to avoid complication in presentation of the model; though such a definition is independently provided in Section 3. For now, consider conditions as boolean predicates. An assignment is valid in a situation if and only if its condition is satisfied.

Formally speaking, the following sets and relations define the purpose-aware access control policy in PuRBAC_B model:

- *USERS*, *ROLES*, *PRPS*, and *DATA*; sets of users, roles, purposes, and data types, respectively.
- *PRMS*; set of pairs of the form $\langle d, a \rangle$ where $d \in DATA$ is a data type, and a is a valid action on that.
- *CNDS*; set of possible conditions.
- $UA \subseteq USERS \times ROLES$; user to role assignment relation.
- $PRPA \subseteq PRPS \times ROLES$; purpose to role assignment relation.
- $PRMA \subseteq PRMS \times PRPS$; permission to purpose assignment relation.
- $CNDA = CNDS \times PRMA$; condition to permission assignment binding relation.

Note that according to the definition of the *CNDA* relation, there exists exactly one condition for each permission assignment. The execution semantics of

the model is as follows. As in standard RBAC, assigned roles to a user can be activated and deactivated by her in corresponding sessions; for a working user there is at least one session, which is unique to her. For exercising privacy-sensitive permissions, a user needs to provide a purpose; while only purposes assigned to the current active roles of a user can be asserted. The user is authorized to exercise permissions assigned to the provided purpose, given their conditions are satisfied. The following sets and functions capture the state of the system at runtime based on user interaction with the system:

- $SESSIONS$; set of sessions created by the users.
- $SessionUser : SESSIONS \rightarrow USERS$; mapping function from a session to its corresponding user.
- $SessionRoles : SESSIONS \rightarrow 2^{ROLES}$; mapping function from a session s to its active roles rs , where $rs \subseteq \{r | \langle SessionUser(s), r \rangle \in UA\}$.

Although the principal requesting access in an access scenario is the user, but in RBAC such requests are mediated through sessions. Therefore, we discuss authorizations for sessions in which authorizations are requested on behalf of a user. The following functions capture runtime authorization for role activation, purpose assertion, and conditional permission exercise in PuRBAC_B:

- $AuthRoles : SESSION \rightarrow 2^{ROLES}$; mapping function from a session to the roles that can be activated in it. Formally: $AuthRoles(s : SESSIONS) = \{r \in ROLES | \langle SessionUser(s), r \rangle \in UA\}$.
- $AuthPurposes : SESSION \rightarrow 2^{PRPS}$; mapping function from a session to the purposes that can be asserted for exercising permissions. Formally: $AuthPurposes(s : SESSIONS) = \{prp \in PRPS | \langle prp, r \rangle \in PRPA \wedge r \in SessionRoles(s)\}$.
- $CAuthPurposePermissions : PRPS \rightarrow 2^{CNDS \times PRMS}$; mapping function from a purpose to the conditional permissions that can be exercised through. Formally: $CAuthPurposePermissions(prp : PRPS) = \{\langle cnd, prm \rangle \in CNDS \times PRMS | \langle prm, prp \rangle \in PRMA \wedge \langle cnd, \langle prm, prp \rangle \rangle \in CNDA\}$.

The access control process of PuRBAC is different from classic access control models that only grant or deny the access request. At runtime, a user access request is submitted as a session, purpose, and requested permission. The access decision function (ADF) either determines a conditional authorization or responds with a denial decision (if no conditional authorization is resolved):

$$ADF(s : SESSIONS, prp : PRPS, prm : PRMS) =$$

$$\begin{cases} cnd & \text{if } prp \in AuthPurposes(s) \\ & \wedge \exists \langle cnd, prm \rangle \in CAuthPurposePermissions(prp) \\ \text{"deny"} & \text{otherwise} \end{cases}$$

If a conditional authorization is resolved by ADF, it will be passed to the access control enforcement function (AEF). The actual enforcement of condition by AEF is dependant on the condition model in use. However, generally it will check some constraints and enforce some obligations that eventually may result in granting or denying the access. We will provide enforcement details in Section 3.

2.2 Hierarchical Model: PuRBAC_H

Hierarchies have been widely employed for propagation of authorization decision in access control models. Role hierarchy in standard RBAC allows senior roles to inherit permissions of junior roles. In PuRBAC_H, senior roles inherit the purposes allowed for junior roles. Hierarchies are also useful for purpose based on generality/specificity concepts [11,12]. If a user can access an object for one purpose, she can also access that object for a more specific purpose. We also consider hierarchy for permissions, specifically based on data hierarchy (e.g. aggregation hierarchy) for the same actions; if a user is authorized for an action on one data type, she is also authorized for the same action on descendents of that data type.

PuRBAC_H considers hierarchies for roles, purposes, and permissions in the policy, defined as follows:

- $RH \subseteq ROLES \times ROLES$; a partial order relation on roles, denoted as $\geq_r$.
- $PRPH \subseteq PRPS \times PRPS$; a partial order relation on purposes, denoted as $\geq_{prp}$.
- $PRMH \subseteq PRMS \times PRMS$; a partial order relation on permissions, denoted as $\geq_{prm}$.

The existence of hierarchies impose some changes to the base model. In addition to directly assigned roles, a user can activate roles that are junior to the assigned roles. Because of role hierarchy the user can assert purposes assigned to not only her active roles, but also junior roles of her active roles. Moreover, because of purpose hierarchy the user can assert any more general purpose than she is entitled through role hierarchy.

Special care should also be taken when dealing with exercising permissions, which are conditionally assigned to purposes. Hierarchies in standard RBAC have only permissive behavior while in PuRBAC they can be both permissive and constraining. Since there are no condition on permission assignment in standard RBAC, inheritance of a previously owned permission by a role through hierarchy does not change the role permissions; therefore, the only effect of hierarchy is inheriting permissions assigned to junior roles by senior roles. In contrast, due to the existence of conditions on permission assignments in our model, if a hierarchical relation for a purpose leads to inheritance of a permission that was previously assigned to the purpose, there will be different conditional assignments for the same permission. In such a situation, there are two possible approaches for authorizing the permission for purpose: whether one of the conditions or all of them should be satisfied. We take the conservative approach and consider all the conditions applicable. Because the administrator can define more general privacy conditions at lower levels of the hierarchies, and be ensured that they are applicable to more fine-grained purposes and permissions. Hierarchical inheritance in this way constrains the permission by putting more conditions for the exercise. We show an example of such a scenario later in this section.

The following functions capture runtime authorization for role activation, purpose assertion, and conditionally assuming permissions in PuRBAC_H:

- $AuthRoles : SESSION \rightarrow 2^{ROLES}$; mapping function from a session to the roles that can be activated in it. Formally: $AuthRoles(s : SESSIONS) = \{r \in ROLES \mid \langle SessionUser(s), r' \rangle \in UA \wedge r \geq_r r'\}$.
- $AuthPurposes : SESSION \rightarrow 2^{PRPS}$; mapping function from a session to the purposes that can be asserted for exercising permissions. Formally: $AuthPurposes(s : SESSIONS) = \{prp \in PRPS \mid \langle prp', r' \rangle \in PRPA \wedge prp \geq_{prp} prp' \wedge r \geq_r r' \wedge r \in SessionRoles(s)\}$.
- $CAuthPurposePermissions : PRPS \rightarrow 2^{CNDS \times PRMS}$; mapping function from a purpose to the conditional permissions which can be exercised through. Formally: $CAuthPurposePermissions(prp : PRPS) = \{\langle \prod cnd_i, prm \rangle \in CNDS \times PRMS \mid \langle prm', prp' \rangle \in PRMA \wedge prp \geq_{prp} prp' \wedge prm \geq_{prm} prm' \wedge \langle cnd_i, \langle prm', prp' \rangle \rangle \in CNDA\}$.

As described earlier in the case there are multiple applicable conditional assignments, all of them should be aggregated to be enforced; the term $\prod cnd_i$ refers to such an aggregation. The aggregation function itself is dependent to condition model in use. We provide the aggregation process of the conditions followed by our condition model in Section 3. Note that the access decision function ADF needs no change compared to the base model.

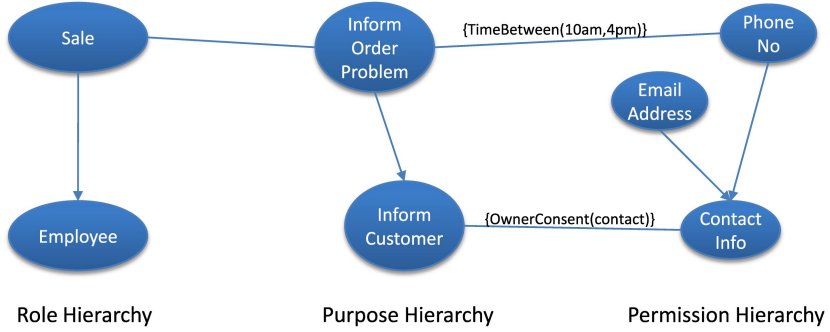


Fig. 2. Example Hierarchies and Their Assignments

For an example of access control process in $PuRBAC_H$, consider Figure 2 as partial role, purpose, and permission hierarchies in an online store system, along with their assignments. For simplicity, only the data types for permissions have been specified, and the action is *read* for all. The hierarchical relations are depicted with a directed line between entities. The role and purpose hierarchies are similar, e.g., role *sale* is senior to role *employee*, and purpose *inform order problem* is senior to (more specific than) purpose *inform customer*. The permission hierarchy is an aggregation hierarchy where the most coarse grained data is depicted at the bottom, e.g., permission to read *email address* is part of (more specific than) *contact info*. Data hierarchies are usually visualized in the opposite direction, but our visualization is aligned with the assignment inheritance, e.g., assignment of *contact info* to a role is also (indirectly) applied to *email*

address. Considering the mentioned relations, suppose that there is a problem with a customer's order and a sale employee wants to contact the customer. The employee first activates her *sale* role, which is assigned to the purpose *inform order problem*. For contacting the customer using email, she would request access to the customer's *email address*, asserting purpose *inform order problem* to the system. According to the policy setting, *contact information* can be accessed for purpose *inform customer* on the condition that owner of contact information has consented for contacting him using it. Based on the purpose and permission hierarchical relations, this authorization will be also applicable to the requested permission/purpose. Alternatively, the employee may call the customer by asserting purpose *inform order problem* to access customer's *phone number*. In that case there are two assignments applicable: there is a direct assignment between the requested purpose/permission with the condition of being daytime, and the previously mentioned assignment is also inherited through hierarchies with the condition of owner having consent. Therefore, the combined condition should be evaluated to true in order that access be granted to the employee.

2.3 Hybrid Hierarchical Model: PuRBAC_{HH}

Hybrid hierarchy have been originally defined in the context of Generalized Temporal RBAC (GTRBAC) [13]. It separates the notion of permission inheritance and activation inheritance in role hierarchy, taking into account three types of relations: inheritance (I), activation (A), and inheritance-activation (IA). If r_1 is I-senior to r_2 , it inherits the permissions of r_2 . If r_1 is A-senior to r_2 , any user assigned to r_1 can activate r_2 , but the role r_1 does not inherit permissions of r_2 . Finally, if r_1 is IA-senior to r_2 , it inherits the r_2 's permissions and also r_2 can be activated by anyone who can activate r_1 . PuRBAC_{HH} leverages hybrid hierarchy for roles and purposes to provide more flexibility and overcome a weakness of PuRBAC_H. Note that the semantics of hybrid hierarchy for purposes slightly differs from their semantic for roles in the way that purposes can be asserted instead of being activated.

One of the strengths of role hierarchy in RBAC is support for the principle of least privilege: a user is able to activate a junior role, which holds less permissions compared to a senior role, in the case she does not need the added permissions of the senior role for her current use of the system. PuRBAC_H model enables activating more junior roles in role hierarchy, and similarly asserting more general purposes in purpose hierarchy. Although purpose hierarchies are very similar to role hierarchies, asserting a more general purposes does not necessarily mean acquiring less privilege as it may even result in more privilege; if a user asserts a more general purpose than what she really intended for, she may have less restrictions for accessing some privacy-sensitive data. For instance, consider the previous example depicted in Figure 2. As mentioned before, if a sale employee asserts purpose *inform order problem* to access a customer's *phone number*, she is only allowed if the customer has consented before and it is daytime. But leveraging the hierarchical relation *inform order problem* $\geq_{prp}$ *inform customer*,

if the employee asserts purpose *inform customer*, she will be no longer restricted by the time constraint for accessing the phone number.

As described in previous section, the reason behind the possibility of such incident is that purpose hierarchies in our model can be either permissive or constraining, while role hierarchies in RBAC are permissive in nature. We can leverage the notion of hybrid hierarchies to overcome this issue, by allowing inheritance-only relation whenever we want to restrict the user's purpose assertions. For instance in the example above, if the relation between *inform customer* and *inform order shipment* is chosen as I-relation, user assigned to purpose *inform order shipment* will not be able to assert *inform customer* anymore, while the constraint for its corresponding assignment is still applied.

PuRBAC_{HH} redefines role and purpose hierarchies in RBAC_H with the notion of hybrid hierarchies (permission hierarchy remains unchanged) as follows:

- *RHH*; hybrid hierarchy over roles includes three relations defined in *ROLES*: inheritance denoted as $\geq_{rI}$, activation denoted as $\geq_{rA}$, and inheritance-activation denoted as $\geq_{rIA}$. Note that $r_1 \geq_{rIA} r_2 \Leftrightarrow r_1 \geq_{rI} r_2 \wedge r_1 \geq_{rA} r_2$.
- *PRPHH*; hybrid hierarchy over purposes includes three relations defined in *PRPS*: inheritance denoted as $\geq_{prpI}$, assertion denoted as $\geq_{prpA}$, and inheritance-assertion denoted as $\geq_{prpIA}$. Note that $prp_1 \geq_{prpIA} prp_2 \Leftrightarrow prp_1 \geq_{prpI} prp_2 \wedge prp_1 \geq_{prpA} prp_2$.

The definitions for functions capturing runtime authorizations in PuRBAC_H are applicable to PuRBAC_{HH} considering a few changes:

- In definition of *AuthRoles*, $\geq_r$ should be substituted with $\geq_{rA}$.
- In definition of *AuthPurposes*, $\geq_r$ and $\geq_{prp}$ should be substituted with $\geq_{rI}$ and $\geq_{prpA}$, respectively.
- In definition of *CAuthPurposePermissions*, $\geq_{prp}$ should be substituted with $\geq_{prpI}$.

3 Condition Model

Conditions that bind to permission assignments in PuRBAC have an important role in configuring access control policy to truly reflect the required privacy requirement. In Section 2, we defined a condition as a boolean predicate where the truth value affects the validity of corresponding permission assignment. In this section, we provide a more detailed approach for conditions defining their components and semantics.

Conditions can impose constraints on the assignment validity by the means of checking some data related to the accessed permission or any other access contexts. For instance, the consent of a data owner may be required to grant an access to the data. Also some data accesses may require that certain actions be properly pursued by the system or user before the access can be granted, referred to as pre-obligations. For instance, the policy may require the system to re-authenticate the user before authorizing access to highly privacy sensitive data such as social security numbers. In such a situation, if the user fails

to re-authenticate properly the permission can not be granted. Some other actions may be required to be carried out based on a data access after an access is granted, referred to as post-obligations. For example a data retention policy would schedule deletion of a data item in one year after its creation in data storage. Therefore, we model three types of conditions: constraints, pre-obligations, and post-obligations.

In order to provide the expected expressiveness for conditions, we define conditional constraints, pre-obligations, and post-obligations as follows:

- *CNDS*; conditions include conditional constraints, pre-obligations, and post-obligations. Formally: $CNDS = \{\mathcal{P}(PRDS \times CONS) \cup \mathcal{P}(PRDS \times PREOBGS) \cup \mathcal{P}(PRDS \times POSTOBGS)\}$, where
 - *PRDS* and *CONS* are sets of boolean predicates based on data variables in the system, representing general conditions and constraints, respectively.
 - *PREOBGS* is the set of all valid pre-obligations in the system. Pre-obligations may require input parameters.
 - *POSTOBGS* is the set of all valid post-obligations in the system. Post-obligations may require input parameters.

Semantically, constraints only query for data and provide a boolean result. Pre-obligations require the system or user to exercise some actions that may influence the access, while returning a boolean value that determines their proper enforcement. If any pre-obligation is not enforced properly, the access should be denied (and probably already-enforced pre-obligations be rollbacked). Post-obligations are enforced after the access is exercised. A special conditional predicate is available for post-obligations: *AccessGranted*, that can be used to control the dependency of enforcement of post-obligations on the result of access authorization. Figure 3 depicts how access control enforcement function (AEF) does the condition enforcement. The steps showing determination of constraints and

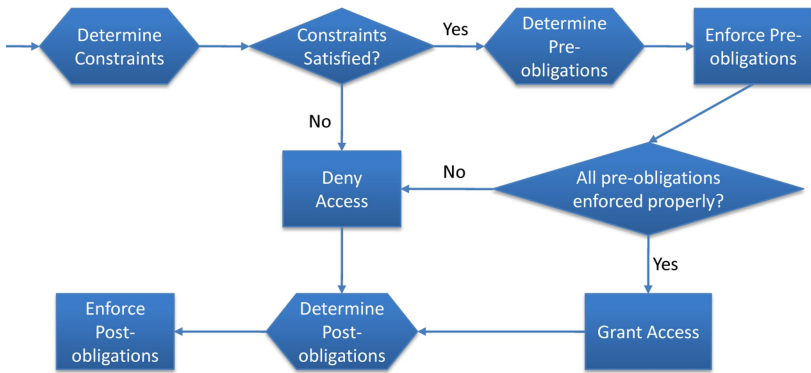


Fig. 3. Flowchart of Condition Enforcement by AEF

obligations check corresponding conditional predicates to decide which of them are applicable. Note that some post-obligations can be enforced even when the access is denied. Generally if a post-obligation does not check the conditional predicate *AccessGranted*, it will be enforced regardless of access decision. For instance, a post-obligation can log access attempt to an highly sensitive data item for operator review.

Different types of conditions can be defined and enforced based on our condition model. In the following examples d and a refer to the data instance being accessed and action on the data, respectively. Note that we use CN , PO , and OP for denoting conditional constraints, pre-obligations, and post-obligations, respectively. A category of important constraints in privacy policies is consent of data owner for collection or use of data. Such a constraint can simply be expressed as $CN(True, OwnerConsent(d, a) = True)$, where d and a refer to the data instance being accessed and type of access, respectively. However, in some situations a constraint is not applied for all instances of a data type. For example, Children Online Privacy Protection Act of 1998 (COPPA) [14] describes policies that are applicable to collection and processing of information about children under age 13. Here, there is a need to define conditional constraint whose condition is $OwnerAge(d) < 13$. In the case of COPPA, the consent of collection and use of data should be provided by a child's parent; therefore, the conditional constraint becomes $CN(OwnerAge(d) < 13, ParentalConsent(d, a) = True)$. Another category of conditions is data accuracy control which can be enforced through pre-obligations. For instance, suppose that a customer support wants to inform a buyer about the actual credit card that is billed, in response to an inquiry by the buyer. For such a purpose she needs only to verify the last 4 digits of the customer's credit card numbers. Such an accuracy can be controlled using an always-true pre-obligation on the permission assignment of accessing the credit card data field: $PO(True, FilterStringData(Length(d) - 4, Length(d)))$. In another example, suppose a child under age 13 wants to register in a Web site. The creation of a record for such a child fails in the first run because of a constraint similar to one mentioned previously. In such a situation, a post-obligation may seek parental consent if it hasn't been sought before: $OP(\neg AccessGranted \wedge ParentConsent = NA, AcquireParentalConsent(d, a))$. Other typical post-obligations for privacy policies are logging access or notifying some party such as the data owner about the access occurrence. Such post-obligations may be chosen to be enforced regardless of the access decision. The mentioned conditions do not constitute a complete list for possible conditions, but show a good variety of conditions for privacy control addressable by our model.

Besides support for different conditions, our model also supports aggregation of conditions. As mentioned in the previous section, in the case of $PuRBAC_H$ and $PuRBAC_{HH}$ models, that multiple assignments may apply to an access request, all the bound conditions should be aggregated and applied to the access. Since conditions are sets of conditional constraint or obligations, aggregation of multiple conditions is the union of those conditions. For instance, consider the following conditions:

$$\begin{aligned}
c_1 &= \{CN(True, OwnerConsent(d, a)), \\
&\quad OP(AccessGranted, SendOwnerNotification())\} \\
c_2 &= \{PO(True, GetUserAcknowledgement()), \\
&\quad OP(Owner(d) \in MonitoredOwners, LogAccess())\} \\
c_3 &= \{CN(True, OwnerConsent(d, a)), \\
&\quad PO(True, GetUserAcknowledgement()), \\
&\quad OP(Owner(d) \in MonitoredOwners, LogAccess()), \\
&\quad OP(AccessGranted, SendOwnerNotification())\}
\end{aligned}$$

Here, condition c_1 includes a constraint (checking owner's consent) and a post-obligation (sending a notification to the data owner if access is granted). Condition c_2 imposes a pre-obligation (getting acknowledgement from user for accessing data) and a post-obligation (logging the access if the data owner is being monitored). Condition c_3 is the union of conditions c_1 and c_2 , which includes all the mentioned constraints and obligations.

As conditions include obligations in addition to constraints, the unified set of conditions can result in inconsistencies or conflicts between obligations for enforcement. Moreover, enforcement of individual obligations may have inconsistencies or conflicts with the ongoing obligations in the system. In the case of two inconsistent obligations, the inconsistency can be resolved by overriding the one that can subsume the other. For instance, if an access results to a data retention post-obligation of one month, and there already exists a data retention obligation for one year, the one-month retention is enforced, discarding the other. But in the case of conflict, where none of the obligations can subsume the other, the system needs a meta-policy according to which it can determine which obligation should override, or possibly the access is being denied if no resolution is possible.

4 Analysis and Discussion

In this section we discuss strengths of the proposed model in Section 2 and some related issues.

4.1 Privacy Policy Management

Many privacy policies, particularly privacy acts and regulations, provide mandatory requirements for organizations to comply with. Those policies usually do not capture the internal system entities. Instead, they are focused on purposes and conditions that private data may be used. Therefore, the assignment of permissions on privacy-sensitive data to purposes along with the constraints can satisfactory model those privacy requirements; while authorization of roles for proper purposes complements the privacy policy.

The separation mentioned enables some independence for administration of the two assignment relations. Permission assignment should strictly consider

privacy regulations, for which the corresponding administrator should be completely informed and have clear understanding. However the latter part, purpose to role assignment, is more of an organizational issue and depend on roles' responsibilities, which possibly needs more modification over time compared to the former part. Therefore, we decrease the possibility of inadvertent manipulation of assignments privacy permission in the process of management of authorized organizational responsibilities and activities. Also, more flexibility for distributed administration is provided.

For instance, for the purpose *shipping order* in a company multiple permissions may be assigned to such as *read customer's shipping address* information. Suppose that shipping is done by sales department for some time, and hence purpose *shipping order* is assigned to role *sale*. Later if due to changes in organizational structure, shipment task is moved to the new shipping department, we need to change only the assignment of the purpose *shipping order* to the role *sale* to the new role *shipping*; there is no need to redefine the privacy permissions required for the purpose of shipping orders. In approaches such as P-RBAC [4], such a change requires manipulation of every privacy permission that is related to the purpose of *shipping order*.

In addition to the separation fact, a numerical comparison of possible assignments between our model and P-RBAC [4] can show the complexity differences. Note that we usually use term *assignment* for relating a pair of entities, and *rule* for relating multiple entities together (tuples). Therefore, assignments can also be considered as a tuple with two entities. The more the number of the rules a model deal with, the more complex it is to administer. For this comparison we exclude conditions (constraints and obligations), since they are present in both models and have similar influence on complexity. Suppose there are n roles, p purposes, and m permissions defined in a system. There can be as many as $n \times m \times p$ different authorization rules for roles in P-RBAC; while in our model we can have at most $p \times n$ assignments in *PRPA* (purpose to role assignments) and $n \times p$ assignments in *PRMA* (permission to purpose assignments), which sum up to $(n + m) \times p$ assignments. Considering that in practice the number of permissions (m) is much higher than the number of roles (n) in a system, our model can have about a factor of n less possible assignments, and hence is much less complex to administer.

4.2 Expressiveness Power

As described in the previous subsection, PuRBAC decreases the policy complexity by avoiding rules involving multiple components, namely role, purpose, and permission. Surely, such complexity decrease comes at the expense of some loss of expressiveness power. An expressiveness challenge may arise in a scenario that different roles with the same purpose can have different accesses. Consider the following scenario in a healthcare institute. The role *senior researcher* can access *complete profiles* of specific patients with the purpose *research*, with previous consent though. However, the role *research assistant* with the same purpose has only access to *limited profiles*. We believe that if purposes are defined

fine-grained enough in the system, there would not be an expression problem most of the time. For the mentioned scenario, although both accesses by roles *senior researcher* and *research assistant* have the purpose *research* at high level, they can be categorized into the more fine-grained purposes *complete research* and *limited research*, respectively.

If fine-grained purposes are not easy to define in a scenario, PuRBAC can cope with the issue by allowing predicates based on role in the conditional constraints on permission to purpose assignment. Therefore, permissions can be dynamically assigned to purposes based on the user's active roles. However, the use of such a role-based constraints should be restricted to special situations to keep the complexity advantage of PuRBAC.

4.3 Control over Purpose

Access control models that support privacy policies usually require the user to indicate the purpose of accessing information as one of the access request parameters. The indicated purpose is then used to check for compliance with policies in the system. The drawback in existing models is that users can indicate any purpose for information access without any restriction. Although the indicated purpose is checked against the policy, but that freedom makes system very vulnerable to misuse of data for purposes not really related to a role. That can happen with the existence of a simple error in the policy rules, that is not unlikely in practice considering the presence of role and purpose hierarchies.

In our model, the user cannot use data for a purpose without first having been authorized for that purpose. Such authorization is possible only for those purposes assigned to the user's currently active roles; and those assignment come from the fact that any role has a restricted set of responsibilities and functionalities which will define purposes for privacy-sensitive information access.

4.4 Role vs. Purpose

As mentioned in Section 1, notions of purpose and role can be very similar. The closeness and similarities between them tend to make it difficult to make distinction in some situations. For example, order processing in a store can be modeled as a specific role compared to widely scoped roles (as described in [15]), as well as a purpose for accessing customer record (as described in [11]). We take advantage of this tight relation to justify their direct assignment in our model, and argue against considering them both as (different type of) roles. Roles are usually derived based on organizational positions and responsibilities. But purposes have no relation to organizational structure, but to functions.

Someone may argue that the role entity itself can support both notions of role and purpose in our model, by having structural and functional roles, respectively. In such an approach, the model needs to deal differently with those two role types by allowing (i) assignment of permissions only to functional roles, (ii) inherence of functional roles by structural roles, and (iii) assignment of users only to structural roles. Moreover, a major difference of authorization model of

PuRBAC, compared to RBAC, is assertion of purpose as a part of access request. That feature is not supported in standard RBAC, as access check is only based on session and requested permission [10]. Therefore, the access checking function in the standard needs to be changed to be aware of the access purpose (functional role). Considering the mentioned differences and also the administration independence for role and purpose hierarchies in our model, we believe that there is enough motivation to consider purpose as a separate entity from role.

4.5 Sticky Policies

A very flexible approach for privacy policies would be the sticky policy paradigm [16]. In that approach, policies are defined and maintained for each data instance. Therefore, privacy policies can be different for different instances of the same data type. Although quite promising, we argue that it is less probable to be followed by organizations. The main drawback is that the organization would lose its centralized control over access control policies once the policy is stuck to the data. That is not preferred since the access control policy may require changes due to revision of high-level policies, or frequent improvement of the access control policy itself. Moreover, the storage and processing of access control policies will be very expensive in the case of using sticky policy approach.

5 Related Work

Enforcement of privacy policies have been studied by several researchers. Karjoth et al. propose Enterprise Privacy Architecture (EPA) as a methodology for enterprises to provide privacy-enabled service to their customers [17]. In [2], Powers et al. investigate privacy concerns from organizations' point of view and existing technologies, and describe an approach for enterprise-wide privacy management. They suggest expressing privacy policy in terms of privacy rules comprising of data type, operation on it, data user, purpose of data access, condition that restricts that access, and obligatory actions taken by the user when she is authorized. The Platform for Enterprise Privacy Practices (E-P3P) policy language, proposed in [5], contains authorization rules. In addition to mentioned components of the privacy policy, each authorization rule in E-P3P contains a parameter that indicates if the rule is either positive or negative authorization, and a precedence value. Tree hierarchies for data categories, users, and purposes are also considered. Another similar work has been done by Karjoth et al. that extends Jajodia's Authorization Specification Language (ASL) [18], to include obligations and user consent [11]. They also discuss a solution to automatically translate inner-enterprise privacy policy stated using E-P3P to publishable P3P policies for customers [19]. The language has been formalized and refined to form IBM Enterprise Privacy Authorization Language (EPAL) [20].

Ni et al. propose P-RBAC [4], a privacy-aware role-based access control model, which incorporates privacy policies into RBAC [3]. They encapsulate data, action, purpose, condition, and obligation as privacy data permission. A permission assignment in P-RBAC is an assignment of a privacy data permission

to a role (equivalent to data user in [2]). Also, more complex conditions have been considered in a conditional version of P-RBAC [21]. In previous sections, especially Section 4, we described and analyzed the advantages of our approach to modeling purposes compared to P-RBAC. From condition model perspective, compared to P-RBAC the support of conditional constraint and obligations in our model enables more concise privacy policy definition. P-RBAC requires specifying explicitly all the conditions when a privacy-sensitive data can be accessed (the notion of condition in P-RBAC is close to constraint in our model). However, conditional constraints in our model allows enforcement of constraints on permissions only when some conditional predicates are met. P-RBAC also lacks clear semantics for enforcement of obligations; our model provides clear semantics for the enforcement process flow of constraints, pre-obligation, and post-obligations. Ni et al. have proposed an obligation model extension for P-RBAC very recently, which includes the notion of conditional obligation (but not conditional constraints), and deals with temporal obligations and obligation dominance [22].

Byun et al. address the notion of purpose-based access control [12], seeking compliance between intended (allowed or prohibited) purposes defined for data and access purposes requested by users at runtime. The strength of their approach is dealing with purpose in complex hierarchical data management systems. However, the approach seems too complicated to be used as a general purpose access control model.

6 Conclusions

We proposed purpose-aware role-based access control (PuRBAC) model as a natural RBAC extension to include privacy policies. In PuRBAC, purpose is defined as an intermediary entity between role and permission. Users can only exercise permissions assigned to an asserted purpose, which itself should be authorized through assignment to the user's active roles. Also, assignments of permissions to purposes are bound with conditions that constrain the assignment validity or impose obligations. We also defined a general model of conditions, providing the enforcement semantics for conditional constraints, pre-obligations, and post-obligations.

The introduction of purpose as a separate high-level entity in RBAC requires further analysis of its impact on other aspects of RBAC paradigm such as separation of duty constraints, policy administration model, etc. Moreover, the assertion of purpose in the access control process needs to be studied in more detail from usability and accountability perspectives. The ultimate goal might be to enable the system intelligently identify the purpose of data access according to the user's tasks, while ensuring the user accountability.

Acknowledgements. This research has been supported by the US National Science Foundation award IIS-0545912. We would like to thank the anonymous reviewers for their helpful comments.

References

1. Antón, A.I., Bertino, E., Li, N., Yu, T.: A roadmap for comprehensive online privacy policy management. *Communications of the ACM* 50(7), 109–116 (2007)
2. Powers, C., Ashley, P., Schunter, M.: Privacy promises, access control, and privacy management: Enforcing privacy throughout an enterprise by extending access control. In: *Proc. 3rd International Symposium on Electronic Commerce*, October 18–19, 2002, pp. 13–21 (2002)
3. Sandhu, R.S., Coyne, E.J., Feinstein, H.L., Youman, C.E.: Role-based access control models. *IEEE Computer* 29(2), 38–47 (1996)
4. Ni, Q., Trombetta, A., Bertino, E., Lobo, J.: Privacy-aware role based access control. In: *Proc. 12th ACM symposium on Access control models and technologies*, pp. 41–50. ACM Press, New York (2007)
5. Ashley, P., Hada, S., Karjoth, G., Schunter, M.: E-P3P privacy policies and privacy authorization. In: *Proc. ACM workshop on Privacy in the Electronic Society*, pp. 103–109. ACM, New York (2002)
6. Sandhu, R.S., Samarati, P.: Access control: Principles and practice. *IEEE Communications Magazine* 32(9), 40–48 (1994)
7. Cranor, L., Langheinrich, M., Marchiori, M., Presler-Marshall, M., Reagle, J.: The platform for privacy preferences 1.0 specification. Technical report, W3C (2002)
8. OECD: Oecd guidelines on the protection of privacy and transborder flows of personal data (1980),
http://www.oecd.org/document/18/0,3343,en_2649_34255_1815186_1_1_1_1,00.html
9. Joshi, J.B.D., Bertino, E., Ghafoor, A., Zhang, Y.: Formal foundations for hybrid hierarchies in gtrbac. *ACM Transactions on Information and System Security* 10(4), 1–39 (2008)
10. Ferraiolo, D., Kuhn, D.R., Chandramouli, R.: Role-based access control. Artech House computer security series. Artech House, Boston (2003)
11. Karjoth, G., Schunter, M.: A privacy policy model for enterprises. In: *Proc. 15th IEEE Computer Security Foundations Workshop*, June 24–26, 2002, pp. 271–281 (2002)
12. Byun, J.W., Bertino, E., Li, N.: Purpose based access control of complex data for privacy protection. In: *Proc. 10th ACM symposium on Access control models and technologies*, pp. 102–110. ACM Press, New York (2005)
13. Joshi, J., Bertino, E., Ghafoor, A.: Hybrid role hierarchy for generalized temporal role based access control model. In: *Proc. 26th Annual International Computer Software and Applications Conference (COMPSAC)*, August 26–29, 2002, pp. 951–956 (2002)
14. FTC: Children’s online privacy protection act of 1998 (coppa) (1998),
<http://www.ftc.gov/ogc/coppa1.htm>
15. Samarati, P., De Capitani di Vimercati, S.: Access control: Policies, models, and mechanisms. In: Focardi, R., Gorrieri, R. (eds.) *FOSAD 2000. LNCS*, vol. 2171, pp. 137–196. Springer, Heidelberg (2001)
16. Karjoth, G., Schunter, M., Waidner, M.: Platform for enterprise privacy practices: Privacy-enabled management of customer data. In: Dingledine, R., Syverson, P.F. (eds.) *PET 2002. LNCS*, vol. 2482, pp. 69–84. Springer, Heidelberg (2003)
17. Karjoth, G., Schunter, M., Waidner, M.: Privacy-enabled services for enterprises. In: *Proc. 13th International Workshop on Database and Expert Systems Applications*, September 2–6, 2002, pp. 483–487 (2002)

18. Jajodia, S., Samarati, P., Sapino, M.L., Subrahmanian, V.S.: Flexible support for multiple access control policies. *ACM Transactions on Database Systems* 26(2), 214–260 (2001)
19. Karjoth, G., Schunter, M., Van Herreweghen, E.: Translating privacy practices into privacy promises: how to promise what you can keep. In: *Proc. 4th IEEE International Workshop on Policies for Distributed Systems and Networks*, June 4–6, 2003, pp. 135–146 (2003)
20. IBM: The enterprise privacy authorization language,
<http://www.zurich.ibm.com/security/enterprise-privacy/epal/>
21. Ni, Q., Lin, D., Bertino, E., Lobo, J.: Conditional privacy-aware role based access control. In: Biskup, J., López, J. (eds.) *ESORICS 2007*. LNCS, vol. 4734, pp. 72–89. Springer, Heidelberg (2007)
22. Ni, Q., Bertino, E., Lobo, J.: An obligation model bridging access control policies and privacy policies. In: *Proc. 13th ACM symposium on Access control models and technologies*, pp. 133–142. ACM, New York (2008)

Uncle-Share: Annotation-Based Access Control for Cooperative and Social Systems

Peyman Nasirifard and Vassilios Peristeras

Digital Enterprise Research Institute
National University of Ireland, Galway
IDA Business Park, Lower Dangan, Galway, Ireland
`firstname.lastname@deri.org`

Abstract. Shared workspaces and Web 2.0 platforms provide lots of services for sharing various objects. Most current shared workspaces and Web 2.0 platforms provide role-based, coarse-grained access control policies which undermine the utility of them in some cases. In this paper, we present Annotation-Based Access Control, an approach towards access control which benefits from user annotations to annotate people using various fixed and desired open vocabulary (tags) and helps to build a more flexible access control mechanism based on relationships among different types of users. We also present a prototype, a gadget called Uncle-Share, which we have developed to enable this access control mechanism and evaluate it.

Keywords: Access Control, Shared Workspace, Annotation, Social Network, Web 2.0.

1 Introduction

Web 2.0 platforms and shared workspaces (e.g. BSCW, Microsoft SharePoint) provide necessary tools and infrastructure for sharing various items. In a shared workspace or social platform, where the people collaborate together and share resources, there should definitely exist some kind of embedded access control mechanisms in order to restrict unauthorized accesses to various resources. In brief, Access Control defines who can access what data [15].

We have analyzed the embedded access control mechanisms within some shared workspaces and Web 2.0 platforms. We signed up to some platforms, uploaded/added some resources (e.g. documents, photos, bookmarks), added some contacts as friends and tried to share our resources with some of our contacts. We noticed that the embedded access control mechanisms were not flexible enough to enable us to share our resources with desired contacts within specific context. For instance, we could not share a specific project-related bookmark with only people that are working on that project. To overcome this situation, we had to send emails to share the bookmark with them. Most current shared workspaces and Web 2.0 platforms provide coarse-grained access control policies which undermine the utility of them in some cases.

In this paper, we present an approach for access control by annotating people and defining access control policies based on the annotations (i.e. Annotation-Based Access Control). We benefit from Semantic Web [2] technologies for annotations, storing and retrieving data. These technologies enable us to do reasoning on top of annotations and also help us to interact with various platforms or even other applications can integrate with our platform.

The rest of this paper proceeds like the following: In the next part, we have an overview of related work regarding access control in social and cooperative systems. In section 3, we introduce Annotation-Based Access Control model. In part 4, we present a prototype that we have developed to enable and evaluate our access control mechanism. After that, we compare our model with some other approaches and finally we conclude and have an overview of future works in section 6.

2 Related Work

There exist plenty of approaches and mechanisms towards controlling access to electronic resources: access control lists, which is probably the simplest access control mechanism, role-based access control [7,17], attribute-based access control [12], etc. Each approach has its own advantages, disadvantages and feasibility scope.

Many researchers try to combine different access control mechanisms to build a more powerful mechanism and decrease the disadvantages of each mechanism. Kern et al. [10] provide an architecture for role-based access control to use different rules to extract dynamic roles. Alotaiby et al. [1] present a team-based access control which is built upon role-based access control. Periorellis et al. [14] introduce another extension to role-based access control which is called task-based access control. They discuss task-based access control as a mechanism for dynamic virtual organization scenarios. Georgiadis et al. [8] provide a model for combining contextual information with team-based access control and they provide a scenario in health care domain, where the model is used. Zhang et al. [21] propose a model for dynamic context-aware role-based access control for pervasive applications. They extend role-based access control and dynamically align role and permission assignments based on context information.

The study of access control mechanisms in Cooperative Systems is not new and was in existence since the birth of e-Collaboration tools in 1980s. Shen et al. [18] studied access control mechanisms in a simple collaborative environment, i.e. a simple collaborative text editing environment. Zhao [22] provides an overview and comparison of three main access control mechanisms in collaborative environments. Tolone et al. [19] have published a comprehensive study on access control mechanisms in collaborative systems and compare different mechanisms based on multiple criteria, e.g. complexity, understandability, ease of use. Jaeger et al. [9] present basic requirements for role-based access control within collaborative systems. Kim et al. [11] propose a collaborative role-based access control (C-RBAC) model for distributed systems which is fine-grained and try to address the conflicts from cross-domain role-to-role translation.

There exist some studies on access control in social networks. Most of the literature focuses on relationships that the people may acquire in a social network. In [16], Kruk et al. suggest a role-based policy-based access control for social networks, where the access rights will be determined based on social links and trust levels between people. In [3], Carminati et al. present the same approach and in [5], they extend their model by adding the concept of private relationships in access control, as they noticed that all relationships within social networks should not be public, due to security and privacy reasons.

3 Annotation-Based Access Control Model

Annotation is a common mechanism which is used nowadays by many social platforms for annotating shared objects to facilitate the discovery of relevant resources. Our access control model is based on annotations. It benefits **partially** from social acquaintances to express the annotation mechanism, however it is not only limited to fixed terms and open vocabularies can be also utilized for annotations. In this approach, end users are able to annotate their **contacts** and define policies based on their **annotations**. In this case, only those annotated contacts, that fulfill the required policies, have access to specified resources. A simple example follows: User A annotates user B which is part of his social network as **supervisor**. User A owns also several resources and defines different policies for them. In this case, all resources that have just **supervisor** in their policies and their policies express that they can be shared with the people that have been tagged as **supervisor**, are automatically accessible to the user B which has been annotated as **supervisor**.

Annotation-based access control is very close to how we share resources in our real-life. We may share the key of our apartments with our parents, but not with our friends. Based on this simple scenario, in annotation-based access control, both our parents and friends are parts of our social network, but our parents have been tagged as **parent** while our friends as **friend** and our keys are resources that we define to be shared only with entities tagged as **parent**.

Our current access control model consists of three main entities and two main concepts: **Person**, **Resource**, and **Policy** are the entities; **Annotation** and **Distance** are the main concepts. A Person is an entity with the RDF type *Person*¹. A Person is connected to zero or more other Persons. Each connection between Persons can be annotated with zero or more Annotations. An Annotation is a term or a set of terms that are connected together and aims to describe the Person. A Person owns zero or more Resources. A Resource is an entity with the RDF type *Resource*² and is owned by (isOwnedBy) one or more Persons. Resources may be in the form of URIs / URLs / short messages. A Resource can be either private or public. A Policy is an entity with the RDF type *Policy*³. A

¹ <http://uncle-share.com/ontology/Person> OR
<http://xmlns.com/foaf/0.1/Person>

² <http://uncle-share.com/ontology/Resource>

³ <http://uncle-share.com/ontology/Policy>

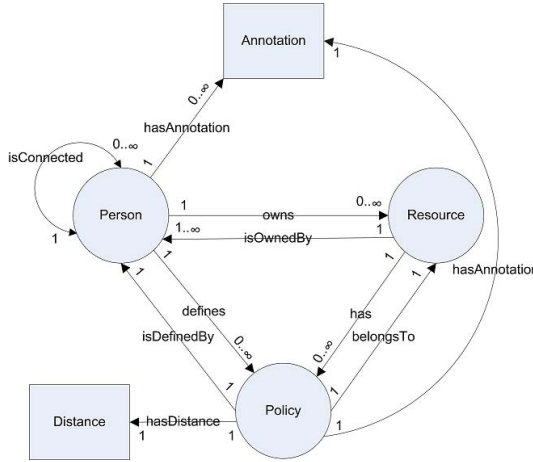


Fig. 1. Main elements in access control mechanism and their relationships

Policy is defined by (isDefinedBy) one Person and belongs to (belongsTo) one Resource. A Policy has one Annotation and one Distance. Again an Annotation is a term or a set of terms that are connected together and aims to describe the Person that the Resource should be shared with. A Distance is a numerical value which determines the **depth** that the Policy is valid. Depth is actually the shortest distance among two Persons with consideration of Annotations. This will be more clear with an example in section 5. A Person defines zero or more Policies. Note that multiple Policies for a Resource that are defined by a Resource owner may have different Distances. Figure 1 demonstrates the main elements of our access control model.

There exist several rules (meta-policies) in our approach:

- Rule 1: A Person acquires access to a Resource, if and only if (iff) s/he meets *all* policies that have been defined by Resource owner for that Resource. It means that the Person has been already annotated with the Annotations which are defined in the Policies and s/he is also in the scope of the Policies (i.e. Distance criteria). A conclusion of this rule follows: Multiple Policies that are defined by a Resource owner for a Resource are *ORed*, if they have different Distances, otherwise the Policies are *ANDed*.
- Rule 2: Only the Resource owner is eligible to define Policies for that Resource.
- Rule 3: If a Person acquires access to a Resource, s/he may *copy/add* the Resource to his/her Resources. In this case, s/he will be the Resource owner. (The original Resource owner will also keep the ownership as well.)
- Rule 4: A private Resource has zero or more Policies, whereas a public resource has at least one Policy.
- Rule 5: The default Distance for Policies is one.

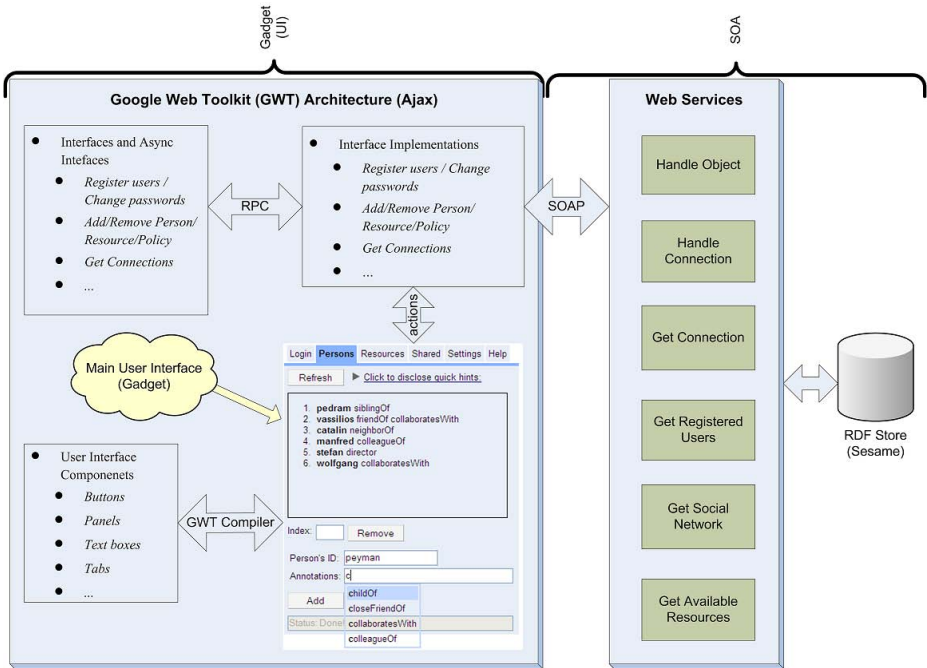


Fig. 2. The general architecture and overview of UI

4 Uncle-Share: Annotation-Based Access Control Prototype

Based on the presented Annotation-Based Access Control model, we have developed a prototype called **Uncle-Share** to validate and test the approach. Figure 2 demonstrates the overall architecture of Uncle-Share plus an overall view of the User Interface (UI). In the following, we describe some important aspects of the architecture and UI.

Uncle-Share is based on Service-Oriented Architecture (SOA). Uncle-Share provides several SOAP-based services to end users. In other words, all functionalities of Uncle-Share (registration, changing password, adding persons and resources, fetching shared resources, etc.) are wrapped as Web services. Following this approach enables developers to utilize all functionalities of Uncle-Share within their own applications. Uncle-Share provides currently the following services:

- **Handle Object**: This service enables end users to register themselves to the system and/or change their passwords.
- **Handle Connection**: This service enables end users to add connections between persons; persons and resources; and persons and policies. This service enables also end users to annotate those connections with closed and open terms.

- **Get Connection:** This service enables end users to get who/what stuff is connected to a specific person.
- **Get Registered Users:** This service returns the list of the registered users on the system.
- **Get Social Network:** This service returns the social network of authenticated user in RDF (based on FOAF⁴).
- **Get Available Resources:** This service returns the available resources to a specific person based on *Distance* input.

We have chosen to build the Uncle-Share user interface as a widget / gadget. These two terms, widget and gadget, are used sometimes to refer to the same concept. There exist currently many gadget / widget platforms, and open source and commercial gadget-building tools. NetVibes⁵ and iGoogle⁶ are two mostly used gadget platforms. Both platforms provide basic tools for building gadgets / widgets. Having **gadgetized** user interface enables end users to have Uncle-Share besides other applications and this can attract more users, as they should not launch a new application or browse a new Web page to utilize Uncle-Share. Our gadget can be embedded into any widget / gadget platform or Web site. We used AJAX [20] (Asynchronous JavaScript and XML) technologies for developing the user interface. The only client-side requirement is that the browser should support JavaScript. The current version of gadget has six main tabs: Login, Persons, Resources, Shared, Settings, and Help.

For annotations and also defining policies, Uncle-Share has a **suggest box**. In the suggest box, end users will get some recommendations / suggestions from Uncle-Share. These suggestions are based on the RELATIONSHIP [6] ontology. It is an extended version of FOAF and a set of terms for describing the general relationships between people.

Uncle-Share gadget can be accessed and tested online^{7 8}. We have successfully embedded the gadget into iGoogle and BSCW shared workspace, in order to enable Annotation-Based Access Control for bookmarks.

We have chosen some specific open source and free software to implement Uncle-Share. We use Sesame 2.0⁹ as RDF store. The SOA backbone is based on Apache CXF¹⁰ which eases the development of Web services. For building the AJAX-based gadget, we used Google Web Toolkit¹¹ (GWT). GWT has a Java to JavaScript compiler which compiles the Java source and generates desired user interface.

⁴ <http://www.foaf-project.org/>

⁵ <http://www.netvibes.com/>

⁶ <http://www.google.com/ig>

⁷ <http://purl.oclc.org/projects/uncle-share-gadget-igoogles>

⁸ <http://purl.oclc.org/projects/uncle-share-gadget-standalone>

⁹ <http://www.openrdf.org/>

¹⁰ <http://incubator.apache.org/cxf/>

¹¹ <http://code.google.com/webtoolkit/>

5 Evaluation and Comparisons

In the first glance, our approach for access control sounds to be similar to Role-Based Access Control [7,17] (RBAC), Generalized Role-Based Access Control [13] (GRBAC) and other family members of RBAC. In brief, in RBAC, a user is assigned one or more roles. Each role has some defined permissions. Users will receive desired permissions through their roles or they inherit the permissions through the role hierarchy. RBAC is a quite successful access control method and is used in many platforms (operating systems, databases, etc.) and organizations. In GRBAC [13], the authors extend RBAC by introducing subject roles, object roles and environment roles.

RBAC, GRBAC and other family members of RBAC works well, if there exists well-structured (and perhaps hierarchy) of roles, permissions (and resources). The main difference between RBAC and our approach is that in RBAC, the roles are already defined by a role engineer, but in our approach, we have decentralized concepts (i.e. annotations) which are not necessary roles (from the semantics point of view). It is the user that defines his/her own annotations and assigns them to his/her contacts which is more user-centric. From the RBAC perspective, our model can be seen as an extension to RBAC through assigning user-centric roles (i.e. annotations) to a person's contacts. The other main difference is the concept of Distance which increases or decreases the scope of policies in sharing resources, as the people are connected together in a graph-like manner (rather than hierarchy-like manner). Where RBAC can be very useful in large and well-structured organizations, our approach fits well for defining access control policies for personal data.

In our model, all relationships are private, as there is no need to publicly announce the relationships between people, due to privacy reasons. However, end users can freely publish their own relationships, if this is needed. While fixed vocabulary is used in approaches like [3], in our model and tool, fixed terms are just suggested to end users, as we do not really force users to exclusively use them. They are allowed to use their own terms as well as fixed terms for annotations. This open vocabulary approach enables end users to express the trust level in a more accurate way as well. As an example, instead of using percentages for expressing the trust level (e.g. friend 80%) like in [16], end users can express degrees of friendship in a more natural way with an annotation like `closeFriendOf`. The model becomes in this way more realistic and expressive, as we don't really label our friends and relationships in real-life with numerical values and percentages.

Moreover, we calculate the distance between two persons taking into account the annotation values. This is important because annotations build a graph among people which may contain several paths between two persons and it is important to consider all paths when we want to reach target person from a source person. For example, if person A is connected to person B and this connection has the annotation `student`, the distance from person A to B (directional) with the consideration of `student` is one. The distance from person A to B (directional) with the consideration of any other annotation (e.g. `friendOf`) is infinity. The

distance from person B to A (directional) is also infinity, if person B has not defined an outgoing link to person A.

6 Conclusion and Future Work

In this paper, we presented an annotation-based access control model and a prototype based on that. Uncle-Share enables end users to annotate their contacts and set different policies for their resources based on their annotations. From the RBAC perspective, our model can be seen as an extension to RBAC, where people are able to define their own roles (i.e. annotations) and assign them to others in a user-centric model.

We are currently working to extend RELATIONSHIP ontology and add more collaboration-based terms to it, as our model and prototype are mainly utilized in collaboration-based environments. RELATIONSHIP ontology and works like REL-X [4] contain the terms that capture the **general** relationships and social acquaintances among people.

One other interesting extensions is using Open Social¹² API to embed the Uncle-Share into the social networking sites like MySpace and Orkut. Open Social follows the idea of **Write once, run anywhere** and enables developers to develop cross-platform applications among social Web sites.

More advanced user model and suggestions / recommendations, and prioritizing the policies are different possible improvements. Due to the small nature of widgets / gadgets, we may develop a full-screen version of user interface and put a snippet of the main interface into the gadget.

Acknowledgments

This work is supported by Ecospace project: FP6-IST-5-352085.

References

1. Alotaiby, F.T., Chen, J.X.: A Model for Team-based Access Control. In: International Conference on Information Technology: Coding and Computing. IEEE Computer Society, Los Alamitos (2004)
2. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web, A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities. *Scientific American* (2001)
3. Carminati, B., Ferrari, E., Perego, A.: Rule-Based Access Control for Social Networks. In: OTM Workshops (2), pp. 1734–1744. Springer, Heidelberg (2006)
4. Carminati, B., Ferrari, E., and Perego, A. The REL-X vocabulary. OWL Vocabulary (accessed June 18, 2008) (2006), <http://www.dicom.uninsubria.it/andrea.perego/vocs/relx.owl>
5. Carminati, B., Ferrari, E., Perego, A.: Private Relationships in Social Networks. In: Proceedings of ICDE Workshops, pp. 163–171 (2007)

¹² <http://opensocial.org/>

6. Davis, I., Vitiello Jr., E.: RELATIONSHIP: A vocabulary for describing relationships between people (accessed June 18, 2008) (2005), <http://vocab.org/relationship/>
7. Ferraiolo, D.F., Kuhn, D.R.: Role Based Access Control. In: 15th National Computer Security Conference, pp. 554–563 (1992)
8. Georgiadis, C.K., Mavridis, I., Pangalos, G., Thomas, R.K.: Flexible team-based access control using contexts. In: SACMAT 2001: Proceedings of the sixth ACM symposium on Access control models and technologies, pp. 21–27. ACM Press, New York (2001)
9. Jaeger, T., Prakash, A.: Requirements of role-based access control for collaborative systems. In: 1st ACM Workshop on Role-based access control. ACM Press, New York (1996)
10. Kern, A., Walhorn, C.: Rule support for role-based access control. In: 10th ACM symposium on Access Control Models and Technologies, pp. 130–138. ACM Press, New York (2005)
11. Kim, H., Ramakrishna, R.S., Sakurai, K.: A Collaborative Role-Based Access Control for Trusted Operating Systems in Distributed Environment. IEICE transactions on fundamentals of electronics, communications and computer sciences 88(1), 270–279 (2005)
12. Kolter, J., Schillinger, R., Pernul, G.: A Privacy-Enhanced Attribute-Based Access Control System. In: Barker, S., Ahn, G.-J. (eds.) Data and Applications Security 2007. LNCS, vol. 4602, pp. 129–143. Springer, Heidelberg (2007)
13. Moyer, M.J., Ahamad, M.: Generalized Role-Based Access Control. In: ICDCS 2001: Proceedings of the The 21st International Conference on Distributed Computing Systems, p. 391. IEEE Computer Society, Los Alamitos (2001)
14. Periorellis, P., Parastatidis, S.: Task-Based Access Control for Virtual Organizations. Scientific Engineering of Distributed Java Applications, 38–47 (2005)
15. Russell, D., Gangemi Sr., G.T.: Computer Security Basics. O'Reilly and Associates, Inc., Sebastopol (1991)
16. Ryszard Kruk, S., Grzonkowski, S., Gzella, A., Woroniecki, T., Choi, H.C.: D-FOAF: Distributed Identity Management with Access Rights Delegation. In: Mizoguchi, R., Shi, Z.-Z., Giunchiglia, F. (eds.) ASWC 2006. LNCS, vol. 4185, pp. 140–154. Springer, Heidelberg (2006)
17. Sandhu, R.S., Coyne, E.J., Feinstein, H.L., Youman, C.E.: Role-Based Access Control Models. IEEE Computer 29(2), 38–47 (1996)
18. Shen, H., Dewan, P.: Access Control for Collaborative Environments. In: Computer-Supported Cooperative Work Conference, pp. 51–58. ACM Press, New York (1992)
19. Tolone, W., Ahn, G., Pai, T., Hong, S.: Access control in collaborative systems. ACM Computing Surveys 37, 29–41 (2005)
20. Zakas, N.C., McPeak, J., Fawcett, J.: Professional Ajax (Programmer to Programmer), 2nd edn. Wiley Publishing, Chichester (2007)
21. Zhang, G., Parashar, M.: Dynamic Context-aware Access Control for Grid Applications. In: GRID 2003: Proceedings of the Fourth International Workshop on Grid Computing, p. 101. IEEE Computer Society, Los Alamitos (2003)
22. Zhao, B.: Collaborative Access Control. In: Seminar on Network Security (NetSec) (2001)

Verifying Extended Criteria for the Interoperability of Security Devices

Maurizio Talamo^{1,3}, Franco Arcieri¹, Giuseppe Della Penna², Andrea Dimitri¹,
Benedetto Intrigila³, and Daniele Magazzeni²

¹ Nestor Lab - University of Roma "Tor Vergata", Italy

² Department of Computer Science, University of L'Aquila, Italy

³ Department of Mathematics, University of Roma "Tor Vergata", Italy

Abstract. In the next years, smart cards are going to become the main personal identification document in many nations. In particular, both Europe and United States are currently working to this aim. Therefore, tens of millions of smart cards, based on hardware devices provided by many different manufacturers, will be distributed all over the world, and used in particular to accomplish the security tasks of *electronic authentication* and *electronic signature*. In this context, the so called *Common Criteria* define the security requirements for digital signature devices. Unfortunately, these criteria do not address any interoperability issue between smart cards of different manufacturers, which usually implement digital signature process in still correct but slightly different ways.

To face the interoperability problem, we realized a complete testing environment whose core is the *Crypto Probing System* ©Nestor Lab, an abstract interface to a generic cryptographic smart card, embedding a standard model of the correct card behavior, which can be used to test the digital signature process behavior, also in the presence of alternate or disturbed command sequences, in conjunction with automatic verification techniques such as *model checking*. The framework allows to verify *abstract behavior models* against *real smart cards*, so it can be used to automatically verify the Common Criteria as well as the extended interoperability criteria above and many other low-level constraints. In particular, in this paper we show how we can verify that the card, in the presence of a sequence of (partially) modified commands, rejects them without any side effect, remaining usable, or accepts them, generating a correct final result.

1 Introduction

Starting from 2010, the *European Citizen Card* [1] is going to be developed and distributed to all European citizens. In the next years, tens of millions of smart cards, based on hardware devices provided by many different manufacturers, will be distributed in Europe, and one of the main purposes of such cards will be to provide an easy and safe way for the European citizens to generate and use security data for *electronic authentication* and for *electronic signature*.

Therefore, many European countries have started projects to test and validate such cards before they are put on the market. In particular, the problem has been faced in the Italian National Project for Electronic Identity Card [2], designed by one of the authors of this paper. Among the other issues addressed by this project, there is the very

important problem of the *interoperability* between smart cards produced by different manufacturers.

In this context, the so called *Common Criteria* (ISO/IEC 15408, [3]) and the CWA-1469 [4] standards define the requirements of security devices for digital signature conforming to the annex III of EU directive 1999/93/CE, and the criteria to be followed to verify this conformance. In particular, they define a set of general rules and formats for the microprocessor of a smart card to work correctly as a digital signature device. Similar standards are going to be defined for other security tasks, like the electronic authentication based on smart cards.

The digital signature process, defined by these standards, is coherent and unitary but its implementation is different from a smart card to another.

Indeed, smart cards should generally communicate with their readers using the APDU protocol defined by ISO 7816-4 and 7816-8, but smart card families can implement the APDU protocol in a variety of ways, and the Common Criteria compliancy only certifies that, given the *specific* sequence of commands implemented by the smart card manufacturer, the card generates the required digital signature respecting the security requirements requested by the Common Criteria.

Much responsibility is left to the card readers and client applications, which must have intimate knowledge of the specific command set of the smart card they are communicating with. Observe that one relevant problem is related to *extra-capabilities* added to the software by some weakness in the production process. In particular, one can consider: new commands not included in the standards, the possibility of inappropriately storing relevant information related to the transaction and so on. It is clear that due to the complexity of the production process, which is moreover not standardized, even a trustable producer cannot certify the complete trustability of the software. This makes the signature of the software only a partial solution: "the signature guarantees the origin of the cardlet, not that it is innocuous" [5]. Another possibility is to make an effort to standardize the production process and make use of suitable software engineering approaches such as *Correctness By Construction* [6], [7]. The current situation, described above, makes however this solution not viable.

So we are faced with the problem of setting up a comprehensive verification process able to severely test a given card and its software applications. The task of this verification is to give (at least partial) answers to the many questions not addressed by the standards. What happens if, for example, a card receives a command sequence that is certified as correct for another card? This may happen if the card client cannot adapt to its specific command set, or does not correctly recognize the card. Obviously, this problem may be also extended to a more general context, where the card is deliberately *attacked* using incorrect commands.

Unfortunately, the common criteria do not address any interoperability issue of this kind. This could lead to many problems, especially when designing the card clients, which should potentially embed a driver for each family of card on the market, and be easily upgradable, too. This would be difficult and expensive, and could slow down the diffusion of the smart cards. Therefore, the interoperability problem must be addressed in detail.

1.1 Our Contribution

To face the interoperability problem, we realized a complete testing environment whose core is the *Crypto Probing System* ©Nestor Lab, an abstract interface to a generic cryptographic smart card, which embeds a standard model of the correct card behavior.

The Crypto Probing System can be used to transparently interface with different physical smart cards and test the digital signature process. Indeed, it will be used by the Italian Government to test the card conformance to the Italian security and operational directives for the Electronic Identity Card project.

However, the Crypto Probing System can be also used to test the digital signature process behavior in the presence of alternate or disturbed command sequences. Indeed, in this paper we will use this feature to check a first interoperability issue, i.e., automatically and systematically verify the card behavior when stimulated with unexpected input signals and/or unexpected sequences of commands or parameters. Note that, despite of this simple level of heterogeneity, the common criteria cannot ensure the smart cards interoperability even in this case.

We decided to exploit *model checking techniques* to automate the verification process. Actually we used the tool CMurφ [8], realized by some of the authors with other researchers, which extends Murφ, among others, with the capability of use external C/C++ functions. The model checker will be interfaced to the Crypto Probing System and through it, transparently, to the smart card.

To validate our integrated verification framework, in the present paper we show an experiment used to check that the card STM-Incrypto34 is actually compliant with the extended criteria mentioned above. This verification can be considered as an instance of a whole new kind of testing processes for the digital signature devices.

1.2 Related Works

Although much research is being done in the field of smart card verification[6,9], most of the works in the literature address the more generic problem of (Java) *byte code verification*. In these works, the verification is performed on the applications that should run over the card hardware and operating system, to check if the code respects a set of safety constraints. From our point of view, the major drawback of this approach is that it assumes the correctness of the smart card hardware/firmware and of the Java Virtual Machine which is embedded in the card. Thanks to our abstract machine, the Crypto Probing System, in this work we show how such application code can be tested directly *on the real card*. In this way, we are able to validate both how the code affects the behavior of the card, and how the card hardware can affect the code execution.

Moreover, currently we are interested in verifying the correctness of the smart card output after a sequence of possibly disturbed or unexpected commands, and in particular we aim to use these experiments to address the smart card interoperability issue.

2 Extending the Common Criteria

The Common Criteria [3] define a set of general rules for a smart card-based digital signature process to work *correctly*. However, there are technical aspects that the Common Criteria do not address at all, being too "high level" to analyze issues related to

the card application code. For instance, no requirements are given for the card behavior when an unexpected command is received during the digital signature process. In this case, what should be defined as "the correct behavior" of the card processor?

We considered the following *extended criteria* for a correct behavior:

- if a result can be obtained it is always the correct one; or
- the wrong command is rejected but the card remains usable.

Observe that the second requirement is needed to avoid denial of service attacks or malfunctioning. Here we assume that an error in the command (as opposed to an error in parameters) should not compromise the availability of the card. This assumption can be of course questioned, and this shows the need of more detailed criteria.

2.1 Robustness of the Signature Process

As a first example of extended smart card security property, in this paper we propose the following problem related to the digital signature process.

The digital signature process can be generally split in eleven main steps [4]. At each step, the card expects a particular set of commands and parameters to proceed to the next step.

Now we want to consider possible random disturbances in the commands sent to the smart card. In the presence of such disturbances, at each step, either the invalid command is refused leaving the process unaltered, or the wrong command is accepted but the final result is *identical* to the one obtained with the right command. The existence of *erroneous accepted command* is due to the presence of bits which are *uninfluential* on the parsing of the command syntax.

In this scenario, we want to check the *smart card robustness*. That is, we want to verify that *any* possible disturbance is *correctly* handled by the card, where *correctness* refers to the model discussed above. It is clear that such a verification cannot be performed manually, but requires an automatic framework where the model can be analyzed and exhaustively compared with the behavior or the real smart card hardware. As we will see in Section 4, model checking is a perfect candidate for this task.

3 The Crypto Probing System

The core of a smart card is its *microprocessor*, which contains, on board, a cryptographic processor, a small EEPROM random access memory (≈ 64 KBytes), an operating system and a memory mapped file system.

The microprocessor can execute a restricted set of operations named APDUs (*Application Protocol Data Units*), which are sent from external software applications through a serial communication line.

The standard ISO 7816 part 4, specifies the set of APDU's that any compatible smart card microprocessor must implement. In particular, an APDU consists of a mandatory header of 4 bytes: the Class Byte (*CLA*), the Instruction Byte (*INS*) and two parameter bytes (*P1,P2*), followed by a conditional body of variable length.

However the microprocessor manufacturers develop the APDUs with some deviations from the standards or, in some cases, they create new APDUs not defined by the standards.

Therefore, in order to interface with any kind of card, the client applications should know in advance their command set: no insurance is given that the same APDU sequence will be accepted by all the cards. To investigate this issue, we developed the Crypto Probing System (*CPS*), an executable abstract smart card model, with a simplified set of commands that can act as a *middleware* between the external applications and the real smart cards.

The CPS is able to translate its simplified instructions to the corresponding sequence of APDUs to be sent to the connected physical smart card and to translate the smart card responses in a common format. In this way, the CPS offers a simple interface for testing applications to verify the process correctness and robustness on different physical devices.

The CPS can be invoked via command line, to interactively test the command sequences, or used as a daemon, which stays in execution and accepts commands on TCP/IP connections. The CPS instruction set is the following.

- `reset`: resets the card.
- `start`: initializes the signature process.
- `next`: executes the next command in the process sequence using the correct *cla* and *ins* values and advances to the next step.
- `stay [cla | ins] [r | s] [leave | restore]`: executes the next command in the process sequence, applying a disturbance to its *CLA* or *INS* parameters, but does not advance to the next step. In particular, the `[r | s]` modifiers specify a random (uniform generator `rand48`) or sequential (starting from the current value of the parameter) disturbance, whereas the `[leave | restore]` modifiers tell the CPS to leave the last value or restore the original value of the other non disturbed parameter.
- `accept`: executes the next command in the process sequence using the same values of *CLA* and *INS* of the last `stay` command and advances to the next step.

All the instructions above return:

1. the current hexadecimal node number (01..0B), representing the reached step in the signature process,
2. the values of *CLA* and *INS* sent to the card by the last command,
3. the result code of the command (where a nonzero value indicates an error condition),
4. the overall status of the signature process, i.e.,
 - `TERMINATED` if the card has correctly reached the final step of the process, obtaining the same result as the right sequence,
 - `UNTERMINATED` if the card has not still reached the final step,
 - or `WRONG` if the card has reached the final step but with an incorrect result, that is with a result different from the one of the right sequence.

If the cards behaves accordingly to the extended criteria mentioned above, either a modified command is rejected but the card remains in the current state and is able to reach a `TERMINATED` status, or the card always remains in the `UNTERMINATED` status. Of course, what we want to exclude is the possibility that a *perturbed command generates a different signed document*: this corresponds to never reach the `WRONG` status.

4 The Role of Model Checking

The compliancy of a smart card to the Common Criteria is a testing problem, i.e., it can be certified by manually or semi-automatically by reproducing the context and events described by each criterion and then verifying the expected card behavior.

However, more complex correctness or security properties, like those proposed in this paper (Section 2.1), which work on a lower level, cannot be handled by testing, and require more powerful verification methods. Using model checking it is possible to exhaustively check the compliance of the smart card w.r.t. an *extended* model such as the one described in Section 2.1.

In our context, a *correct* smart card is modeled as a finite automaton which includes the disturbances. This is enough to check the digital signature robustness property, but the model may be extended and enriched to support the verification of almost any property: indeed, model checking is able to deal with very extended systems, having millions of states [10].

Then, the possible disturbances (or - if we think to malicious disturbances - smart card *attacks*) are also modeled within the verifier: i.e., we have a “card model” and “a disturbance model” (or an “attacker model”) that will be run in parallel by the verifier. Finally, the verifier is interfaced with a real smart card (see Section 5 for details).

In this framework, model checking will be used to generate all the possible disturbances (e.g., unexpected commands) that can be carried on the smart card (or, at least, a large subset of such possible disturbances). Then, these actions will be performed on the real smart card, and their results compared with the ones of the correct smart card model. The property to verify is clearly that the real card has the desired correct behavior in any possible situation.

5 Integrating the Crypto Probing System with the CMur φ Model Checker

Having clarified the role of model checking in this extended smart card verification framework, in this Section we describe how this technique has been actually integrated with a smart card system to check the digital signature robustness property. However, as we will see, the presented methodology is very general, so it could be used to verify many other extended smart card properties.

5.1 The CMur φ Model Checker

In this paper we use the CMur φ tool [8]. Of course, our framework could also be used with different model checkers (however, as observed before, they must use explicit algorithms). The most useful aspect of CMur φ is its extension [11], that allows to embed externally defined C/C++ functions in the modeling language. This feature will be used to interface CMur φ with the real smart card device through a suitable interface library.

The state of the system is represented by the set of the *state variables*. Since our system represents the *interaction between the user and the smart card*, we consider the internal state of the smart card (i.e. its current node within the authentication process

and its status) and the possible actions of the user (i.e. the type of disturbance to apply and the node to be disturbed).

In order to describe the evolution of the system, we define three *guarded transition rules*:

- **next**: this rule simply models a normal command sent to the smart card;
- **disturb**: this rule sends a disturbed command to the smart card. Moreover, depending on the returned value, it sends the command `accept` or `next` according to their semantics given in Section 3;
- **end authentication**: this rule is executed at the end of the authentication process and checks the final status of the smart card (i.e. if the signature is correct or not).

Note that the rules are mutual exclusive, thanks to the use of the guards.

Finally, we have to define the *invariant*, that is the property which has to be satisfied in each state of the system. In our case, we want the status of the smart card to be different from `WRONG` during each step of the authentication process.

5.2 The Integrated Framework

In our verification framework, the CMur φ model must be able to access and drive a real smart card.

To this aim, we set up an environment where a smart card is connected to a computer-based host running the Linux operating system. The CPS daemon runs on the same machine and interfaces with the card, whereas a simple TCP/IP connection library, whose functions are exported into the CMur φ model, allows the verifier to talk with the CPS.

At this point, given a model of the digital signature process, we can program CMur φ to exhaustively test the card behavior by simulating all the possible scenarios. In this way, we are able to verify the compliance of the smart card w.r.t. the model.

6 Experimentation

To verify the smart card behavior in the presence of erroneous commands during the digital signature process described in Section 2, we used the CMur φ -based model presented in Section 5 and the framework as described in the following.

Let $s_1, \dots, s_{11}$ be the steps of the digital signature process. Moreover, let $c(s_i)$ be the command that should be sent to the card at the step s_i to correctly proceed to the next step s_{i+1} . Finally, let $disturb(x)$ a function that, given any binary data x , returns it with an added random disturbance.

Then, according to the CMur φ model described in the previous section, the verification procedure is summarized by the algorithm shown in Figure 1.

In a first experiment, we sent *only one* disturbed command in each authentication session, obtaining the results shown in Table 1 (row “Exp. 1”). The smart card behavior was acceptable, since it did not produce any incorrect results.

However, we may note that the 1.9% of the modified commands sent to the smart card was accepted as a correct one and the execution proceeded to the next step in the digital signature process. Apparently, this did not cause any problem, since the final result

```

for i = 1 to 11 {
  for k = 1 to MAX_TESTS {
    Start a new signature session
    for j = 1 to i-1
      send  $c(s_j)$  to the card
      /* now we are at step  $s_i - 1$  */
      let  $cr = \text{disturb}(c(s_j))$ 
      send  $cr$  to the card
      let resp = the current card status
      if (resp == Error) /* we are still at step  $s_i - 1$  */
        send  $c(s_j)$  to the card
        /* otherwise the disturbed command has been accepted, so we are at
           step  $s_i$  */
      for j = i+1 to 11
        send  $c(s_j)$  to the card
      /* now we should be at the final step */
      verify the card output validity
  }
}

```

Fig. 1. Experiment 1: one disturbed command in each authentication session

was correct, but leaves some doubts about the card software. We may suppose that the card does not support only one digital signature process, but several variants triggered by alternate commands in some steps. In the worst case, however, these modified commands may create a “hidden damage” to the card, that may show its consequences only later.

To further investigate this issue, as second experiment, we stressed the smart card in a more intensive way. Namely, we sent a disturbed command *at each step* of the authentication process obtaining the results shown in row “Exp. 2” of Table 1.

Table 1. Experimental results

	Total # of commands sent	# of modified commands	# of rejected modified commands	# of accepted modified commands	# of incorrect results
Exp. 1	99396	9036	8864 (98.1%)	172 (1.9%)	0
Exp. 2	11000	9000	8907 (98.7%)	93 (1.3%)	0

Again, the card has a correct behavior, but continues to accept a small percentage of modified commands as correct. However, an analysis of the reasons of this strange behavior is beyond the scope of the present paper. With this last experiment, we achieved our aims, showing that the card under analysis is robust with respect to any altered command sequence.

Observe that the full interaction between the model checker and the smart card, related to a single command, requires on the average 0.8 seconds. Thus, the two experiments took about 22 hours and 2 hours, respectively. However, to accelerate more complex verification tasks, we plan to make use of distributed verification architectures, which have been already developed for the Murphi verifier [12].

7 Conclusions

In this work, we have shown an integrated environment to perform the verification of smart card based signature devices, w.r.t. models of correct behavior which can be much

more detailed than those considered in the Common Criteria. Since this verification task goes beyond simple black box testing we integrated a model checker in the verification environment.

We tested a commercial signature device, systematically targeted with wrong commands while executing the signature of a fixed document. While the card has shown a correct behavior, w.r.t. a reasonable model, even in this simple experimentation it has been possible to point out some anomalous behavior such as the acceptance of wrong commands. Many other verifications can be performed on smart cards using our integrated framework, addressing aspects not covered by the common criteria: for instance, currently we do not know if and how a card microprocessor would react to *concurrent* signing sessions.

Beyond such specific verifications, our general objective is to set up a whole family of behavioral models - defining in a (hopefully) complete way the correct behavior of a card-based digital signature device, as well as a verification environment able to prove or, at least to give strong evidence, that the system is compliant w.r.t. all models.

We think that if this task is accomplished, this will be a very relevant step towards the solution of the interoperability problem, as the cards so certified would perform well even in the most challenging situations.

References

1. CEN: TC224 WG15
2. D.M. S.O. n. 229 della G.U. 261 del 9/11/2007 Regole tecniche della Carta d'identità elettronica (Technical Rules for the Eletronic Identity Card) (November 8, 2007)
3. Common Criteria for Information Technology Security Evaluation (September 2006)
4. CEN WORKSHOP AGREEMENT: Cwa 14169 (March 2004)
5. Leroy, X.: Computer security from a programming language and static analysis perspective. In: Degano, P. (ed.) ESOP 2003. LNCS, vol. 2618, pp. 1–9. Springer, Heidelberg (2003)
6. Toll, D.C., Weber, S., Karger, P.A., Palmer, E.R., McIntosh, S.K.: Tooling in Support of Common Criteria Evaluation of a High Assurance Operating System. IBM Thomas J. Watson Research Center Report (2008)
7. Chapman, R.: Correctness by construction: a manifesto for high integrity software. In: SCS 2005: Proceedings of the 10th Australian workshop on Safety critical systems and software, pp. 43–46. Australian Computer Society, Inc. (2006)
8. CMurphi Web Page,
<http://www.di.univaq.it/gdellape/murphi/cmurphi.php>
9. Michael, C., Radosevich, W.: Black box security testing tools. Cigital (2005)
10. Della Penna, G., Intrigila, B., Melatti, I., Tronci, E., Venturini Zilli, M.: Integrating ram and disk based verification within the Mur ϕ verifier. In: Geist, D., Tronci, E. (eds.) CHARME 2003. LNCS, vol. 2860, pp. 277–282. Springer, Heidelberg (2003)
11. Della Penna, G., Intrigila, B., Melatti, I., Minichino, M., Ciancamerla, E., Parisse, A., Tronci, E., Venturini Zilli, M.: Automatic verification of a turbogas control system with the Mur ϕ verifier. In: Maler, O., Pnueli, A. (eds.) HSCC 2003. LNCS, vol. 2623, pp. 141–155. Springer, Heidelberg (2003)
12. Melatti, I., Palmer, R., Sawaya, G., Yang, Y., Kirby, R.M., Gopalakrishnan, G.: Parallel and distributed model checking in eddy. In: Valmari, A. (ed.) SPIN 2006. LNCS, vol. 3925, pp. 108–125. Springer, Heidelberg (2006)

Generating a Large Prime Factor of $p^4 \pm p^2 + 1$ in Polynomial Time

Maciej Grześkowiak*

Adam Mickiewicz University,
Faculty of Mathematics and Computer Science,
Umultowska 87, 61-614 Poznań, Poland
maciejg@amu.edu.pl

Abstract. In this paper we present a probabilistic polynomial-time algorithm for generating a large prime p such that $\Phi_m(p^2)$ has a large prime factor, where $\Phi_m(x)$ is the m -th cyclotomic polynomial and $m = 3$ or $m = 6$. An unconditionally polynomial time algorithm for generating primes of the above form is not yet known. Generating primes of such form is essential for the GH and the CEILIDH Public Key Systems, since they are key parameters in these cryptosystems.

Keywords: The Gong-Harn Public Key System, CEILIDH Public Key System, Torus-Based Cryptography, primes of a special form.

1 Introduction and Background

Many new cryptosystems have been introduced in recent years which require generating primes of special forms as key parameters. For instance of interest is generating of a large prime p such that $\Phi_m(p^k)$ is divisible by a large prime q , where k is a fixed positive integer and $\Phi_m(x)$ is the m -th cyclotomic polynomial. From the security point of view it is essential to find a prime p such that $m \log p^k \approx 2048$ to obtain a level of security equivalent to factoring a positive integer having 2048 bits. The prime q should have at least 160 bits to make solving DLP in subgroup of order q of $\mathbf{F}_{p^k}^*$ impossible in practice. For $m = 3$, in 1998 Gong and Harn presented a public key system called GH [4], [5]. In 2003 Rubin and Silverberg introduced the idea of Torus-Based Cryptography [10]. In particular, they proposed a public key system called CEILIDH, which requires the generation of special primes p, q for $m = 6$. There exist two main approaches for generating primes of the above form. The first approach was proposed by Gong and Giuliani [6]. The second approach for generating desired primes was proposed by Lenstra and Verheul [9]. We next give an illustration of this algorithm in the case $m = 3$ and $k = 1$. The algorithm randomly selects a prime $q \equiv 7 \pmod{12}$ and computes r_i for $i = 1, 2$ roots of $\Phi_6(x) = x^2 - x + 1 \pmod{q}$. Alternatively the algorithm finds a positive integer r_3 such that $\Phi_6(r_3) = q$ is a prime. Next the algorithm selects a prime p such that $p \equiv r_i \pmod{q}$ for one

* Supported by Ministry of Science and Higher Education, grant N N201 1482 33.

of r_i $i = 1, 2, 3$ (from this q divides $\Phi_6(p)$). It is worth pointing out that the above algorithm works perfectly well in practice. However in the general case, we can encounter some problems. For example let $k = 2$. Consider the naive way of computing the root of the polynomial $\Phi_6(x^2) = x^4 - x^2 + 1 \pmod{q}$, where q is a prime. Substituting $y = x^2$ we reduce the degree of $\Phi_6(x^2) \pmod{q}$ to 2. Next we compute y_1, y_2 roots of $y^2 - y + 1 \pmod{q}$, which requires computing $\sqrt{-3} \pmod{q}$. In the end we compute the square root $\pmod{q}$ of y_i for $i = 1$ or $i = 2$. There are two difficulties, which we can encounter in practice while computing the roots of $\Phi_6(x^2) \pmod{q}$. The first is that we have to use algorithm to compute the square root $\pmod{q}$. Computing of the square root $\pmod{q}$ is basically simple, except for the case where $q \equiv 1 \pmod{8}$, which takes at most $O(\log^4 q)$ [2]. The second problem lies in the handling of square roots when these are not in $\mathbf{F}_q$ (they are then in $\mathbf{F}_{q^2}$ and $\mathbf{F}_{q^4}$ respectively). However the alternative method in the abovementioned algorithm, involving finding a positive integer r_3 such that $\Phi_6(r_3) = q$ is a prime, causes theoretical problem. We do not know if there exist infinitely many primes of the form $\Phi_6(r_3)$. This is an extremely hard, still unproven mathematical problem. The second part of the algorithm also seems problematic. When the modulus q is close to x there are not sufficiently many primes $p \leq x$, to warrant the equidistribution among the residue classes $a \pmod{q}$. To be more precise, let $\pi(x; a, q)$, $1 \leq a \leq q$, $(a, q) = 1$, denote the number of primes $p \equiv a \pmod{q}$ with $p \leq x$. By the Siegel-Walfisz theorem [3] we get that for any fixed $N > 0$, the formula $\pi(x; a, q) = x/(\phi(q) \log x) \{1 + o(1)\}$ holds uniformly throughout the range $q \leq (\log x)^N$ and $(a, q) = 1$. Therefore we cannot apply Siegel-Walfisz theorem to estimate the running time of the second procedure, when q is close to p . Analysis and theoretical estimation of computational complexity of the Lenstra and Verheul algorithm under the assumption of some unproven conjectures can be found in [7]. However an unconditionally polynomial time algorithm for generating desired primes p and q is not yet known.

In this paper we present a new probabilistic algorithm for generating large primes p and q such that $q|\Phi_6(p^2)$ or $q|\Phi_3(p^2)$, which is faster than those previously considered. We prove that the algorithm for finding such a primes is random and executes in polynomial time. We also present the developments and improvements of ideas proposed by Lenstra and Verheul. In particular, we improve the method of finding root of polynomials $\Phi_m(x^2) \pmod{q}$, where $m = 3, 6$ and q is a prime, by reducing the number of computed square roots. Our method require computing only $\sqrt{3} \pmod{q}$ in order to find the root of $\Phi_m(x^2) \pmod{q}$, which is a big improvement over the Lenstra-Verheul method [9]. Achieving the described goals is made possible by generating a prime q , which is a value of a primitive quadratic polynomial of two variables with integer coefficients. We prove that the procedure for finding such prime is random and executes in polynomial time. Moreover we prove Lemma 2, which is slightly weaker than the above Siegel-Walfisz result, but can be applied to estimate computational complexity of finding prime $p \equiv a \pmod{q}$, where p is close to q . Therefore we can prove that our algorithm executes in polynomial time.

2 Generating a Large Prime Factor of $\Phi_m(p^2)$

Our algorithm consists of two procedures. Let us fix $F(x, y) = 144x^2 + 144y^2 + 24y + 1 \in \mathbb{Z}[x, y]$. The first procedure generates positive integers $a \in \left[\frac{n}{12\sqrt{2}}, \frac{cn}{12\sqrt{2}} \right]$ and $b \in \left[\frac{n-\sqrt{2}}{12\sqrt{2}}, \frac{cn-\sqrt{2}}{12\sqrt{2}} \right]$ such that $F(a, b) = q$ is a prime, where $n \in \mathbb{N}$ and c is some positive number. The second procedure computes $r \pmod{q}$ and next finds a positive integer $k \in \left[1, \left\lceil \frac{n^6-r}{q} \right\rceil \right]$ such that the number $qk + r$ is prime.

Algorithm 1. Generating primes p and q , such that $q | \Phi_m(p^2)$ and $m = 3, 6$

```

1: procedure FINDPRIMEQ( $n, F(x, y)$ ) ▷ Input  $n$  and  $F(x, y)$ 
2:    $q \leftarrow 1$ 
3:   while not IsPrime( $q$ ) do
4:      $a \leftarrow \text{Random}(n)$  ▷ Randomly select  $a \in \left[ \frac{n}{12\sqrt{2}}, \frac{cn}{12\sqrt{2}} \right]$ 
5:      $b \leftarrow \text{Random}(n)$  ▷ Randomly select  $b \in \left[ \frac{n-\sqrt{2}}{12\sqrt{2}}, \frac{cn-\sqrt{2}}{12\sqrt{2}} \right]$ 
6:      $q \leftarrow F(a, b)$ 
7:   end while
8:   return ( $a, b, q$ )
9: end procedure

10: procedure FINDPRIMEPMODULOQ( $a, b, q, m$ ) ▷ Input  $a, b, q$  and  $m$ 
11:    $r \leftarrow (\sqrt{3}(12b + 1) - 12a)(-2(12b + 1))^{-1} \pmod{q}$ 
12:   if  $m = 3$  then
13:      $r \leftarrow -r$ 
14:   end if
15:    $p \leftarrow 1$ 
16:   while not IsPrime( $p$ ) do
17:      $k \leftarrow \text{Random}(n)$  ▷ Randomly select  $k \in \mathbb{N}, k \in \left[ 1, \left\lceil \frac{n^6-r}{q} \right\rceil \right]$ 
18:      $p \leftarrow qk + r$ 
19:   end while
20:   return ( $p$ )
21: end procedure

22: return ( $p, q$ )

```

Theorem 1. Let us fix $m = 3$ or $m = 6$. Then Algorithm 1 generates primes p and q such that q divides $\Phi_m(p^2)$. Moreover $q = F(a, b) = N(\gamma)$ and $\Phi_m(p^2) = N(\xi)$, where $\gamma, \xi \in \mathbb{Z}[i]$, $\gamma \mid \xi$ and $\gamma = 12a + (12b + 1)i$ and $\xi = (p^2 - 1) + pi$.

Proof. Let $\mathbb{Z}[i] = \{x + yi : x, y \in \mathbb{Z}, i = \sqrt{-1}\}$. Let $\mathbb{Q}(i)$ be the corresponding quadratic number field with the ring of integers $\mathbb{Z}[i]$. Let $\alpha \in \mathbb{Z}[i]$. We denote by $N(\alpha) = x^2 + y^2$ the norm of α relative to $\mathbb{Q}$. Assume that the procedure FINDPRIMEQ finds positive integers a, b such that $F(a, b) = q$ is prime. Then there exists $\gamma = 12a + (12b + 1)i \in \mathbb{Z}[i]$ such that $F(a, b) = (12a)^2 - ((12b + 1)i)^2 =$

$N(\gamma)$. Let $\xi \in \mathbf{Z}[i]$, $\xi = (p^2 - 1) + pi$, where p is a prime. We have $N(\xi) = \Phi_6(p^2)$. Assume that γ divides ξ . Then there exists $\delta \in \mathbf{Z}[i]$, $\delta = x + yi$, $x, y \in \mathbf{Z}$ such that

$$\gamma\delta = (12a + (12b + 1)i)(x + yi) = (p^2 - 1) + p = \xi, \quad (1)$$

and

$$N(\gamma)N(\delta) = N(\xi) = \Phi_6(p^2). \quad (2)$$

We show how one can find elements δ , $\xi \in \mathbf{Z}[i]$, and a prime p satisfying (1). By (1) it follows that

$$\begin{cases} 12ax - (12b + 1)y = p^2 - 1 \\ (12b + 1)x + 12ay = p, \end{cases} \quad (3)$$

where $12a$, $12b + 1$ are given. Squaring the second equation and substituting to the first one we get

$$Ax^2 + Bxy + Cy^2 + Dx + Ey + 1 = 0, \quad (4)$$

where

$$A = -(12b + 1)^2, \quad B = -2(12a)(12b + 1), \quad (5)$$

$$C = -(12a)^2, \quad D = 12a, \quad E = -(12b + 1). \quad (6)$$

Now we find solutions of (4). We write $\Delta = B^2 - 4AC$. Trivial computation show that $\Delta = 0$. Multiplying (4) by $2A$ we obtain $(2Ax + By)^2 + 4ADx + 4AEy + 4A = 0$. Let $2Ax + By = T$ then $T^2 + 2(2AE - BD)y + 4A + 2DT = 0$ and $(T + D)^2 = 2(BD - 2AE)y + D^2 - 4A$. Consequently equation (4) is equivalent to

$$(2Ax + By + D)^2 + \alpha y = \beta, \quad (7)$$

where

$$\alpha = 2(BD - 2AE) = 4(12b + 1)((12a)^2 + (12b + 1)^2) = -4(12b + 1)q \quad (8)$$

and

$$\beta = D^2 - 4A = (12a)^2 + 4(12b + 1)^2. \quad (9)$$

Let

$$X = 2Ax + By + D, \quad Y = -\alpha y. \quad (10)$$

By (7)

$$X^2 - \beta = Y, \quad (11)$$

we see that a necessary condition for existence of integers solution of (7) is solubility of the congruence

$$Z^2 \equiv \beta \pmod{\alpha}. \quad (12)$$

Let z_0 be the solution of (12). From (8) and (9) it follows that

$$\begin{aligned} z_0 &\equiv 0 \pmod{4} \\ z_0 &\equiv 12a \pmod{(12b+1)} \\ z_0 &\equiv \sqrt{3}(12b+1) \pmod{q}. \end{aligned} \quad (13)$$

Since $q \equiv 1 \pmod{3}$ then 3 is quadratic residue modulo q and, in consequence, $z_0 \pmod{\alpha}$ exists. It can be easily found by the Chinese Remainder Theorem. By (10), (11) we have $y = (z_0^2 - \beta)/(-\alpha)$, $y \in \mathbf{N}$. Now we prove that in this case x is integer as well. By (10) we have

$$z_0 - D = 2Ax + By \quad (14)$$

Since $q = F(a, b) = (12a)^2 + (12b+1)^2$ is a prime then $(2A, B) = 2(12b+1)$. Hence $z_0 - D \equiv 0 \pmod{2(12b+1)}$ and so (14) has integer solutions. Consequently, solutions of (7) are integers. This observation works for general solutions of (12) $z \equiv z_0 \pmod{\alpha}$. Our computation shows that integers solutions x, y of (7) have the form

$$x = \frac{z - By - D}{2A}, \quad y = \frac{z^2 - \beta}{-\alpha}, \quad x, y \in \mathbf{Z}, \quad (15)$$

where $z = \alpha t + z_0$, $t \in \mathbf{Z}$. Substituting the above x to the second equation of (3) we obtain

$$(12b+1) \left(\frac{\alpha t + z_0 - D}{2A} \right) + \left(12a - \frac{B(12b+1)}{2A} \right) y = p$$

Since $(12a - ((12b+1)B)/2A)y = 0$ then putting (8), (5) we get

$$2qt + \frac{z_0 - 12a}{-2(12b+1)} = p, \quad t \in \mathbf{Z}.$$

Hence

$$p \equiv (z_0 - 12a)(-2(12b+1))^{-1} \pmod{q}$$

and consequently by (13)

$$p \equiv (\sqrt{3}(12b+1) - 12a)(-2(12b+1))^{-1} \pmod{q}, \quad (16)$$

Taking (compare steps 11-14 of procedure FINDPRIMEPMODULOQ)

$$r = (\sqrt{3}(12b+1) - 12a)(-2(12b+1))^{-1} \pmod{q}$$

then from (2) and (16) we get

$$\Phi_6(p^2) \equiv \Phi_6(r^2) \equiv 0 \pmod{q}.$$

Therefore if we find prime p in the arithmetic progression $p \equiv r \pmod{q}$ (compare steps 16-20 of procedure FINDPRIMEPMODULOQ), then $q|\Phi_6(p^2)$. Since $\Phi_6(r) = \Phi_3(-r)$ then if we find $p \equiv -r \pmod{q}$, then $q|\Phi_3(p^2)$. This finishes the proof.

3 Run-Time Analysis of the Algorithm

Let us adopt the standard notation used in the theory of primes. We denote by $\pi(x, q, a)$ the number of primes $p \equiv a \pmod{q}$ not exceeding x , where $x \geq 1$, $a, q \in \mathbf{N}$, $1 \leq a \leq q$, $(a, q) = 1$. We write also $\pi(x)$ in place of $\pi(x, 1, 1)$. Moreover we write

$$\psi(x; q, a) = \sum_{\substack{n \leq x \\ n \equiv a \pmod{q}}} \Lambda(n),$$

where

$$\Lambda(n) = \begin{cases} \log p, & \text{if } n = p^k \\ 0, & \text{otherwise} \end{cases}$$

With the notation as above we recall some theorems which are related to distributions of primes.

Theorem 2 (de la Vallée Poussin). *For some positive number A*

$$\pi(x) = \text{li } x + O(x \exp(-A\sqrt{\log x})).$$

Proof. see [3].

Theorem 3 (Bombieri-Vinogradov). *Let $A > 0$ be fixed. Then*

$$\sum_{q \leq Q} \max_{y \leq x} \max_{\substack{a \\ (a, q) = 1}} \left| \psi(y; q, a) - \frac{y}{\phi(q)} \right| \ll x^{\frac{1}{2}} Q (\log x)^5,$$

provided that $x^{\frac{1}{2}} (\log x)^{-A} \leq Q \leq x^{\frac{1}{2}}$.

Proof. see [3].

Theorem 4 (Iwaniec). *Let $P(x, y) = ax^2 + bxy + cy^2 + ex + fy + g \in Z[x, y]$, $\deg P = 2$, $(a, b, c, e, f, g) = 1$, $P(x, y)$ be irreducible in $\mathbf{Q}[x, y]$, represent arbitrary large number and depend essentially on two variables. Then $\frac{N}{\log N} \ll \sum_{\substack{q \leq N \\ q = P(x, y)}} 1$, if $D = af^2 - bef + ce^2 + (b^2 - 4ac)g = 0$ or $\Delta = b^2 - 4ac$ is a perfect square.*

Proof. see [8].

3.1 Analysis of the Procedure FINDPRIMEQ

We denote by $\mathcal{PT}$ the number of bit operations necessary to carry out the deterministic primality test [1]. For simplicity, assume that $\mathcal{PT} \gg \log^4 n$.

Theorem 5. *Let $F(x, y) = 144x^2 + 144y^2 + 24y + 1 \in \mathbb{Z}[x, y]$. Then there exist constants c and $b_0 = b_0(c, n)$, n_0 such that for every integer $n \geq n_0$ and an arbitrary real $\lambda \geq 1$, the procedure FINDPRIMEQ finds $a \in \left[\frac{n}{12\sqrt{2}}, \frac{cn}{12\sqrt{2}}\right]$ and $b \in \left[\frac{n-\sqrt{2}}{12\sqrt{2}}, \frac{cn-\sqrt{2}}{12\sqrt{2}}\right]$ such that $q = F(a, b)$ is a prime, $q \in [n^2, (cn)^2]$, with probability greater than or equal to $1 - e^{-\lambda}$ after repeating $\lfloor b_0 \lambda \log n \rfloor$ steps 3–7 of the procedure. Every step of the procedure takes no more than $\mathcal{PT}$ bit operations.*

Proof. We start with an estimate for the number of primes in the interval $[n^2, (cn)^2]$ which are of the form $F(a, b)$, where $F(x, y) = 144x^2 + 144y^2 + 24y + 1 \in \mathbb{Z}[x, y]$. We apply the Theorem 4. We say that F depends essentially on two variables if $\partial F/\partial x$ and $\partial F/\partial y$ are linearly independent. We use the Lemma

Lemma 1. *Let $F(x, y) = ax^2 + bxy + cy^2 + ex + fy + g \in \mathbb{Z}[x, y]$, $\Delta = b^2 - 4ac$, $\alpha = bf - 2ce$, $\beta = be - 2af$. Then $\partial F/\partial x$ and $\partial F/\partial y$ are linearly dependent if and only if $\Delta = \alpha = \beta = 0$.*

Proof. see [8]

Since $\Delta = 288^2$ and $(144, 144, 24, 1) = 1$ then $F(x, y)$ satisfies assumptions of Theorem 4. We define the set

$$\mathcal{Q} = \{n^2 \leq q \leq (cn)^2 : F(x, y) = q - \text{prime}, x, y \in \mathbb{N}\},$$

where $c > 0$. Denote by $|\mathcal{Q}|$ the number of the elements of $\mathcal{Q}$. Since Δ is a perfect square then by Theorem 4 there exists $c_0 > 0$ such that $|\mathcal{Q}| \geq (c_0(cn)^2)(2 \log n)^{-1} - \pi(n^2)$. By Theorem 2 and the above there exists c_1 such that for sufficiently large n we have

$$|\mathcal{Q}| \geq c_1 \frac{n^2}{\log n} + O\left(\frac{n^2}{\log^2 n}\right), \quad (17)$$

where $c_1 = (c_0 c^2 - 1)/2$ with $c \geq \sqrt{3/c_0}$. Denote by A_F the event that a randomly chosen pair of natural numbers a and b satisfying

$$a \in \left[\frac{n}{12\sqrt{2}}, \frac{cn - \sqrt{2}}{12\sqrt{2}}\right], \quad b \in \left[\frac{n - \sqrt{2}}{12\sqrt{2}}, \frac{cn - \sqrt{2}}{12\sqrt{2}}\right]$$

is such that the number $F(a, b) \in [n^2, (cn)^2]$ is a prime. Hence by (17) there exists $c_2 = c_1 - \varepsilon(n)$, where $\varepsilon \rightarrow 0$ as $n \rightarrow \infty$ such that for sufficiently large n , the probability that in l trials A_F does not occur is

$$\left(1 - \frac{c_2}{\log n}\right)^l = \exp\left(l \log\left(1 - \frac{c_2}{\log n}\right)\right) \leq \exp\left(\frac{-c_2 l}{\log n}\right) \leq e^{-\lambda}$$

for an arbitrary real $\lambda \geq 1$ and $l = b_0 \lambda \log n$, where $b_0 = c_2^{-1}$. Hence the probability that in l trials A_F does occur is greater or equal to $1 - e^{-\lambda}$. So after repeating $[b_0 \lambda \log n]$ steps, the procedure finds integers a and b and primes $q = F(a, b)$ with probability greater than or equal to $1 - e^{-\lambda}$. The most time-consuming step of the algorithm is the deterministic primality test for number q which takes no more than $\mathcal{PT}$ operations. This finished the proof.

3.2 Analysis of the Procedure FINDPRIMEPMODULOQ

Theorem 6. *Let q be the output of the procedure FINDPRIMEQ. The procedure FINDPRIMEPMODULOQ with the input consisting of the prime q and a, b has the following properties. There exists b_1 and n_1 such that for every integer $n \geq n_1$ and an arbitrary real $\lambda \geq 1$, the procedure finds a positive integer $k \in \left[1, \left\lceil \frac{n^6 - r}{q} \right\rceil\right]$ such that $p = qk + r$ is prime, $q \ll p \ll n^6$, with probability greater than or equal to $1 - e^{-\lambda}$ after repeating $[b_1 \lambda \log n]$ steps of the procedure with the possible exception of at most $O(n^2(\log n)^{-C_0-1})$ values of q . Every step of the procedure takes no more than $\mathcal{PT}$ bit operations.*

Proof. We use the lemma

Lemma 2. *Let $q \leq (cn)^2$ be a positive integer. Then there exist constants $0 < C_0 < B < 1$ and n_0 such that for every $n > n_0$ and for all residue classes $a \pmod{q}$*

$$\pi(n^6; q, a) = \frac{n^6}{6\phi(q) \log n} + O\left(\frac{n^6}{\phi(q)(\log n)^{B-C_0+1}}\right)$$

with the possible exception of at most $O(n^2(\log n)^{-C_0-1})$ values of q .

Proof. See section 3.3

Denote by A_p the event that a randomly chosen positive integer $k \in \left[1, \frac{n^6 - r}{q}\right]$ is such that the number $qk + r$ is a prime. It follows by Lemma 2 that there exist $0 < C_0 < B < 1$ and n_1 such that for every $n > n_1$ we have $A_p \geq (6 \log n)^{-1} + O((\log n)^{-B+C_0-1})$ for all q with the possible exception of at most $O(n^2(\log n)^{-C_0-1})$ values of q . Hence there exists $c_1 = \frac{1}{6} - \varepsilon(n)$, where $\varepsilon \rightarrow 0$ as $n \rightarrow \infty$ such that for sufficiently large n the probability that in l trials A_p does not occur is

$$\left(1 - \frac{c_1}{\log n}\right)^l = \exp\left(l \log\left(1 - \frac{c_1}{\log n}\right)\right) \leq \exp\left(\frac{-lc_1}{\log n}\right) \leq e^{-\lambda}$$

for an arbitrary real $\lambda \geq 1$ and $l = b_1 \lambda \log n$, where $b_1 = c_1^{-1}$. Hence the probability that in l trials A_P does occur is greater than or equal to $1 - e^{-\lambda}$. So after repeating $[b_1 \lambda \log n]$ steps, the procedure finds a positive integer k such that $p = qk + r$ is prime with probability greater than or equal to $1 - e^{-\lambda}$ for all q with the possible exception of at most $O(n^2(\log n)^{-C_0-1})$ values of q . The most time-consuming step of the algorithm is the deterministic primality test for number p which takes no more than $\mathcal{PT}$ operations. This finishes the proof.

3.3 Proof of Lemma 2

Proof. We apply Theorem 3 with $x = n^6$ and $A = 2B + 6$, $0 < B < 1$

$$\sum_{q \ll n^3 (\log n)^{-2B-6}} \max_{y \leq n^6} \max_{\substack{a \\ (a,q)=1}} \left| \psi(y; q, a) - \frac{y}{\phi(q)} \right| \ll \frac{n^6}{(\log n)^{2B+1}}. \quad (18)$$

Let

$$\tilde{\mathcal{Q}} = \left\{ q \leq (cn)^2 : \max_{n^6 (\log n)^{-C} \leq y \leq n^6} \max_{\substack{a \\ (a,q)=1}} \left| \psi(y; q, a) - \frac{y}{\phi(q)} \right| \geq \frac{y}{\phi(q) (\log n)^B} \right\},$$

where $C > 0$. Then

$$\frac{n^6}{(\log n)^{2B+1}} \geq \sum_{q \in \tilde{\mathcal{Q}}} \frac{y}{\phi(q) (\log n)^B} \gg \frac{n^6}{(\log n)^{B+C}} \sum_{q \in \tilde{\mathcal{Q}}} \frac{1}{\phi(q)} \gg \frac{n^6 |\tilde{\mathcal{Q}}|}{n^2 (\log n)^{B+C}}.$$

Hence

$$|\tilde{\mathcal{Q}}| \ll \frac{n^2}{(\log n)^{B-C+1}} = \frac{n^2}{(\log n)^{C_0+1}}, \quad (19)$$

where $C = B - C_0$ and $B < 2C_0$. Consequently

$$\max_{\substack{n^6 \\ (\log n)^{B-C_0} \leq y \leq n^6}} \max_{\substack{a \\ (a,q)=1}} \left| \psi(y; q, a) - \frac{y}{\phi(q)} \right| \leq \frac{n^6}{(\log n)^B}.$$

and

$$\psi(y; q, a) = \frac{y}{\phi(q)} (1 + O((\log n)^{-C_0})) \quad (20)$$

for all reduced residue classes $a \pmod{q}$, and for all $q \leq (cn)^2$ with the possible exception of at most $O(n^2 (\log n)^{-C_0-1})$ values of q . We have

$$\begin{aligned} \pi(n^6; q, a) &= \sum_{\substack{m \leq n^6 \\ m \equiv a \pmod{q}}} \frac{\Lambda(m)}{\log m} - \sum_{t \geq 2} \sum_{\substack{p^t \leq n^6 \\ p^t \equiv a \pmod{q}}} \frac{1}{t} \\ &= \sum_{\substack{m \leq n^6 \\ m \equiv a \pmod{q}}} \frac{\Lambda(m)}{\log m} + O\left(\frac{n^3}{\log n}\right) \end{aligned}$$

By (20) and Abel's summation formula

$$\begin{aligned} \sum_{\substack{2 \leq m \leq n^6 \\ m \equiv a \pmod{q}}} \frac{\Lambda(m)}{\log m} &= \frac{\psi(n^6; q, a)}{6 \log n} + \int_2^{n^6} \frac{\psi(y; q, a) dy}{y \log^2 y} = \frac{\psi(n^6; q, a)}{6 \log n} + J_1 + J_2 \\ &= \frac{n^6}{6 \phi(q) \log n} + O\left(\frac{n^6}{\phi(q) (\log n)^{C_0+1}}\right) + J_1 + J_2, \end{aligned}$$

where

$$J_1 = \frac{n^6}{(\log n)^{B-C_0}} \int_2^{\frac{n^6}{(\log n)^{B-C_0}}} \frac{\psi(y; q, a) dy}{y \log^2 y}, \quad J_2 = \int_{\frac{n^6}{(\log n)^{B-C_0}}}^{n^6} \frac{\psi(y; q, a) dy}{y \log^2 y}.$$

Since

$$\psi(y; q, a) = \sum_{\substack{2 \leq m \leq y \\ m \equiv a \pmod{q}}} \Lambda(m) \ll \log y \sum_{\substack{2 \leq m \leq y \\ m \equiv a \pmod{q}}} 1 \ll \frac{y \log y}{q} + O(\log y)$$

Hence there exists n_1 such that for every positive integer $n \geq n_1$

$$J_1 \ll \frac{1}{q} \int_2^{\frac{n^6}{(\log n)^{B-C_0}}} \frac{dy}{\log y} + \int_2^{\frac{n^6}{(\log n)^{B-C_0}}} \frac{dy}{y \log y} \ll \frac{1}{q} \frac{n^6}{(\log n)^{B-C_0+1}}.$$

By (20) there exists n_2 such that for every positive integer $n \geq n_2$

$$J_2 \ll \frac{1}{\phi(q)} \int_{\frac{n^6}{(\log n)^{B-C_0}}}^{n^6} \frac{dy}{\log^2 y} + \frac{1}{\phi(q)} \int_{\frac{n^6}{(\log n)^{B-C_0}}}^{n^6} \frac{dy}{(\log y)^{C_0+2}} \ll \frac{n^6}{\phi(q) \log^2 n}.$$

This finishes the proof.

References

1. Agrawal, M., Kayal, K., Saxena, N.: Primes is P. *Ann. of Math.* 160, 781–793 (2004)
2. Cohen, H.: *A Course in Computational Algebraic Number Theory*. Springer, New York (1993)
3. Davenport, H.: *Multiplicative Number Theory*. Springer, New York (1980)
4. Gong, G., Harn, L.: Public-Key Cryptosystems Based on Cubic Finite Field Extension. *IEEE Transactions on Information Theory* 45, 2601–2605 (1999)
5. Gong, G., Harn, L.: A New Approach on Public-key Distribution. In: *Proceedings of China - Crypto*, Chengdu, China, pp. 50–55 (1998)
6. Giuliani, K., Gong, G.: Generating Large Instances of the Gong-Harn Cryptosystem. In: *Proceedings of Cryptography and Coding: 8th International Conference Cirencester*. LNCS, vol. 2261, pp. 111–133. Springer, Heidelberg (2002)
7. Grześkowiak, M.: Analysis of Algorithms of Generating Key Parameters for the XTR Cryptosystem. In: *Proceedings of Wartacrypt 2004*, pp. 1–12. Tatra Mountains Mathematical Publications (2006)
8. Iwaniec, H.: Primes Represented by Quadratic Polynomials in Two Variables. *Acta Arith.* 24, 435–459 (1974)
9. Lenstra, A.K., Verhuel, E.R.: The XTR Public Key System. In: Bellare, M. (ed.) *CRYPTO 2000*. LNCS, vol. 1880, pp. 1–19. Springer, Heidelberg (2000)
10. Rubin, K., Silverberg, A.: Torus-based cryptography. In: Boneh, D. (ed.) *CRYPTO 2003*. LNCS, vol. 2729, pp. 349–365. Springer, Heidelberg (2003)

ODBASE 2008 PC Co-chairs' Message

Welcome to the proceedings of the 7th International Conference on Ontologies, Databases, and Applications of Semantics (ODBASE 2008) held in Monterrey, Mexico, November 11 – 13, 2008.

We are now moving towards meaningful Internet systems and ubiquitous computing, earmarking the significant transitions of semantic technologies in the years since the first ODBASE conference in 2002. Recent methods allow us to scale semantic technologies to handle trillions of triples; to compose intriguing semantic applications within a few days; to address target applications from science up to e-commerce and e-health; and to develop or choose among plenty of ontologies and RDF stores, inferencing engines, ontology mapping, maintaining, and evaluation systems, to name but a few.

The ODBASE conferences provide a forum for the sharing of original research results and practical development experiences in the areas of ontologies, databases, and applications of data semantics. This year's conference included technical sessions organized around such important subtopics as semantic matching and similarity measuring, semantic searching, ontology development, maintenance and evaluation, applications of ontologies to Semantic Web and intelligent networked systems, etc.

This high-quality program would not have been possible without the authors who chose ODBASE as a venue for their publications. Out of 78 submitted papers, we selected 19 full papers, 5 short papers, and 6 posters.

To round up this excellent program, Richard Hull from IBM T.J. Watson Research Center agreed to be our keynote speaker.

A conference such as this can only succeed as a team effort. We would like to thank the program committee members and reviewers for their invaluable efforts. We are grateful to Robert Meersman and Zahir Tari for their support in organizing this event. Finally, we are deeply indebted to Vidura Gamini Abhaya, who was immensely helpful in facilitating the review process and making sure that everything stayed on track.

We hope that you enjoyed ODBASE 2008 and had a wonderful stay in Monterrey, Mexico.

November 2008

Fausto Giunchiglia

Ling Feng

Malu Castellanos

Artifact-Centric Business Process Models: Brief Survey of Research Results and Challenges

Richard Hull*

IBM T.J. Watson Research Center, Yorktown Heights, NY, USA

Abstract. A data-centric approach to business process and workflow modeling has been emerging over the past several years. This short paper presents a structured framework for a class of data-centric business process models, which are based on “business artifacts”. The paper provides a brief survey of research results on artifact-centric business process, and identifies a broad array of remaining research challenges.

1 Introduction

Businesses and other organizations increasingly rely on business process management, and in particular the management of electronic workflows underlying business processes. While most workflow is still organized around relatively flat process-centric models, over the past several years a *data-centric* approach to workflow has emerged. A key paper in this area is [37], which introduces the *artifact-centric* approach to workflow modeling. This approach focuses on augmented data records, known as “business artifacts” or simply “artifacts”, that correspond to key business-relevant objects, their lifecycles, and how/when services (a.k.a. tasks) are invoked on them. This approach provides a simple and robust structure for workflow, and has been demonstrated in practice to permit efficiencies in business transformation. As this approach has been applied, both internal to IBM and with IBM customers [5,6] a family of new requirements has emerged which are not easily addressed by the largely procedural artifact-centric model introduced in [37]. To address these requirements, variations of the original artifact-centric model are now being explored. This short paper presents a framework that can be used to help structure this exploration and more clearly expose the implications of different modeling choices in artifact-centric business process. The paper also highlights a broad array of research challenges raised by the artifact-centric approach. The field of artifact-centric business process is still in its infancy; the research results and challenges described here are not intended to be comprehensive, but rather to reflect the author’s view on some of the most pressing and useful issues to study.

The basic challenge in business process modeling is to find mechanisms whereby business executives, analysts, and subject matter experts can specify, in an intuitive yet concise way, the framework and specifics of how the operations of a business are to be conducted. The specification should make it easy to develop IT infrastructures to automate the operations as much as possible. At the same time the framework must support

* Supported in part by NSF grants IIS-0415195 and CNS-0613998.

flexibility at two important levels. The first is at the level of *individual enactments of the workflow* – with the increasing intricacy of modern day corporate and personal life, and the emergence of the “mass market of one” [20], it is essential that workflows permit a dramatic improvement in permitting highly varied operation for different customers, products, contexts and regions. The second is flexibility in enabling rich *evolution of the workflow schema*. This is increasingly important as “internet speed” becomes a reality in the definition and transformation of new markets, new competitors, and new government regulations. While both forms of flexibility must be supported, *monitoring and reporting* continue to be critical, and mechanisms need to be found so that the reports are meaningful in spite of highly varied enactments and continually evolving workflow schemas.

The emerging challenge of *generic/specialized* is focused on the need to permit the specification of a generic workflow schema that can have multiple specializations, for use in different circumstances. The specializations might be required for different regions with different governmental regulations, or for different kinds of customers and/or products, or because of different outsourcing partners being used to accomplish similar objectives. It should be easy to specify these specializations, and to provide reporting at different levels of generality, to permit both high-level comparisons across the specializations and detailed reports within the specializations.

The emergence of the web and increasing reliance on outsourcing calls for a qualitative improvement in how business processes are *componentized*, to enable *re-use* and workflow *composition* (automated if possible). Analogous to the goals of semantic web services [34], it should be possible to establish a library of business process components which are easy to refine and compose together, to enable quick construction of workflows with desired capabilities and properties.

It is our contention that the elegant and intuitively natural decomposition of business process modeling provided by the constructs of the artifact-centric approach can enable substantial advances in addressing the aforementioned requirements.

2 Four “Dimensions”

Traditional process-centric business process and workflow models, whether flat or hierarchical, are essentially uni-dimensional. They focus almost entirely on the process model, its constructs and its patterns, and provide little or no support for understanding the structure or life-cycle of the data that underlies and tracks the history of most workflows. In contrast, the artifact-centric approach can provide four explicit, inter-related but separable “dimensions” in the specification of business process, as first described in [8]. These four dimensions are illustrated in Figure 1 and described now.

Business Artifacts. Business artifacts, or simply ‘artifacts’, are intended to hold all of the information needed in completing business process execution. Speaking broadly, the notion of “business artifact” includes not only the data associated with a business object, but also aspects of its overall lifecycle and relationships to other artifacts and the business in general. In the more formal discussion here, we use ‘artifact’ to focus on the data held by the artifact, and add some of the other contextual information with explicit constructs.

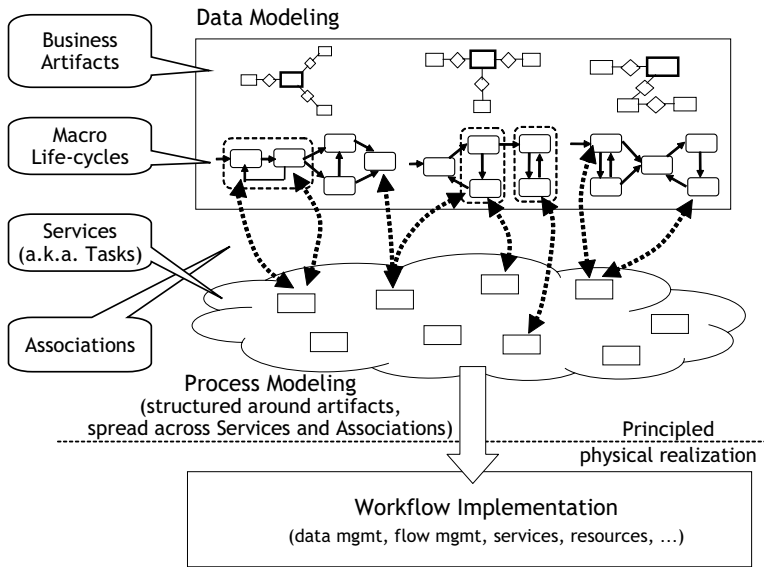


Fig. 1. Four “dimensions” in artifact-centric business process modeling

The artifacts themselves are discovered by subject matter experts, who identify key real and conceptual entities that are managed by the business. Typical artifacts include purchase orders, sales invoices, bills of lading, insurance claims, and a structured history of interactions with a customer. The artifacts should incorporate the information needed to (i) capture business process goals, and (ii) allow for evaluating how thoroughly these goals are achieved. Example data found in artifacts include data that are received during the business process execution from the external world, data that are produced by the execution, and data that record the decisions taken in the execution. A business artifact has an identity and can be tracked as it progresses through the workflow. It can have a set of attributes and other data; most of this data is uninitialized at when an artifact is created, and is initialized and possibly modified as the artifact runs through its lifecycle. In business terms, an artifact should represent the explicit knowledge concerning progress toward a business operational goal at any instant. At a more IT level, this implies that at any time of execution, the runtime state of a business process is determined by the snapshot of all artifacts.

(Macro-Level) Lifecycle. In the artifact-centric methodology, the discovery of the key business artifacts goes hand-in-hand with the discovery of the macro-level lifecycle of these artifacts. In most cases the business stakeholders can describe this macro-level lifecycle in terms of key, business-relevant *stages* in the possible evolution of the artifact, from inception to final disposition and archiving. The different artifact classes may have different “life expectancies”; in some cases the artifact is relatively short-lived (e.g., a customer order), in other cases relatively long-lived (e.g., an ongoing log of services to a customer and the customer’s satisfaction), and in yet other cases the artifact is essentially permanent (e.g., an artifact which holds the full inventory for a

given warehouse). Conceptually, it is natural to represent the macro-level lifecycle of a given class of artifacts by using a variant of finite state machines, where each state of the machine corresponds to a possible stage in the lifecycle of an artifact from this class. In this variant of state machines, little or nothing is indicated about why or how an artifact might move from one stage to another, although “guard” conditions may be attached to transitions in the machine. The use of hierarchy in the state machines may have benefits for representing specializations of generic business process schemas.

Services. A service (or ‘task’) in a business process encapsulates a unit of work meaningful to the whole business process in at least two aspects. First, the potential changes made by the service should reflect a measurable step (or steps) of progress towards the business goal. Second, the division of the business process into some collection of services should be able to accommodate (expected) administrative organization structures, IT infrastructures, customer-visible status, etc. A service may be fully automated, or may incorporate human activity and judgement. In the latter case, the outcome of a service may be essentially non-deterministic, but constrained to lie within a post-condition that captures a business policy.

Technically, a service makes changes to one or more business artifacts, and the changes should be transactional, i.e., a service should have (the effect of having) exclusive control over the involved artifacts when making these changes. The term ‘service’ rather than ‘task’ is used here, to emphasize the close correspondence between the kind of services used here and the kinds of services found in the Services Oriented Architecture (SOA) and in web services in general. This is especially relevant as workflows will become increasingly distributed in the future, both across sub-organizations of a single organization, and via the web across multiple independent organizations.

Associations. At a philosophical level, services in a business process make changes to artifacts in a manner that is restricted by a family of constraints. These constraints might stem from a procedural specification, e.g., a flowchart, or by associating services with the transitions of a finite state machine or a Petri net. Alternatively, the constraints might stem from a declarative specification, e.g., a set of rules and/or logical properties that must be satisfied. Some common types of constraints include precedence relationships among the services, between services and external events (e.g., receiving a request), and between services and internal events (e.g., timeout). In some cases the constraints involve relative or absolute times, and are thus temporal constraints.

This four-dimensional framework is referred to as “BALSA” – Business Artifacts, Lifecycles, Services, Associations.

By varying the model and constructs used in each of the four dimensions one can obtain different artifact-centric business process models with differing characteristics. We provide some simple examples here; many others are possible. The data models used for artifacts might be focused on nested name-value pairs as in [37], might be nested relations or restricted Entity-Relationship schemas as in [8], or might be based on XML, RDF, OWL, or some other Description Logic [10]. The lifecycle might be a high-level state machine with just a handful of states, it might be a hierarchical state machine or possibly a state chart. Or, it might give a very detailed view of the lifecycle, as in [37], where essentially every service moves the artifact from one stage to the next. The services might be specified as flowcharts, by simply listing their input and output artifacts

and/or artifact attributes, or by providing pre- and post-conditions as in [7,15,17], essentially in the spirit of OWL-S and semantic web services [13]. Finally, with regards to the associations of services to artifacts, there is a spectrum of possibilities from largely procedural [37] to largely declarative [7,15,17].

The impact of different combinations of models and constructs in the four dimensions, on ease of business process design, flexibility, componentatization, and reporting, are largely unexplored at present. We believe that structuring of business process models around the four dimensions can provide a rich and valuable basis for addressing these issues. For example, with regards to flexibility, the four dimensions offer four different areas in which a workflow can evolve – at the level of the data schema of artifacts, their lifecycle, the properties of available services, and the ways that the services are tied to the artifacts. We anticipate that the artifact schemas and macro-level lifecycles will be relatively stable over time, and that these can provide the basis for reasonably detailed reporting that cuts across much of the variability of enactments and schemas.

The declarative style appears quite promising in terms of supporting rich flexibility, and enabling the specification of a generic schema with multiple specializations. This is because a declarative specification describes the characteristics of *what* a business process should achieve without specifying unnecessary details about *how* it should be achieved. As a simple example, if the order of application of three services is irrelevant to the success of a business process, this is easily expressed in a declarative style, but rather clumsy to express in typical procedural models. More generally, a declarative style makes it easier to build up a business process schema from piece-parts, because the “glue” can be provided at a high level by logical assertions, rather than needing to explicitly specify detailed orderings for the different steps of the combined process.

Recent formal work [7,15,17] on the declarative approach has focused on condition-action rules that give conditions under which it is permitted (but not required) that a service be invoked or a transition from one stage to the next be permitted. It would also be useful to adapt the relatively simple temporal-logic inspired declarative language of [38] to the artifact-centric setting. Another approach is to use the the rich and extensive Semantics of Business Values and Rules (SBVR) standard to specify declarative properties that the association should satisfy. In these cases, the artifact schemas and lifecycles provide a skeleton that helps to provide a basic structure that the declarative specification can build upon. It may also be interesting to incorporate assertions that indicate “penalties” for situations where a process is taking longer than it should (e.g., “if the time from process start to manager decision is longer than 2 weeks, then raise a red alert”). The style of complex event processing constraints might be useful here [32].

3 Research Directions and Challenges

Through experiences at IBM, with both internal and external customers, we have found that the artifact-centric approach to business process specification and management has a great intuitive appeal to business managers, can bring substantial new insights to the managers, and can greatly facilitate communication about business processes between different divisions and regions of an enterprise. Central concepts from the artifact-centric approach are already being incorporated into the mainstream of IBM

professional service offerings [39], and are influencing new features in IBM's business process modeling tools. We anticipate that the artifact-centric approach and its extensions will continue to grow in usage and impact.

On the research side, the artifact-centric approach has the potential of providing elegant and pragmatic solutions for the emerging requirements on business process models, but significant research will be needed along several fronts, from both theoretical and applied perspectives. At a deeper level, the artifact-centric approach may provide a useful basis for the development of a fundamental understanding of the relationship of process and data [10]. In particular, the approach allows exploration of relatively simple notions of process (e.g., high-level state machines and/or declarative rules for service invocation, where the services are specified at a relatively abstract level) as combined with relatively simple forms of data (e.g., name-value pairs, nested relations, or restricted Entity-Relationship schemas).

We provide now a brief listing of selected research directions and challenges, and indicate how the artifact-centric approach might provide a useful basis in their investigation.

Models and views. The data-centric approach to business process has been the subject of several research investigations. Some key roots of the artifact-centric approach are present in adaptive objects [30], adaptive business objects [35], business entities, and "document-driven" workflow [43]. The notion of documents as in document engineering [21] is focused on certain aspects of artifacts, namely the artifact data itself and how it can be used to facilitate communication between sub-organizations when doing workflow processing. The Vortex workflow framework [16,25,26] is also data centric, and provides a declarative framework for specifying if and when workflow services are to be applied to a given business object. The Active XML framework [1,2] also provides an approach to mixing data and process, in a manner that is tied closely to the tree-based structure of XML. Finally, techniques and results from scientific workflow (e.g., [14]), which is also very data-centric, may have substantial impact on our understanding of artifact-centric business process. Scientific workflow is typically functional in nature, and so the techniques will need adaptation to the business process setting where side effects and overwriting of data can occur.

In spite of almost a decade of research into data-centric approaches for business process, a consensus has not yet emerged with regards to the "best" data-centric model, and none of the existing models can adequately handle the broad requirements mentioned in Section 1. It is for this reason that continued exploration of, and experimentation with, different models is so important.

In the end, no one model will be appropriate for all users or applications. It is thus important to develop a theory of *views* of business processes, along with an understanding of how to map between them. Examples of such work are [29,31], which explore mechanisms for mapping between a view or model involving primarily artifacts and their life-cycles and a view or model that is largely process centric. More generally, we envision the development of several different ways of viewing artifact-centric workflow schemas, so that different kinds of process designers will be able to easily see aspects of a schema that are relevant to their current activities. The ability of the artifact-centric approach to

explicitly model data, lifecycles, individual processes, and the mapping of processes to data provides a rich basis for the development of the different kinds of views.

Design principles in support of usability and flexibility. The field of databases has a rich literature and practice in the area of “good” database designs, with techniques such as Entity-Relationship (ER) modeling and normal forms for relational database schemas, and tools to map from ER schemas into normal form relational schemas. An analogous theory for data-centric business process models is still in a nascent stage at best. While results in, e.g., [29,31,39], can provide some insight into business process schema design, they are focused primarily on a procedural style rather than a declarative style. A fundamental question is: when is it better to use a procedural approach to specifying part of a business process, and when is it better to use a declarative approach? Another question is: how can we measure whether one schema is “more flexible” than another schema (perhaps relative to a pre-determined family of possible variations)? Is there a trade-off between the “amount” of flexibility that is incorporated into a schema and the ability to execute at high scale with fast response times, and if so, how can we quantify it?

Componentization and Composition. The paradigm of Service-Oriented Architecture (SOA) provides an approach for partitioning software into natural components, services in this case, that offer the possibility of building many different combinations of the components. The field of business process componentization (e.g., [9]), has grown from a similar need, due to the need for flexibility in schema evolution, and increased globalization and out-sourcing. Business process components also have implications in connection with efficiency: anecdotal evidence suggests that reducing the number of “hand-offs” in a business process (that is, situations where one person stops working on a given enactment and another person starts working on it) can lead to better efficiencies and fewer exceptions.

It appears that a data-centric perspective, and the four-dimensional structure of artifact-centric business process modeling, can provide a useful structure for defining process components and mechanisms to compose them. More specifically, the four dimensions allow for experimentation with componentization at each of the four dimensions, either individually or in combination, to determine the most effective approaches.

Implementation and optimization. Efficient implementation of a procedural variant of the artifact-centric paradigm is described in [28]. It is less clear, however, how to implement declarative variants in an efficient manner. One approach to efficient implementation would be to use a relatively direct, interpretive approach, similar to how some rule engines are implemented (e.g., [23,24]). In this case, the declarative specification would be processed primarily to build up indices and other support structures, and each step of processing would focus on identifying services eligible for invocation. The interpretive approach is probably not feasible if the artifact-centric workflow is to be distributed across geographic regions or across organizations. In this case an approach based on a transformation to a more procedural model that explicitly includes distribution of data and message passing is probably appropriate. An initial exploration of this approach is presented in [8]. The various dimensions of the artifact-centric approach

may support optimizations in this context that would not be as obvious when working with a process-centric workflow. Many problems remain open, including issues around concurrency control, indexing, data staleness, efficient performance monitoring and dashboards, graceful handling of workflow schema evolution, and incorporation of specializations of generic schemas.

Foundations, static analysis, and synthesis. From the formal perspective, little is understood about artifact-centric (and other data-centric) business process modeling. As noted above, the artifact-centric approach has the potential of supporting a new theory that enables the study of data and process in a richly interconnected setting.

There have been a handful of formal investigations into the analysis of artifact-based systems: [18,19,33] provide techniques and results for procedural variants, and [7,15] focus on declarative variants. Analysis tools are especially important in the declarative case, both because process designers may be less familiar with declarative specifications, and because declarative specifications have strong potential for use in the area of business process composition. Analysis questions are undecidable if artifact invention is permitted, or if general set-valued artifact attributes are permitted. Restrictions on set-valued attributes that enable decidability results are presented in [15], and useful extensions of these results can probably be obtained by taking into account the ways that set-valued attributes might be used in practice. Given the undecidability of analysis in the general setting, it will also be extremely useful to develop analysis results based on abstractions of the model, to provide conservative tests that can be used in practice.

Another fundamental question for artifact-centric business process concerns the automatic construction, or synthesis, of workflow schemas from high-level goals. Citation [17] presents a first investigation in this direction, and explores the use of weakest preconditions (a.k.a. “regressions”) for building declarative artifact-centric schemas from artifact schemas, services specified using pre- and post-conditions, and high-level goals that the schema’s workflow enactments are to satisfy. That work is inspired in part by work on automatic composition of semantic web services [3,4,36], but uses different techniques stemming from unique characteristics of the artifact-centric setting.

The study of a variety of other foundational questions can provide fruitful insights about the artifact-centric approach. As noted earlier, the study of different views of artifact-centric workflows as in [31] will be important in helping to design and examine workflow schemas, by giving designers different perspectives on the schemas. Notions of dominance and equivalence between artifact-centric workflow schemas in terms of their expressive or representational power, analogous to corresponding research on database schemas [22], will be fundamental to understanding optimization and evolution of workflow schemas. Another key area, mentioned above, where foundational work is needed concerns decomposition of workflow schemas. For example, can we develop a theory for understanding the relationship between different decompositions, how data is shared or transported between components, and the amount or quality of “hand-offs” between process users?

User-centric aspects. Business process design and usage are intensely human-centered activities, perhaps rivaled only computer games and browser-based web services for the level of tight interaction between human and machine. A key challenge for any business process model is that there be a natural approach for representing schemas in the

model to business executives, analysts and subject-matter experts. In many cases these folks are non-technical but nevertheless need to understand the schemas, sometimes at a quite detailed level. While an approach for visualization and manipulation has been developed [28] for a procedural variant of artifact-centric business process, developing an analog for declarative variants remains an open challenge.

Turning to process users, citations [28,41,40] describe an end-to-end user-centric approach for designing and implementing the user interfaces for the services associated with a procedural artifact-centric workflow schema. The artifact schemas, and the input and output artifacts and attributes of individual services, provide key information that can be used to automatically generate the sequences of web pages that are presented to the process users. One interesting challenge in this area is to extend these techniques to be able to provide to process users an understanding of how their activities in connection with specific services relate to the overall flow and performance of the enactments running through a workflow.

Monitoring and tuning. Central to successful business process management is the ability to monitor the performance of an operation, to provide both immediate feedback (typically through dashboards) and deeper, more extensive insights (typically through reporting and data mining on logs), and to enable tuning of the process based on objective data obtained. The artifact-centric approach provides useful structure for these activities, because the artifacts hold, at any time, all business-relevant information about their status and position within the overall workflow. Citations [12,27,28] describe how this has been exploited in practical situations. Techniques are needed to extend the approach to situations where the underlying workflow schema is evolving over time, and where there is a generic schema with multiple specializations. More generally, it would be very useful to develop mechanisms whereby information from performance monitoring could be used to modify how resources (and in particular, the people performing services) are allocated, and more fundamentally, how a business process is organized [11]. In another direction, it would be useful to extend techniques for “process mining” [42] to an artifact-centric setting, where the artifacts might be discovered, or might be viewed as known prior to the mining activity.

Linkage to business strategy. In works such as [5,28] a four-tier view of business process management is described. At the top level is the specification of business strategy. This is used as the starting point and driver for the second level, the specification of the business operations model, for which the artifact-centric approach seems quite relevant. The third level is termed “solution composition”, and refers to the specification at a logical level of what services and capabilities will need to be brought together to support the operations model, and how these might be optimized. At the fourth level is the platform-specific physical implementation.

While the business strategy can be used informally to drive the development of an artifact-based schema for the operations, it would be quite valuable to develop a much tighter linkage between these two. A key ingredient here would be to develop an approach to specifying business strategy that is flexible enough to encompass the full range of techniques and factors that might arise in the development of a strategy, but that uses a relatively small number of conceptual constructs. A next step would be the

development of a framework for mapping from strategy to operations specification, and for specifying how the operations should be tuned based on observed performance.

Evaluation of paradigms, tools, and methods. A final, and very challenging, area for research in business process management concerns determining the relative merit of different modeling paradigms along with their associated families of methods for designing and realizing processes, and tools that support those methods. At present there is anecdotal evidence that the artifact-centric approach brings insights and efficiencies to businesses [5,6], but this is notoriously hard to measure in an objective fashion. Further, while introducing a declarative flavor into the artifact-centric approach appears useful for solving some of the emerging requirements, again there is no objective evidence to support this view.

Developing a compelling and systematic approach for comparing the costs and benefits of different approaches to business process management would substantially simplify the challenge of deciding what approaches should be used, and more important to the research community, what approaches should be extended and developed more fully.

Acknowledgements. This survey of research results and directions is based on discussions with many people, both within IBM and from several universities. The author especially thanks Kamal Bhattacharya and Jianwen Su for extensive and inspiring conversations. The author thanks the participants of the “Zurich workshop” held in July, 2008, which included David Cohn, Pankaj Dhoolia, Amit Fisher, Adrian Flatgard, Terry Heath, Anil Nigam, Jana Koehler, Jochen Küster, Rong (Emily) Liu, Yohai Makbili, Prabir Nandi, Jorge Sanz, John Vergo, and Ksenia Wahler; many of the research challenges described here came into sharper focus during that workshop. The author also thanks Diego Calvanese, Henry Chang, Giuseppe de Giacomo, Alin Deutsch, Christian Fritz, Stacy Hobson, Mark Linehan, Fabio Patrizi, Noi Sukaviriya, Victor Vianu, and Fred Wu for many stimulating ideas and discussions around artifact-centric business process.

References

1. Abiteboul, S., Benjelloun, O., Milo, T.: The Active XML project: An overview. *Very Large Databases Journal* 17(5), 1019–1040 (2008)
2. Abiteboul, S., Segoufin, L., Vianu, V.: Static analysis of active XML systems. In: *Proc. Intl. Symp. on Principles of Database Systems (PODS)*, pp. 221–230 (2008)
3. Berardi, D., Calvanese, D., De Giacomo, G., Hull, R., Mecella, M.: Automatic composition of transition-based semantic web services with messaging. In: *Intl. Conf. on Very Large Databases (VLDB)*, pp. 613–624 (2005)
4. Berardi, D., Calvanese, D., De Giacomo, G., Lenzerini, M., Mecella, M.: Automatic service composition based on behavioral descriptions. *Int. J. Cooperative Inf. Syst.* 14(4), 333–376 (2005)
5. Bhattacharya, K., Caswell, N.S., Kumaran, S., Nigam, A., Wu, F.Y.: Artifact-centered operational modeling: Lessons from customer engagements. *IBM Systems Journal* 46(4), 703–721 (2007)

6. Bhattacharya, K., et al.: A model-driven approach to industrializing discovery processes in pharmaceutical research. *IBM Systems Journal* 44(1), 145–162 (2005)
7. Bhattacharya, K., Gerede, C.E., Hull, R., Liu, R., Su, J.: Towards formal analysis of artifact-centric business process models. In: *Proc. Int. Conf. on Business Process Management (BPM)*, pp. 288–304 (2007)
8. Bhattacharya, K., Hull, R., Su, J.: A Data-centric Design Methodology for Business Processes. In: Cardoso, J., van der Aalst, W.M.P. (eds.) *Handbook of Research on Business Process Management* (to appear, 2009)
9. Bohrer, K., Johnson, V., Nilsson, A., Rubin, B.: Business process components for distributed object applications. *Commun. ACM* 41(6), 43–48 (1998)
10. Calvanese, D., de Giacomo, G.: Private communication (August 2008)
11. Chang, H.: Private communication (August 2008)
12. Chowdhary, P., et al.: Model driven development for business performance management. *IBM Systems Journal* 45(3), 587–605 (2006)
13. OWL Services Coalition. OWL-S: Semantic markup for web services (November 2003)
14. Davidson, S.B., Freire, J.: Provenance and scientific workflows: Challenges and opportunities. In: *Proc. ACM SIGMOD Int. Conf. on Management of Data*, pp. 1345–1350 (2008)
15. Deutsch, A., Hull, R., Patrizi, F., Vianu, V.: Automatic verification of data-centric business processes (submitted for publication, 2008)
16. Dong, G., Hull, R., Kumar, B., Su, J., Zhou, G.: A framework for optimizing distributed workflow executions. In: Connor, R.C.H., Mendelzon, A.O. (eds.) *DBPL 1999. LNCS*, vol. 1949, pp. 152–167. Springer, Heidelberg (2000)
17. Fritz, C., Hull, R., Su, J.: Automatic construction of simple artifact-based workflows (submitted for publication, 2008)
18. Gerede, C.E., Bhattacharya, K., Su, J.: Static analysis of business artifact-centric operational models. In: *IEEE International Conference on Service-Oriented Computing and Applications* (2007)
19. Gerede, C.E., Su, J.: Specification and verification of artifact behaviors in business process models. In: Krämer, B.J., Lin, K.-J., Narasimhan, P. (eds.) *ICSOC 2007. LNCS*, vol. 4749, pp. 181–192. Springer, Heidelberg (2007)
20. Gilmore, J.H., Pine, B.J. (eds.): *Markets of One: Creating Customer-Unique Value through Mass Customization*. Harvard Business Review, Cambridge (1988)
21. Glushko, R.J., McGrath, T.: *Document Engineering: Analyzing and Designing Documents for Business Informatics and Web Services*. MIT Press, Cambridge (2005)
22. Hull, R.: Relative information capacity of simple relational database schemata. *SIAM J. Comput.* 15(3), 856–886 (1986)
23. Hull, R., et al.: Everything personal, not just business: Improving user experience through rule-based service customization. In: Orlowska, M.E., Weerawarana, S., Papazoglou, M.P., Yang, J. (eds.) *ICSOC 2003. LNCS*, vol. 2910, pp. 149–164. Springer, Heidelberg (2003)
24. Hull, R., et al.: Enabling context-aware and privacy-conscious user data sharing. In: *IEEE Intl. Conf. on Mobile Data Management (MDM)* (2004)
25. Hull, R., Llirbat, F., Kumar, B., Zhou, G., Dong, G., Su, J.: Optimization techniques for data-intensive decision flows. In: *Proc. Int. Conf. on Data Engineering* (2000)
26. Hull, R., Llirbat, F., Simon, E., Su, J., Dong, G., Kumar, B., Zhou, G.: Declarative workflows that support easy modification and dynamic browsing. In: *Proc. Int. Joint Conf. on Work Activities Coordination and Collaboration* (1999)
27. Kapoor, S., et al.: Sense-and-respond supply chain using model-driven techniques. *IBM Systems Journal* 46(4), 685–702 (2007)
28. Kumaran, S., et al.: Using a model-driven transformational approach and service-oriented architecture for service delivery management. *IBM Systems Journal* 46(3), 513–529 (2007)

29. Kumaran, S., Liu, R., Wu, F.Y.: On the duality of information-centric and activity-centric models of business processes. In: Bellaïssène, Z., Léonard, M. (eds.) CAiSE 2008. LNCS, vol. 5074. Springer, Heidelberg (2008)
30. Kumaran, S., Nandi, P., Heath, T., Bhaskaran, K., Das, R.: ADoc-oriented programming. In: Symp. on Applications and the Internet (SAINT), pp. 334–343 (2003)
31. Küster, J., Ryndina, K., Gall, H.: Generation of BPM for object life cycle compliance. In: Proceedings of 5th International Conference on Business Process Management (BPM) (2007)
32. Linehan, M.: Private communication (July 2008)
33. Liu, R., Bhattacharya, K., Wu, F.Y.: Modeling business contexture and behavior using business artifacts. In: Krogstie, J., Opdahl, A., Sindre, G. (eds.) CAiSE 2007 and WES 2007. LNCS, vol. 4495, pp. 324–339. Springer, Heidelberg (2007)
34. McIlraith, S.A., Son, T.C., Zeng, H.: Semantic web services. *IEEE Intelligent Systems* 16(2), 46–53 (2001)
35. Nandi, P., Kumaran, S.: Adaptive business objects – a new component model for business integration. In: Proc. Intl. Conf. on Enterprise Information Systems, pp. 179–188 (2005)
36. Narayanan, S., McIlraith, S.: Simulation, verification and automated composition of web services. In: Intl. World Wide Web Conf. (WWW 2002) (2002)
37. Nigam, A., Caswell, N.S.: Business artifacts: An approach to operational specification. *IBM Systems Journal* 42(3), 428–445 (2003)
38. Pesic, M., Schonenberg, H., van der Aalst, W.M.P.: Declare: Full support for loosely-structured processes. In: IEEE Intl. Enterprise Distributed Object Computing Conference (EDOC), pp. 287–300 (2007)
39. Strosnider, J.K., Nandi, P., Kumarn, S., Ghosh, S., Arsanjani, A.: Model-driven synthesis of SOA solutions. *IBM Systems Journal* 47(3), 415–432 (2008)
40. Sukaviriya, N., Sinha, V., Ramachandra, T., Mani, S.: Model-driven approach for managing human interface design life cycle. In: Engels, G., Opdyke, B., Schmidt, D.C., Weil, F. (eds.) MODELS 2007. LNCS, vol. 4735, pp. 226–240. Springer, Heidelberg (2007)
41. Sukaviriya, N., Sinha, V., Ramachandra, T., Mani, S., Stolze, M.: User-centered design and business process modeling: Cross road in rapid prototyping tools. In: Proc. Intl. Conf. Human-Computer Interaction (INTERACT), Part I, pp. 165–178 (2007)
42. van der Aalst, W.M.P., de Beer, H.T., van Dongen, B.F.: Process mining and verification of properties: An approach based on temporal logic. In: Proc. On the Move Confederated International Conferences CoopIS, DOA, and ODBASE 2005, Part I, pp. 130–147 (2005)
43. Wang, J., Kumar, A.: A framework for document-driven workflow systems. In: Business Process Management, pp. 285–301 (2005)

Ten Challenges for Ontology Matching

Pavel Shvaiko¹ and Jérôme Euzenat²

¹ TasLab, Informatica Trentina S.p.A., Trento, Italy

`pavel.shvaiko@infotn.it`

² INRIA & LIG, Grenoble, France

`Jerome.Euzenat@inrialpes.fr`

Abstract. This paper aims at analyzing the key trends and challenges of the ontology matching field. The main motivation behind this work is the fact that despite many component matching solutions that have been developed so far, there is no integrated solution that is a clear success, which is robust enough to be the basis for future development, and which is usable by non expert users. In this paper we first provide the basics of ontology matching with the help of examples. Then, we present general trends of the field and discuss ten challenges for ontology matching, thereby aiming to direct research into the critical path and to facilitate progress of the field.

1 Introduction

The progress of information and communication technologies has made available a huge amount of disparate information. The number of different information resources is growing significantly, and therefore, the problem of managing heterogeneity among them is increasing. As a consequence, various solutions have been proposed to facilitate dealing with this situation, and specifically, of automating integration of distributed information sources. Among these, semantic technologies have attracted significant attention. For example, according to Gartner¹, semantic technologies is in the list of top ten disruptive technologies for 2008-2012. In this paper we focus on a particular part of semantic technologies, which is ontology matching.

An ontology typically provides a vocabulary that describes a domain of interest and a specification of the meaning of terms used in the vocabulary. Depending on the precision of this specification, the notion of ontology encompasses several data and conceptual models, for example, sets of terms, classifications, database schemas, or fully axiomatized theories. However, when several competing ontologies are in use in different applications, most often they cannot interoperate as is, though the fact of using ontologies rises heterogeneity problems to a higher level.

Ontology matching is a solution to the semantic heterogeneity problem. It finds correspondences between semantically related entities of ontologies. These correspondences can be used for various tasks, such as ontology merging, query

¹ <http://www.gartner.com/it/page.jsp?id=681107>

answering, data translation, etc. Thus, matching ontologies enables the knowledge and data expressed in the matched ontologies to interoperate [25].

Many diverse solutions of matching have been proposed so far, see [49, 67] for some contributions of the last decades and [14, 46, 64, 68, 73] for recent surveys². Finally, ontology matching has been given a book account in [25]. However, despite the many component matching solutions that have been developed so far, there is no integrated solution that is a clear success, which is robust enough to be the basis for future development, and which is usable by non expert users.

This is a prospective paper and its key contribution is a discussion of the main trends in the ontology matching field articulated along ten challenges accompanied for each of these with an overview of the recent advances in the field. This should direct research into the critical path and accelerate progress of the ontology matching field. The challenges discussed are: (i) large-scale evaluation, (ii) performance of ontology-matching techniques, (iii) discovering missing background knowledge, (iv) uncertainty in ontology matching, (v) matcher selection and self-configuration, (vi) user involvement, (vii) explanation of matching results, (viii) social and collaborative ontology matching, (ix) alignment management: infrastructure and support, and (x) reasoning with alignments.

The remainder of the paper is organized as follows. Section 2 provides, with the help of an example, the basics of ontology matching. Section 3 outlines ontology matching applications and discusses the role of final users in defining application requirements. Section 4 presents a market watch for the ontology matching field. Sections 5-14 discuss ten challenges of the field and for each of these briefly overview the corresponding recent advances. Finally, Section 15 reports the major findings of the paper.

2 The Ontology Matching Problem

In this section we first discuss a motivating example (§2.1), then we provide some basic definitions of ontology matching (§2.2), and finally we describe the alignment life cycle (§2.3).

2.1 Motivating Example

Let us use two simple XML schemas (see Figure 1), which can be viewed as a particular type of ontology, in order to exemplify the ontology matching problem.

Let us suppose that an e-commerce company needs to acquire another one. Technically, this acquisition may require the integration of the databases of these companies. The documents of both companies are stored according to XML schemas *O1* and *O2*, respectively. A first step in integrating the schemas is to identify candidates to be merged or to have taxonomic relationships under an integrated schema. This step refers to a process of matching. For example, the elements with labels *Price* in *O1* and in *O2* are candidates to be merged,

² See <http://www.ontologymatching.org> for a complete information on the topic, e.g., publications, tutorials, relevant events.

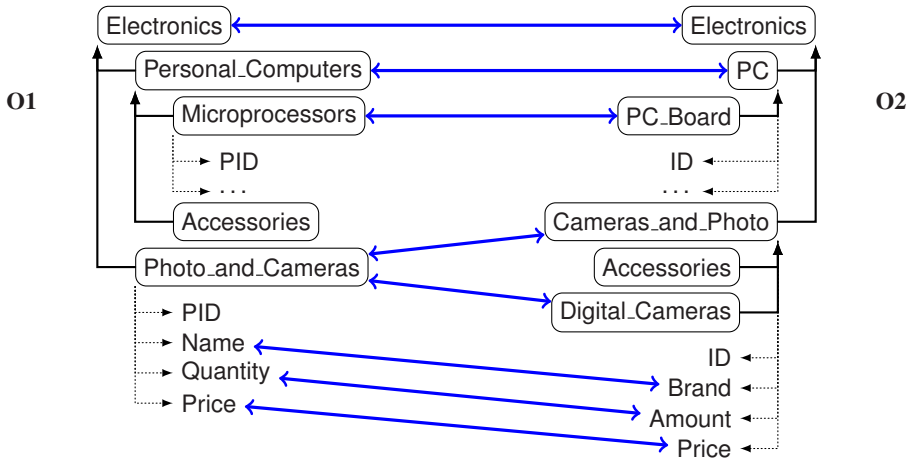


Fig. 1. Two simple XML schemas. XML elements are shown in rectangles with rounded corners, while attributes are shown without the latter. The correspondences are expressed by arrows.

while the element with label `Digital_Cameras` in *O2* should be subsumed by the element with label `Photo_and_Cameras` in *O1*. Once the correspondences between two schemas have been determined, the next step has to generate, for example, query expressions that automatically translate data instances of these schemas under an integrated schema [73].

2.2 Problem Statement

The *matching* operation determines the alignment A' for a pair of ontologies *O1* and *O2*, each of which consisting of a set of discrete entities, such as classes, properties or individuals. There are some other parameters that can extend the definition of the matching process, namely: (i) the use of an input alignment *A*, which is to be completed by the process; (ii) the matching parameters, for instance, weights, thresholds; and (iii) external resources used by the matching process, for instance, common knowledge and domain specific thesauri.

Alignments express correspondences between entities belonging to different ontologies. Given two ontologies, a *correspondence* is a 5-uple: $\langle id, e_1, e_2, n, r \rangle$, where: *id* is a unique identifier of the given correspondence; e_1 and e_2 are entities (e.g., tables, XML elements, properties, classes) of the first and the second ontology, respectively; *n* is a confidence measure (typically in the $[0, 1]$ range) holding for the correspondence between e_1 and e_2 ; *r* is a relation (e.g., equivalence ($=$), more general ($\sqsupseteq$), disjointness ($\perp$), overlapping ($\sqcap$)) holding between e_1 and e_2 . The correspondence $\langle id, e_1, e_2, n, r \rangle$ asserts that the relation *r* holds between the ontology entities e_1 and e_2 with confidence *n*. The higher the confidence, the higher the likelihood that the relation holds.

For example, in Figure 1, according to some matching algorithm based on linguistic and structure analysis, the confidence measure (for the fact that the equivalence relation holds) between entities with labels *Photo_and_Cameras* in *O1* and *Cameras_and_Photo* in *O2* could be 0.67. Suppose that this matching algorithm uses a threshold of 0.55 for determining the resulting alignment, i.e., the algorithm considers all the pairs of entities with a confidence measure higher than 0.55 as correct correspondences. Thus, our hypothetical matching algorithm should return to the user the following correspondence: $\langle id_{3,3}, Photo\_and\_Cameras, Cameras\_and\_Photo, 0.67, = \rangle$. The relation between the same pair of entities, according to another matching algorithm which is able to determine that both entities mean the same thing, could be exactly the equivalence relation (without computing the confidence measure). Thus, returning to the user $\langle id_{3,3}, Photo\_and\_Cameras, Cameras\_and\_Photo, n/a, = \rangle$.

2.3 Alignment Life Cycle

Like ontologies, alignments have their own life cycle [23] (see Figure 2). They are first created through a matching process, which may be manual. Then they can go through an iterative loop of evaluation and enhancement. Evaluation consists of assessing properties of the obtained alignment. It can be performed either manually or automatically. Enhancement can be obtained either through manual change of the alignment or application of refinement procedures, e.g., selecting some correspondences by applying thresholds. When an alignment is deemed worth publishing, then it can be stored and communicated to other parties interested in such an alignment. Finally, the alignment is transformed into another form or interpreted for performing actions, like mediation or merging.

As Figure 2 indicates, creating an alignment is only the first step of the process. Very often these alignments have to be evaluated, improved and finally transformed into some executable procedure before being used by applications: transforming an ontology in order to integrate it with another one, generating a set of bridge axioms that will help the identification of corresponding concepts, translating messages sent from one agent to another, translating data circulating among heterogeneous web services, mediating queries and answers in peer-to-peer systems and federated databases.

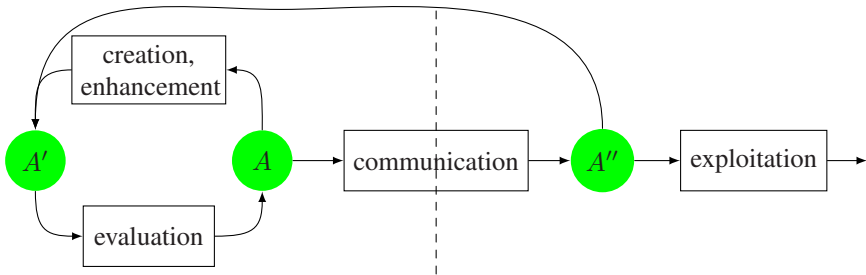


Fig. 2. The ontology alignment life cycle [23]

3 Applications and Use Cases

Ontology matching is an important operation in traditional applications, such as ontology evolution, ontology integration, data integration, and data warehouses. Typically, these applications are characterized by heterogeneous structural models that are analyzed and matched either manually or semi-automatically at design time. In such applications, matching is a prerequisite to running the actual system.

There are some emerging applications that can be characterized by their dynamics, such as peer-to-peer information sharing, web service integration, multi-agent communication, query answering and semantic web browsing. Such applications, contrary to traditional ones, require (ultimately) a run time matching operation and take advantage of more explicit conceptual models. A detailed description of these applications can be found in [25]. Let us now discuss the role of final users in defining application requirements.

User-Oriented Approach. Many research projects devoted to ontology matching correctly identify an application 'in which prototypes they develop can be eventually exploited. However, it is far rarely the case that *final users* are directly involved in the definition of requirements and use cases instantiating the applications under consideration within those projects. This is so because research projects are not usually concerned with bringing the original ideas developed within them down to the actual exploitation of these by the (expected) final users. Also enterprises that are often involved in larger research projects (e.g., of 4 years with about 1K man-month effort) are primarily interested in acquiring know-how to be later exploited in their internal projects. Hence, in order to foster an early practical exploitation of the research prototypes, it is necessary to directly involve final users in the research and development cycles. An example of such user-oriented open innovation methodologies includes Living Labs³.

Below we exemplify a use case that has been elaborated together with final users, namely a public administration and more specifically, the Urban Planning and Environment Protection department of the Autonomous Province of Trento. Notice that involving final users into the research and development cycles requires addressing a *social challenge* of integrating relevant actors and facilitating the cross-fertilization among research centers, technology providers and user institutions, see [30] for a discussion of these in the context of the semantic heterogeneity problem. An example of undertaking this challenge includes Trentino as a Lab⁴[31].

Emergency Response. Within the OpenKnowledge⁵ project there has been analyzed the organizational model of the distributed GIS agency infrastructure of Trentino that includes: civilian protection, urban planning, forestry, viability, etc. Each GIS agency is responsible for providing a subset of the geographic

³ <http://www.cdt.ltu.se/projectweb/4421cddc626cb/Main.html>

⁴ <http://www.taslab.eu>

⁵ OpenKnowledge (FP6-027253): <http://www.openk.org>

information for the local region. Let us focus on the most frequent use case, i.e., map request service, and in turn, on the most typical request, such as a digital map request. A service requestor - both in an emergency or normal situation - needs to visualize a map of a region with geo-referenced information selected by a user. Therefore, the required map is a composition of different geographic layers offered by one of the service provider agents.

The OpenKnowledge project developed a peer-to-peer infrastructure which was used within the emergency response domain [56]. At the core of this approach is a specific view on semantics of both web service and agent coordination as proposed in [69]. Peers share explicit knowledge of the *interactions* in which they are engaged and these models of interaction are used operationally as the anchor for describing the semantics of the interaction. Instead of requiring a universal semantics across peers we require only that semantics is consistent (separately) for each instance of an interaction. These models of interactions are developed locally by peers and are shared on the network. Then, since there is no a priori semantic agreement (other than the interaction model), matching is needed to automatically make semantic commitments between the interacting parts. In particular, it is used to identify peers, which are suitable to play a particular role in an interaction model.

In the context of formalization of a digital map request scenario mentioned above (see for details [56]), consider i -th interaction model IM_i , where a constraint on playing m -th role r_m in IM_i is as follows $C1$: `getMap(MapFile, Version, Layers, Width, Height, Format, XMinBB, YMinBB, XMaxBB, YMaxBB)`, which can be viewed as a web service description. In turn, the `getMap` message will contain the URL of the requested map (`MapFile`), the version of the service (`Version`), the requested geographic layers (`Layers`), the dimensions of the map (`Width`, `Height`), its graphic format (`Format`), and finally its spatial coverage (`XMinBB`, `YMinBB`, `XmaxBB`, `YMaxBB`). Let us suppose that $C2$: `getMap(Dimension(Width, Height), MapFile, Edition, Layers)` is a description of the capabilities of k -th peer, p_k . Then, p_k wants to subscribe to r_m in IM_i , and thus, its capabilities should be matched to the constraints of r_m . If the matching between $C1$ and $C2$ is good enough, then, peer p_k can be allowed to play role r_m . Notice that matching between constraints of a role in an interaction model and peer capabilities should be performed at run time. A matching solution for this use case has been developed in [33].

4 Market Watch

Let us make several observations concerning the development of the ontology matching field as such. With this respect, an important work has been conducted within the Knowledge Web project⁶. It concerned with the analysis of the Gartner hype curve⁷ and placement of the various semantic web technologies along it, see Figure 3. In order to build this curve various distinct groups of researchers and practitioners have been involved, see [10] for details. On the

⁶ KnowledgeWeb (IST-2004-507482): <http://knowledgeweb.semanticweb.org/>

⁷ <http://www.gartner.com/pages/story.php.id.8795.s.8.jsp>

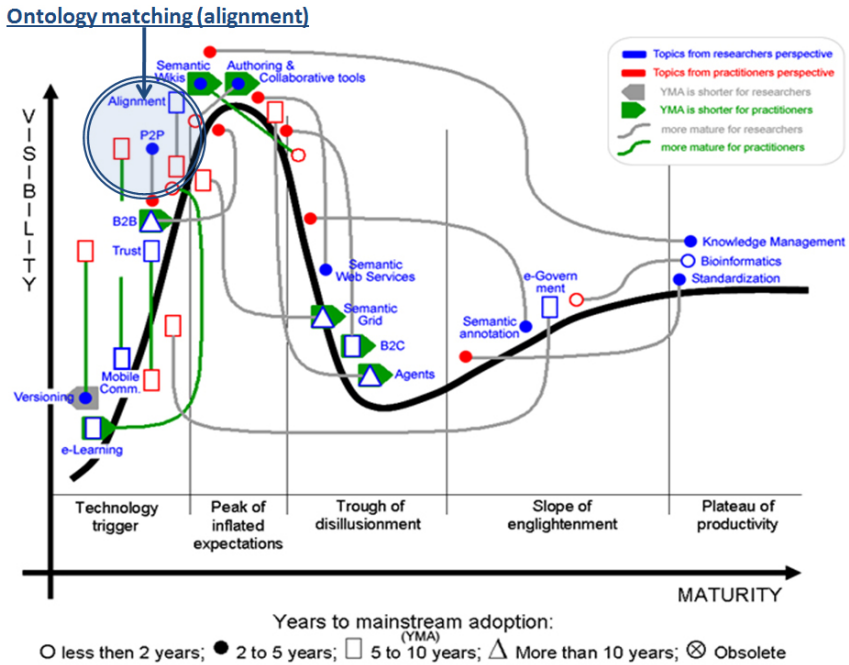


Fig. 3. Hype curve: comparison between researchers' and practitioners' points of view on semantic web technologies. Adapted from [10].

one side, the topics addressed in Figure 3 are specific to the semantic web domain. These cannot be directly compared with any Gartner's counterpart, and, hence, the latter are not taken into account. On the other side, topics of Figure 3 include ontology matching, referred to as alignment.

The first observation is that for what concerns ontology matching, both researchers and practitioners agree on locating this topic just before the peak of inflated expectation, with the same long term duration (5 to 10 years) to mainstream adoption. Hence, there are still many challenges to be addressed before ontology matching technology can be seen among the mainstream components.

Let us now consider dynamics of papers devoted to ontology matching and published in the major conferences and journals⁸, which is as follows (year:number of publications): ≤2000:18, 2001:15, 2002:13, 2003:17, 2004:29, 2005:54, 2006:60, and 2007:71. Another observation is that the dynamics of papers devoted to ontology matching reconfirm the overall trend indicated in Figure 3 that the ontology matching field keeps growing.

Based on the analysis above, we expect that, as the ontology matching technology is becoming more mature, practitioners will increase their expectations and will want to experiment with it more intensively.

⁸ <http://www.ontologymatching.org/publications>. Access date: 18.08.2008.

In Sections 5-14 we discuss ten challenges for ontology matching together with a brief overview of the recent advances in the field for each of these challenges.

5 Large-Scale Evaluation

The rapid growth of various matching approaches makes the issues of their evaluation and comparison more severe. In order to address these issues, in 2005 the Ontology Alignment Evaluation Initiative - OAEI⁹ was set up, which is a coordinated international initiative that organizes the evaluation of the increasing number of ontology matching systems. The main goal of OAEI is to support the comparison of the systems and algorithms on the same basis and to allow anyone to draw conclusions about the best matching strategies. Two first events were organized in 2004 [76]. Then, unique OAEI campaigns occurred in 2005 [2], 2006 [24], 2007 [22] and at the moment of writing of this paper OAEI-2008 is under way.

There are many issues to be addressed in ontology matching evaluation in order to empirically prove the matching technology to be mature and reliable.

- OAEI campaigns gave only some preliminary evidence of the scalability characteristics of the ontology matching technology. Therefore, larger tests involving 10.000, 100.000, and 1.000.000 entities per ontology (e.g., UMLS¹⁰ has about 200.000 entities) are to be designed and conducted. In turn, this raises the issues of a wider automation for acquisition of reference alignments, e.g., by minimizing the human effort while increasing an evaluation dataset size.
- There is a need for more accurate evaluation quality measures (initial steps towards these have already been done in [21]). In particular, application specific measures are needed in order to assess whether the result of matching is good enough for an application.
- There is a need for evaluation methods grounded on a deep analysis of the matching problem space in order to offer semi-automatic test generation methods of desired test hardness by addressing a particular point of this space (initial steps towards this line have already been done in [39]).
- Despite efforts on meta-matching systems, composing matchers [18, 48, 50] and on Alignment API [19], ontology matching largely lacks interoperability benchmarks between tools.

6 Performance of Ontology-Matching Techniques

Beside quality of matching results, there is an issue of performance, see, e.g., [6]. Performance is of prime importance in many dynamic applications, for example, where a user can not wait too long for the system to respond. Execution time

⁹ <http://oaei.ontologymatching.org/>

¹⁰ <http://www.nlm.nih.gov/research/umls/>

indicator shows scalability properties of the matchers and their potential to become industrial-strength systems. Also, referring to [41], the fact that some systems run out of memory on some test cases, although being fast on the other test cases, suggests that their performance time is achieved by using a large amount of main memory. Therefore, usage of main memory should also be taken into account.

Optimizations are worth been done only once the underlying basic techniques are stable. For example, in the case of S-Match [35, 38, 41], when dealing with lightweight ontologies [32, 42], the matching problem was reduced to the validity problem for the propositional calculus. The basic version of S-Match was using a standard DPLL-based satisfiability procedure of SAT4J¹¹. Once it has been realized that the approach is promising (based on the preliminary evaluation in [34]), the efficiency problems have been tackled. Specifically, for certain and quite frequent in practise cases, e.g., when matching formula is Horn, satisfiability became resolved in linear time, while standard SAT solver would require quadratic time, see [40] for details. Beside S-Match, several other groups, for example, Falcon [44] and COMA++ [13], have started addressing seriously the issues of performance. However, this fact cannot be still considered as a trend in the field, see, e.g., the results of the anatomy track of OAIE-2007 [22], where only several systems, such as Falcon, took several minutes to complete this matching task, while other systems took much more time (hours and even days).

7 Discovering Missing Background Knowledge

One of the sources of difficulty for the matching tasks is that ontologies are designed with certain background knowledge and in a certain context, which unfortunately do not become part of the ontology specification, and, thus, are not available to matchers. Hence, the lack of background knowledge increases the difficulty of the matching task, e.g., by generating too many ambiguities. Various strategies have been used to attack the problem of the lack of background knowledge. These include: (i) declaring the missing axioms manually as a pre-match effort [12, 54]; (ii) reusing previous match results [12]; (iii) querying the web [43]; (iv) using domain specific corpus [1, 52]; (v) using domain specific ontologies [1, 77]; and (vi) using ontologies available on the semantic web [71]. In addition, the work in [36] discussed an automatic approach to deal with the lack of background knowledge in matching tasks by using semantic matching [35, 37] iteratively. While the work in [9] proposed to automatically revise a mediated schema (which can be viewed as a background knowledge in data integration applications) in order to improve matchability.

The techniques mentioned above have helped improving the results of matchers in various cases. Moreover, these techniques can undergo different variations based on the way the background knowledge sources are selected, the way the ontology entities are matched against the background knowledge sources and the combination of the results obtained from the various external sources; though

¹¹ <http://www.sat4j.org/>

they still have to be systematically investigated, combined in a complementary fashion and improved.

Finally, it is worth noting that discovering missing background knowledge is particularly important in dynamic settings, where the matching input is often much more shallow (especially when dealing with fragmented descriptions), and therefore, incorporates fewer clues. To this end, it is vital to identify the minimal background knowledge necessary to resolve a particular problem with good enough results [74] and how to compute this minimal background knowledge.

8 Uncertainty in Ontology Matching

The issue of dealing with uncertainty in ontology matching has been addressed in [8, 16, 28, 29, 53, 63]. A way of modeling ontology matching as an uncertain process is by using similarity matrices as a measure of certainty. A matcher then is measured by the fit of its estimation of a certainty of a correspondence to the real world. In [29], such a formal framework was provided, attempting to answer the question of whether there are good and bad matchers. Uncertainty can also be reduced iteratively. In such a setting, initial assumptions are strengthened or discarded, thereby refining the initial measures of imperfection. In [28], uncertainty is refined by a comparison of K alignments, each with its own uncertainty measure (modeled as a fuzzy relation over the two ontologies) in order to improve precision of the matching results. Finally, the work in [16] introduced the notion of probabilistic schema mappings (correspondences), namely a set of mappings with a probability attached to each mapping; and, used it to answer queries with uncertainty about semi-automatically created mappings. Imprecise mappings can be further improved over time as deemed necessary, for example, within the settings of approximate data integration, see, e.g., [72].

Beside the work done along this line, there is still a need to understand better the foundations of modeling uncertainty in ontology matching in order to improve detection of mappings causing inconsistencies, e.g., via probabilistic reasoning, or to identify where the user feedback is maximally useful. In the dynamic applications it often occurs that there is no precise correspondence or a correspondence identified is not specific enough, hence, there is a need to choose a good enough one (with respect to application needs). In turn, this requires formalizing a link between ontology matching tools and information integration systems that support uncertainty.

9 Matcher Selection and Self-configuration

There are many matchers that are available nowadays. Often these perform well in some cases and not so well in some other cases. This makes the issues of (i) matcher selection, (ii) matcher combination and (iii) matcher tuning of prime importance.

Matcher Selection. The work on evaluation (§5) can be used in order to assess the strengths and the weaknesses of individual matchers by comparing their

results with task requirements. Often, there are many different constraints and requirements brought by the matching tasks, e.g., correctness, completeness, execution time, main memory, thereby involving multi-decision criteria. This problem has been addressed so far through, e.g., analytic hierarchy process [62] and ad hoc rules [45].

Matcher Combination. Beside matcher selection, another issue is the combination of individual matchers and libraries of matchers. This increases the complexity of the previous problem by allowing to put several matchers together and to combine them adequately. So far, only design time toolboxes allow to do this manually [13]. Another approach involves ontology meta-matching [50], i.e., a framework for combining a set of selected ontology matchers. The main issue here is the semi-automatic combination of matchers by looking for complementarities, balancing the weaknesses and reinforcing the strengths of the components.

Matcher Tuning. In dynamic settings, such as the web, it is natural that applications are constantly changing their characteristics. Therefore, approaches that attempt to tune and adapt automatically matching solutions to the settings in which an application operates are of high importance. This may involve the run time reconfiguration of a matcher by finding its most appropriate parameters, such as thresholds, weights, and coefficients. The work in [50] proposed an approach to tune a library of schema matchers at design time; while the work in [15] discussed consensus building after many methods have been used. The challenge, however, is to be able to perform matcher self-tuning at run time, and therefore, efficiency of the matcher configuration search strategies becomes crucial.

The above mentioned problems share common characteristics: the search space is very large and the decision is made involving multiple criteria. Notice that resolving these simultaneously at run time makes the problem even harder.

10 User Involvement

In traditional applications automatic ontology matching usually cannot deliver high quality results, especially on large datasets, see, e.g., [39]. Thus, for traditional applications, semi-automatic matching is a way to improve the effectiveness of the results. So far, there have only been few studies on how to involve users in ontology matching. Most of these efforts have been dedicated to design-time matcher interaction [13, 66].

Some recent work, however, has focussed on the ergonomic aspect of elaborating alignments, either for designing them manually or for checking and correcting them. The work in [27] proposed a graphical visualization of alignments based on cognitive studies. In turn, the work in [60, 61] has provided an environment for manually designing complex alignments through the use of connected perspective that allows to quickly deemphasize non relevant aspects of the ontologies

being matched while keeping the connections between relevant entities. This line of work must be still consolidated and it should be possible to seamlessly plug the results obtained here into an alignment management system (see §13). With the development of interactive approaches the issues of their usability will become more severe. This includes scalability of visualization [70] and better user interfaces in general, which are expected to bring big productivity gains; as from [5] even bigger than from more accurate matching algorithms.

There remains an interesting path to follow concerning user involvement: relying on the application users in order to learn from them what is useful in the alignments under consideration. This can be exploited either at the matcher level by adjusting its parameters and providing new (partial) input alignments, or at the alignment level by experimenting with confidence weights to improve the results given to the users. Another promising direction in this respect is what we call “implicit matching”, i.e., by serendipitously contributing to improve available alignments. For instance, in a semantic peer-to-peer system, if a user poses a query and there is no alignment in the system leading to an answer, this user may be willing to help the system by providing several correspondences that are necessary for answering the query. These correspondences can be collected by the system and, over time, the system will acquire enough knowledge about the useful correspondences. The example discussed can be also viewed as a part of typical interactions in a collaborative environment (see §12). The issue here is, both for design time and run time matching, to design interaction schemes which are burdenless to the user. At design time, interaction should be both natural and complete; at run time, it should be hidden in the user task.

Finally, let us note that dynamic applications have a specific feature that traditional applications have not: since there are multiple parties (agents) involved in the process, mismatches (mistakes) could be negotiated (corrected) in a fully automated way. This has already been considered in the field of multi-agent systems where raw alignments are refined by agent negotiation [17, 47]. Therefore, explanations of matching (see §11), being an argumentation schema, become crucial.

11 Explanation of Matching Results

In order for matching systems to gain a wider acceptance, it will be necessary that they can provide arguments for their results to users or to other programs that use them. In fact, alignments produced by matching systems may not be intuitively obvious to human users, and therefore, they need to be explained. Having understood the alignments returned by a matching system, users can deliberately edit them manually, thereby providing the feedback to the system, see [11, 47, 75] for the solutions proposed so far and [25] for their in-depth analysis.

A more recent work introduced the notion of a matchability score (computed via a synthetic workload), which quantifies how well on average a given schema matches future schemas [9]. Using the matchability score, different types of matching mistakes can be analyzed. Based on them a matchability report is

generated, thereby guiding users in revising the correspondences by addressing the reported mistakes together with the suggested revisions to be made.

Generally, the key issue here is to represent explanations in a simple and clear way to the user in order to facilitate informed decision making. In a longer term, it would be useful to standardize explanations/proofs of matching results in order to facilitate the interaction of matching systems with other programs.

12 Social and Collaborative Ontology Matching

Another way to tackle the matching task is to take advantage of the network effect: if it is too cumbersome for one person to come up with a correct alignment between several pairs of ontologies, this can be more easily solved by many people together. This comes from three aspects: (i) each person has to do a very small amount of work, (ii) each person can improve on what has been done by others, and (iii) errors remain in minority.

The work in [78] reported on early experiments with community-driven ontology matching in which a community of people can share alignments and argue about them by using annotations. Later the work in [65] proposed a collaborative system in the area of bio-informatics for sharing both ontologies and mappings (i.e., correspondences). It allows users to share, edit and rate these mappings. The strengths of this system are a user friendly interface with the possibility to annotate alignments and the direct connection with the ontologies which helps users to navigate. In turn, [57] proposed to enlist the multitude of users in a community to help match schemas in a Web 2.0 fashion by asking users simple questions and then learn from the answers to improve matching accuracy. Finally, the work on alignment server in [20] supported alignment storing, correspondence annotation and sharing, though it was more closely designed as a middleware component rather than a collaborative tool.

Collaborative and social approaches to ontology matching rely on infrastructures allowing for sharing alignments and annotating them in a rich way. These features can be used to facilitate alignment reuse. The current challenge in collaborative ontology matching is thus to find the right annotation support and the adequate description units to make it work at a large scale. In particular, contradictory and incomplete alignments should be dealt with in a satisfactory way. Other issues include understanding how to deal with malicious users, and what would be the promising incentive schemas to facilitate user participation.

13 Alignment Management: Infrastructure and Support

Alignments, like ontologies, must be supported during their life cycle phases by adequate tools and standards. These required functions can be implemented as services, the most notable of which are: (i) *match two ontologies* possibly by selecting an algorithm to be used and its parameters (including an initial alignment, see §2.2); (ii) *store an alignment* in a persistent storage; (iii) *retrieve an alignment* based on its identifier; (iv) *retrieve alignment metadata* such that its

identifier can be used to choose between specific alignments; (v) *find (stored) alignments* between two specific ontologies; (vi) *edit an alignment* by adding or discarding correspondences (this is typically the result of a graphic editing session); (vii) *trim alignments* based on a threshold; (viii) *generate code* implementing ontology transformations, data translations or bridge axioms based on a particular alignment; and (ix) *translate a message* with regard to an alignment.

This functional support must be complemented with rich metadata allowing users and systems to select the adequate alignments based on various criteria. It should also support permanent storage and identification of alignments in order to reliably use the existing alignments. In databases, several systems have been designed for offering a variety of matching methods and a library of mappings [4]. However, these were meant only as a component for design time integration and not as a service that can be used at run time. In turn, the alignment server [20] has been designed with this goal in mind. Notice that in the context of collaborative matching (§12) the above mentioned needs are vital.

We can distinguish two levels in alignment management: (i) the infrastructure middleware and (ii) the support environments that provide task related access to alignments. The support environments may be dedicated to alignment edition [60, 66], alignment processing, alignment sharing and discussing [65], or model management [59]. These two levels may be mixed in a single system [65] or kept clearly separated [20].

One of the challenges here is to provide an alignment support infrastructure at the web scale, such that tools and, more importantly, applications can rely on it in order to share, i.e., publish and reuse, alignments.

Moreover, the alignment life cycle (§2.3) is tightly related to the ontology life cycle: as soon as ontologies evolve, new alignments have to be produced following the ontology evolution. This can be achieved by recording the changes made to ontologies and transforming those changes into an alignment (from one ontology version to the next one). This can be used for computing new alignments that will update the previous ones. In this case, previously existing alignments can be replaced by their composition with the ontology update alignment (see Figure 4). As demonstrated by this evolution example, alignment management can rely on composition of alignments which, in turn, requires to reason about alignments (see §14).

14 Reasoning with Alignments

The ultimate goal of matching ontologies is to use alignments. For this purpose, they should be attributed a semantics. There have been developed various kinds of semantics [7, 51, 80] that allow to define the consequences of the aligned ontologies or distributed systems, i.e., several ontologies and several alignments.

At the level of alignments, an important question is what correspondences are the consequences of the aligned ontologies or distributed systems (α -consequences). This is important because it allows systems using alignments to take advantage of these correspondences, e.g., for transforming ontologies or translating messages. Computing α -consequences is used for finding missing alignments between two

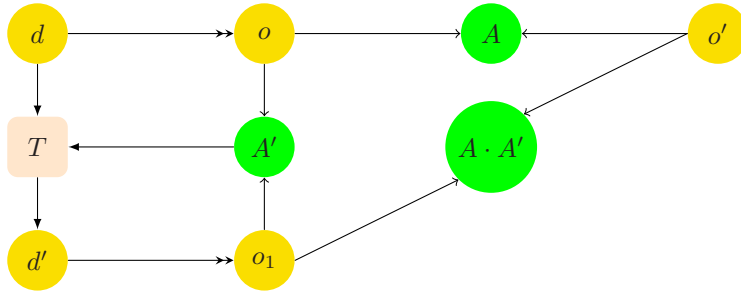


Fig. 4. Evolution of alignments [23]. When an ontology o evolves into a new version o_1 , it is necessary to update the instances of this ontology (d) and the alignment(s) (A) it has with other ontologies (o'). To this extent, a new alignment (A') between the two versions can be established and used for generating the necessary instance transformation (T) and for linking ($A \cdot A'$) the ontologies o_1 and o' .

ontologies or strengthening the existing alignments. This is useful for: (i) deducing alignments; (ii) evolving alignments (see Figure 4); (iii) checking alignment consistency and repairing alignments [58]; and (iv) evaluating alignments [21].

A weaker level of reasoning that can be implemented is an alignment composition. It consists of deducing correspondences holding between two ontologies from alignments involving other ontologies. We can distinguish between two kinds of alignment composition: full alignment composition and ontology-free alignment composition. The latter composes alignments without any access to ontologies. Hence, it cannot, in general find all consequences of ontologies, but only the so-called quasi-consequences [79]. All these kinds of reasoning are correct but not semantically complete, i.e., they will not find all α -consequences of a set of alignments. This can however be useful because they may be faster to obtain.

In database schema matching, the notion of mapping composition is prominent and has been thoroughly investigated [3, 55]. The problem here is to design a composition operator that guarantees that the successive applications of two mappings yields the same results as the application of their composition [26]. Similar studies should be performed in the context of ontology alignments with various ontology and alignment languages.

15 Conclusions

We discussed ten challenges for ontology matching, accompanied for each of these with an overview of the recent advances in the field. We believe that challenges outlined are on the critical path, hence, addressing them should accelerate progress of ontology matching. Moreover, these challenges are not isolated from each others: collaborative matching requires an alignment infrastructure; alignment evolution and other operations of alignment management require reasoning with alignments; user involvement would benefit from and contribute to collaborative matching; etc. Hence, these challenges, even if clearly identified will certainly have to be considered in prospective relation with each other.

Beside the mentioned challenges, much more work is needed in order to bring the matching technology to the plateau of productivity. This includes dealing with multilinguism, spatial matching for GIS applications, etc.

Acknowledgements. The first author appreciates support from the Trentino as a Lab (TasLab) project of the European Network of the Living Labs. The second author has been partially supported by the European integrated project NeOn (IST-2005-027595) and the RNTL project WebContent. We are thankful to the TasLab group members: Isabella Bressan, Ivan Pilati, Valentina Ferrari, Luca Mion and Marco Combetto for many fruitful discussions on the living labs methodology for innovation. We are grateful to Fausto Giunchiglia, Maurizio Marchese, Mikalai Yatskevich, Roberta Cuel (University of Trento), Lorenzo Vaccari (Urban Planning and Environment Protection department of the Autonomous Province of Trento), Marta Sabou (Open University), Antoine Zimmermann (INRIA), Zharko Aleksovski (Vrije Universiteit Amsterdam), and Malgorzata Mochol (Free University of Berlin) for the insightful comments on various aspects of ontology matching covered in this paper.

References

1. Aleksovski, Z.: Using background knowledge in ontology matching. Ph.D thesis, Vrije Universiteit Amsterdam (2008)
2. Ashpole, B., Ehrig, M., Euzenat, J., Stuckenschmidt, H. (eds.) Proceedings of the workshop on Integrating Ontologies at K-CAP (2005)
3. Bernstein, P., Green, T., Melnik, S., Nash, A.: Implementing mapping composition. *The VLDB Journal* (2008)
4. Bernstein, P., Halevy, A., Pottinger, R.: A vision of management of complex models. *ACM SIGMOD Record* (2000)
5. Bernstein, P., Melnik, S.: Model management 2.0: manipulating richer mappings. In: *Proceedings of SIGMOD* (2007)
6. Bernstein, P., Melnik, S., Petropoulos, M., Quix, C.: Industrial-strength schema matching. *ACM SIGMOD Record* (2004)
7. Borgida, A., Serafini, L.: Distributed description logics: Assimilating information from peer sources. *Journal on Data Semantics* (2003)
8. Castano, S., Ferrara, A., Lorusso, D., N  th, T., M  ller, R.: Mapping validation by probabilistic reasoning. In: Bechhofer, S., Hauswirth, M., Hoffmann, J., Koubarakis, M. (eds.) *ESWC 2008. LNCS*, vol. 5021. Springer, Heidelberg (2008)
9. Chai, X., Sayyadian, M., Doan, A., Rosenthal, A., Seligman, L.: Analyzing and revising mediated schemas to improve their matchability. In: *Proceedings of VLDB* (2008)
10. Cuel, R., Delteil, A., Louis, V., Rizzi, C.: Knowledge Web white paper: The Technology Roadmap of the Semantic Web (2007), <http://knowledgeweb.semanticweb.org/o2i/menu/KWTR-whitepaper-43-final.pdf>
11. Dhamankar, R., Lee, Y., Doan, A., Halevy, A., Domingos, P.: iMAP: Discovering complex semantic matches between databases schemas. In: *Proceedings of SIGMOD* (2004)
12. Do, H., Rahm, E.: COMA – a system for flexible combination of schema matching approaches. In: Bressan, S., Chaudhri, A.B., Li Lee, M., Yu, J.X., Lacroix, Z. (eds.) *CAiSE 2002 and VLDB 2002. LNCS*, vol. 2590. Springer, Heidelberg (2003)
13. Do, H., Rahm, E.: Matching large schemas: Approaches and evaluation. *Information Systems* (2007)

14. Doan, A., Halevy, A.: Semantic integration research in the database community: A brief survey. *AI Magazine* (2005); Special issue on Semantic integration
15. Domshlak, C., Gal, A., Roitman, H.: Rank aggregation for automatic schema matching. *IEEE Transactions on Knowledge and Data Engineering* (2007)
16. Dong, X., Halevy, A., Yu, C.: Data integration with uncertainty. In: *Proceedings of VLDB* (2007)
17. dos Santos, C., Moraes, M., Quaresma, P., Vieira, R.: A cooperative approach for composite ontology mapping. *Journal on Data Semantics* (2008)
18. Ehrig, M., Staab, S., Sure, Y.: Bootstrapping ontology alignment methods with APFEL. In: Gil, Y., Motta, E., Benjamins, V.R., Musen, M.A. (eds.) *ISWC 2005*. LNCS, vol. 3729, pp. 186–200. Springer, Heidelberg (2005)
19. Euzenat, J.: An API for ontology alignment. In: McIlraith, S.A., Plexousakis, D., van Harmelen, F. (eds.) *ISWC 2004*. LNCS, vol. 3298, pp. 698–712. Springer, Heidelberg (2004)
20. Euzenat, J.: Alignment infrastructure for ontology mediation and other applications. In: *Proceedings of the workshop on Mediation in Semantic Web Services* (2005)
21. Euzenat, J.: Semantic precision and recall for ontology alignment evaluation. In: *Proceedings of IJCAI* (2007)
22. Euzenat, J., Isaac, A., Meilicke, C., Shvaiko, P., Stuckenschmidt, H., Šváb, O., Svátek, V., van Hage, W., Yatskevich, M.: Results of the ontology alignment evaluation initiative 2007. In: *Proceedings of the workshop on Ontology Matching at ISWC/ASWC* (2007)
23. Euzenat, J., Mocan, A., Scharffe, F.: Ontology alignments: an ontology management perspective. In: *Ontology management: semantic web, semantic web services, and business applications*. Springer, Heidelberg (2008)
24. Euzenat, J., Mochol, M., Shvaiko, P., Stuckenschmidt, H., Svab, O., Svatek, V., van Hage, W., Yatskevich, M.: Results of the ontology alignment evaluation initiative 2006. In: *Proceedings of the workshop on Ontology Matching at ISWC* (2006)
25. Euzenat, J., Shvaiko, P.: *Ontology matching*. Springer, Heidelberg (2007)
26. Fagin, R., Kolaitis, P., Popa, L., Tan, W.: Composing schema mappings: Second-order dependencies to the rescue. *ACM Transactions on Database Systems* (2005)
27. Falconer, S., Storey, M.: A cognitive support framework for ontology mapping. In: *Proceedings of ISWC/ASWC* (2007)
28. Gal, A.: Managing uncertainty in schema matching with top-k schema mappings. *Journal on Data Semantics* (2006)
29. Gal, A., Anaby-Tavor, A., Trombetta, A., Montesi, D.: A framework for modeling and evaluating automatic semantic reconciliation. *The VLDB Journal* (2005)
30. Giunchiglia, F.: Managing diversity in knowledge. Keynote talk at *ECAI* (2006)
31. Giunchiglia, F.: Il ruolo degli enti di ricerca per lo sviluppo dell'ICT del Trentino (English translation: The role of the research centers in the development of Trentino). In: *Le Tecnologie Digitali nell'economia del Trentino* (2008)
32. Giunchiglia, F., Marchese, M., Zaihraye, I.: Encoding classifications into lightweight ontologies. *Journal of Data Semantics* (2007)
33. Giunchiglia, F., McNeill, F., Yatskevich, M., Pane, J., Besana, P., Shvaiko, P.: Approximate structure preserving semantic matching. In: *Proceedings of ODBASE* (2008)
34. Giunchiglia, F., Shvaiko, P., Yatskevich, M.: S-Match: an algorithm and an implementation of semantic matching. In: Bussler, C.J., Davies, J., Fensel, D., Studer, R. (eds.) *ESWS 2004*. LNCS, vol. 3053, pp. 61–75. Springer, Heidelberg (2004)
35. Giunchiglia, F., Shvaiko, P., Yatskevich, M.: Semantic schema matching. In: *Proceedings of CoopIS* (2005)

36. Giunchiglia, F., Shvaiko, P., Yatskevich, M.: Discovering missing background knowledge in ontology matching. In: Proceedings of ECAI (2006)
37. Giunchiglia, F., Shvaiko, P., Yatskevich, M.: Semantic matching. *Encyclopedia of Database Systems* (to appear, 2009)
38. Giunchiglia, F., Yatskevich, M.: Element level semantic matching. In: Proceedings of the workshop on Meaning Coordination and Negotiation at ISWC (2004)
39. Giunchiglia, F., Yatskevich, M., Avesani, P., Shvaiko, P.: A large scale dataset for the evaluation of ontology matching systems. *The Knowledge Engineering Review* (to appear, 2008)
40. Giunchiglia, F., Yatskevich, M., Giunchiglia, E.: Efficient semantic matching. In: Gómez-Pérez, A., Euzenat, J. (eds.) *ESWC 2005*. LNCS, vol. 3532, pp. 272–289. Springer, Heidelberg (2005)
41. Giunchiglia, F., Yatskevich, M., Shvaiko, P.: Semantic matching: Algorithms and implementation. *Journal on Data Semantics* (2007)
42. Giunchiglia, F., Zaihrayeu, I.: Lightweight ontologies. *Encyclopedia of Database Systems* (to appear, 2009)
43. Gligorov, R., Aleksovski, Z., ten Kate, W., van Harmelen, F.: Using google distance to weight approximate ontology matches. In: Proceedings of WWW (2007)
44. Hu, W., Qu, Y., Cheng, G.: Matching large ontologies: A divide-and-conquer approach. *Data and Knowledge Engineering* (to appear, 2008)
45. Huza, M., Harzallah, M., Trichet, F.: OntoMas: a tutoring system dedicated to ontology matching. In: Proceedings of the workshop on Ontology Matching (2006)
46. Kalfoglou, Y., Schorlemmer, M.: Ontology mapping: the state of the art. *The Knowledge Engineering Review* (2003)
47. Laera, L., Blacoe, I., Tamma, V., Payne, T., Euzenat, J., Bench-Capon, T.: Argumentation over ontology correspondences in MAS. In: Proceedings of AAMAS (2007)
48. Lambrix, P., Tan, H.: A tool for evaluating ontology alignment strategies. *Journal on Data Semantics* (2007)
49. Larson, J., Navathe, S., Elmasri, R.: A theory of attributed equivalence in databases with application to schema integration. *IEEE Transactions on Software Engineering* (1989)
50. Lee, Y., Sayyadian, M., Doan, A., Rosenthal, A.: eTuner: tuning schema matching software using synthetic scenarios. *The VLDB Journal* (2007)
51. Lenzerini, M.: Data integration: A theoretical perspective. In: Proceedings of PODS (2002)
52. Madhavan, J., Bernstein, P., Doan, A., Halevy, A.: Corpus-based schema matching. In: Proceedings of ICDE (2005)
53. Madhavan, J., Bernstein, P., Domingos, P., Halevy, A.: Representing and reasoning about mappings between domain models. In: Proceedings of AAAI (2002)
54. Madhavan, J., Bernstein, P., Rahm, E.: Generic schema matching with Cupid. In: Proceedings of VLDB (2001)
55. Madhavan, J., Halevy, A.: Composing mappings among data sources. In: Proceedings of VLDB (2003)
56. Marchese, M., Vaccari, L., Shvaiko, P., Pane, J.: An application of approximate ontology matching in eResponse. In: Proceedings of ISCRAM (2008)
57. McCann, R., Shen, W., Doan, A.: Matching schemas in online communities: A web 2.0 approach. In: Proceedings of ICDE (2008)
58. Meilicke, C., Stuckenschmidt, H., Tamilin, A.: Repairing ontology mappings. In: Proceedings of AAAI (2007)
59. Melnik, S., Rahm, E., Bernstein, P.: Developing metadata-intensive applications with Rondo. *Journal of Web Semantics* (2003)

60. Mocan, A.: *Ontology-based data mediation for semantic environments*. Ph.D thesis, National University Ireland Galway (2008)
61. Mocan, A., Cimpian, E., Kerrigan, M.: Formal model for ontology mapping creation. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L.M. (eds.) *ISWC 2006*. LNCS, vol. 4273, pp. 459–472. Springer, Heidelberg (2006)
62. Mochol, M., Jentzsch, A., Euzenat, J.: Applying an analytic method for matching approach selection. In: *Proceedings of the workshop on Ontology Matching* (2006)
63. Nottelmann, H., Straccia, U.: *Information retrieval and machine learning for probabilistic schema matching*. Information Processing and Management (2007)
64. Noy, N.: *Semantic integration: A survey of ontology-based approaches*. ACM SIGMOD Record (2004)
65. Noy, N., Griffith, N., Musen, M.: Collecting community-based mappings in an ontology repository. In: *Proceedings of ISWC* (2008)
66. Noy, N., Musen, M.: The PROMPT suite: interactive tools for ontology merging and mapping. *International Journal of Human-Computer Studies* (2003)
67. Parent, C., Spaccapetra, S.: *Issues and approaches of database integration*. Communications of the ACM (1998)
68. Rahm, E., Bernstein, P.: A survey of approaches to automatic schema matching. *The VLDB Journal* (2001)
69. Robertson, D.: A lightweight coordination calculus for agent systems. In: *Declarative Agent Languages and Technologies* (2004)
70. Robertson, G., Czerwinski, M., Churchill, J.: Visualization of mappings between schemas. In: *Proceedings of CHI* (2005)
71. Sabou, M., d'Aquin, M., Motta, E.: Exploring the semantic web as background knowledge for ontology matching. *Journal on Data Semantics* (to appear, 2008)
72. Sarma, A., Dong, X., Halevy, A.: Bootstrapping pay-as-you-go data integration systems. In: *Proceedings of SIGMOD* (2008)
73. Shvaiko, P., Euzenat, J.: A survey of schema-based matching approaches. *Journal on Data Semantics* (2005)
74. Shvaiko, P., Giunchiglia, F., Bundy, A., Besana, P., Sierra, C., van Harmelen, F., Zaihrayeu, I.: *OpenKnowledge Deliverable 4.2: Benchmarking methodology for good enough answers* (2008), <http://www.cisa.informatics.ed.ac.uk/OK/Deliverables/D4.2.pdf>
75. Shvaiko, P., Giunchiglia, F., Pinheiro da Silva, P., McGuinness, D.: Web explanations for semantic heterogeneity discovery. In: Gómez-Pérez, A., Euzenat, J. (eds.) *ESWC 2005*. LNCS, vol. 3532, pp. 303–317. Springer, Heidelberg (2005)
76. Sure, Y., Corcho, O., Euzenat, J., Hughes, T. (eds.): *Proceedings of the workshop on Evaluation of Ontology-based tools* (2004)
77. Zhang, S., Bodenreider, O.: Experience in aligning anatomical ontologies. *International Journal on Semantic Web and Information Systems* (2007)
78. Zhdanova, A., Shvaiko, P.: Community-driven ontology matching. In: Sure, Y., Domingue, J. (eds.) *ESWC 2006*. LNCS, vol. 4011, pp. 34–49. Springer, Heidelberg (2006)
79. Zimmermann, A.: *Sémantique des connaissances distribuées*. Ph.D thesis, Université Joseph-Fourier, Grenoble (FR) (2008)
80. Zimmermann, A., Euzenat, J.: Three semantics for distributed systems and their relations with alignment composition. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L.M. (eds.) *ISWC 2006*. LNCS, vol. 4273, pp. 16–29. Springer, Heidelberg (2006)

Dynamic Labelling Scheme for XML Data Processing

Maggie Duong and Yanchun Zhang

School of Computer Science and Mathematics
Victoria University, Australia
`maggie.duong1@live.vu.edu.au`
`yanchun.zhang@vu.edu.au`

Abstract. Extensive research has been conducted on labelling schemes, however, most of proposed labelling schemes are costly due to the need of re-calculating or re-labelling existing nodes whenever XML documents being updated. In our view, an effective labelling scheme needs to be (i) Compact, total lengths of labels are as small as possible. (ii) Dynamic, being able to update XML data dynamically without re-labelling or re-calculating value of existing nodes. (iii) Last but not least, facilitating the identification of various relationships between nodes. In this paper, we develop a labelling scheme, the Compressed Dynamic Labelling scheme which meets the above requirements. Furthermore, with our compressed labelling scheme, total lengths of labels are reduced significantly comparing with some existing labelling schemes. Our experimental works have shown advantages of the proposed scheme.

Keywords: XML labeling scheme, XML update, query processing.

1 Introduction

With the advanced characteristics of XML such as platform independent, etc. more and more XML documents have been used to display and exchange information on the web. To response to this emergence, several works have been developed to store and query XML data. XPath and XQuery [4,9] are query languages developed by W3C group for processing XML data. XPath and XQuery are both strongly typed as declarative queries. They use path expressions to traverse XML data irregularly.

Since queries navigate XML data via path expressions, indexes can be used to accelerate queries. To index an XML data, each node in an XML tree is given a unique code. It is also necessary to label nodes in such a way that they can show relationship between any two given nodes [23]. Once these are done, structural queries can be simply answered by using the developed index. This will help to reduce the number of nodes that would otherwise need to be accessed for the search.

For this reason, many researchers have been trying to develop effective indexes for XML data [5,19,20,22,26,28,31]. Their techniques vary from path indexing to labelling/ numbering scheme. For example, the works [3,8,15,24,25] use path

indexing. [2] and [33] use region - based numbering scheme. [7,19,20,34] use prefix - based labelling schemes. [22,29] use Dewey prefix-based numbering. [21] employs Dietz's numbering scheme.

While above indexing techniques can facilitate query processing, problems still exist. XML data have an intrinsic order. That means XML data orders its nodes corresponding to the order in which a sequential read of the textual XML would encounter the nodes [15]. When a new node is being inserted in XML data, labels of existing nodes are affected. Re-labelling or re-calculating values for existing nodes are then required. Several works have been developed to eliminate this problem, such as [11,19,23,26,31]. However, sizes of labels of [11,19,20,23,26] are not compact. [31] needs to recalculate values of exiting nodes when order - sensitive XML data is updated. [22] is not suitable for the update because the labels of whole nodes and the finite state transducer must be reconstructed after data insertions. Similar for [5], when new insertions occur, the global position and the length of each segment must be relabelled.

To the best of our knowledge, none of existing labeling schemes are compact and support updating XML dynamically when inserting order-sensitive nodes. In this paper, we propose a dynamic labelling scheme that has the following advantages:

- Compact and space efficient; total lengths of labels are relatively small.
- Dynamic; being able to update XML data dynamically without re-labelling or re-calculating value for existing nodes.
- Last but not least, supporting the representation of the ancestor - descendant relationships and sibling relationships between nodes such as parent, child, ancestor, descendant, previous - sibling, following - sibling, previous, following.

```

<catalog>
  <book>
    <title>A First Book of C++</title>
    <author>
      <last>Bronson</last>
      <first>Bronson</first>
    </author>
    <publisher>PWS Publishing</publisher>
    <year>1997</year>
  </book>
  <book>
    <title>Usability Engineering</title>
    <author>
      <last>Rosson</last>
      <first>Mary</first>
    </author>
    <author>
      <last>Carroll</last>
      <first>John</first>
    </author>
    <publisher>Morgan Kaufmann</publisher>
    <year>2002</year>
  </book>
  <book>
    <title>An Overview of Computer Science </title>
    <author>
      <last>Brookshear</last>
      <first>J.</first>
    </author>
    <publisher>Addison-Wesley</publisher>
    <year>2005</year>
  </book>
</catalog>

```

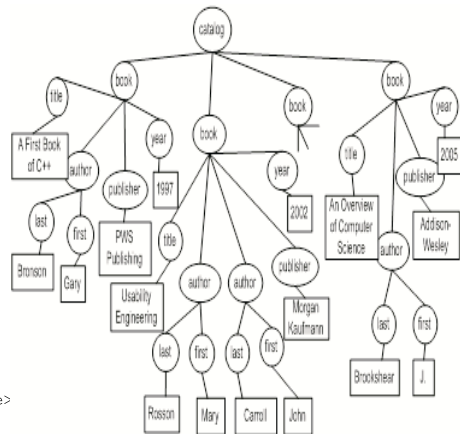


Fig. 1. An XML document & tree structure

The rest of this paper is organized as follows, in section 2, we shall discuss related works in path indexing, and labelling schemes for XML data. Section 3 presents our dynamic labelling schemes, the loose LSDX and Com-D schemes. Section 4 will present some experiments based on the Com-D technique. Section 5 concludes this paper.

2 Related Works

There are several path-indexing, labelling or numbering schemes that have been developed to facilitate query processing for XML data. For example, [15] uses pre-order traversal and post-order traversal of the data tree to label nodes. This technique still requires re-indexing when the XML data is updated. It is because they assign the code for each node in the order where the XML data is entered, post - order is reversed. Thus, once the XML data is changed, all the codes of related nodes also need to be changed. A similar technique is used in [3].

[19,20] proposed prefix labelling schemes using binary string to label nodes. They compare codes based on the lexicographical order rather than the numerical order. In general, it works as follows.

The root node is labelled with an empty string. The self-label of the first (left) child node is "01". The self-label of the last (right) child node is "011". The purpose of choosing to use "01" and "011" as the first and last sibling self-labels because they want to insert nodes before the first sibling and after the last sibling. Once they have assigned the left and right self-labels, they label the middle self - label using these two rules:

Case (a): IF left self-label size $\leq$ right self-label size. When adding the middle self - label, they change the last character of the right self - label to "0" and concatenate one more "1".

Case (b): IF left self-label size $>$ right self-label size. They directly concatenate one more "1" after the left self label.

Thus, they label the middle child node, which is the third child, i.e. $\lfloor (1 + 5) / 2 \rfloor = 3$. The size of the 1st (left) self-label ("01") is 2 and the size of the 5th (right) self label ("011") is 3 which satisfies Case (a), thus the self label of the third child node is "0101" ("011" $\rightarrow$ "010" $\rightarrow$ "0101").

Next, they label the two middle child nodes between "01" and "0101", and between "0101" and "011". For the middle node between "01" (left self-label) and "0101" (right self-label), i.e. the second child node $\lfloor (1 + 3) / 2 \rfloor = 2$, the left self-label size 2 is smaller than the right self label size 4 which satisfies Case (a), thus the self label of the second child is "01001" ("0101" $\rightarrow$ "0100" $\rightarrow$ "01001").

For the middle node between "0101" (left self-label) and "011" (right self-label), i.e. the fourth child $\lfloor (3 + 5) / 2 \rfloor = 4$, the left self-label size 4 is larger than the right self-label size 3 which satisfies Case (b), thus the self label of the fourth child is "01011" ("0101" + "1" $\rightarrow$ "01011").

This prefix labelling scheme uses following theorems:

- The sibling self-labels of ImprovedBinary are lexically ordered.

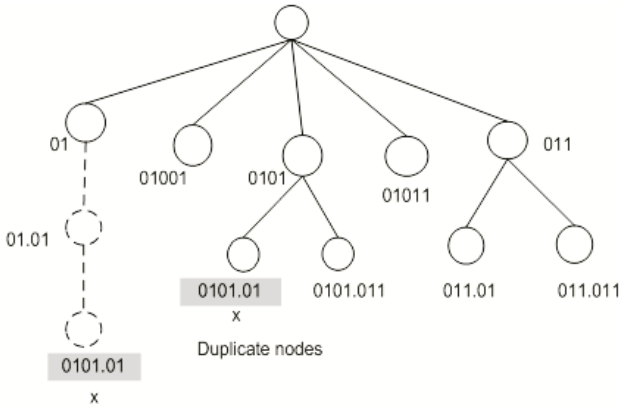


Fig. 2. ImprovedBinary Scheme - Updating Problem

- The labels (prefix-label + delimiter + self-label) of ImprovedBinary are lexically ordered when comparing the labels component by component.

For example, self-labels of the five child nodes of the root (first level child nodes from the root) in Figure 2 are lexically ordered, i.e. "01" < "01001" < "0101" < "01011" < "011" lexically. Similarly, "0101.011" < "011.01" lexically.

Let us trying to add some more nodes to the existing nodes, let's say we want to add a child node to the node "01", as it is the first (left) child node, the code for the new node is "01.01". It looks fine. Then if we adding further child node for this node, the code of the new node will be "0101.01". This causes a conflict with the existing node. See Figure 2. We have duplicate nodes here.

[29] uses Dewey prefix - based numbering scheme. To minimize renumbering cost, gaps are used when assigning labels for nodes. Similar characteristics can be found in the work of [34]. They propose a prefix - based PBi (Perfect Binary) Tree scheme, which uses preserved codes between every two - child nodes to reduce the possibility of renumbering all siblings and their descendants when updating is needed. Although these approaches support updating XML data, this technique is not flexible because codes must be reserved before hand. In addition to that, when all reserved codes are used up, renumbering has to be done again.

[7] and [34] use prefix - based labelling schemes. [7] proposes two prefix - based labelling schemes to assign a specific code to each child of a node v . The first approach is one-bit growth. For instance, the first child's code of the node v is "0" which is labelled as $L(v).0$. The second child's code of the node v is "10" which is labelled as $L(v).10$. The third child's code is "110" which is labelled as $L(v).110$. Hence, the i th child's code is repeated with '1' for each child's code that ends with "1", together with a "0" attached at the end.

The second approach is double-bit growth, suppose that u_i 's code is $L(v).L'(u_i)$ where $L(v)$ is its direct parent code, it assigns its children nodes as $L'(u_1) = 0$, $L'(u_2) = 10$, $L'(u_3) = 1100$, $L'(u_4) = 1101$, $L'(u_5) = 1110$, $L'(u_6) = 11110000$,

etc. In general, it increases the binary code represented by $L'(ui)$ by 1, that means to assign $L'(ui + 1)$. However, if the representation of $L'(ui)$ consists of all ones, it doubles its length by adding a sequence of zeros.

Due to the ways prefix based numbering scheme assigning bits as a prefix to a node, sometimes renumbering is still required. For instance, when a new node v is added as the i th position, the code for those nodes originally at v_i and v_i+1 , $v_i+2 \dots v_n$ need to reallocate by one position. Therefore, all nodes in the sub trees rooted at and v_i+1 , $v_i+2 \dots v_n$ need to be renumbered.

To overcome that disadvantage, [23] proposes a labelling scheme using Group Based Prefix (GRP) labelling. Its technique is to divide a big tree into many small groups and using prefix labelling scheme to label them.

[26] also uses Dewey - like numbering scheme (ORDPaths) to label each nodes. The difference is that it starts with the odd numbers for initial load. Such as 1.1, 1.3, 1.5, 1.3.1, 1.3.3, 1.5.1, 1.5.3, etc. When new nodes are inserted, it uses even "careting-in" between sibling nodes without re-labelling. However, this scheme is not space efficient. With an XML tree that increasing in depth and fan-out, the size of labels generated by this scheme will increase fast.

[31] uses Prime numbers to label XML nodes. The label of a node is the product of its parent label (a prime number) and its self label (the next available prime number). Ancestor - descendant relationship between two nodes is determined if one can exactly divide the other. Such as node u is ancestor of node v if and only if $\text{label}(v) \bmod \text{label}(u) = 0$.

In order to maintain document order, Prime employs a table of Simultaneous Congruence (SC) values to keep order for each element. Thus, the order of a node is calculated by $\text{SC} \bmod \text{self-label}$. For instance, the document order of the node which labeled as '5' is 3, which is calculated by 29243 (a SC value) mod 5.

Prime does not need to re-label any existing nodes when new nodes are inserted in XML tree. However, it needs to re-calculate the SC value to keep the new ordering of the nodes. Furthermore, Prime has to skip a lot of numbers to obtain a prime number, as the result, products of primes can become quite big. Additionally, re-calculating Simultaneous Congruence values every time a new node is inserted is quite time consuming.

In [21], advantages of Dietz's numbering scheme, which is to use pre-order and post-order to determine the ancestor - descendant relationship between any pair of tree nodes, are taken into account. They then propose a new numbering scheme that uses pre-order and a range of descendants to reserve additional number space for future insertions. Their proposed numbering scheme associates each node with a pair of numbers $\langle \text{order}, \text{size} \rangle$. Comparing to Dietz's scheme, their numbering scheme is more flexible and can somehow deal with the issue of dynamic updating XML data. This can be possible because extra spaces are reserved before hand. Nevertheless, when all the reserved spaces are used up, renumbering affected nodes shall be required.

Similar technique is used in [2], nodes are labelled with 'start' and 'end' of the interval by using floating point values. When a new node is inserted, this technique may not need to re-label existing nodes due to the available values between two

floating point numbers. However, there is finite number of values between any two floating point numbers. When all the available values are used up, re-labelling has to be done. Thus, this technique can not quite solve the problem.

The essential difference between our Com-D labelling scheme and other existing labelling schemes is that it is a dynamic labelling scheme for dynamic XML data and it is compact. It does not matter where new nodes should be inserted or how many of new nodes are added. It is guaranteed that none of existing nodes needs to be re-labelled and no re-calculation is required. Moreover, as proved in our experiments, the total lengths of labels of our Com-D labelling scheme is space efficient, more compact than other labelling schemes. In addition to those advantages, our Com-D labelling scheme also supports the representation of the ancestor - descendant relationships and sibling relationships between nodes.

3 Dynamic Labelling Schemes

A dynamic labelling scheme differs from other labelling schemes as it supports updating XML data dynamically without the need of re-labelling existing nodes. In this section, we will present two dynamic labelling schemes. In the following subsections, we will introduce a primitive labelling scheme. The improved one will be presented in the next section. We called the primitive scheme a loose LSDX labelling scheme and the improved one, Com-D labelling scheme.

3.1 Loose LSDX Labelling

The loose labelling scheme is demonstrated in Figure 3. We first give document element "catalog" an "a". As there is no parent node for the document element, we assign "0" at the front of that "a". "0a" is the unique code for the document element "a" is the unique code for the document. For the children nodes of "0a", we continue with the next level of the XML tree which is "1" then the code of its parent node which is "a" and a concatenation ".". We then add a letter "b" for the first child, letter "c" for the second child, "d" for the third child and so on. Unique codes for children nodes of "0a" shall be "1a.b", "1a.c", "1a.d", etc.

From level 1 of the XML tree and downwards, we choose to use letter "b" rather than "a" for the first child of the document element. The concatenation "." is employed to help users figure out relationship between ancestor and descendant nodes. For example, just by looking at node "1a.b", one will realize that it is a descendant of node "0a".

There is a little difference in using the concatenation "." at the third (level 2) and other lower levels of the XML tree. As shown in Figure 3 the unique code for the first child of node "1a.b" is "2ab.b" rather than "1a.b.b". We intentionally choose to name it that way because we do not want to confuse users with too many ".".

Highlights of Loose Labelling Scheme. Loose LSDX labelling scheme is a dynamic labelling scheme, when XML data is required for updating, there is

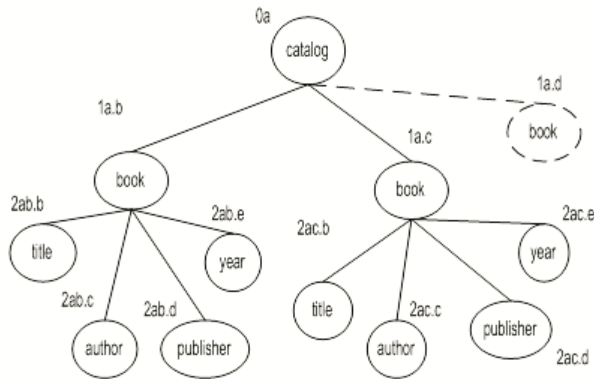


Fig. 3. Loose LSDX

no need to re-label affected nodes, hence facilitating fast update. It can also identify all important axes in XQuery and XPath, such as ancestor - descendant relationships, sibling relationships between nodes. However, regarding to the length of labels, there's room for improvement for this labelling to become more compact and space efficient. We will discuss this in the next section.

3.2 Com-D Labelling Scheme

We develop a new technique to label XML tree to make them smaller and more compact. We name our improve version of the loose LSDX as a compressed dynamic labelling scheme or Com-D for short.

The basic concept for adding new nodes of our Com-D scheme works similarly to primitive scheme. However, there are slight differences in this technique to make this scheme compact, space efficient than primitive version.

Let's consider a large XML document with hundreds of siblings, there are chances that some part of the node label will be repeated. For example, for node labelled "0101010111", "01" is repeated four times, for node "0110110011", "011" is repeated two times. Similarly, nodes "bcbcbcb", "bbbc", "abcabcddd" all have repetitive labels.

Therefore, we can make our loose labelling more compact by rewriting these labels using the following technique. Below are some of the examples that labels can be rewritten.

Example 1. "aaaaabbbbcdde" $\rightarrow$ "5a3bc2de"

Example 2. "aaabbcdddddddee" $\rightarrow$ "3a2bc8d2e"

Example 3. "kkkkkkkkkkokkkkkkkkkkobbabb" $\rightarrow$ "2(10ko)5b"

Example 4. "eaabcbcbcaabcbcbcx" $\rightarrow$ "e2a3(bc)2a3(bc)x" $\rightarrow$ "x2(2a3(bc))x"

Example 5. "eeebbbbcbbbbcbbbbcbbbbcbbbbcbbbbc" → "3e7(4bc)"

Com-D Labelling. We start with empty string "0" for the root document. For the children of the root document, we start with its tree level + ".", + a letter "b" for the first child, letter "c" for the second child, "d" for the third child and so on. Unique codes for children nodes of root document shall be "1,b", "1,c", and shall continue up to "1,z", "1,zb" to "1,zz". Since there are repetitive letters, "zz", we can replace them by number of occurrence of "z" and the letter "z" itself. That means "zz" shall be replaced by "2z". It can be continued with "2zb" to "3z". Repetitive letter is not counted if the concatenation "." stand in middle. For example, "b.bc" is considered as having no repetitive letter.

Similarly for all child nodes of node "b", the unique code for the first child is its tree level + ".", + code of its parent node concatenating with the "." and a letter "b" for the first child, letter "c" for the second child, "d" for the third child and so on. For instance, the first child of node "b" is "1,b.b", the second child of "b" is "1,b.c", and the third child of "b" is "1,b.d" and so on.

It is important to keep in mind that when generating unique code for child nodes, the "." from the code of parent node shall be removed. This is done to minimize the number of "." needed while maintaining its advantage in showing relationship between nodes. In general, generating unique codes works as follows:

Suppose node u is the first child of node v. Rule for generating unique code for node u will consist of these three following steps:

1. *Get the code of node v, remove "." if have, check for repetitive letters. If any letter appears more than once, it shall be accumulated and replaced by number of its occurrence + the letter itself.*
2. *Add concatenation dot "."*
3. *Add "b" if it is the first child of node v. Add "c" for the second child and add "d" for the third child of node v. The labeling continues for the rest of child nodes in alphabetical order. If any repetitive letter occurs again, it shall be replaced by number of its occurrence + the letter itself.*

3.3 Updating

For updating XML data, our labelling scheme can generate unique codes for every new node without re-labelling existing nodes. It does not matter where new nodes shall be. The rule for generating unique codes for new nodes is described below.

Updating rule: If there is no node standing before the place that a new node shall be added, unique code of new node is the code of its following sibling node minus one value from the last letter. If the last letter of the code of the new node is "a", attach "b" at the end. Otherwise, keep counting from the code of its preceding sibling so that the code for the new node will be greater than the code of its preceding sibling and less than the code of its following sibling (if have) in alphabetical order. If the code of its preceding node ends with "z", attach "b" at the end.

To get into more details of this compact labelling, let us use some insertions to show how it works. We categorize two insertion situations, one is to insert a new node that have no preceding-sibling and the other, have preceding-sibling. We call these two situations as Insert Before and Insert After operation respectively.

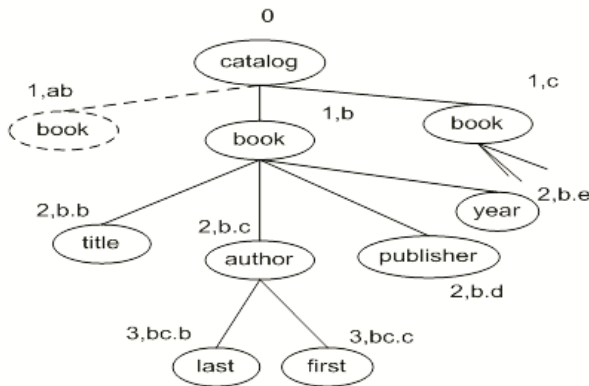


Fig. 4. Insert a node before a given node

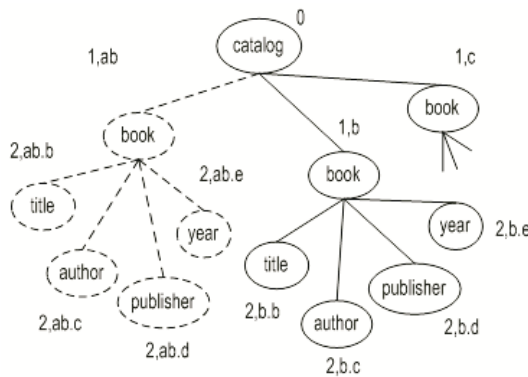


Fig. 5. Insert a sub tree before a given node

3.4 Insert Before

Insert before is inserting a node/sub tree before any given node which have no preceding-sibling. For instance, Figure 4 shows inserted node with dot lines. If we want to add a node before the node "b", we will just follow the updating rule. In this case, there is no node standing before the node "b", thus we get code "b" minus by one value, which is "a". As our rule said, if the last letter is "a", attach "b" at the end. Thus, the code for the new node shall be "ab". See Figure 4.

All children of the new node of "ab" will have "ab." attached at front, then a letter "b" for the first child, "c" for second child, "d" for the third child and so on. See Figure 5.

The need for more insertions might continue to arise in the future. Just simply apply updating rules to generate unique label for each new node. For example, if we need to insert another new node before the node "ab", the unique code for the new node will be "aab" or "2ab" after compressed. See Figure 6. Nodes from

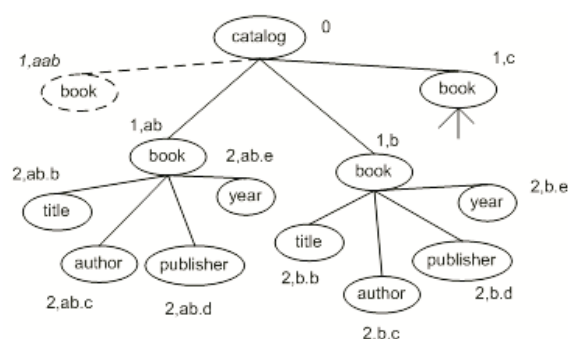


Fig. 6. Inserting a node before a given node

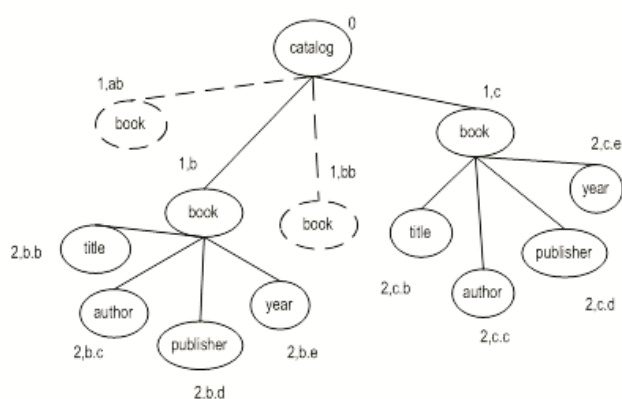


Fig. 7. Example of inserting a node after a given node

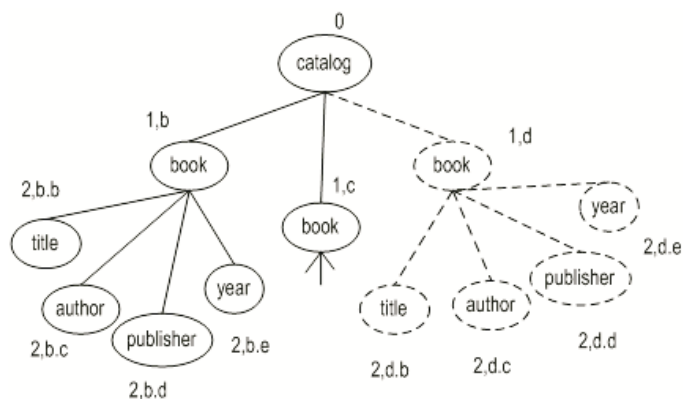


Fig. 8. Example of inserting a sub tree after a given node

"aab" to "aaz" can be used when more insertions are needed. This technique can be utilized repeatedly.

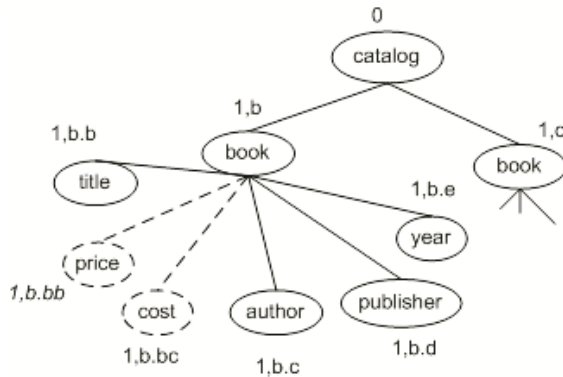


Fig. 9. Order-sensitive updates - Adding new elements

3.5 Insert After

Insert After is inserting a new node after any given node. Insert after differs from insert before because there must be a preceding node before the space that is intended for Insert After operation. However, there might be no following sibling at all. Example is given in Figure 7 with dot lines. If we want to add a new node after the node "c", in this case, the preceding node is "c". There is no following node. Thus, we need to continue counting from "c" to generate a code, which shall be greater than "c" in alphabetical order. The code for the new node will be "d". If another new node is needed for insertion after "d", its code will be "e" and shall continue up to "z", "zb" to "zz", etc.

Suppose now we want to store two more fields, *price* and *cost* of each book in our database. These new two fields are intrinsic ordered, say after the *Title* and before the *Author* node. Using our rules for adding new nodes, unique codes can be generated for the two new elements without re-labelling enormous nodes already existed in data file and still maintain order of data. Figure 9 illustrates this situation.

3.6 Deleting

Deleting single node or sub tree does not affect any other nodes in XML data tree as long as that single node has no children. In the other word, deleting a node that has child nodes means that all its child nodes will also be deleted. Code of deleting node/sub tree can be used again when a new node/sub tree is inserted in its place.

3.7 Relationships between Nodes

By adding codes of the parent nodes to the codes of child nodes, it helps us to determine all important relationships between nodes: parent, child, ancestor, descendant, previous - sibling, following - sibling, previous, following.

For instance, in Figure 9, by knowing a node called "1,b.b", we can understand that its parent is "b" and all the nodes beginning with "1,b." are its siblings. Precisely, for all other nodes that start with "1,b." and the remaining letters of their codes (after the ".") is less than "b" in alphabetical order, those nodes are preceding - siblings of node "1,b.b". If the remaining letters of their codes are greater than "b" in alphabetical order, they will be following - siblings of node "1,b.b". All children nodes of node "1,b.b" shall have "2,bb." attached at front.

3.8 Order-Sensitive Queries

Com-D labelling scheme can be used in all kinds of ordered queries. Ordered queries like Position = n, Preceding, Following, Preceding-sibling and Following-sibling can be answered by evaluating labels of nodes. For instance, the query "/play/act[3]" can be retrieved by first select all act nodes that are descendants of "play". Then return the third act.

Preceding and Following queries like "/play/act[3] /preceding::*" or "following::*" can be answered by comparing the order of all node labels occur before or after with the act[3] node label respectively, descendants of act[3] are ignored.

Preceding-sibling and Following-sibling queries such as "/play/act[3]/following-sibling::act" or "preceding-sibling:: act" retrieve all acts that are sibling of act[3] and then output all nodes after act[3] or before act[3] respectively in document order.

4 Experiments

We have conducted experimental works to compare our proposed Com-D labelling scheme with some other schemes such as ORDPaths scheme [26], Dewey scheme [29], GRP and SP labelling scheme [17] to observe its performance. All

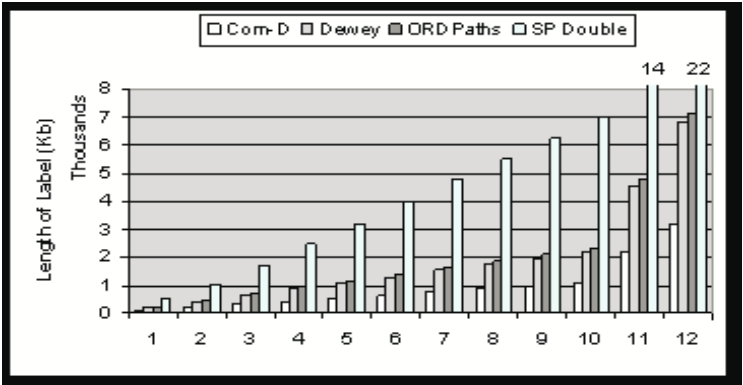


Fig. 10. Space requirements for each labelling scheme

Table 1. Query performance

	Test Queries	Number of nodes returned	Response Time (ms)
Q1	/play/act[5]	185	16
Q2	/play/act/scene[2]preceding::scene	855	15
Q3	/play/act	925	0
Q4	/play/act/scene/speech[4]	3545	250
Q5	/play/act/scene	3740	0
Q6	/play/act/scene/speech[3]preceding-sibling::speech	7280	266
Q7	/play/act/scene/speech/line[2]	85445	1343
Q8	/play/act/scene/speech[2]following-sibling::speech	147275	412
Q9	/play/act/scene/speech	154665	110
Q10	/play/act/scene/speech/line	534410	422

of our experiments are performed on the Pentium IV 2.4G with 512MB of RAM running on windows XP with 25G hard disk.

We use Java 1.4.2 and SAX from Sun Microsystems as the XML parser. For the database, [27] giving a balanced XML document which usually comes across in real - world situations. We use XMark datasets to create from various sizes of data for experimental purposes.

We carry out three sets of experiments to evaluate the performance of four labelling schemes. The first set compares the storage requirements of four schemes. The second set examines the query performance and the last set investigates the order-sensitive update and study the numbers of nodes which might require re-labelling.

4.1 Storage Requirement

Our first experiment is to compare storage space of labels with some other labelling schemes such as ORDPaths, SP double bit growth and Dewey labelling.

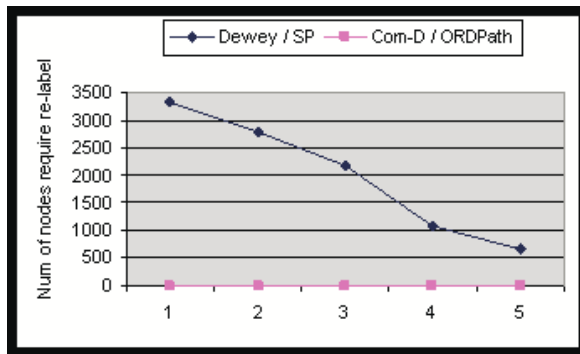


Fig. 11. Numbers of nodes need re-labelling

Table 2. Documents used in experiments

Document#	File Size (Kb)	Total Number of Nodes
D1	1155	17132
D2	2330	33140
D3	3485	50266
D4	4727	67902
D5	5736	83533
D6	6984	100337
D7	8145	118670
D8	9371	134831
D9	10483	151289
D10	11597	167865
D11	23365	336244
D12	34866	501498

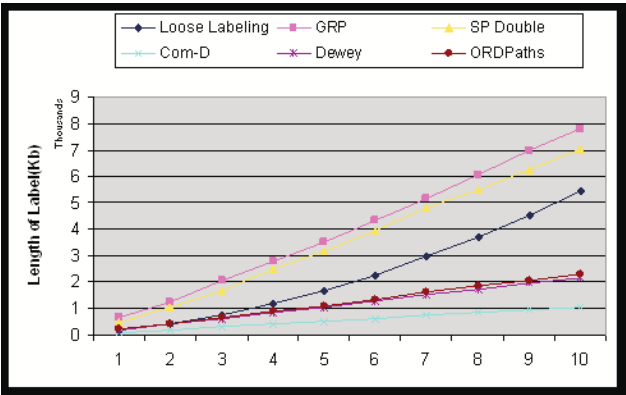


Fig. 12. Total length of label

These experiments indicate that our Com-D labelling scheme is superior to all ORDPPaths, SP one bit and double bit growth, GRP, LSDX and Dewey labelling. Results of these experiments show in Figure 10. To avoid graph clustering, SP one bit, GRP and LSDX results are not included in this paper.

4.2 Query Performance

In this experiment, we test the query performance using our new labelling scheme. We use Shakespeare’s play dataset [18] for this purpose. In order to see its real performance on huge XML data, we increase the Shakespeare’s play dataset 5 times. Ten queries used in this experiment are shown in Table 1 with number of nodes returned by these queries and theirs response time.

4.3 Update Performance

In this experiment, we run several updates to an XML file to measure order-sensitive update performance among several labelling schemes. We use the Dream XML file in Shakespeare's play since all elements in the file are order - sensitive. Dream contains 5 acts; we add a new act before and between existing acts. We then calculate number of nodes that need to be re-labelled for each case. Figure 11 shows the number of node that requires re-labelling. In all 5 cases, Com-D and ORDPaths labelling schemes need not to re-label any existing nodes. Dewey and SP schemes need to re-label a huge amount of nodes. Figure 11 shows the result of this experiment.

For Prime labelling scheme, there is no node which needs to be re-labelled, however, in order to maintain the order - sensitive of XML nodes, there is a number of nodes required re-calculating SC value. Since the performance of Prime is reported in [31], we omit it in this experiment.

5 Conclusions

In this paper, we proposed dynamic labelling schemes that support updating XML data dynamically without the need of re-labelling existing nodes, hence facilitating fast update. Our dynamic labelling schemes support all important axes in XPath such as parent, child, ancestor, descendant, previous - sibling, following - sibling, previous, following. Moreover, our proposed Com-D labelling scheme is more compact than existing ones. Our experimental works show that, Com-D labelling scheme is superior to all ORDPaths, GPR, Dewey, SP one bit and double bit schemes.

In addition to those advantages, using our dynamic labelling scheme as an index structure shall reduce the number of nodes that would otherwise need to be accessed for searching or querying purposes.

In the future, we will conduct more comprehensive experimental works to compare results of this labelling scheme with others. Especially, more order - sensitive queries and updates performances will be presented.

References

1. Alstrup, S., Rauhe, T.: Improved Labeling Scheme for Ancestor Queries. In: Proceedings of the 13th annual ACM-SIAM Symposium on Discrete Algorithm (2002)
2. Amagasa, T., Yoshikawa, M., Uemura, S.: QRS: A Robust Numbering Scheme for XML Documents. In: ICDE (2003)
3. Amato, G., Debole, F., Rabitti, F., Zezula, P.: Yet Another Path Index for XML Searching. In: Koch, T., Sølvsberg, I.T. (eds.) ECDL 2003. LNCS, vol. 2769, pp. 176–187. Springer, Heidelberg (2003)
4. Boag, S., Chamberlin, D., Fernández, M., Florescu, D., Robie, J., Siméon, J.: XQuery 1.0: An XML Query Language. W3C Recommendation (2007), <http://www.w3.org/TR/xquery/>

5. Catania, B., Ooi, B., Wang, W., Wang, X.: Lazy XML Updates: Laziness as a Virtue of Update and Structural Join Efficiency. In: Proc. of the ACM SIGMOD (2005)
6. Chen, Y., Mihaila, G., Bordawekar, R., Padmanabhan, S.: L-Tree: a Dynamic Labelling Structure for Ordered XML Data
7. Cohen, E., Kaplan, H., Milo, T.: Labelling dynamic XML trees. In: Proceedings of PODS 2002 (2002)
8. Cooper, F.B., Sample, N., Franklin, J.M., Hjalton, R.G., Shadmon, M.: A Fast Index for Semistructured Data. In: Proceedings of VLDB Conference (2001)
9. Draper, D., Fankhauser, P., Fernández, M., Malhotra, A., Rose, K., Rys, M., Siméon, J., Wadler, P.: XQuery 1.0 and XPath 2.0 Formal Semantics. W3C Recommendation (2007), <http://www.w3.org/TR/xquery- semantics/>
10. Duong, M., Zhang, Y.: An Integrated Access Control for Securely Querying and Updating XML Data. In: Proceedings of 19th Australasian Database Conference (ADC 2008), Wollongong, Australia, vol. 75 (2008)
11. Duong, M., Zhang, Y.: LSDX: A New Labelling Scheme for Dynamically Updating XML Data. In: Proceedings of 16th Australasian Database Conference, Newcastle, Australia, vol. 39 (2005)
12. Elmasri, Navathe: The fundamental of Database Systems, 4th edn., ch.14, 15
13. El-Sayed, M., Dimitrova, K., Rundensteiner, E.: Efficiently Supporting Order in XML Query Processing, Efficiently supporting order in XML query processing. Data & Knowledge Engineering 54(3), 355–390 (2003)
14. Fisher, D., Lam, F., Wong, R.: Algebraic Transformation and Optimization for XQuery. In: Yu, J.X., Lin, X., Lu, H., Zhang, Y. (eds.) APWeb 2004. LNCS, vol. 3007, pp. 201–210. Springer, Heidelberg (2004)
15. Grust, T.: Accelerating XPath Location Steps. In: Proceedings of the 2002 ACM SIGMOD International Conference on Management of Data, Madison, Wisconsin. ACM, New York (2002)
16. Kaelin, M.: Database Optimization: Increase query performance with indexes and statistics, TechRepublic (2004), http://techrepublic.com.com/5100-6313_11-5146588
17. Kaplan, H., Milo, T., Shabo, R.: A Comparison of Labelling Schemes for Ancestor Queries, <http://www.math.tau.ac.il/~haimk/papers/comparison.ps>
18. Niagara Project, <http://www.cs.wisc.edu/niagara/>
19. Li, C., Ling, W.T., Hu, M.: Efficient processing of updates in dynamic xml data. In: International Conference on Data Engineering, ICDE 2006 (2006)
20. Li, C., Ling, W.T.: An Improved Prefix Labeling Scheme: A Binary String Approach for Dynamic Ordered XML. In: Zhou, L.-z., Ooi, B.-C., Meng, X. (eds.) DASFAA 2005. LNCS, vol. 3453, pp. 125–137. Springer, Heidelberg (2005)
21. Li, Q., Moon, B.: Indexing and Querying XML Data for Regular Path Expressions. In: Proceedings of VLDB (2001)
22. Lu, J., Ling, T., Chan, C., Chen, T.: From Region Encoding To Extended Dewey: On Efficient Processing of XML Twig Pattern Matching. In: Proc. of the VLDB (2005)
23. Lu, J., Ling, W.T.: Labelling and Querying Dynamic XML Trees. In: Yu, J.X., Lin, X., Lu, H., Zhang, Y. (eds.) APWeb 2004. LNCS, vol. 3007, pp. 180–189. Springer, Heidelberg (2004)
24. Meuss, H., Strohmaier, M.C.: Improving Index Structures for Structured Document Retrieval. In: 21st BCS IRSG Colloquium on IR, Glasgow (1999)
25. Milo, T., Suciu, D.: Index Structures for Path Expression. In: Proceedings of 7th International Conference on Database Theory (1999)

26. O'Neil, P., O'Neil, E., Pal, S., Cseri, S., Schaller, G., Westbury, N.: ORDPaths: Insert-Friendly XML Node Labels. In: Proceedings of the 2004 ACM SIGMOD, Paris, France (2004)
27. Schmidt, A., Waas, F., Kersten, M., Carey, J.M., Manolescu, I., Busse, R.: XMark: A Benchmark for XML Data Management. In: Bressan, S., Chaudhri, A.B., Li Lee, M., Yu, J.X., Lacroix, Z. (eds.) CAiSE 2002 and VLDB 2002. LNCS, vol. 2590. Springer, Heidelberg (2003)
28. Silberstein, A., He, H., Yi, K., Yang, J.: BOXes: Efficient maintenance of order-based labeling for dynamic XML data. In: The 21st International Conference on Data Engineering (ICDE) (2005)
29. Tatarinov, I., Viglas, S., Beyer, K., Shanmugasundaram, J., Shekita, E., Zhang, C.: Storing and Querying Ordered XML Using a Relational Database System. In: Proceedings of SIGMOD 2002 (2002)
30. Wang, W., Jiang, H., Lu, H., Yu, X.J.: PBiTree Coding and Efficient Processing of Containment Joins. In: 19th International Conference on Data Engineering, 2003, Bangalore, India (2003)
31. Wu, X., Lee, M., Hsu, W.: A Prime Number Labeling Scheme for Dynamic Ordered XML Trees. In: Proceedings of the 20th International Conference on Data Engineering (ICDE 2004) (2004)
32. Yokoyama, S., Ohta, M., Katayama, K., Ishikawa, H.: An Access Control Method Based on the Prefix Labeling Scheme for XML Repositories. In: Proceedings of 16th Australasian Database Conference, Australia, vol. 39 (2005)
33. Yoshikawa, M., Amagasa, T.: XRel: A Path-Based Approach to Storage and Retrieval of XML Documents using Relational Databases. ACM, New York (2001)
34. Yu, X.J., Luo, D., Meng, X., Lu, H.: Dynamically Updating XML Data: Numbering Scheme Revisited. In: World Wide Web: Internet and Web Information System, vol. 8(1) (2005)

Real-Time Enterprise Ontology Evolution to Aid Effective Clinical Telemedicine with Text Mining and Automatic Semantic Aliasing Support

Jackei H.K. Wong, Wilfred W.K. Lin, and Allan K.Y. Wong

Department of computing, The Hong Kong Polytechnic University,
Hung Hom, Kowloon, Hong Kong S.A.R.
{cshkwong, cswklin, csalwong}@comp.polyu.edu.hk

Abstract. A novel approach is proposed in this paper to aid real-time enterprise ontology evolution in a continuous fashion. Automatic semantic aliasing (ASA) and text mining (TM) are the two collaborating mechanisms (together known as ASA&TM) that support this approach. The text miner finds new knowledge items from open sources (e.g. the web or given repertoires), and the ASA mechanism associates all the canonical knowledge items in the ontology and those found by text mining via their degrees of similarity. Real-time enterprise ontology evolution makes the host system increasingly smarter because it keeps the host system's ontological knowledge abreast of the contemporary advances. The ASA&TM approach was verified in the Nong's mobile clinics based pervasive TCM (Traditional Chinese Medicine) clinical telemedicine environment. All the experimental results unanimously indicate that the proposed approach is definitively effective for the designated purpose.

Keywords: enterprise TCM onto-core, automatic semantic aliasing, text mining, real-time enterprise ontology evolution, clinical telemedicine, D/P system.

1 Introduction

The *enterprise ontology driven information system development* (EOD-ISD) approach has contributed to the successful development of the Nong's TCM (Traditional Chinese Medicine) clinical telemedicine diagnosis/prescription (D/P) system [Lin08]. There are many D/P system variants customized from the same Nong's master proprietary enterprise TCM ontology core (TCM onto-core). Telemedicine refers to administering medicine over the mobile Internet that supports wireless and wireline communications intrinsically [Kaar99, Lacroix99, JWong08]. For example, the Nong's telemedicine environment, which has been deployed successfully in the Hong Kong SAR for clinical practice and treats hundreds of patients daily, is made up of many mobile clinics (MC) that collaborate wirelessly. On board every Nong's MC the essential elements include: a pervasive TCM D/P system, a physician to operate it, a paramedic, a dispenser, and local pharmacy. The physician treats patients locally in a computer-aided manner via the GUI (graphical user interface) of the D/P system and

dispenses the prescriptions electronically. The GUI supports only the “select & key-in” procedure, and in this way all the keyed-in words come from the system and are therefore “standard”. These standard words are canonical terms that were extracted from formal available TCM classics, treatises, and case histories and enshrined in the Nong’s master enterprise ontological vocabulary by TCM experts in a consensus certification process [Rifaieh06]. They enable the experience of every medical/patient case to be potentially used as feedback to enrich the D/P system. Since the TCM onto-core of the system is considered “local” and was customized from the proprietary Nong’s enterprise TCM onto-core, new information (including feedback of case experience) cannot be incorporated directly into the MC D/P system. This prevents ad hoc, instant incorporation that would render the Nong’s MCs incompatible and lead to uncontrollable collaboration failures. At this moment all clinical cases are only treated as physicians’ personal experience. To allow such experience and other new scientific findings data-mined over the web to be incorporated into the local TCM onto-core, we propose in this paper a novel approach to support automatic real-time enterprise ontology evolution. This approach has two components: automatic semantic aliasing, and text mining (ASA&TM). It was verified successfully in the Nong’s MC based clinical TCM telemedicine environment in a controlled fashion.

2 Related Work

The Nong’s MC based clinical TCM telemedicine D/P system was developed by the enterprise ontology driven information system development (EOD-ISD) approach [Uschold07]. Traditional software engineering usually starts with the user/functional specification and follows through the Waterfall model, which may involve the fast prototyping process that allows user input in different development phases. From the functional specification the design specification is derived, verified and implemented. Effective change control may be achieved by using entity-relationships so that variables and functions would not be inadvertently modified, causing errors in system operation. Yet, the semantics of the program modules are not explicit making their reusability and debugging difficult. This is particular so if the software is for distributed processing because traditional debugging tools designed for sequential processing are not applicable.

EOD-ISD is a new paradigm to take advantage of the intrinsic explicit semantics expressed in the enterprise ontology. The success of the Nong’s D/P telemedicine system is an evidence of the EOD-ISD effectiveness. The D/P system development process differentiates three conceptual levels of ontological constructs: i) the global ontology; ii) the local enterprise ontology core (onto-core); and iii) the local system onto-core. In this light, the details of the three conceptual levels for clinical TCM practice are as follows:

- a) Global TCM ontology: This is a school that includes all the relevant TCM classics, medical theories (syllogistic – logically valid but may not have been verified yet), case histories, and treatises. It is an explicit conceptualization of concepts, entities, and their associations [Gruber93], and this conceptualization may be organized into the corresponding subsumption hierarchy of sub-ontology constructs [Guarino95].

- b) Local TCM enterprise ontology (or simply enterprise ontology): This is a subset of the global TCM ontology extracted with a specific clinical purpose in mind. For example, the Nong's TCM onto-core is an extraction for clinical telemedicine deployment and practice. Therefore, the Nong's vocabulary reflects only clinical facts and no syllogistic assumptions.
- c) Local MC D/P system TCM onto-core: It is customized directly from the Nong's proprietary or "master" TCM onto-core for the meta-interface (MI) specification given; thus it is a subset of the master TCM onto-core. The customization of a D/P system is automatic and the first step is to create the MI specification by selecting desired icons from the Nong's master icon library. Therefore, the local TCM onto-cores for all the MC systems customized for a particular client would be the same, but would differ from those of other clients. Neither the Nong's master TCM onto-core nor the local MC D/P system TCM onto-cores evolve automatically. At their present forms they all risk the danger of being stagnated with ancient TCM knowledge derived from old classics, treatises, and case histories. The proposal of the novel ASA&TM (automatic semantic aliasing and text mining) approach in this paper will remove this stagnation danger.

2.1 Nong's D/P Operation Walkthrough

Figure 1 depicts the EOD-ISD details as following: i) the MI specification of selected icons is first created by the client; ii) the EOD-ISD generator extracts the portion of the master/enterprise TCM onto-core of Nong's to form the D/P TCM onto-core to be customized for the local system; iii) the EOD-ISD generator constructs the semantic net (or DOM tree) and inserts the standard ready-made Nong's parser; and iv) the EOD-ISD mechanism constructs the GUI, which is the syntactical layer of the semantic net.

The semantic net is the machine processable form of the local TCM onto-core and the syntactical layer is the query system, which abstracts the semantic net, for human understanding and manipulation. For the Nong's MC system variants the syntactical layer is a graphical user interface (GUI), which has the same appearance as the MI specification. All the symptoms keyed-in via the GUI by the physician (e.g. s_1, s_2, s_3) are captured as actual parameters for the query (e.g. $Q(s_1, s_2, s_3)$) to be implicitly (i.e. user-transparently) constructed by the GUI system as input to the parser. The parsing mechanism draws the logical conclusion from the DOM tree (i.e. the corresponding illness for the $Q(s_1, s_2, s_3)$ query). To be precise, the customized local system's ontological layer defines the ambit of the D/P operation. This layer is the vocabulary and the operation standard of the entire customized system. If a MI-based local D/P system has been correctly customized by the EOD-ISD approach, the three layers should be intrinsically and semantically transitive [Ng08]. With semantic transitivity, given an item from any layer the corresponding ones in the other layers always surface consistently.

Figure 2 is a GUI example which was customized from the given MI specification (the GUI and MI have the same appearance). In this example the master enterprise TCM onto-core is the Nong's proprietary version (used with permission). The GUI

- g) **Section (IX)** – Specific questions (e.g. Do you loathe cold ambience conditions (“ / ”)?), and general physical inspection (e.g. complexion (“ ”) – pale, red or dark).
- h) **Section (X)** – Tongue diagnosis (“ ”) (e.g. texture and coating color).

In Figure 2 the “Symptoms (病)” window echoes the symptoms obtained from the patient by the normal four-step diagnostic procedure: Look (), Listen&Smell (), Question (問), and Pulse-diagnosis (). Table 1 shows an example of how these four steps would be applied by the physician to reach a diagnostic conclusion in the traditional and manual way; in this case Influenza (冒) is concluded. With the D/P interface a physician follows the same four steps, but in a “key-in”, computer-aided fashion. The “key-in” D/P operation is standard and potentially allows the D/P results to be used as immediate feedback to enrich the local TCM onto-core. This is possible because all the TCM terms in the “select & key-in procedure” are standard in the

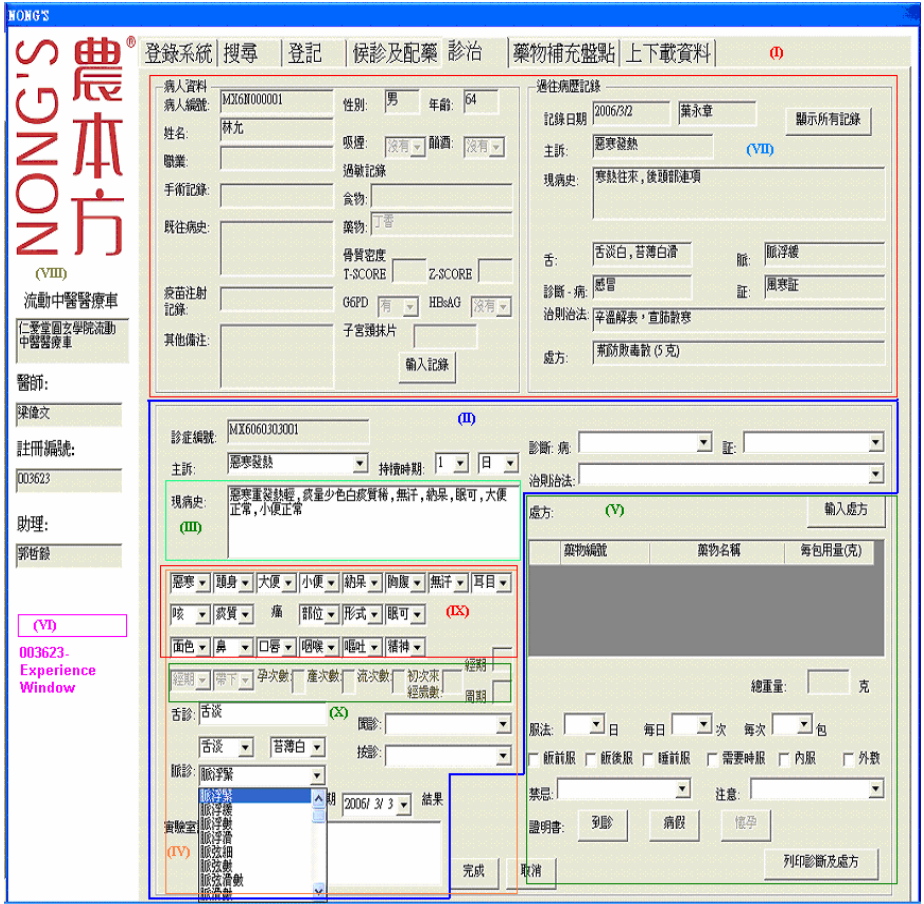


Fig. 2. A D/P GUI generated automatically from the given MI by EOD-ISD

Table 1. A traditional “望， 問， ” diagnosis example (manual conclusion)

Look ()	Listen&Smell ()	Question (問)	Pulse-diagnosis ()	Illness Concluded
pale face	cough, bad breath	headache, fever, loathe cold ambience conditions (/)	taut and fast	Influenza (冒)

enterprise ontology or vocabulary and thus the local system ontology customized by the EOD-ISD process. All the translations of Chinese terms into English are based on the WHO (World Health Organization) TCM standard [WHO07].

The master Nong’s TCM onto-core is annotated in the XML metadata format. Figure 3 is the partial XML display that pertains to the D/P operation in Figure 2.

```
<?xml version="1.0" encoding="Big5" ?>
<咳 >
<風 >
< 證>
  <咳 >
    <咳 >id="1"
    </咳 >
  </咳 >
  < >
    <稀薄 白 >id="1"
    </稀薄 白 >
  </ >
</ 證>
<兼證>
  <風 束 證>id="1"
</風 束 證>
```

Fig. 3. Partial XML annotation pertaining to the D/P operation in Figure 2

Although all the known MC D/P system variants customized from the master Nong’s enterprise TCM onto-core are successfully deployed, they all risk the danger of being stagnated with the ancient TCM knowledge enshrined in the master enterprise TCM onto-core. The desire to remove this danger prompts the proposal of the novel ASA&TM approach, which aids real-time ontology evolution of the host D/P system.

3 The Novel ASA&TM Approach

In the process of automatic semantic aliasing (ASA) the similarity between two entities are computed. For example, $A(x_1, x_2)$ and $B(x_1, x_3)$ mean that the entities

A and B are defined by two separate sets of attributes, (x_1, x_2) and (x_1, x_3) respectively. These two entities are semantically similar because they have the common parameter x_1 . In the ASA context two entities are semantically the same if there are defined by exactly the same set of parameters, and they became aliases to each other if they were redefined later by only some common attributes such as x_1 . In TCM this kind of aliases is rife because the same illness, which was canonically enshrined in ancient formal classics, may be redefined by some different parameters due to geographical and epidemiological needs and differences. For example, if A was canonically enshrined (i.e. a canonical term or context) and now regarded as the reference, B is then its alias. If both A and B are canonical, the context not being taken as the reference is the alias. The ASA mechanism computes the degree of similarity or relevance index (RI) of an alias to the reference context (e.g. the similarity or RI of A to B if the latter is regarded as reference context). Since the local TCM onto-core of a customized Nong's D/P system was customized from the master enterprise onto-core, which is canonical in nature (constructed by consensus certification from formal TCM text, treatises, and case histories), all its entities are canonical. But, this local TCM onto-core risks the danger of stagnation with ancient canonical TCM knowledge because it is not equipped to evolve automatically with time. The proposed ASA mechanism removes this danger by absorbing new TCM information that includes new treatment cases by physicians and scientific reports found on the open web. This can only be achieved with text mining (TM) in an incessant, real-time manner. The tasks by the ASA mechanism include:

- a) Master Aliases Table (MAT) construction: If the local D/P system is equipped with the ASA&TM capability, then the MAT table will be built as part of the system initialization phase. For every canonical reference context/illness (RC) four tables/vectors are built as shown in Figure 4: i) CAT (context aliases table) to record all the aliases (canonical in nature) – a reference context may be an alias to another; ii) CAV (context attributes vector) to record all the defining symptoms; iii) RIT (relevance index table) to record the RI that defines the “non-transitive” degree of similarity of the alias to the RC; and iv) the PPT (possible prescriptions table) to expand the number of usable prescription beyond the canonical ones enshrined for the RC; achieved with the “SAME” principle (to be explained later). Therefore, the MAT serves as the entry point (or directory) to the complete set of four RC vectors/tables in the local D/P system. These vectors are the catalytic structures, which facilitate real-time onto-core evolution in the ASA&TM context. Initially the set size is equal to the number of canonical RC enshrined in the local TCM onto-core of the customized D/P system. In fact, the ASA&TM is directly applicable to the Nong's master enterprise TCM onto-core if the customized system adopts the whole master enterprise onto-core. Thus, in a generic sense the ASA&TM approach aids any enterprise onto-core for real-time evolution.
- b) Text mining invocation: The text miner, namely, the WEKA is invoked by default once the ASA&TM mechanism has finished its MAT initialization. The reason for

choosing the WEKA will be explain later. The WEKA ploughs through the open web as well as the given “repertoire” of treatment cases from different TCM physicians over time to find aliases for the canonical RC in the MAT. For every new alias the corresponding information in the four catalytic structures will be filled. The RI for the new alias with respect to the RC will be computed by the designated algorithm (explained later). Via text mining and automatic semantic aliasing the local D/P system would become smarter and clinically more effective, in light of the “SAME” principle (to be explained later).

The ASA mechanism and the text miner can be switched off with the following implications:

- a) If they were switched off right before the MAT initialization, the local D/P system operates with only the original canonical knowledge extracted from the master enterprise TCM onto-core.
- b) If text miner is switched off after running for some time, then the onto-core evolution respective to the MAT contents stopped and the system operates with the original canonical knowledge plus whatever new information acquired before text mining was stopped.

The ASA mechanism computes the similarity of two terms with the given algorithm. For the two terms Ter_1 and Ter_2 , $Ter_1 = Ter_2$ logically indicates that they are synonyms. But, for the $P(Ter_1 \cup Ter_2) = P(Ter_1) + P(Ter_2) - P(Ter_1 \cap Ter_2)$ expression, where $\cup / \cap$ for union/intersection, $Ter_1 \neq Ter_2$ means that Ter_1 and Ter_2 as aliases (not synonyms). $P(Ter_1 \cap Ter_2)$ is the probability or degree of the similarity; $P(Ter_1)$ and $P(Ter_2)$ are probabilities of the multi-representations (other meanings). For example, the English word “errand” has two meanings (multi-representations): “*a short journey*”, and “*purpose of a journey*”. Figure 5 summarizes the ASA rationale. The illnesses **Illness (A, a)**, **Illness (A, b)** and **Illness (A, c)** are basically aliases to another because they have some common attributes (e.g. x_1 and x_3). If the attributes are weighted, the degree of similarity between any two of them in terms of the RI value can be computed. In the ASA&TM convention, if **Illness (A, a)** is the referential context (RC), **Illness (A, b)** is an aliais, and similarity by the respective RI (e.g. $RI = 0.7$) indicates that **Illness (A, b)** is 70% similar to **Illness (A, a)**. If the attributes are categorized, for example: primary attributes (PA) of weight 0.5, secondary attributes (SA) of weight 0.3, tertiary attributes of weight 0.2, and nice-to-know attributes/ones (NKA/O) of weight 0.0, the RI scores of all the aliases (i.e. **Illness (A, b)**, **Illness (A, c)** and **Illness (X, x)** (as a new aliais found by the text miner over the open web)) for the chosen RC **Illness (A, a)** can be computed; this is shown by Table 2. The total set of attributes for the four illnesses in Figure 5 is called the *corpus* in this research. The second attribute **b** (e.g. in **Illness (A, b)**) is our covnetion to show geographical and/or epidemiological significance; the illness **A** can be defined by somewhat different attribute(s) in different locations (e.g. **b** and **c**).

Referential Context: Weighted Possible Common Cold Prescriptions			
Aliases	Attributes	Relevance indices	Traditional contextual prescriptions (1)
Pneumonia	Fever	0.7	Pneumonia prescriptions (0.7)
Influenza	Cough	0.45	Influenza prescriptions (0.45)
	Headache		
CAT	CAV	RIT	PPT

Domain of referential context - prescription view

Fig. 4. A set of our catalytic data structures for the Common Cold RC

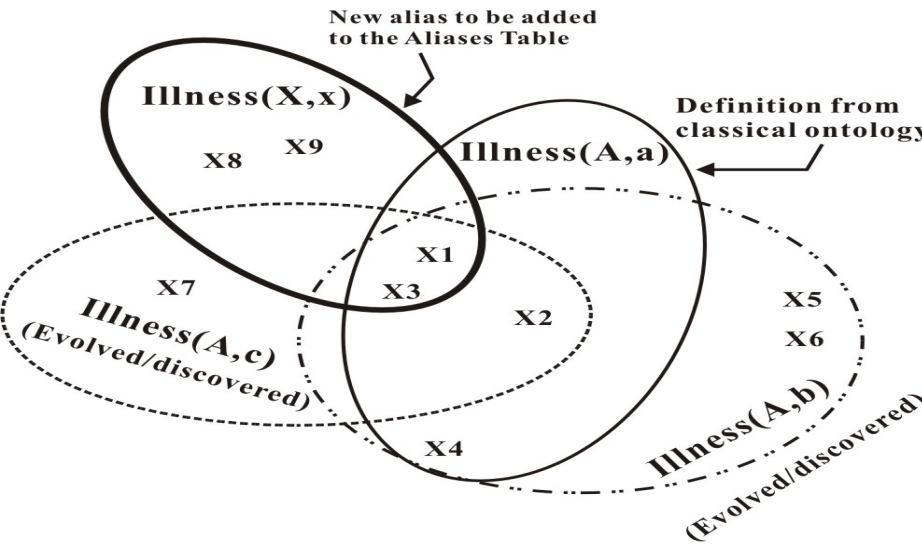


Fig. 5. Illness (X, x) is a new alias for the Illness (A, a) RC

Table 2. Aliases' RI scores for the Illness (A, a) as the RC in Figure 5

Illness/context	Attributes, corpus or alias's set	Attribute classes (for the referential context)	RI scores, { 50% -PA, 30% -SA, 20%-TA, 0%- NKA}	Remarks
Illness (A, a); Common Cold	<i>corpus</i> : $\{x_1, x_2, x_3, x_4, x_8, x_9\}$	PA - $\{x_1, x_2\}$, SA- $\{x_3\}$ TA- $\{x_4\}$, NKA - $\{x_8, x_9\}$	$RI = \frac{0.5}{2}(1+1) + 0.3(1) + 0.2(1) + \frac{0}{2}(1+1) = 1$	Referential context (RC)
Illness (A, b)	<i>alias's set</i> : $\{x_1, x_r, x_r, x_\varepsilon, x_\circ, x_\gamma\}$	PA - $\{x_1, x_r\}$, SA- $\{x_r\}$ TA- $\{x_4\}$, NKA - $\{x_5, x_6\}$	$RI = \frac{0.5}{2}(1+1) + 0.3(1) + 0.2(1) + \frac{0}{2}(1+1) = 1$	Alias of 100% relevance
Illness (A, c)	<i>alias's set</i> : $\{x_1, x_2, x_3, x_7\}$	PA - $\{x_1, x_2\}$, SA- $\{x_3\}$ NKA - $\{x_7\}$	$RI = \frac{0.5}{2}(1+1) + 0.3(1) + 0.2(0) + \frac{0}{2}(1) = 0.8$	Alias of 80% relevance
Illness (X, x)	<i>alias's set</i> : $\{x_1, x_r, x_r, x_v\}$	PA - $\{x_1, x_r\}$, NKA - $\{x_\lambda, x_\lambda\}$	$RI = \frac{\cdot \circ}{\gamma}(\cdot) + \cdot \cdot \gamma(\cdot) + \cdot \cdot \gamma(\cdot) + \frac{\cdot}{\gamma}(\cdot + \cdot) = \cdot \cdot \gamma \circ$	Alias of 25% relevance

Text mining is fundamental to the success of the proposed ASA&TM approach. It addresses the issue of how to find useful information patterns from a preprocessed text effectively [Bloehdorn05, Holzman03]. Preprocessing involves structuring the input text in the predefined way dictated by the technique/tool used. During the structuring process parsing may be applied along with the addition of derived linguistic features and removal of others. The successfully preprocessed input text (e.g. in the Attribute-Relation File Format (ARFF) for the WEKA text mining tool) is saved in the database for subsequent and repeated uses. Text preprocessing may involve the following tasks, in an alone or combined fashion: text categorization, text clustering, concept/entity extraction, granular taxonomy, sentiment analysis, document summarization, and entity relationship modeling. A "high quality" text mining process usually yields useful results in terms of relevance combination, novelty, interest, and discovery. A common text mining usage is to measure and quantify statistically how important a term/word is in a document or a corpus

(a bundle of documents). The interpretation of importance, however, remains with the domain experts. For example, the following two quantification parameters are commonly used in various problem domains including our research with modifications:

- a) *Term frequency* (or simply tf): This weighs how important the i^{th} term t_i is by its occurrence frequency in the j^{th} document (or d_j) within the corpus of K documents. If the frequency of t_i is tf_i and $tf_{l,j}$ is the frequency of any other term t_l in d_j , then the relative importance of t_i in d_j is $tf_{i,j} = \frac{tf_i}{\sum_{l=1}^L tf_{l,j}}$, for $l = 1, 2, \dots, L$ and $\sum_{l=1}^L tf_{l,j}$ includes tf_i .
- b) *Inverse document frequency* (idf): This measures the general importance of the t_i term in the corpus of size K as $idf_{i,k} = \frac{K}{|\{d : t_i \in d\}|}$, where d is the set of documents that contain t_i .

Our in-house experience shows that the WEKA is more effective than other in mining textual patterns, and therefore it is adopted for the proposed ASA&TM prototypes to be verified in the Nong's mobile clinics based diagnosis/prescription (D/P) system environment. Table 3 compares WEKA with some popular text mining tools in the field.

The RI of an alias with respect to the given reference context (RC) is computed based on the idf concept with some modifications; it is redefined as

$$RI_i = \frac{SAC}{|\{MAS \in [V]\}|} . \text{ In practice, } RI_i, \text{ its inverse, } \frac{|\{MAS \in [V]\}|}{SAC} \text{ or the}$$

normalized form $0 < RI_i \leq 1$ can be used. RI_i is the weight or degree of similarity of the i^{th} alias in terms of the ratio defined by “size of the attribute corpus (SAC) of a reference context (e.g. illness) over the mined attribute set (MAS)”. The attributes in MAS are conceptually canonical or standard terms in the background ontology. The RI values help improve the curative chance of a patient because it associates usable prescriptions by the “SAME” principle. This principle, which was enshrine in the core of TCM sine its dawn, is defined as: “If the symptoms are the same or similar, different conditions could be treated in the same way medically, independent of whether they come from the same illness or different ones [WHO07]”; in Chinese terminology it is the “病異，異病” principle. For example, if the three different sets of prescriptions for the **Illness (A, a)**, **Illness (A, b)** and **Illness (A, c)** illnesses are PAA , PAB and PAC respectively, then by the SAME principle the

total set usable prescriptions in the context of automatic semantic aliasing for **Illness** ($\mathbf{A}, \mathbf{a}$) is $PAb_{byRI} = PAa \cup PAb \cup PAc$, where $\cup$ means union of sets. This is based on the presence of common attributes among the three illnesses. The RI values computed for the aliases PAb and PAc indicate their curative efficacies for PAa ; for example, from Table 2 their efficacies are 100% and 80% respectively.

Table 3. Strengths and weaknesses of the some text mining tool examples

Tools	Strengths	Weaknesses
Clementine	● Visual interface ● Algorithm breadth	● Scalability
Darwin	● Efficient client-server ● Intuitive interface options	● No unsupervised algorithm ● Limited visualization
Data Cruncher	● Ease of use	● Single Algorithm
Enterprise Miner	● Depth of algorithms ● Visual interface	● Harder to use ● New product issues
Mine Set	● Data visualization	● Few algorithms ● No model export
CART	● Depth of tree options	● Difficult file I/O ● Limited visualization
Scenario	● Ease of use	● Narrow analysis path
Neuro Shell	● Multiple neural network architectures	● Unorthodox interface ● Only neural networks
S-Plus	● Depth of algorithms ● Visualization ● Programmable or extendable	● Limited inductive methods ● Steep learning curve
Wiz Why	● Ease of use ● Ease of model understanding	● Limited Visualization
WEKA	● Ease of use ● Ease of understanding ● Depth of algorithms ● Visualization ● Programmable or extendable	(not obvious for our purpose)

4 Experimental Results

Many experiments were carried out in Nong's MC based pervasive TCM telemedicine D/P system environment. Different local D/P system prototypes were

WONG'S 農本方

登錄系統 | 搜尋 | 登記 | 候診及配藥 | 診治 | 藥物補充盤點 | 上下載資料

(VIII) 流動中醫醫療車
仁孝堂醫學院流動中醫醫療車
醫師: 陳偉文
註冊編號: 003623
助理: 郭晉鋒

(VI) 003623-Experience Window

病人資料
病人編號: MX6N000001 性別: 男 年齡: 64
姓名: 林尤 吸煙: 沒有 酗酒: 沒有
職業: 過敏記錄: 食物: 藥物: 丁香
手術記錄: 既往病史: 骨質密度 T-Score Z-Score
疫苗注射記錄: G6PD 有 HbA1c 沒有
其他備注: 子宮頸抹片 輸入記錄

過往病歷記錄
記錄日期: 2006/3/2 葉永章 顯示所有記錄
主訴: 惡寒發熱 (VII)
現病史: 寒熱往來, 後頭部連項
舌: 舌淡白, 苔薄白滑 脈: 脈浮緩
診斷-病: 感冒 証: 風寒証
治則治法: 辛溫解表, 宣肺散寒
處方: 荆防敗毒散 (5克)

(II) 診症編號: MX6060303001
主訴: 惡寒發熱 持續時間: 1 日
現病史: 惡寒重發熱輕, 痰量白色白痰質稀, 無汗, 納呆, 腹可, 大便正常, 小便正常
診斷-病: 感冒 証: 風寒証
治則治法: 辛溫解表, 宣肺散寒
處方: 荆防敗毒散 (5克)

(III) 惡寒 身重 大便 小便 納呆 胸腹 無汗 耳目
咳 痰質 痛 部位 形式 腹可 (IX)
面色 鼻 口唇 咽喉 嘔吐 精神
經期 帶下 孕次數 產次數 流產數 初次來經年齡 周數
舌診: 舌淡 脈診: 脈浮緩
按診: 按診
結果: 2006/3/3
實驗室 (IV)
完成 取消 列印診斷及處方

總重量: 克
服法: 日 每日 次 每次 包
斷前服 斷後服 睡前服 需要時服 內服 外敷
禁忌: 孕婦 痛經 懷孕
證明書: 孕婦 痛經 懷孕
列印診斷及處方

Fig. 6. ASA has established a larger set of prescriptions for treating the RC

customized from the Nong's master enterprise TCM onto-core and equipped with the ASA&TM mechanism with the WEKA text miner. The different experimental results unanimously indicate that the novel ASA&TM approach can indeed drive real-time enterprise ontology evolution for clinical telemedicine practice effectively. In this section one of the experimental results that show how the SAME principle is actually realized for clinical practice is shown in Figure 6. The key to this realization is automatic semantic aliasing that computes the RI values. From RI scores of the aliases the total set of prescriptions for treating the RC is concluded (e.g. $P_{A,a}^{total} = PAa \cup PAb \cup PAc \cup PXx$).

The essence of this experimental result is as follows:

- a) Patient's complaint (訴): The patient complained of "loathing cold ambience and had fever –".

- b) *Symptoms* (): The eight symptoms provided by the patient were keyed-in via the D/P interface (e.g. “no perspiration – 汗” symptom).
- c) *Prescriptions* (): Four usable prescriptions were suggested by the D/P system, as a result of ASA; they had different RI scores: 1.0 (directly from the RC), 0.7 (from 1st alias of RC), 0.5 (from 2nd alias of RC), and 0.2 (from 3rd alias of RC). The RI scores indicate the relative efficacy of the alias’s prescription for treating the RC, as shown in the (IV) section of Figure 6. In the original Nong’s MC telemedicine medicine system, which does not has the ASA capability, the set of prescriptions for treating a RC is restricted to the one that was initially established by the consensus certification process in the master enterprise TCM onto-core (e.g. the PA_a set RC **Illness (A, a)**).

The union of the RC’s and aliases’ prescription sets produces a much bigger usable set for curative purposes (e.g. $P_{A,a}^{total} = PAa \cup PAb \cup PAc \cup PXx$). In the ASA&TM context this is a clinical intelligence discovery, for the $PAb \cup PAc \cup PXx$ subset suggested by the D/P system to treat the RC might have never been enshrined in any TCM classics.

5 Conclusion

In this paper the novel ASA&TM approach is proposed for aiding real-time enterprise ontology evolution in a continuous fashion. This approach is supported by two collaborating mechanisms: automatic semantic aliasing (ASA) and text mining (TM); therefore, it is called the ASA&TM approach. The text miner finds new knowledge items from open sources such as the web and given repertoires of medical cases. Then, the ASA mechanism associates all the canonical knowledge items in the ontology and those found by text mining by their degrees of similarity, known as the relevance indices. The capability of real-time enterprise ontology evolution makes the host system increasingly smarter because it keeps ontological knowledge abreast of contemporary advances. The ASA&TM approach was verified in the Nong’s mobile clinics based pervasive TCM (Traditional Chinese Medicine) clinical telemedicine environment. All the experimental results unanimously indicate that the proposed approach is indeed effective for the designated purposes. The next logical step is to let physicians evaluate the ASA&TM approach vigorously in a controlled environment of clinical telemedicine practice.

Acknowledgments. The authors thanks the Hong Kong Polytechnic University and the PuraPharam Group of Hong Kong SAR for the respective research funding, A-PA9H and ZW93.

References

- [JWong08] Wong, J.H.K., Wong, A., Lin, W.W.K., Dillon, T.S.: Dynamic Buffer Tuning: An Ambience-Intelligent Way for Digital Ecosystem. In: Proc. Of the 2nd IEEE International Conference on Digital Ecosystems and Technologies (IEEE-DEST 2008), Phitsanulok, Thailand (February 2008)

- [Lin08] Lin, W.W.K., Wong, J.H.K., Wong, A.K.Y.: Applying Dynamic Buffer Tuning to Help Pervasive Medical Consultation Succeed. In: Proc. of the 1st International Workshop on Pervasive Digital Healthcare (PerCare), IEEE PerCom 2008, Hong Kong, March 2008, pp. 184–191 (2008)
- [Lacroix99] Lacroix, A., Lareng, L., Rossignol, G., Padeken, D., Bracale, M., Ogushi, Y., Wootton, R., Sanders, J., Preost, S., McDonald, I.: G-7 Global Healthcare Applications Sub-project 4. *Telemedicine Journal* (March 1999)
- [Kaar99] Kaar, J.F.: International Legal Issues Confronting Telehealth Care. *Telemedicine Journal* (March 1999)
- [Rifaieh06] Rifaieh, R., Benharkat, A.: From Ontology Phobia to Contextual Ontology Use in Enterprise Information System. In: Taniar, D., Rahayu, J. (eds.) *Web Semantics & Ontology*. Idea Group Inc. (2006)
- [Uschold07] Uschold, M., King, M., Moralee, S., Zorgios, Y.: The Enterprise Ontology, Artificial Intelligence Applications Institute, University of Edinburg, UK, <http://citeseer.ist.psu.edu/cache/papers/cs/11430/ftp:zSzzSzftp.ai.ai.ed.ac.ukzSzpubzSzdocumentszSz1998zSz98-ker-ent-ontology.pdf/uschold95enterprise.pdf>
- [Gruber93] Gruber, T.R.: A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition* 5(2), 199–220 (1993)
- [Guarino95] Guarino, N., Giaretta, P.: Ontologies and Knowledge Bases: Towards a Terminological Clarification. In: *Towards very large knowledge bases: Knowledge building and knowledge sharing*, pp. 25–32. ISO Press, Amsterdam (1995)
- [Ng08] Ng, S.C.S., Wong, A.K.Y.: RCR – A Novel Model for Effective Computer-Aided TCM (Traditional Chinese Medicine) Learning over the Web. In: *The International Conference on Information Technology in Education (CITE 2008)*, Wuhan, China (July 2008)
- [WHO07] WHO International Standard Terminologies on Traditional Medicine in the Western Pacific Region, World Health Organization (2007) ISBN 978 92 9061 248 7
- [Holzman03] Holzman, L.E., Fisher, T.A., Galitsky, L.M., Kontostathis, A., Pottenger, W.M.: A Software Infrastructure for Research in Textual Data Mining. *The International Journal on Artificial Intelligence Tools* 14(4), 829–849 (2004)
- [Bloehdorn05] Bloehdorn, S., Cimiano, P., Hotho, A., Staab, S.: An Ontology-based Framework for Text Mining. *LDV Forum – GLDV Journal for Computational Linguistics and Language Technology* 20(1), 87–112 (2005)

Engineering OODA Systems: Architectures, Applications, and Research Areas

Dimitrios Georgakopoulos

Telcordia, USA

Abstract. The majority of today's software systems and organizational/business structures have been built on the foundation of solving problems via long-term data collection, analysis, and solution design. This traditional approach of solving problems and building corresponding software systems and business processes, falls short in providing the necessary solutions needed to deal with many problems that require agility as the main ingredient of their solution. For example, such agility is needed in responding to an emergency, in military command control, physical security, price-based competition in business, investing in the stock market, video gaming, network monitoring and self-healing, diagnosis in emergency health care, and in a plethora of other areas. The concept of Observe, Orient, Decide, and Act (OODA) loops is a guiding principal that captures the fundamental issues and approach for engineering information systems that deal with many of these problem areas. However, there are currently few software systems that are capable of supporting OODA. In this talk, we describe an OODA architecture and provide a tour of the research issues, approaches, and sample state of the art solutions we have developed for supporting OODA in the domains of video surveillance and emergency response.

Speaker Bio

Dr. Dimitrios Georgakopoulos is a Senior Scientist in Telcordia's Applied Research. He received his PhD and MS degrees in Computer Science from the University of Houston in 1990 and 1986, respectively, and his BS degree from the Aristotle University in Greece. In Telcordia, Dimitrios has led several multi-million, multi-year research projects in areas that include large-scale collaboration, complex event processing in networks of multimedia sensors, and operation support systems for advanced broadband services involving high speed data, VoIP, and video gaming. Before coming to Telcordia, Dimitrios was the Technical Manager and Chief Architect of the "Collaboration Management Infrastructure (CMI)" consortial project at MCC. Before MCC, Dimitrios was a Principal Scientist at GTE (currently Verizon) Laboratories Inc., where he led several multi-year R&D projects in the areas of workflow management and distributed object management systems. Before GTE/Verizon Dimitrios was Member of Technical Staff in Bellcore and worked as an independent consultant in the Houston Medical Center where he designed, developed, and marketed one of the first commercial systems for performing computerized patient diagnosis.

Dimitrios has received a GTE (Verizon) Excellence Award in 1997, two IEEE Computer Society Outstanding Paper Awards in 1994 and 1991, and has been nominated for the Computerworld Smithsonian Award in Science in 1994. He has published more than eighty journal and conference papers. He was the Program Chair of the 2007 International Conference of Web Information Systems Engineering (WISE) in Nancy France, and CollaborateCom 2007 in New York. In 2005, he was the General chair of WISE in New York. In 2002, he served as the General Chair of the 18th International Conference on Data Engineering (ICDE) in San Jose, California. In 2001, he was the Program Chair of the 17th ICDE in Heidelberg, Germany. Before that he was the Program Chair of 1st International Conference on Work Activity Coordination (WACC) in San Francisco, California, 1999, and has served as Program Chair in half a dozen smaller conferences and workshops.

Approximate Structure-Preserving Semantic Matching

Fausto Giunchiglia¹, Fiona McNeill², Mikalai Yatskevich¹, Juan Pane¹,
Paolo Besana², and Pavel Shvaiko³

¹ University of Trento, Povo, Trento, Italy

{fausto,yatskevi,pane}@dit.unitn.it

² University of Edinburgh, Scotland

f.j.mcneill@ed.ac.uk, p.besana@sms.ed.ac.uk

³ TasLab, Informatica Trentina, Italy

Pavel.Shvaiko@infotn.it

Abstract. Typical ontology matching applications, such as ontology integration, focus on the computation of correspondences holding between the nodes of two graph-like structures, e.g., between concepts in two ontologies. However, for applications such as web service integration, we need to establish whether full graph structures correspond to one another globally, preserving certain structural properties of the graphs being considered. The goal of this paper is to provide a new matching operation, called *structure-preserving semantic matching*. This operation takes two graph-like structures and produces a set of correspondences, (i) still preserving a set of structural properties of the graphs being matched, (ii) only in the case if the graphs are *globally* similar to one another. Our approach is based on a formal theory of abstraction and on a tree edit distance measure. We have evaluated our solution in various settings. Empirical results show the efficiency and effectiveness of our approach.

1 Introduction

Ontology matching is a critical operation in many applications, such as Artificial Intelligence, the Semantic Web and e-commerce. It takes two graph-like structures, for instance, lightweight ontologies [9], and produces an alignment, that is, a set of correspondences, between the nodes of those graphs that correspond semantically to one another [6].

Many varied solutions of matching have been proposed so far; see [6,29,24] for recent surveys¹. In this paper we introduce a particular type of matching, namely *Structure-preserving semantic matching (SPSM)*. In contrast to conventional ontology matching, which aims to match single words through considering their position in hierarchical ontologies, structure-preserving semantics matching aims to match complex, structured terms. These terms are not structured according to their semantics, as terms are in an ontology, but are structured to express relationships: in the case of our approach, first-order relationships. This structure-preserving matching is therefore a two-step process, the first step of which is to match individual words within the terms through techniques used for conventional ontology matching, and the second - and

¹ See, <http://www.ontologymatching.org> for a complete information on the topic.

novel - step of which is to match the structure of the terms. For example, consider a first-order relation *buy(car, price)* and another, *purchase(price, vehicle, number)*, both expressing buying relations between vehicles and cost. If the words used in these terms are from known ontologies, then we can use standard ontology matching techniques to determine, for example, that *buy* is equivalent to *purchase* and that *car* is a sub-type of *vehicle*. If they are not from known ontologies we can still use WordNet to gather this information. Our work is concerned with understanding and using this information about how the words are related to determine how the full structured terms are related. Therefore, SPSM needs to preserve a set of structural properties (e.g., vertical ordering of nodes) to establish whether two graphs are globally similar and, if so, how similar they are and in what way. These characteristics of matching are required in web service integration applications, see, e.g., [21,23,18].

More specifically, most of the previous solutions to web service matching employ a single ontology approach, that is, the web services are assumed to be described by the concepts taken from a shared ontology. This allows for the reduction of the matching problem to the problem of reasoning within the shared ontology [21,27]. In contrast, following the work in [1,26,31], we assume that web services are described using terms from different ontologies and that their behavior is described using complex terms; we consider first-order terms. This allows us to provide detailed descriptions of the web services' input and output behavior. The problem becomes therefore that of matching two web service descriptions, which in turn, can be viewed as first-order terms and represented as tree-like structures. An alignment between these structures is considered as successful only if two trees are *globally* similar, e.g., *tree*₁ is 0.7 similar to *tree*₂, according to some measure in [0 1]. A further requirement is that the alignment must preserve certain structural properties of the trees being considered. In particular, the syntactic types and sorts have to be preserved: (i) a function symbol must be matched to a function symbol and (ii) a variable must be matched to a variable. We are mainly interested in approximate matching, since two web service descriptions may only rarely match perfectly.

The contributions of this paper include: (i) a new approach to approximate web service matching, called *Structure-preserving semantic matching (SPSM)*, and (ii) an implementation and evaluation of the approach in various settings (both with automatically generated tests and real-world first-order ontologies) with encouraging results. SPSM takes two tree-like structures and produces an alignment between those nodes of the trees that correspond semantically to one another, preserving the above mentioned two structural properties of the trees being matched, and only in the case that the trees are globally similar. Technically, the solution is based on the fusion of ideas derived from the theory of abstraction [11,12] and tree edit distance algorithms [3]. To the best of our knowledge, this is the first work taking this view.

The rest of the paper is organized as follows. Section 2 explains how calls to web services can be viewed as first-order trees. It also provides a motivating example. We overview the approximate SPSM approach in Section 3, while its details, such as abstraction operations, their correspondence to tree edit operations as well as computation of global similarity between trees are presented in Section 4 and Section 5, respectively. Evaluation is discussed in Section 6. Section 7 relates our work to similar approaches. Finally, Section 8 summarizes the major findings.

2 Matching Web Services

Our hypothesis is that we can consider web services inputs and outputs as trees and therefore apply SPSM to calls to web services. This kind of structural matching can then allow us to introduce flexibility to calls to web services so that we no longer need to rely on (i) terms used in these calls coming from a global ontology; instead local ontologies adapted to purpose can be used; (ii) the structuring of these calls being fixed.

The structure is important because each argument in a call to a web service is defined according to its position in the input or output. However, expecting this structure to be fixed is just as problematic as expecting a global ontology. Individual web service designers will use different structure just as they will use different vocabulary and changes to web service descriptions over time will be mean that previous calls to web services become inappropriate. In order to remove the need both for a global ontology and a fixed structure for every web service call, we therefore need to employ structured matching techniques for matching between web service calls and returns and web service inputs and outputs.

The first-order terms that we match do not distinguish between inputs and outputs in the same manner as, for example, Web Service Description Language (WSDL). Instead, both inputs and outputs are arguments of the same predicate. In Prolog notation, this is indicated by using a + for an input and a - for an output. Thus the term:

$$purchase(-Price, +Vehicle, +Number)$$

indicates that *Vehicle* and *Number* are inputs and *Price* is an output. During run-time, we can distinguish between inputs and outputs because inputs must be instantiated and outputs must be uninstantiated. In order to use our tree matching techniques for web services, we therefore make use of an automated translation process we have created that will map between a first-order term such as the above and a standard WSDL representation of the same information. This approach can also be used for other kinds of services in addition to web services; all that is required is that a translation process is created to convert between the representation of the service and first-order terms.

We make the assumption that web services written in WSDL will contain some kind of semantic descriptions of what the inputs and outputs are: that arguments are labelled descriptively and not merely as 'input1' and so on. This is after all what WSDL, as a description language, is designed to do. We appreciate that in practice designers of web services adopt a lazy approach and label inputs and outputs with terms that do not describe their semantics, especially when the WSDL files are generated automatically from classes or interfaces written in a programming language. In such cases, our techniques will have a very low success rate. However, such web services are of little value for any automated process and do not make use of the full potential of WSDL. We believe that as they become more widely used, the need for them to be properly descriptive becomes imperative so that they can be located and invoked automatically. In the meantime, any mark-up that is used to provide semantics for web services outside of the WSDL can also be amenable to our techniques, provided, as is usually the case, that descriptions of inputs and outputs can be expressed as a tree.

Let us consider an example of approximate SPSM between the following web services: `get_wine(Region, Country, Color, Price, Number_of_bottles)` and `get_wine(Region`

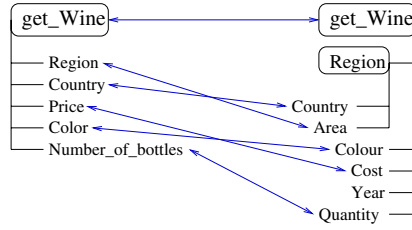


Fig. 1. Two approximately matched web services represented as trees: $T1$: `get_wine(Region, Country, Color, Price, Number_of_bottles)` and $T2$: `get_wine(Region(Country, Area), Colour, Cost, Year, Quantity)`. Functions are in rectangles with rounded corners; they are connected to their arguments by dashed lines. Node correspondences are indicated by arrows.

(Country, Area), Colour, Cost, Year, Quantity), see Figure 1. In this case the first web service description requires the fourth argument of the `get_wine` function (Color) to be matched to the second argument (Colour) of the `get_wine` function in the second description. Also, `Region` in $T2$ is defined as a function with two arguments (Country and Area), while in $T1$, `Region` is an argument of `get_wine`. Thus, `Region` in $T1$ must be passed to $T2$ as the value of the `Area` argument of the `Region` function. Moreover, `Year` in $T2$ has no corresponding term in $T1$. Notice that detecting these correspondences would have not been possible in the case of exact matching by its definition.

In order to guarantee successful web service integration, we are only interested in the correspondences holding among the nodes of the trees underlying the given web services in the case when the web services themselves are similar enough. At the same time the correspondences have to preserve two structural properties of the descriptions being matched: (i) functions have to be matched to functions and (ii) variables to variables. Thus, for example, `Region` in $T1$ is not linked to `Region` in $T2$. Finally, let us suppose that the correspondences on the example of Figure 1 are aggregated into a single similarity measure between the trees under consideration, e.g., 0.62. If this global similarity measure is higher than empirically established threshold (e.g., 0.5), the web services under scrutiny are considered to be similar enough, and the set of correspondences showed in Figure 1 is further used for the actual web service integration.

3 Overview of the Approach

The matching process is organized in two steps: (i) node matching and (ii) tree matching. Node matching solves the semantic heterogeneity problem by considering only labels at nodes and contextual information of the trees. We use here the S-Match system [14]. Technically, two nodes $n_1 \in T1$ and $n_2 \in T2$ match iff: $c@n_1 R c@n_2$ holds, where $c@n_1$ and $c@n_2$ are the concepts at nodes n_1 and n_2 , and $R \in \{=, \sqsubseteq, \sqsupseteq\}$. In semantic matching [10] as implemented in the S-Match system [14] the key idea is that the relations, e.g., equivalence and subsumption, between nodes are determined by (i) expressing the entities of the ontologies as logical formulas and by (ii) reducing the matching problem to a logical validity problem. Specifically, the entities are translated

Table 1. Element level matchers. The first column contains the names of the matchers. The second column lists the order in which they are executed. The third column introduces the matcher's approximation level. The relations produced by a matcher with the first approximation level are always correct. Notice that matchers are executed following the order of increasing approximation. The fourth column reports the matcher's type, while the fifth column describes the matcher's input, see [14] for details.

Matcher name	Execution order	Approximation level	Matcher type	Schema info
WordNet	1	1	Sense-based	WordNet senses
Prefix	2	2	String-based	Labels
Suffix	3	2	String-based	Labels
Edit distance	4	2	String-based	Labels
Ngram	5	2	String-based	Labels

into logical formulas which explicitly express the concept descriptions as encoded in the ontology structure and in external resources, such as WordNet [8]. Besides WordNet, the basic version of S-Match also uses four string-based matchers, see Table 1. This allows for a translation of the matching problem into a logical validity problem, which can then be efficiently resolved using sound and complete state of the art satisfiability solvers [13]. Notice that the result of this stage is the set of one-to-many correspondences holding between the nodes of the trees. For example, initially Region in T_1 is matched to both Region and Area in T_2 .

Tree matching, in turn, exploits the results of the node matching and the structure of the trees to find if these globally match each other. Specifically, given the correspondences produced by the node matching, the abstraction operations (§4) are used in order to select only those correspondences that preserve the desired properties, namely that functions are matched to functions and variables to variables. Thus, for example, the correspondence that binds Region in T_1 and Region in T_2 should be discarded, while the correspondence that binds Region in T_1 and Area in T_2 should be preserved. Then, the preserved correspondences are used as allowed operations of a tree edit distance in order to determine global similarity (§5) between trees under consideration. If this global similarity measure is higher than an empirically established threshold, the trees are considered to be similar enough, and not similar otherwise. Technically, two trees T_1 and T_2 approximately match iff there is at least one node n_{1i} in T_1 and a node n_{2j} in T_2 such that: (i) n_{1i} approximately matches n_{2j} , and (ii) all ancestors of n_{1i} are approximately matched to the ancestors of n_{2j} , where $i=1, \dots, N_1$; $j=1, \dots, N_2$; N_1 and N_2 are the number of nodes in T_1 and T_2 , respectively.

Semantic heterogeneity is therefore reduced to two steps: (i) matching the web services, thereby obtaining an alignment, and (ii) using this alignment for the actual web service integration. This paper focuses only on the matching step.

4 Matching Via Abstraction

In this section we first discuss the abstraction operations (§4.1), then discuss how these operations are used in order to drive a tree edit distance computation (§4.2), and, finally, discuss the implementation details (§4.3).

4.1 Abstraction Operations

The work in [12] categorizes the various kinds of abstraction operations in a wide-ranging survey. It also introduces a new class of abstractions, called TI-abstractions (where TI means “Theorem Increasing”), which have the fundamental property of maintaining completeness, while loosing correctness. In other words, any fact that is true of the original term is also true of the abstract term, but not vice versa. Similarly, if a ground formula is true, so is the abstract formula, but not vice versa. Dually, by taking the inverse of each abstraction operation, we can define a corresponding refinement operation which preserves correctness while loosing completeness. The second fundamental property of the abstraction operations is that they provide *all and only* the possible ways in which two first-order terms can be made to differ by manipulations of their signature, still preserving completeness. In other words, this set of abstraction/refinement operations defines all and only the possible ways in which correctness and completeness are maintained when operating on first-order terms and atomic formulas. This is the fundamental property which allows us to study and consequently quantify the semantic similarity (distance) between two first-order terms. To this extent it is sufficient to determine which abstraction/refinement operations are necessary to convert one term into the other and to assign to each of them a cost that models the semantic distance associated to the operation.

The work in [12] provides the following major categories of abstraction operations:

Predicate: Two or more predicates are merged, typically to the least general generalization in the predicate type hierarchy, e.g., $Bottle(X) + Container(X) \mapsto Container(X)$. We call $Container(X)$ a predicate abstraction of $Bottle(X)$ or $Container(X) \sqsupseteq_{Pd} Bottle(X)$. Conversely, we call $Bottle(X)$ a predicate refinement of $Container(X)$ or $Bottle(X) \sqsubseteq_{Pd} Container(X)$.

Domain: Two or more terms are merged, typically by moving the functions or constants to the least general generalization in the domain type hierarchy, e.g., $Micra + Nissan \mapsto Nissan$. Similarly to the previous item we call $Nissan$ a domain abstraction of $Micra$ or $Nissan \sqsupseteq_D Micra$. Conversely, we call $Micra$ a domain refinement of $Nissan$ or $Micra \sqsubseteq_D Nissan$.

Propositional: One or more arguments are dropped, e.g., $Bottle(A) \mapsto Bottle$. We call $Bottle$ a propositional abstraction of $Bottle(A)$ or $Bottle \sqsupseteq_P Bottle(A)$. Conversely, $Bottle(A)$ is a propositional refinement of $Bottle$ or $Bottle(A) \sqsubseteq_P Bottle$.

Let us consider the following pair of first-order terms ($Bottle\ A$) and ($Container$). In this case there is no abstraction/refinement operation that makes them equivalent. However, consequent applications of propositional and domain abstraction operations make the two terms equivalent:

$$(Bottle\ A) \mapsto \sqsupseteq_P (Bottle) \mapsto \sqsupseteq_D (Container)$$

In fact the relation holding among the terms is a composition of two refinement operations, namely $(Bottle\ A) \sqsubseteq_P (Bottle)$ and $(Bottle) \sqsubseteq_D (Container)$.

The abstraction/refinement operations discussed above allow us to preserve the desired properties: that functions are matched to functions and variables to variables. For

example, predicate and domain abstraction/refinement operations do not convert a function into a variable. Therefore, the one-to-many correspondences returned by the node matching should be further filtered based on the allowed abstraction/refinement operations: $\{=, \sqsupseteq, \sqsubseteq\}$, where $=$ stands for equivalence; $\sqsupseteq$ represents an abstraction relation and connects the precondition and the result of a composition of arbitrary number of predicate, domain and propositional abstraction operations; and $\sqsubseteq$ represents a refinement relation and connects the precondition and the result of a composition of arbitrary number of predicate, domain and propositional refinement operations.

Since abstractions and refinements cover every way in which first-order terms can differ (either in the predicate, in the number of arguments or in the types of arguments), we can consider every relation between terms that are in some way related as a combination of these six basic refinements and abstractions. Therefore, every map between first-order trees can be described using these operations. The only situation in which we cannot use these techniques is if there is no semantic relation between the predicates of the two terms, but in this situation, a failed mapping is the appropriate outcome since we do not consider them to be related even though the arguments may agree. Note that we can match non-related arguments using these operations by applying propositional abstraction and then propositional refinement.

4.2 Tree Edit Distance Via Abstraction Operations

Now that we have defined the operations that describe the differences between trees, we need some way of composing them so that we can match entire trees to one another. We look for a composition of the abstraction/refinement operations allowed for the given relation R (see §3) that are necessary to convert one tree into another. In order to solve this problem we propose to represent abstraction/refinement operations as tree edit distance operations applied to the term trees.

In its traditional formulation, the tree edit distance problem considers three operations: (i) vertex deletion, (ii) vertex insertion, and (iii) vertex replacement [32]. Often these operations are presented as rewriting rules:

$$(i) \ v \rightarrow \lambda \quad (ii) \ \lambda \rightarrow v \quad (iii) \ v \rightarrow \omega$$

where v and ω correspond to the labels of nodes in the trees while λ stands for the special blank symbol.

Our proposal is to restrict the formulation of the tree edit distance problem in order to reflect the semantics of the first-order terms. In particular, we propose to redefine the tree edit distance operations in a way that will allow them to have one-to-one correspondence to the abstraction/refinement operations. Table 2 illustrates the correspondence between abstraction/refinement and tree edit operations. Let us focus for the moment on the first three columns of Table 2. The first column presents the abstraction/refinement operations. The second column lists corresponding tree edit operations. The third column describes the preconditions of the tree edit operation use.

Let us consider, for example, the first line of Table 2. The predicate abstraction operation applied to first-order term t_1 results with term t_2 ($t_1 \sqsupseteq_{Pd} t_2$). This abstraction operation corresponds to a tree edit replacement operation applied to the term t_1 of the

Table 2. The correspondence between abstraction operations, tree edit operations and costs

Abstraction operations	Tree edit operations	Preconditions of operations	$Cost_{T1=T2}$	$Cost_{T1 \sqsubseteq T2}$	$Cost_{T1 \supseteq T2}$
$t_1 \sqsupseteq_{Pd} t_2$	$a \rightarrow b$	$a \sqsupseteq b$; a and b correspond to predicates	1	∞	1
$t_1 \sqsupseteq_D t_2$	$a \rightarrow b$	$a \sqsupseteq b$; a and b correspond to functions or constants	1	∞	1
$t_1 \sqsupseteq_P t_2$	$a \rightarrow \lambda$	a corresponds to predicates, functions or constants	1	∞	1
$t_1 \sqsubseteq_{Pd} t_2$	$a \rightarrow b$	$a \sqsubseteq b$; a and b correspond to predicates	1	1	∞
$t_1 \sqsubseteq_D t_2$	$a \rightarrow b$	$a \sqsubseteq b$; a and b correspond to functions or constants	1	1	∞
$t_1 \sqsubseteq_P t_2$	$a \rightarrow \lambda$	a corresponds to predicates, functions or constants	1	1	∞
$t_1 = t_2$	$a = b$	$a = b$; a and b correspond to predicates, functions or constants	0	0	0

first tree that replaces the node a with the node b of the second tree ($a \rightarrow b$). Moreover, the operation can be applied only in the case that: (i) label a is a generalization of label b and (ii) both nodes with labels a and b in the term trees correspond to predicates in the first-order terms.

4.3 Implementation

We have implemented our approximate SPSM solution in Java. Many existing tree edit distance algorithms allow the tracking of the nodes to which a replace operation is applied. According to [32], the minimal cost correspondences are: (i) one-to-one, (ii) horizontal order preserving between sibling nodes, and (iii) vertical order preserving. The alignment depicted in Figure 1 complies with (i), (iii) and violates (ii). In fact, the fourth sibling Color in $T1$ is matched to the second sibling Colour in $T2$ (see below for an explanation).

For the tree edit distance operations depicted in Table 2, we propose to keep track of nodes to which the tree edit operations derived from the replace operation are applied. In particular, we consider the operations that correspond to predicate and domain abstraction/refinement ($t_1 \sqsupseteq_{Pd}$, $t_1 \sqsubseteq_{Pd}$, $t_1 \sqsupseteq_D$, $t_1 \sqsubseteq_D$). This allows us to obtain an alignment among the nodes of the term trees with the desired properties, i.e., that there are only one-to-one correspondences in it and that functions are matched to functions and variables are matched to variables. This is the case because (i) predicate and domain abstraction/refinement operations do not convert, for example, a function into a variable and (ii) the tree edit distance operations, as from Table 2, have a one-to-one correspondence with abstraction/refinement operations.

At the same time, an alignment used in a tree edit distance computation preserves the horizontal order among the sibling nodes, but this is not a desirable property for the web service integration purposes. In fact, we would want the fourth sibling Colour in $T1$ to match the second sibling Color in $T2$ of Figure 1. However, as from Table 2, the tree edit operations corresponding to predicate and domain abstraction/refinement ($t_1 \sqsupseteq_{Pd}$, $t_1 \sqsubseteq_{Pd}$, $t_1 \sqsupseteq_D$, $t_1 \sqsubseteq_D$) can be applied only to those nodes of the trees

whose labels are either generalizations or specializations of each other, as computed by the S-Match node matching algorithm. Therefore, given the alignment produced by the S-Match node matching algorithm, we identify the cases when the horizontal order between sibling nodes is not preserved and change the ordering of the sibling nodes to make the alignment horizontal order preserving. For example, swapping the nodes *Cost* and *Colour* in T_2 of Figure 1 does not change the meaning of these terms but it allows the correspondence holding between *Colour* and *Color* in Figure 1 to be included in the alignment without increasing the cost during the tree edit distance computation. This switching means that the original horizontal order of siblings is not preserved in most cases. If there are arguments with identical names, such cases are resolved with the help of indexing schemes.

5 Global Similarity between Trees

Our goal now is to compute the similarity between two term trees. Since we compute the composition of the abstraction/refinement operations that are necessary to convert one term tree into the other, we are interested in a minimal cost of this composition. Therefore, we have to determine the minimal set of operations which transforms one tree into another, see Eq. 1:

$$Cost = \min \sum_{i \in S} k_i * Cost_i \quad (1)$$

where, S stands for the set of the allowed tree edit operations; k_i stands for the number of i -th operations necessary to convert one tree into the other and $Cost_i$ defines the cost of the i -th operation. Our goal here is to define the $Cost_i$ in a way that models the semantic distance.

A possible uniform proposal is to assign the same unit cost to all tree edit operations that have their abstraction theoretic counterparts. The last three columns of Table 2 illustrate the costs of the abstraction/refinement (tree edit) operations, depending on the relation (equivalence, abstraction or refinement) being computed between trees. Notice that the costs for estimating abstraction ($\sqsupseteq$) and refinement ($\sqsubseteq$) relations have to be adjusted according to their definitions. In particular, the tree edit operations corresponding to abstraction/refinement operations that are not allowed by definition of the given relation have to be prohibited by assigning to them an infinite cost. Notice also that we do not give any preference to a particular type of abstraction/refinement operations. Of course this strategy can be changed to satisfy certain domain specific requirements.

Let us consider, for example, the first line of Table 2. The cost of the tree edit distance operation that corresponds to the predicate abstraction ($t_1 \sqsupseteq_{Pd} t_2$) is equal to 1 when used for the computation of equivalence ($Cost_{T_1=T_2}$) and abstraction ($Cost_{T_1 \sqsupseteq T_2}$) relations between trees. It is equal to ∞ when used for the computation of refinement ($Cost_{T_1 \sqsubseteq T_2}$) relation.

Eq. 1 can now be used for the computation of the tree edit distance score. However, when comparing two web service descriptions we are interested in similarity rather than in distance. We exploit the following equation to convert the distance produced by a tree edit distance into the similarity score:

$$TreeSim = 1 - \frac{Cost}{\max(T1, T2)} \quad (2)$$

where *Cost* is taken from Eq. 1 and is normalized by the size of the biggest tree. Note that for the special case of *Cost* equal to ∞ , *TreeSim* is estimated as 0. Finally, the highest value of *TreeSim* computed for $Cost_{T1=T2}$, $Cost_{T1 \sqsubseteq T2}$ and $Cost_{T1 \sqsupseteq T2}$ is selected as the one ultimately returned. For example, in the case of example of Figure 1, when we match *T1* with *T2* this would be 0.62 for both $Cost_{T1=T2}$ and $Cost_{T1 \sqsubseteq T2}$.

6 Evaluation

On top of the implementation discussed in §4.3 we exploited a modification of simple tree edit distance algorithm from [34]. The evaluation set-up is discussed in §6.1, while the evaluation results are presented in §6.2.

6.1 Evaluation Set-Up

Ontology and web service engineering practices suggest that often the underlying trees to be matched are derived or inspired from one another. Therefore, it is reasonable to compare a tree with another one derived from the original one. We have evaluated efficiency and quality of the results of our matching solution on two test cases. Note that this is not the largest data set we have access to; a larger set is described, for example, in [15]. However, such data sets are not useful to us in this instance because they do not allow us to evaluate our approximate matching.

Test case 1: real-world ontologies. We used different versions of the Standard Upper Merged Ontology (SUMO)² and the Advance Knowledge Transfer (AKT)³ ontologies. We extracted all the differences between versions 1.50 and 1.51, and between versions 1.51 and 1.52 of the SUMO ontology and between versions 1 and 2.1, and 2.1 and 2.2 of the AKT-portal and AKT-support ontologies⁴. These are all first-order ontologies (hence, their expressivity is far beyond generalization/specialization hierarchies), so many of these differences matched well to the potential differences between terms that we are investigating. However, some of them were more complex, such as differences in inference rules, and had no parallel in our work; therefore, these were discarded, and our tests were run on all remaining differences. Specifically, 132 pairs of trees (first-order logic terms) were used. Half of the pairs were composed of the equivalent terms (e.g., *journal(periodical-publication)* and *magazine (periodical-publication)*) while the other half was composed from similar but not equivalent terms (e.g., *web-reference(publication-reference)* and *thesis-reference (publication-reference)*).

Test case 2: systematic benchmarks. Different application programming interfaces (APIs) suggest that the terms within a tree are likely not to be semantically related

² <http://ontology.teknowledge.com/>

³ <http://www.aktors.org>

⁴ See <http://dream.inf.ed.ac.uk/projects/dor/> for full versions of these ontologies and analysis of their differences.

to each other. Examples from the Java API include: `set(index, element)` and `put(key, value)`. Thus, trees can be considered as being composed of nodes whose labels are random terms.

This test case was composed of trees that are alterations of the original trees. Unlike the work on systematic benchmarks in Ontology Alignment Evaluation Initiative-OAEI [5], the original trees here were generated automatically. We have generated 100 trees. For each original tree, 30 altered ones were created, see Table 3. Pairs composed of the original tree and one varied tree were fed to our SPSM solution. The experiment described above was repeated 5 times in order to remove noise in the results.

For tree generation, node labels were composed of a random number of words, selected from 9000 words extracted from the Brown Corpus⁵. The average number of nodes per tree was 8; in fact, functions usually have fewer parameters. In turn, the tree alterations were inspired by the approach in [5]. These are summarized in Table 3 and include: (i) syntactic alterations, such as adding or removing characters, and (ii) semantic alterations, word addition in labels by using related words (e.g., synonyms) extracted from the Moby thesaurus⁶. The probabilities used for these two types of alterations represent the fact that in most of the cases (0.8) the modifications made during an evolution process concern the altering in meaning, while syntactic modifications, such as introducing acronyms, usually have less occurrences (0.3).

Table 3. Parameters used for generating and modifying the trees

Parameter	Syntactic	Semantic	Combined
Number of trees	100	100	100
Number of modifications per tree	30	30	30
Average number of nodes per tree	8	8	8
Probability of replacing a word in a node label for a related one	0.0	0.8	0.8
Probability of making a syntactic change in a word of a node label	0.3	0.0	0.3

Since the tree alterations made are known, these provide the ground truth, and hence, the reference results are available for free by construction, see also [5,22]. This allows for the computation of the matching quality measures. In particular, the standard matching quality measures, such as *Recall*, *Precision* and *F-measure* for the similarity between trees have been computed [6]. In computation of these quality measures we considered the correspondences holding among first-order terms rather than the nodes of the term trees. Thus, for instance, `journal(periodical-publication1)=magazine(periodical-publication2)` was considered as a single correspondence rather than two correspondences, namely `journal=magazine` and `periodical-publication1=periodical-publication2`.

The evaluation was performed on a standard laptop Core Duo CPU-2Ghz, 2GB RAM, with the Windows Vista operating system, and with no applications running but a single matching system.

⁵ <http://icame.uib.no/brown/bcm.html>

⁶ <http://www.mobysaurus.com/>. Since the SPSM node matching uses WordNet 2.1, an alternative thesaurus was used here.

6.2 Evaluation Results

The matching quality results for the first test case are shown in Figure 2. Quality measures depend on the cut-off threshold values and the SPSM solution demonstrates high matching quality on the wide range of these values. In particular, F-Measure values exceed 70% for the given range.

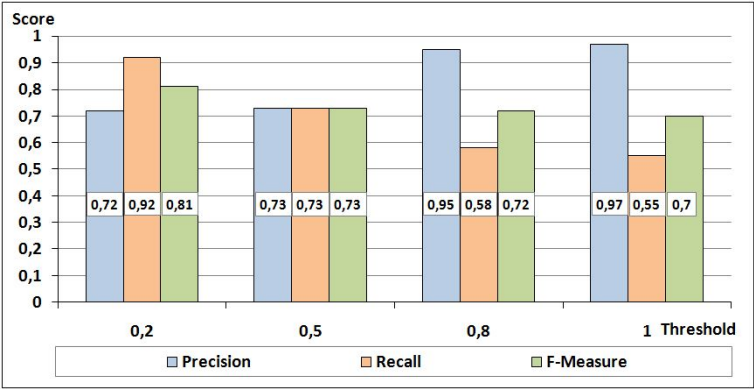


Fig. 2. Test case 1: Evaluation results

The evaluation results for the second test case are summarized in Figures 3, 4 and 5. In order to obtain those results there have been used: (i) the tree matcher discussed in §4 and §5 and (ii) the various matchers used in isolation (namely, edit distance, NGram, prefix, suffix and WordNet) and all these matchers as combined by S-Match, see Table 1. Figures 3, 4 and 5 are composed of four plots (from top to bottom): (i) standard precision-recall plot, (ii) recall vs various cut-off threshold values in [0 1], (iii) precision vs various cut-off threshold values in [0 1], and (iv) F-measure vs various cut-off threshold values in [0 1].

In particular, Figure 3 shows that for the syntactic alterations, as expected, string-based matchers outperform the WordNet matcher. Also, edit distance performs as well as S-Match. The best performance in terms of F-Measure (which is 0.52) is reached at the threshold of 0.8. In turn, Figure 4 shows that for the semantic alterations, as expected, the WordNet matcher outperforms the string-based matchers. The best performance in terms of F-Measure (which is 0.73) is demonstrated by S-Match and is reached at the threshold of 0.8. Finally, Figure 5 shows that when both types of alterations, namely syntactic and semantics, are applied the best performance in terms of F-Measure (which is 0.47) is demonstrated by S-Match and is reached at the threshold of 0.8.

The efficiency of our solution is such that the average execution time per matching task in the two test cases under consideration was 93ms. The quantity of main memory used by SPSM during matching did not rise more than 3Mb higher than the standby level. Finally, the evaluation results show that conventional ontology matching technology that we previously applied to matching classifications and XML schemas

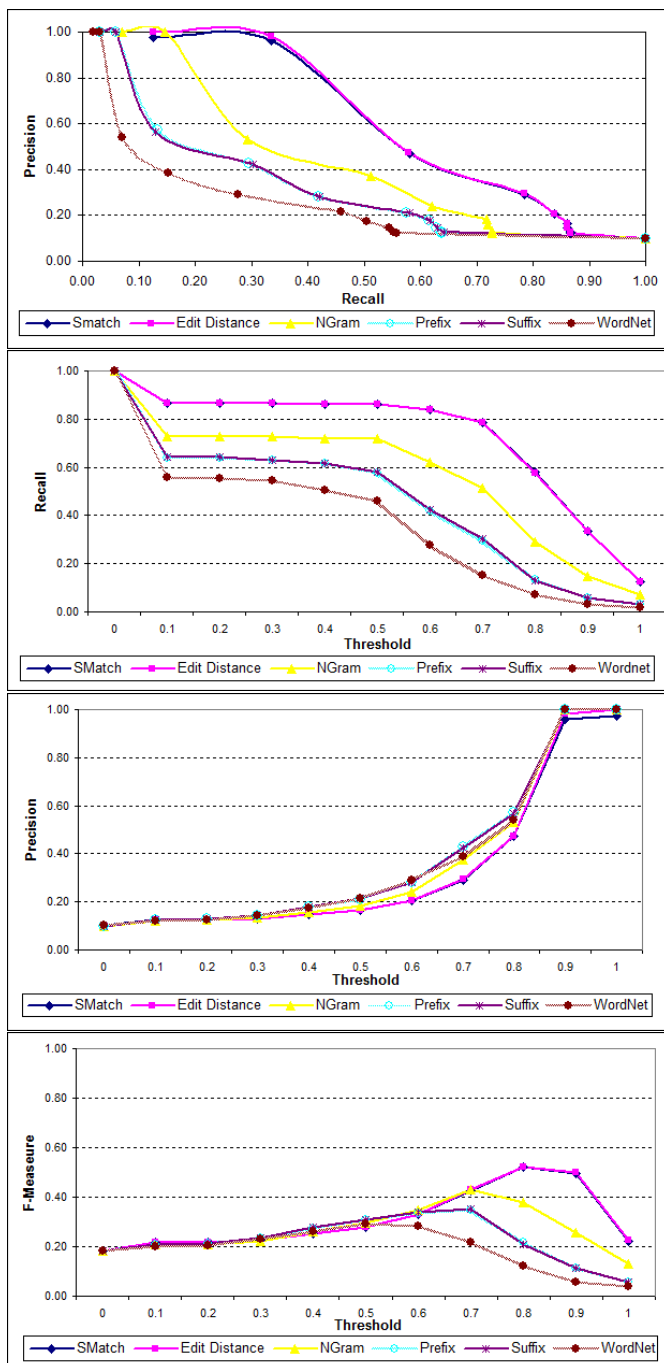


Fig. 3. Test case 2: Evaluation results for syntactic changes

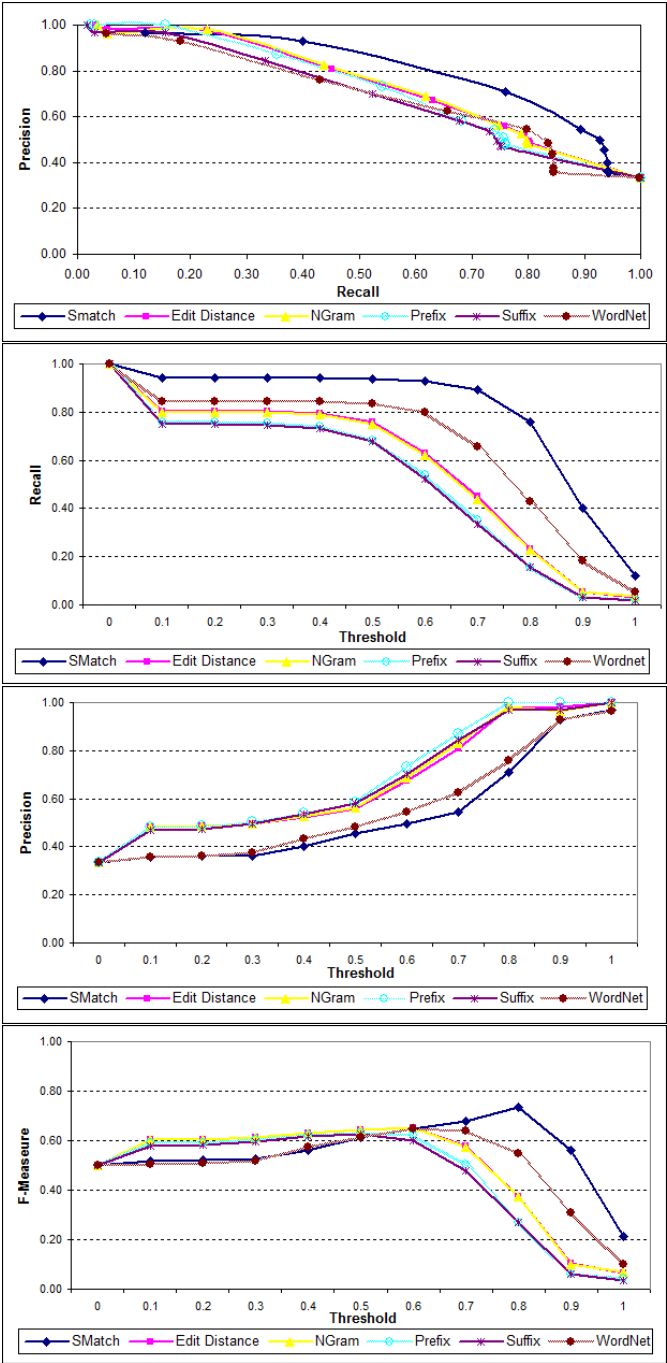


Fig. 4. Test case 2: Evaluation results for semantic changes

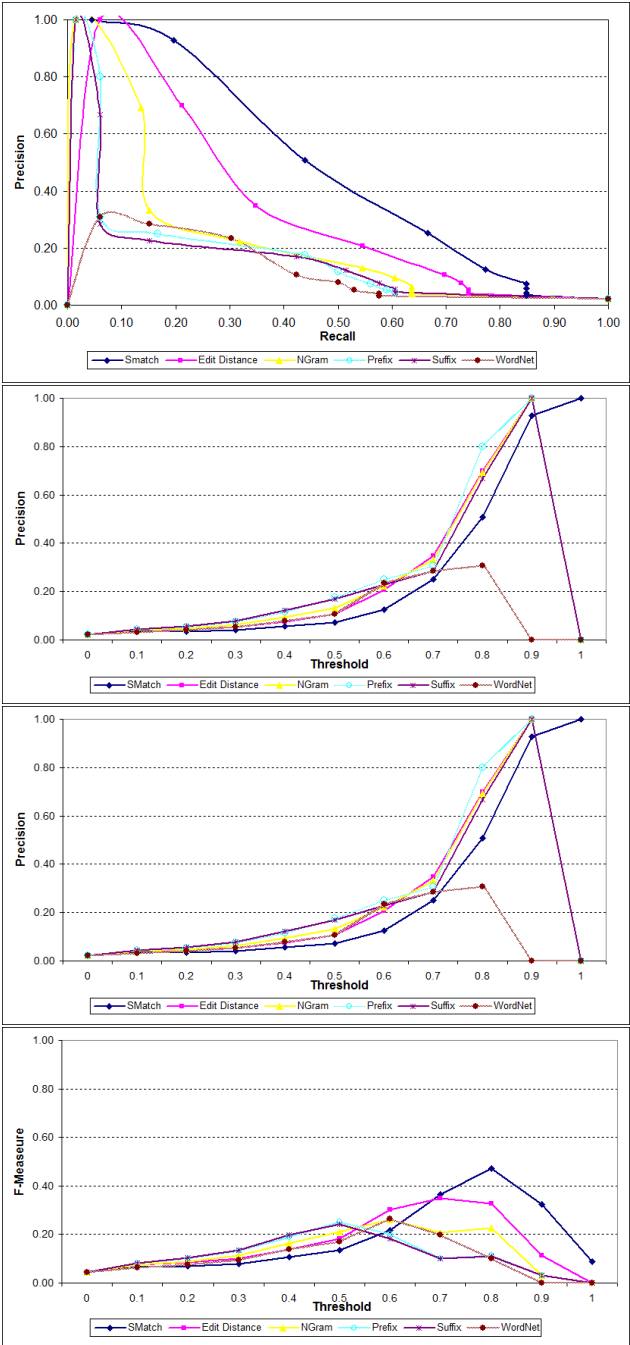


Fig. 5. Test case 2: Evaluation results for combined changes

(see [14]) can also provide encouraging results in the web services domain. Of course, additional extensive testing is needed, especially with WSDL services, for example as done in [31].

7 Related Work

We believe that this approach to structured matching is unique and therefore it is difficult to perform any comparative analysis. In order to demonstrate that we make use of powerful ontology matching tools for the standard ontology matching step of the process, we can compare S-Match against other ontology matching tools. However, the full structure-preserving semantic matching addresses a previously unsolved problem. In this section, we discuss other methods that address similar problems.

Our work builds on standard work in tree-edit distance measures, for example, as espoused by [28]. The key difference with our work is the integration of the semantics that we gain through the application of the abstraction and refinement rules. This allows us to consider questions such as *what is the effect to the overall meaning of the term (tree) if node a is relabelled to node b?*, or *how significant is the removal of a node to the overall semantics of the term?* These questions are crucial in determining an intuitive and meaningful similarity score between two terms, and are very context dependent. Altering the scores given in Table 2 enables us to provide different answers to these questions depending on the context, and we are working on giving providing even more subtle variations of answers reflecting different contexts (see Section 8).

Work based on these ideas, such as Mikhael and Stroudi's work on HTML differencing [16], tends to focus only on the structure and not on the semantics. This work never considers what the individual nodes in their HTML trees mean and only considers context in the sense that, for example, the cost of deleting a node with a large subtree is higher than the cost of deleting a leaf node; the semantic meanings of these nodes is not considered.

The problem of location of web services on the basis of the capabilities that they provide (often referred as the matchmaking problem) has recently received considerable attention. Most of the approaches to the matchmaking problem so far employed a single ontology approach (i.e., the web services are assumed to be described by the concepts taken from the shared ontology). See [21,23,27] for example. Probably the most similar to ours is the approach taken in METEOR-S [1] and in [26], where the services are assumed to be annotated with the concepts taken from various ontologies. Then the matchmaking problem is solved by the application of the matching algorithm. The algorithm combines the results of atomic matchers that roughly correspond to the element level matchers exploited as part of our algorithm. In contrast to this work, we exploit a more sophisticated matching technique that allows us to utilise the structure provided by the first order term.

Many diverse solutions to the ontology matching problem have been proposed so far. See [29] for a comprehensive survey and [7,25,4,17,2,20,30] for individual solutions. However most efforts has been devoted to computation of the correspondences holding among the classes of description logic ontologies. Recently, several approaches allowed computation of correspondences holding among the object properties (or binary

predicates) [33]. The approach taken in [19] facilitates the finding of correspondences holding among parts of description logic ontologies or subgraphs extracted from the ontology graphs. In contrast to these approaches, we allow the computation of correspondences holding among first order terms.

In summary, much work has been done on structure-preserving matching and much has been done on semantic matching, and our work depends heavily on the work of others in these fields. The novelty of our work is in the combination of these two approaches to produce a structure-preserving semantic matching algorithm, thus allowing us to determine fully how structured terms, such as web service calls, are related to one another.

8 Conclusions and Future Work

We have presented an approximate SPSM approach that implements the *SPSM* operation. It is based on a theory of abstraction and a tree edit distance. We have evaluated our solution on test cases composed of hundreds of trees. The evaluation results look promising, especially with reference to the efficiency indicators.

Future work proceeds at least along the following directions: *(i)* studying a best suitable cost model, *(ii)* incorporating preferences in order to drive approximation, thus allowing/prohibiting certain kinds of approximation (e.g., not approximating red wine with white wine, although these are both wines), and *(iii)* conducting extensive and comparative testing in real-world scenarios.

Acknowledgements. We appreciate support from the OpenKnowledge European STREP (FP6-027253).

References

1. Aggarwal, R., Verma, K., Miller, J.A., Milnor, W.: Constraint driven web service composition in METEOR-S. In: Proceedings of IEEE SCC (2004)
2. Bergamaschi, S., Castano, S., Vincini, M.: Semantic integration of semistructured and structured data sources. *SIGMOD Record* 28(1) (1999)
3. Chen, W.: New algorithm for ordered tree-to-tree correction problem. *Journal of Algorithms* 40(2) (2001)
4. Ehrig, M., Staab, S., Sure, Y.: Bootstrapping ontology alignment methods with APFEL. In: Gil, Y., Motta, E., Benjamins, V.R., Musen, M.A. (eds.) *ISWC 2005*. LNCS, vol. 3729, pp. 186–200. Springer, Heidelberg (2005)
5. Euzenat, J., Isaac, A., Meilicke, C., Shvaiko, P., Stuckenschmidt, H., Šváb, O., van Hage, W., Yatskevich, M.: Results of the ontology alignment evaluation initiative 2007. In: Proceedings of the ISWC + ASWC International Workshop on Ontology Matching (OM) (2007)
6. Euzenat, J., Shvaiko, P.: *Ontology matching*. Springer, Heidelberg (2007)
7. Euzenat, J., Valtchev, P.: Similarity-based ontology alignment in OWL-lite. In: Proceedings of ECAI (2004)
8. Fellbaum, C.: *WordNet: an electronic lexical database*. MIT Press, Cambridge (1998)
9. Giunchiglia, F., Marchese, M., Zaihrayeu, I.: Encoding classifications into lightweight ontologies. *Journal on Data Semantics VIII* (2007)
10. Giunchiglia, F., Shvaiko, P.: Semantic matching. *The Knowledge Engineering Review* 18(3) (2003)

11. Giunchiglia, F., Walsh, T.: Abstract theorem proving. In: 11th international joint conference on artificial intelligence (IJCAI 1989), Detroit, Mich., vol. 1, August 20-25 (1989)
12. Giunchiglia, F., Walsh, T.: A theory of abstraction. *Artificial Intelligence* 57(2-3) (1992)
13. Giunchiglia, F., Yatskevich, M., Giunchiglia, E.: Efficient semantic matching. In: Gómez-Pérez, A., Euzenat, J. (eds.) *ESWC 2005*. LNCS, vol. 3532, pp. 272–289. Springer, Heidelberg (2005)
14. Giunchiglia, F., Yatskevich, M., Shvaiko, P.: Semantic matching: Algorithms and implementation. *Journal on Data Semantics IX* (2007)
15. Giunchiglia, F., Yatskevich, M., Avesani, P., Shvaiko, P.: A large scale dataset for the evaluation of ontology matching systems. *The Knowledge Engineering Review Journal* (to appear, 2008)
16. Gligorov, R., Aleksovski, Z., ten Kate, W., van Harmelen, F.: Accurate and efficient html differencing. In: *Proceedings of the 13th IEEE International Workshop on Software Technology and Engineering Practice (STEP)*, pp. 163–172. IEEE Press, Los Alamitos (2005)
17. Gligorov, R., Aleksovski, Z., ten Kate, W., van Harmelen, F.: Using google distance to weight approximate ontology matches. In: *Proceedings of WWW* (2007)
18. Gooneratne, N., Tari, Z.: Matching independent global constraints for composite web services. In: *Proceedings of WWW*, pp. 765–774 (2008)
19. Hu, W., Qu, Y.: Block matching for ontologies. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L.M. (eds.) *ISWC 2006*. LNCS, vol. 4273, pp. 300–313. Springer, Heidelberg (2006)
20. Kalfoglou, Y., Schorlemmer, M.: IF-Map: an ontology mapping method based on information flow theory. *Journal on Data Semantics I* (2003)
21. Klusch, M., Fries, B., Sycara, K.: Automated semantic web service discovery with OWLS-MX. In: *Proceedings of AAMAS* (2006)
22. Lee, Y., Sayyadian, M., Doan, A., Rosenthal, A.: eTuner: tuning schema matching software using synthetic scenarios. *VLDB Journal* 16(1) (2007)
23. Li, L., Horrocks, I.: A software framework for matchmaking based on semantic web technology. In: *Proceedings of WWW* (2003)
24. Noy, N., Doan, A., Halevy, A.: Semantic integration. *AI Magazine* 26(1) (2005)
25. Noy, N., Musen, M.: The PROMPT suite: interactive tools for ontology merging and mapping. *International Journal of Human-Computer Studies* 59(6) (2003)
26. Oundhakar, S., Verma, K., Sivashanugam, K., Sheth, A., Miller, J.: Discovery of web services in a multi-ontology and federated registry environment. *Journal of Web Services Research* 2(3) (2005)
27. Paolucci, M., Kawamura, T., Payne, T., Sycara, K.: Semantic matching of web services capabilities. In: Horrocks, I., Hendler, J. (eds.) *ISWC 2002*. LNCS, vol. 2342. Springer, Heidelberg (2002)
28. Shasha, D., Zhang, K.: Approximate tree pattern matching. In: *Pattern Matching Algorithms*, pp. 341–371. Oxford University Press, Oxford (1997)
29. Shvaiko, P., Euzenat, J.: A survey of schema-based matching approaches. *Journal on Data Semantics IV* (2005)
30. Straccia, U., Troncy, R.: oMAP: Combining classifiers for aligning automatically OWL ontologies. In: Ngu, A.H.H., Kitsuregawa, M., Neuhold, E.J., Chung, J.-Y., Sheng, Q.Z. (eds.) *WISE 2005*. LNCS, vol. 3806, pp. 133–147. Springer, Heidelberg (2005)
31. Stroulia, E., Wang, Y.: Structural and semantic matching for assessing web-service similarity. *International Journal of Cooperative Information Systems* 14(4), 407–438 (2005)
32. Tai, K.-C.: The tree-to-tree correction problem. *Journal of the ACM* 26(3) (1979)
33. Tang, J., Li, J., Liang, B., Huang, X., Li, Y., Wang, K.: Using Bayesian decision for ontology mapping. *Journal of Web Semantics* 4(1) (2006)
34. Valiente, G.: *Algorithms on Trees and Graphs*. Springer, Heidelberg (2002)

Ontology-Based Relevance Assessment: An Evaluation of Different Semantic Similarity Measures

Michael Ricklefs and Eva Blomqvist

Jönköping University, Jönköping, Sweden
`{rimi,blev}@jth.hj.se`

Abstract. Ontology-based relevance assessment of documents is an important task. When viewing this in a business context, approaches are commonly based on the use of one ontology describing the enterprise. A more specific problem is then how to assess the relevance of a set of ontology concepts with respect to a user profile expressed within the same ontology. Semantic similarity measures have been widely used and described in literature, but few experiments have been performed to show the benefits and drawbacks of certain measures. In this paper we describe how a set of measures have been combined, tested, and evaluated. The evaluation was performed through a rank correlation coefficient comparing the measured results with a manually constructed “gold standard”. Conclusions are that certain combinations of measures seem less suitable for solving this type of problem. On the other hand a set of combinations perform quite well, although the detailed performance of most combined measures seem to depend heavily on the structure of the ontology used.

1 Introduction

Relevance assessment of documents is a task that has been addressed in many areas, including Information Retrieval (IR) and the Semantic Web. In IR the focus is often on statistical methods whereas on the Semantic Web more knowledge intensive approaches are suggested. Ontology-based relevance assessment is an important task both on the web and in more business-related areas, such as information systems. In a business context approaches are commonly based on the use of one ontology describing the enterprise. With the use of such an enterprise ontology, relevant information and documents within the company can be described, as well as the information demands of individuals or groups.

If such an ontology is present (or is constructed) and information content and user profiles are described, the remaining problem is concerned with assessing the relevance of information against those profiles. A part of this problem is then how to assess the relevance of a set of ontology concepts (describing a document or a piece of information) with respect to a user profile expressed within the same ontology. Semantic similarity measures have been widely used for assessing the similarity of concepts within an ontology. Still, very few experiments have been

performed to show the benefits and drawbacks of certain measures when used in settings closely related to real-world problems. In this paper we describe how a set of measures have been combined and evaluated for the specific case of relevance assessment of concept sets in ontologies.

In the following section an example project application case is outlined, and in sect. 1.2 the problem addressed in this paper is formulated. Sect. 2 describes the notion of semantic similarity, existing semantic similarity measure and aggregation methods, together with related work. In sect. 3 the selected methods are discussed, then in sect. 4 our evaluation setup is outlined, and results are presented. Finally, we provide conclusions and future work in sect. 5.

1.1 Example Case - The MediaILOG Project

As an application scenario for such relevance assessments we will briefly describe a research project and a specific application within that project. The results of the evaluation presented later in this paper are not specific to this case, but the description will connect the evaluation to a real-world scenario and provide intuition on the kind of ontologies of interest. The project Media Information Logistics (MediaILOG)¹ is a project funded by the Jenz and Carl Olof Hamrin Research Foundation. The project aims at reducing information overflow and introducing demand-oriented information supply through providing new and innovative approaches for information logistics based on enterprise ontologies. Information logistics is a field focusing on the interplay of information demand, information content, and information provision channels [1]. MediaILOG is specifically focused on the media industry, as an especially information intense business, and one project partner is a local newspaper called Jönköpings-Posten. Such a newspaper faces many challenges, including finding and selecting the right information to publish.

An example of an issue is the amount of incoming e-mails to the news sections from the general public. Over one hundred e-mails arrive each day to a designated e-mail address and are then manually sorted, assessed for relevance and possibly forwarded to the appropriate reporter by the news chief on duty. By applying information logistical techniques we envision that parts of this process could be automated, for example by first performing a relevance assessment and sorting a subset of the e-mails directly to the reporters (as described in previous research [2]). If no reporter profile fits to the content of the e-mail, it may be manually processed, but this would still reduce the workload substantially.

With respect to this paper the focus is on the specific problem of assessing the relevance of a set of concepts within the enterprise ontology (representing the information of the incoming e-mail) to another set of concepts of the same ontology (representing a profile of user interests). This does not solve the complete problem of relevance assessment described above, many challenges remain, but the evaluation presented in this paper is an important step towards measures that really conform to the intuition and expectations of human users.

¹ <http://infoeng.hj.se/research/mediailog/mediailog.php>

1.2 Problem Formulation

Our interest lies mostly in comparing sets of concepts in one ontology when treating the ontology as a graph structure (as will be described further in section 2), we can then distinguish three parts of the problem:

1. determining the weight of one edge in the network,
2. determining how to use the edges to determine similarity between two individual concepts,
3. and determining how to aggregate these results to the similarity between concept sets in the network.

In previous research [2] we have studied applied combinations of measures to a specific task, but so far we have not attempted to evaluate the quality of the relevance assessments with respect to human judgement. We believe that it is of importance in order to assess their performance when supporting real-world applications. In this paper we attempt to address the following questions:

- How well does a set of combined edge-based measures conform to the way human users assess relevance?
- Can we distinguish some characteristics of the different combinations of measures that may help when choosing a measure for a specific application?

2 Background and Related Work

In this section we introduce the notion of semantic similarity, present existing similarity measures, and discuss related work evaluating such measures.

2.1 Semantic Similarity

Semantic similarity has been studied from many different perspectives, in computer science, psychology, and cognitive science for example. Some different views exist on what the term denotes and what the properties of semantic similarity are. Similarity is related to the notion of distance. When mathematically expressed common characteristics of a distance function d are usually:

- Minimality - $d(x, y) \geq d(x, x)$
- Symmetry - $d(x, y) = d(y, x)$
- Triangle inequality - $d(x, y) + d(y, z) \geq d(x, z)$

Fulfilling these properties would also make the distance function a distance *metric*. In this paper we are interested in similarity, but many approaches exist to calculate the distance between objects or sets. Intuitively we would state that the smaller the distance the greater the similarity. We would then also like a similarity measure s to conform to the following:

- $s(x, y) \leq s(x, x)$
- $s(x, y) = s(y, x)$

We do not put any requirement on the similarity being a metric, and thereby it does not have to fulfil the triangle inequality. We will thereby generally be able to treat semantic similarity as the inverse of semantic distance (if a distance $d(x, y)$ is normalised to values between 0 and 1, the similarity $s(x, y)$ can then be expressed as $s(x, y) = (1 - d(x, y))$), as long as the distance measure conforms to minimality and symmetry as stated above.

There are views that oppose some of these assumptions, for example Rodríguez and Egenhofer [3] and Andreassen et al. [4] argue that semantic similarity needs to be asymmetric. Rodríguez and Egenhofer give the example that more people tend to agree that “a hospital is similar to a building” than to the statement “a building is similar to a hospital”. Asymmetry might be a correct assumption for some cases, but in our work we are evaluating similarity of concepts in ontologies, whereby a statement like “a building is similar to a hospital” should really be interpreted as something like “the general concept of a building is similar to the concept of hospital”, which does not give the same associations in terms of similarity as the first sentence. In our research we have applied the assumption that semantic similarity is a symmetric property. Based on the above discussion the terms semantic similarity and distance will be used without further explanation.

Additionally Rodríguez and Egenhofer [3] argue that semantic similarity measures most often need to be context dependent, i.e. with respect to what criteria are we considering similarity. Janowicz [5] specifies in more detail what kind of contexts might be present. Six types of context are mentioned:

- the user context,
- the noise context,
- the application context,
- the discourse context,
- the representation context,
- and the interpretation context.

The user context involves the cognitive and psychological processes of the user. Two users will not always have the same opinion about how similar two objects are due to their cultural background, language etc. This type of context in our case affects the reliability of the constructed “gold standard” (see section 4.1). The noise context also affects the “gold standard”, since people use not only the intended information but other “noise” when judging the similarity. An example could be that the size and colours of illustrations may impact the result even though this is not intended as relevant information.

The application context is concerned with how the similarity measure is used. Not all of this context can be controlled, but by restricting our focus to a specific case of concept set similarity assessment for application cases like the one described previously, we sufficiently restrict our context. The discourse context defines in what ontology environment the similarity should be assessed. In our case this is inherent in the problem, since the enterprise ontology is already assumed to be present and the concept sets are predefined. The representation context concerns how the compared concepts are treated and described. This

context is set by using a semantic net-like graph representation of the ontology, and the inclusion of both taxonomic and non-taxonomic relations in this structure. Finally, the interpretation context is concerned with interpreting the similarity result. This problem is outside the scope of this paper, but in the application case it is solved by setting a lower threshold for human assistance and otherwise send the e-mails to the reporter with the highest rank.

When considering semantic similarity specifically for ontologies, Blanchard et al. [6] describe some characteristics of semantic similarity measures. There are three kinds of semantic measures according to this classification, semantic relatedness and semantic distance are the inverse of each other and consider all semantic connections between two concepts. Semantic similarity is by Blanchard et al. described as a specific variant of semantic relatedness, considering only a subset of the semantic links. For our research we focus on the semantic links that can be determined from the ontology specification itself. Most of the existing similarity measures (like in [6]) are based on a semantic network-like representation of the ontology (a graph structure, similar to the one proposed by Andreassen et al. [4]), and consider shortest paths between concepts, depth of the concepts in the taxonomy, density of relations etc.

In general there exist three kinds of measures, with regard to the structures and input considered. As described in [7] one can distinguish between

- edge-based,
- information content-based, and
- feature-based approaches.

As mentioned above we focus on the semantic net-like graph representation of an ontology, which leads us to focus on the edge-based methods. Information content-based methods additionally use corpus texts or other information related to each concept and feature-based methods focus on the property definitions of the concepts.

2.2 Semantic Similarity between Single Concepts

Several measures exist in order to measure edge-based semantic similarity between a pair of concepts. We have divided this problem into two parts, one being the calculation of similarity between two adjacent concepts in the graph (the weighting of each relation) and the other is to extend this to arbitrary pairs of concepts. Existing approaches are described below.

Weighting Methods. The first task is to determine the weight, or the distance, of each relation present in the semantic net-like representation of the ontology. Many applications take a naive approach and assign the weight 1.0 to every relation in the ontology. This means that each step in the graph will have a distance of 1.0, and if paths are used the number of steps in a path can simply be counted and summed up, which yields a simple and efficient solution. This method will be called “simple weighting” throughout the rest of this paper.

A more elaborate measure to calculate the distance of each relation is the method proposed by Sussna [8]. The method takes into account how many relations each of the corresponding concepts have and how “deep” down in the taxonomical hierarchy a concept resides. The intuition is that the deeper in the taxonomical hierarchy, the closer related two adjacent concepts are, as also noted by Andreassen et al. [4] (in their specificity cost property). Additionally the intuition for the number of relations is that the more relations connected to a certain concept, the less the importance of each one of those relations, e.g. the concept “nail” might be “part of” the concept “house”, but it is additionally part of a number of other types of constructions. On the other hand the concept “roof” might only be “part of” the concept “house” and should thereby be deemed more closely related to “house” than “nail” is.

Furthermore in the measure proposed by Sussna different minimal and maximal weight values for each relation type can be defined, so that every relation type can have a different weight range depending on its importance for the similarity of concepts in a specific application case. Sussna’s distance is computed in two steps [8]. First let’s assume that we have two concepts c_1 and c_2 with a relation r , then the first step is to calculate a weight defined by:

$$\omega(c_1 \rightarrow_r c_2) = \max_r - \frac{\max_r - \min_r}{\eta_r(c_1)}$$

Where $\omega(c_1 \rightarrow_r c_2)$ is the initial weight of a relation out of the interval $[\min_r; \max_r]$. $\eta_r(c_1)$ is the number of relations leaving the concept c_1 . This initial weight is then used in the following calculation to compute the complete distance measure proposed by Sussna:

$$\text{dist}_s(c_1, c_2) = \frac{\omega(c_1 \rightarrow_r c_2) + \omega(c_2 \rightarrow_{r'} c_1)}{2 \cdot \max \left[\min_{p \in \text{pths}(c_1, rt)} \text{len}_e(p); \min_{p \in \text{pths}(c_2, rt)} \text{len}_e(p) \right]}$$

Where $\text{len}_e(p)$ is the length in number of edges of the path p and $\text{pths}(c_2, rt)$ are all paths from c_2 to the taxonomical root rt .

As can be seen in the formula, if the concepts at both ends have many relations the fraction in the first formula gets smaller and $\omega(c_1 \rightarrow_r c_2)$ approaches $\max_r$. Thereby the fraction in the second formula increases and the distance, as well. This is a distance measure, which means that as the weight increases the similarity is reduced. If adjacent concepts have many relations the weight of the relation in between them decreases.

Path Calculation. To get the distance between two concepts that are not adjacent there is a need to find a path between the concepts (in case of edge-based methods). Commonly the shortest path between those concepts is used. The shortest path in this sense would be the path where the sum of weights (distances) of the relations (see the last section) on that path is minimum, taking all relations of the graph into account. This will be called the “shortest path method” throughout the rest of this paper.

There are some more elaborate methods available to calculate the distance between pairs of two non-adjacent concepts [6], still without taking any additional information into account (only relying on the ontology structure). These methods commonly also rely on shortest paths but modify or restrict the calculations slightly. Rada et al. proposes a distance measure [9] that only uses taxonomic links and then computes the shortest path between two concepts where all relations have the same weight (of course this could be combined with other weighting methods). If $len_e(p)$ is the length of the path p in number of edges and $pths(c_1, c_2)$ are all possible paths between the concepts c_1 and c_2 the distance is calculated with the following formula:

$$dist_{rmbb}(c_1, c_2) = \min_{p \in pths(c_1, c_2)} len_e(p)$$

The Leacock-Chodorow's similarity [10] is a variant of Rada et al.'s distance but it is instead a measure of similarity. It uses only the shortest path to calculate the similarity. Unlike Rada et al.'s method it uses the length of the path p in number of nodes $len_n(p)$. The measure is normalised with the help of the concept c out of all concepts $cpts$ which has the longest path to the root rt . The resulting similarity is computed through:

$$sim_{lc}(c_1, c_2) = -\log \frac{\min_{p \in pths(c_1, c_2)} len_n(p)}{2 * \max_{c \in cpts} \left(\max_{p \in pths(c, rt)} len_n(p) \right)}$$

The similarity measure proposed by Wu and Palmer [11] uses the shortest path between the two concepts c_1 and c_2 and the depth of their most specific common subsumer $mscs(c_1, c_2)$. The most specific common subsumer is the most specific concept in the taxonomical hierarchy that is a superconcept of both the concepts at hand. The resulting similarity measure is computed as follows:

$$sim_{wp}(c_1, c_2) = \frac{2 * \min_{p \in pths(mscs(c_1, c_2), rt)} len_e(p)}{\min_{p \in pths(c_1, c_2)} len_e(p) + 2 * \min_{p \in pths(mscs(c_1, c_2), rt)} len_e(p)}$$

2.3 Similarity of Sets

Aggregation of semantic similarity between concepts into a similarity measure between sets of concepts can be done in several ways. No ontology-specific measures are needed but instead general aggregation methods for sets may be used, for example to measure distance between geographical point sets.

Hausdorff Distance. As a simple and efficient alternative the measurement method of Hausdorff [12] can be noted. It takes only the most distant objects of each set into account in order to calculate the distance. With X a set of points:

$$d_h : 2^X \times 2^X \rightarrow \mathbb{R}$$

$$d_h(A, B) = \max \left(\max_{a \in A} (\min \{d(a, b) | b \in B\}), \max_{b \in B} (\min \{d(a, b) | a \in A\}) \right)$$

A and B are the sets of objects that should be compared. First the shortest distances from every object of A to the closest object of B will be calculated, $\min \{d(a, b) | b \in B\}$, using some pairwise distance measure. Next, the longest distance is chosen from the previously calculated $\max_{a \in A}(\dots)$. The same will be done for the second set of objects $\max_{b \in B}(\min \{d(a, b) | a \in A\})$. Finally, the larger of the two values (the longest distance) will be selected as the overall distance between the two sets.

Sum of Minimum Distances. The sum of minimum distances method uses one connection from every object in the first set to the other set of objects, for every object in the first set the closest (with respect to the pairwise distance) object in the other set of objects is considered.

$$d(X, Y) = \frac{1}{2} \left(\sum_{x \in X} \left(\min_{y \in Y} d(x, y) \right) + \sum_{y \in Y} \left(\min_{x \in X} d(x, y) \right) \right)$$

For both sets of objects the distance from every object to the closest object in the other set of objects will be calculated and summed up: $\sum_{x \in X} \left(\min_{y \in Y} d(x, y) \right)$. Afterwards both sums are aggregated through taking the average of the results, which is then the resulting distance between the two sets.

Surjection Distance. The surjection distance [12] measures the distance between two sets by using surjections (η) from the larger of the two sets to the smaller one:

$$d_s(S_1, S_2) = \min_{\eta} \sum_{(e_1, e_2) \in \eta} \Delta(e_1, e_2)$$

This means that every object in the larger set will be mapped to some object in the smaller set. The surjection can be minimised using the pairwise distance values computed through comparing individual objects, as described previously.

Fair Surjection Distance. The fair surjection distance [12] uses also surjections like the surjection distance, but they need to be fair. Fair surjections (η') map the larger set evenly onto the smaller set, where evenly refers to the number of mappings of the objects in the smaller set (as a maximum the number of mappings can differ with 1 mapping between the object with most mappings and the one with least mappings).

$$d_{fs}(S_1, S_2) = \min_{\eta'} \sum_{(e_1, e_2) \in \eta'} \Delta(e_1, e_2)$$

Link Distance. The link distance [12] between two sets is the sum of all links to connect every object in both sets to at least one object in the other set. The sum of path weights of the linkings out of all relations R needs to be minimal.

$$d_l(S_1, S_2) = \min_R \sum_{(e_1, e_2) \in R} \Delta(e_1, e_2)$$

Matchings. Additionally Ramon and Bruynooghe [13] have proposed to use matchings for aggregating the distances. A matching occurs when each object in the first set is associated to at most one object in the second set, and the other way around. A maximal matching is a matching when no more associations can be added, and an optimal matching is minimised with respect to distance. Assuming that $m^m(A, B)$ are all matchings between A and B :

$$d^m(A, B) = \min_{r \in m^m(A, B)} d(r, A, B)$$

Average Linkage Based Similarity. A measure considering even more links between the sets is the one proposed by Kalousis et al. [14]. In this measure all possible pairs of objects v_i and v_j from the two sets S_I and S_J are considered and the average similarity is computed by dividing by the number of objects in both sets n_i and n_j .

$$\text{sim}_{AL}(S_I, S_J) = \frac{1}{n_i n_j} \sum_{ij} (\text{sim}(v_i, v_j))$$

Single Linkage-Based Similarity. Another version of the linkage based similarity [14] is to only consider the maximum similarity of any pair comprised of objects v_i and v_j from the two sets S_I and S_J .

$$\text{sim}_{SL}(S_I, S_J) = \max_{ij} (\text{sim}(v_i, v_j))$$

2.4 Experiments on Semantic Similarity

Several projects exist where similarities within ontologies have been measured and evaluated. Many of these projects are limited to using the Gene Ontology², which describes gene products. Different approaches use different ways to measure the similarity of concepts, in addition to semantic similarity one approach uses fuzzy set theory and fuzzy measure theory [15]. Another paper is comparing semantic similarity measures on the GeneOntology and Pfam³, a database of protein families. It also introduces a new similarity measure, GraSM (Graph-based Similarity Measure), which is optimised for the GeneOntology [16]. Still others are testing different measures for searching within the GeneOntology [17].

² <http://www.geneontology.org/>

³ <http://pfam.sanger.ac.uk/>

A study from the Technical University of Crete (TUC) [18] compares several semantic similarity measures within WordNet⁴ and the MeSH Ontology⁵. This study not only measures the similarity between concepts in one ontology, it also measures the similarity of concepts in two different ontologies. To do this they implemented their own hybrid measurement method, called X-Similarity and compare it to existing ones. For evaluating the semantic similarity methods they asked medical experts to enter their judgements, and 12 experts worldwide did that via the Internet. They implemented all methods and integrated them into a similarity assessment system which is available on the Internet⁶.

Additionally there exist related research that tries to evaluate similarity measures that do not conform to our assumptions of similarity, as stated in section 2.1. For example Andreassen et al. [4] evaluate their similarity measure, but this is an asymmetric measure. To the best of our knowledge we are not aware of any study that evaluates the use of point set measurements (as described in section 2.3) to calculate the similarity between concept sets within ontologies. And most of the existing projects evaluating the similarity of two concepts are restricted to one specific ontology (e.g. the GeneOntology).

3 Semantic Similarity within Ontologies - Selected Methods

Considering the application case we have in mind and the available methods as described in past sections, some combinations solving the different parts of the problem were selected for evaluation. The selection was made both on the basis of our specific needs in the project at hand and on the information found in related literature. The methods selected are the following:

- Edge-weighting methods
 - Simple weighting
 - Sussna's weighting
- Path calculation methods
 - Shortest path
- Concept set aggregation methods
 - Hausdorff's method
 - Sum of minimum distances
 - Surjection
 - Fair surjection
 - Linking

The resulting 10 combinations (each using one edge-weighting method, one path selection method and one aggregation method) were selected for use in the evaluation. We decided to use *Sussna's weighting-method* because it does not

⁴ <http://wordnet.princeton.edu>

⁵ <http://www.nlm.nih.gov/mesh>

⁶ <http://www.intelligence.tuc.gr/similarity>

need any additional information, aside from the actual ontology, and it relies on the position in the taxonomy for weighting relations as well as incorporates the possibility to differentiate the weighting of different types of relations. In this evaluation though, we used the same range of weights for all relation types. The range was set to be between 1 and 2, as recommended by Sussna [8]. The *Simple weighting* was introduced as the naive approach to weighting, and in order to have a weighting method that is fast to calculate for all relations (e.g. in case of large ontologies). It is additionally interesting to compare this weighting method to Sussna's weighting method. In our implementation of the simple weighting, every relation got the weight 1.0.

For selecting paths between two concepts in the network the simple shortest path method is used. It might be considered as a version of the method proposed by Rada et al., but extending the method to all kinds of relations. Calculating shortest paths between two concepts within the ontology is done using a modified version of the Dijkstra algorithm. It would also be possible to implement and test the original method proposed by Rada et al., with the restriction to taxonomic links, as an alternative but this is left to future work.

For the aggregation methods five different methods are selected. The method proposed by Hausdorff is very simple and efficient, and is primarily selected based on this fact. The other four methods apply different ways of selecting the set of connections to include in the aggregated value and are selected to give a broad coverage of aggregation possibilities. We used the "Relational WEKA" system⁷, which extends the "WEKA" toolkit⁸, to calculate the surjections, the fair surjections and the minimal linking. There exists also the possibility to calculate Hausdorff's method and the sum of minimum distances through that software, but these were already implemented in our own software tools. Also the edge-weighting and path selection methods tested were implemented during this work.

4 Experiments

In this section we present our experiment setup, for evaluating the combinations of measures presented in the last section. Also the results of the evaluation are presented and discussed.

4.1 Experiment Setup

For the evaluation of the combined semantic similarity measures the resulting 10 combinations of measures were tested with 2 ontologies and 4 profiles (2 for each ontology). Within each ontology 10 concept sets were selected. Those concept sets represent documents in which the user might be interested. The reason for this small data set was the desire to use human subjects to construct a reliable "gold standard" (see below), if another evaluation method would be applied a larger data set could also be used.

⁷ http://cui.unige.ch/~woznica/rel_weka/

⁸ <http://www.cs.waikato.ac.nz/ml/weka/>

One of the ontologies used was the partial enterprise ontology constructed for the sports section of the local newspaper within the MediaILOG project. This ontology contains general concepts concerned with different kinds of sports, sports clubs, athletes, and sport events. More specific concepts concerned with local instantiations of the previously mentioned concepts are also present. Additionally people involved in producing sport news are included, like sport reporters. The ontology is to a large extent built as a taxonomy of sports concepts. The ontology contains 449 named concepts and 150 properties, that were all used in the semantic net-like graph representation for calculating similarity.

The second ontology is an adaptation of a tourism ontology downloaded from the web (quite heavily adapted, the downloaded ontology was only used as a starting point and as inspiration). The ontology treats concepts concerned with accommodation, events and activities as well as facilities for those activities. The ontology also contains instances concerned with tourism, like specific hotels. This ontology has a much smaller taxonomy than the sports ontology described above, instead it contains a larger number of instances. The ontology contains 58 named concepts, 12 properties and 200 instances.

Within both these ontologies two interest profiles were selected manually, with the intention to represent the interest of some supposed person using the ontology (in the sports case this would be a sports reporter, and in the tourism case it might be a person working at a travel agency). The first profile in the sports ontology included the concepts of ball sports, ice hockey and some concepts concerned with facilities for those sports, in total 5 concepts. The second profile instead contained the concepts representing orienteering and “combination” sports (like triathlon and multisport), in total 2 concepts. In the tourism ontology the first profile contained all concepts concerned with sports events and sports facilities (13 concepts and instances), while the second profile contained concepts connected to hostels as well as the English language (as spoken by a certain accommodation provider), in total 5 concepts and instances.

The construction of the 10 concept sets (representing documents) for each ontology was conducted slightly differently for the two ontologies. For the sports ontology documents were already present in the form of e-mails discussing sports events that had been used previously for testing approaches for e-mail ranking in the project. 10 of these e-mails were randomly selected to be used for the experiment, and then manually mapped to a set of concepts in the ontology, resulting in 10 sets of concepts ranging from 1 to 7 concepts. For the tourism ontology texts were selected from the Internet, e.g. from various travel websites. These were then manually mapped to concepts in the ontology, resulting in 10 concept sets ranging from 1 to 11 concepts in each set.

Construction of a “Gold Standard”. To construct a reference to which to compare the results generated by the combined algorithms, we tried to capture the intuition of human users and construct a “gold standard” ranking of the concept sets with respect to the profiles. For each of the four experiment setups (one ontology with one profile and the 10 concept sets representing the documents) an experiment session with human assessors was conducted. For each

session three persons assessed the relevance of the concept sets, and in the end a score was computed based on the number of “votes” from the participants.

To make the task easier for the experiment subjects, and to increase the reliability of the results, the concept sets were shown to the human subjects in pairs. This method was selected early in the experiment design, the consequence being that only a small number of ontologies and concept sets could be used for the experiment (due to the exponential growth of human assessments when increasing the number of sets). The subjects were then in each case asked to pick the one set (from the two) that was considered more relevant than the other to that specific profile. This was repeated for all possible combinations of the 10 concept sets, in each session. When aggregated the selections from all three subjects and for all 45 combinations yield a ranking of the concept sets for that particular profile. This procedure was repeated four times, once for each profile in each of the ontologies, and four “gold standard” rankings were received, representing an approximation of a consensus human judgement of the relevance of the concept sets with respect to the profiles.

As discussed in section 2.1 several kinds of contexts may influence the reliability of such a “gold standard”. The cultural background and language of the subjects and noise, such as the presentation format of questions and ontologies, may influence the results. In this experiment the subjects used were 6 researchers in our university (4 PhD students, one research assistant, and one assistant professor). Three of the subjects do research in areas closely related to ontologies, while the other three do research in areas where they are familiar with conceptual modelling but not ontologies in particular. The subjects were distributed over the four sessions so that each time a different combination of persons were used. Additionally the subjects were given thorough instructions on how to answer the questions, and for example to try and disregard the presentation format and limitations in visualisation of the ontologies. The subjects were given unlimited time to answer each question, also in order to reduce the effect of noise taking overhand and introducing an incorrect first impression.

Result Analysis Setup. After running the combinations of semantic similarity algorithms on the same data as given to the human assessors, resulting in a set of concept set rankings, the results needed to be compared to the previously constructed “gold standard”. For this a statistical non-parametric rank correlations measure was selected. Available measures are for example Spearman’s rho and Kendall’s tau (see for example the book by Siegel [19]). Kendall’s tau was selected due to the fact that the results are easier to interpret than Spearman’s rho and that this measure additionally works well with a small sample size (Siegel [19] suggests that 9 is a sufficient size, thereby our sample size of 10 should also be sufficient). Kendall’s tau (in the presence of ties) can be expressed as:

$$\tau = \frac{S}{\sqrt{\frac{1}{2}N(N-1) - T_x} \sqrt{\frac{1}{2}N(N-1) - T_y}}.$$

Where S is a sum of scores for each possible pair of ranked items, where a score of -1 denotes that they are in the “wrong” order and +1 that they are in the

“correct” order (with respect to the standard used), N is the number of ranked items and T_x and T_y are the number of ties in each compared ranking order. The resulting rank correlation value will be a score between -1 and +1, where -1 denotes an inverse correlation, +1 a perfect correlation and 0 that the two rankings are completely unrelated.

4.2 Experimental Results

The results from the above experiments can be seen in Table 1. The four settings are shown in the left column, and for each one two different weighting methods have been applied, the simple weighting and the weighting based on Sussna’s method, both described earlier. Each of those are then combined with an aggregation method (Hausdorff’s method, sum of minimum distances, surjections, fair surjections, and link distances) and the value of Kendall’s rank correlation coefficient is presented in the table. We remind the reader that the coefficient can take values between -1 and +1, and that 0 denotes no correlation and +1 denotes perfect correlation. Based on this we can note that most combinations show a quite high correlation value, except for some experiments using Hausdorff’s method. Additionally the statistical significance of the measurements are for all values on at least the 0.03 significance level, except for the Hausdorff method where the significance drops to the 0.11 level.

Table 1. Values of the Kendall ranking correlation coefficient for the different evaluation settings

	Hausdorff	SMD	Surj	FairSurj	LinkDist
Tourism - Profile 1					
Simple	0,854	0,899	0,750	0,719	0,899
Sussna	0,405	0,809	0,719	0,719	0,899
Tourism - Profile 2					
Simple	0,519	0,675	0,612	0,612	0,629
Sussna	0,500	0,675	0,582	0,582	0,675
Sport - Profile 1					
Simple	0,481	0,694	0,763	0,667	0,667
Sussna	0,424	0,566	0,754	0,613	0,613
Sport - Profile 2					
Simple	0,814	0,698	0,797	0,845	0,651
Sussna	0,659	0,584	0,764	0,809	0,674

One limitation of these experiments is of course the construction of the “gold standard”. As described previously there are several kinds of contexts that affect this construction and that introduces a certain level of uncertainty with respect to the correctness of the rankings produced by the human subjects. Additionally it is accepted that humans are not always able to completely agree when considering ontologies. Previous experiments, originally conducted by Miller and

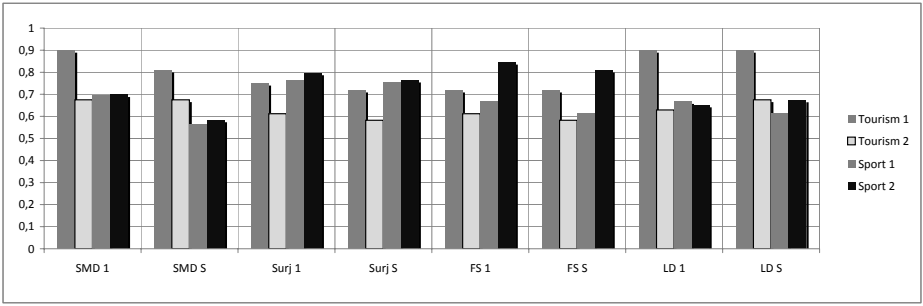


Fig. 1. The correlation coefficients for the different combinations

Charles [20] and replicated by Resnik [21], have shown that a human agreement above 0.9⁹ cannot be reached. Thereby a complete correlation (the value 1.0) can probably never be reached, due to the fact that the “gold standard” will never represent a completely “correct” ranking.

Despite this fact, when analysing the table it can be noted that Hausdorff’s method, although simple and computationally inexpensive, seems to give very varying results. Although the statistical significance of the rank correlation coefficient is lower than for the other cases we may conclude that this method is too unreliable to be usable in cases like ours. This is most likely due to the fact that it only incorporates one distance value from the set of possible connections between concepts in the sets and thereby it is a too crude measure to reliably create an aggregated distance between the complete sets. An intuitive explanation is that since it incorporates only the distance of the most distant concepts, then one “odd” concept may increase the distance drastically, although the rest of the set might be very similar to the other. Additionally, using it together with Sussna’s weighting seems to make the results even more unreliable.

Disregarding Hausdorff’s method and focusing on the rest of the methods, none can be distinguished as the “best” method for all cases. Instead we can look at the illustration of the results in Figure 1 to graphically see that the results differ a lot based on the ontology at hand. In the figure the aggregation method names are abbreviated and 1 stands for the simple weighting whereas S stands for the weighting suggested by Sussna. When analysing the performance in terms of the rank correlation coefficient, it can be noted that there is generally a connection between the ontology used in the experiment and the performance of the measure. For example using the sum of minimal distances aggregation method gives better results than using surjections in both experiments using the tourism ontology, while the situation is the opposite when considering the experiments using the sports ontology. This leads us to the conclusion that which method is really the “best” will have to be determined for each case using a different kind of ontology. Possibly future work could develop guidelines as to

⁹ Note that the method for correlation estimation differs between those experiment and our research.

when and for what ontologies a certain combination is most suitable. Note though that all aggregation methods (except Hausdorff's method) perform reasonably well and any one of them could probably be used in practise, but an application could of course benefit from being tuned to its best performance.

Concerning the weighting methods evaluated, the simple weighting usually performs better or at least as well as the method proposed by Sussna. The only case where we have the opposite situation is for three out of four cases when combined with the link distance aggregation method. From such a small set of experiments we cannot really draw any general conclusions based on this result, but we note that this is an interesting characteristic that should be subject to further investigation. Additional investigations could also be made into differentiating the weights for relation types in the ontologies, since this possibility is already available in Sussna's measure.

If we take into account the computational complexity of computing the similarity with the different method combinations, then sum of minimum distances combined with the simple weighting is the most inexpensive combination. As can be noted this combination also performs reasonably well with respect to the ranking correlation coefficient (it stays above 0.67 for all experiment settings), whereby it could be recommended as the best choice for applications where time consumption for calculation is crucial. If the application is not time critical and the ontology is not envisioned to change drastically during the application lifetime it might be a good approach to test the ontology with a set of methods and do a similar evaluation as the one reported in this paper, since the results above seem to indicate that the combined methods might perform differently depending on the structure of the ontology.

5 Conclusions and Future Work

When analysing the evaluation results above the following conclusions can be drawn. The aggregation method proposed by Hausdorff is the only method that does not produce reasonably good results. When combined with the simple weighting it performs considerably worse than the other methods in half of the cases and when combined with Sussna's weighting, in three out of four cases. Analytically we can explain this by the fact that the aggregation method takes too few concepts in the set into account when aggregating the distance.

The main conclusion concerning the rest of the tested combinations is that they all perform reasonably well, but that the added complexity of using Sussna's weighting method seem to give nothing in return. It might even be the case that it performs worse than simply giving all relations the weight 1.0, but to state this conclusion in the general case we would need more testing of different variants of Sussna's method where all the variables are studied (such as the weight ranges for different relations). Simple weighting seems to at least perform as well and in addition is much more computationally inexpensive.

An additional conclusion is that the performance in terms of rank correlation of the combined methods seems to be connected to the ontology used. We can

see that many of the rank correlation coefficient values are similar for a measure as long as the ontology is the same (even though the concept sets change), but switching to another ontology gives different results. As already stated, all combinations except the ones including Hausdorff's method could be reasonable to use in a real world application, but if necessary this experiment could be repeated with the actual ontology in order to verify the choice in a specific case.

In our plans for future work we would like to perform more experiments with different kinds of ontologies to distinguish more clearly why there are differences in performance that are not easily explained analytically. The evaluations will also be extended to include the complete method of e-mail sorting suggested in section 1.1 together with end-users, but this has not been done yet, it will be part of another research project in the near future. Additional future work is to use larger and more complex ontologies for evaluating the measures, although this would probably yield the construction of a "gold standard" in the manner presented here unfeasible. Next steps are to offer the implemented methods as downloadable software. An API will also be provided at our Jönköping Ontology Toolkit (JOT) site¹⁰ in order to let researchers develop their own similarity measures. Finally, the set of methods available will be increased to cover as many available semantic similarity measures as possible.

Acknowledgements

This work was partly done within the MediaLog research project, financed by the foundation *Carl-Olof och Jenz Hamrins Stiftelse*. Special thanks to the three anonymous reviewers for valuable comments on how to improve this paper.

References

1. Sandkuhl, K.: Information logistics in networked organisations: Selected concepts and applications. In: Cardoso, J., Cordeiro, J., Filipe, J. (eds.) *Enterprise Information Systems VIII*. LNBIP. Springer, Heidelberg (to appear, 2008)
2. Billig, A., Blomqvist, E., Lin, F.: Semantic matching based on enterprise ontologies. In: Meersman, R., Tari, Z. (eds.) *OTM 2007, Part I*. LNCS, vol. 4803, pp. 1161–1168. Springer, Heidelberg (2007)
3. Rodríguez, M.A., Egenhofer, M.J.: Comparing geospatial entity classes: An asymmetric and context-dependent similarity measure. *International Journal of Geographical Information Science* 18(3) (April-May 2004)
4. Andreasen, T., Bulskov, H., Knappe, R.: From ontology over similarity to query evaluation. In: *Proceedings of the 2nd CoLogNET-ElsNET Symposium - Questions and Answers: Theoretical and Applied Perspectives*, Amsterdam, Holland, December 18, 2003, pp. 39–50 (2003)
5. Janowicz, K.: Kinds of contexts and their impact on semantic similarity measurement. In: *Proceedings of the Sixth Annual IEEE International Conference on Pervasive Computing and Communications* (2008)

¹⁰ <http://infoeng.hj.se/jot/>

6. Blanchard, E., Harzallah, M., Briand, H., Kuntz, P.: A typology of ontology-based semantic measures. In: Proc. of the Open Interop Workshop on Enterprise Modelling and Ontologies for Interoperability (June 2005)
7. Raftopoulou, P., Petrakis, E.: Semantic Similarity Measures: a Comparison Study. Technical report, Technical University of Crete. Department of Electronic and Computer Engineering (January 2005)
8. Sussna, M.: Word sense disambiguation for free-text indexing using a massive semantic network. In: Proceedings of the second international conference on Information and Knowledge Management. ACM Press, New York (1993)
9. Rada, R., Mili, H., Bicknell, E., Blettner, M.: Development and application of a metric on semantic nets. *IEEE Transactions on Systems, Man, and Cybernetics* 19, 17–30 (1989)
10. Leacock, C., Chodorow, M.: Combining local context and wordnet similarity for word sense identification. In: WordNet: An electronic lexical database, pp. 265–283. MIT Press, Cambridge (1998)
11. Wu, Z., Palmer, M.: Verb semantics and lexical selection. In: Proceedings of the 32nd Annual Meeting of the Associations for the Computational Linguistics, pp. 133–138 (1994)
12. Eiter, T., Mannila, H.: Distance measures for point sets and their computation. *Acta Informatica* (1997)
13. Ramon, J., Bruynooghe, M.: A polynomial time computable metric between point sets. *Acta Informatica* 37(10) (July 2001)
14. Kalousis, A., Hilario, M.: Representational issues in meta-learning. In: Proceedings of the Twentieth International Conference on Machine Learning (ICML 2003), Washington, DC (2003)
15. Cross, V., Sun, Y.: Semantic, fuzzy set and fuzzy measure similarity for the gene ontology. In: IEEE International Fuzzy Systems Conference, 2007. FUZZ-IEEE 2007, pp. 1–6 (2007)
16. Couto, F.M., Silva, M.J., Coutinho, P.M.: Measuring semantic similarity between gene ontology terms. *Data & Knowledge Engineering* 61(1), 137–152 (2007)
17. Lord, P., Stevens, R., Brass, A., Goble, C.: Semantic similarity measures as tools for exploring the gene ontology. In: Proceedings of the 8th Pacific Symposium on Biocomputing (2003)
18. Petrakis, E.G., Varelas, G., Hliaoutakis, A., Raftopoulou, P.: Design and evaluation of semantic similarity measures for concepts stemming from the same or different ontologies. In: 4th Workshop on Multimedia Semantics (WMS 2006), Chania, Crete, Greece, pp. 44–52 (2006)
19. Siegel, S.: Nonparametric statistics for the behavioral sciences. McGraw-Hill, New York (1956)
20. Miller, G.A., Charles, W.G.: Contextual correlates of semantic similarity. *Language and Cognitive Processes* 6, 1–28 (1991)
21. Resnik, P.: Using information content to evaluate semantic similarity in a taxonomy. In: Proceedings of IJCAI 1995 (1995)

Equivalence of XSD Constructs and Its Exploitation in Similarity Evaluation

Irena Mlýnková

Department of Software Engineering, Charles University
Malostranské nám. 25, 118 00 Prague 1, Czech Republic
`irena.mlynkova@mff.cuni.cz`

Abstract. In this paper we propose a technique for evaluating similarity of XML Schema fragments. Firstly, we define classes of structurally and semantically equivalent XSD constructs. Then we propose a similarity measure that is based on the idea of edit distance utilized to XSD constructs and enables one to involve various additional similarity aspects. In particular, we exploit the equivalence classes and semantic similarity of element/attribute names. Using preliminary experiments we show the behavior and advantages of the proposal.

1 Introduction

The eXtensible Markup Language (XML) [3] has become a standard for data representation and, thus, it appears in most of areas of information technologies. A possible optimization of XML-based methods can be found in exploitation of similarity of XML data. In this paper we focus on similarity of XML schemas that can be viewed from two perspectives. We can deal with either *quantitative* or *qualitative* similarity measure. In the former case we are interested in the degree of difference of the schemas, in the latter one we also want to know how the schemas relate, e.g. which of the schemas is more general. In this paper we deal with quantitative measure that is the key aspect of schema mapping [4, 5], i.e. searching for (sub)schemas that describe the same reality.

In this area the key emphasis is currently put on the semantic similarity of element/attribute names reflecting the requirements of corresponding applications. And if the approaches consider schema structure, they usually analyze only simple aspects such as, e.g., leaf nodes or child nodes. In addition, most of the approaches deal with XML schemas expressed in simple DTD language [3]. Hence, in this paper we focus on similarity of XML schema fragments expressed in XML Schema language [2, 11]. In particular, we cover all key XML Schema constructs and we deal with their structural and semantic equivalence. We propose a similarity measure that is based on the idea of classical edit distance utilized to XSD¹ constructs and enables one to involve various additional similarity aspects. In particular, we exploit the equivalence classes of XML constructs

¹ XML Schema Definition.

and semantic similarity of element/attribute names. Using various experiments we show the behavior and advantages of the proposed approach.

The paper is structured as follows: Section 2 describes the related works. In Section 3 we overview possible XML Schema constructs and we define their structurally and semantically equivalent classes. In Section 4 we describe the proposed approach and in Section 5 we overview results of related experiments. Finally, Section 6 provides conclusions and outlines future work.

2 Related Work

The number of existing works in the area of XML data similarity is nontrivial. We can search for similarity among XML documents, XML schemas or between the two groups. We can distinguish several levels of similarity, such as, e.g., structural level, semantic level or constraint level. Or we can require different precision of the similarity.

In case of document similarity we distinguish techniques expressing similarity of two documents D_x and D_y using edit distance, i.e. by measuring how difficult is to transform D_x into D_y (e.g. [9]) and techniques which specify a simple and reasonable representation of D_x and D_y , such as, e.g., using a set of paths, that enables efficient comparison and similarity evaluation (e.g. [12]). In case of similarity of a document D and a schema S there are also two types of strategies – techniques which measure the number of elements which appear in D but not in S and vice versa (e.g. [1]) and techniques which measure the closest distance between D and “all” documents valid against S (e.g. [8]). And finally, methods for measuring similarity of two XML schemas S_x and S_y combine various supplemental information and similarity measures such as, e.g., predefined similarity rules, similarity of element/attribute names, equality of data types, similarity of schema instances or previous results (e.g. [4, 5]). But, in general, the approaches focus mostly on semantic aspects, whereas structural ones are of marginal importance. And what is more, most of the existing works consider only DTD constructs, whereas if the XML Schema language is supported, the constructs beyond DTD expressive power are often ignored.

3 XML Schema Constructs and Their Equivalence

The most popular language for description of the allowed structure of XML documents is currently the Document Type Definition (DTD) [3]. For simple applications it is sufficient, but more complex ones the W3C proposed a more powerful tool – the XML Schema language [2, 11]. A self-descriptive example of an XSD is depicted in Figure 1.

The constructs of XML Schema can be divided into *basic*, *advanced* and *auxiliary*. The basic constructs involve simple data types (`simpleType`), complex data types (`complexType`), elements (`element`), attributes (`attribute`), groups of elements (`group`) and groups of attributes (`attributeGroup`). Simple data types involve both built-in data types (except for `ID`, `IDREF`, `IDREFS`), such as,

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="employees">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="person" minOccurs="1" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:element name="person">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="name"/>
        <xs:element name="email" type="xs:string" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element ref="relationships" minOccurs="0" maxOccurs="1"/>
      </xs:sequence>
      <xs:attribute name="id" type="xs:ID" use="required"/>
      <xs:attribute name="note" type="xs:string"/>
      <xs:attribute name="holiday" default="no">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="yes"/>
            <xs:enumeration value="no"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:complexType>
  </xs:element>

  <xs:element name="name">
    <xs:complexType>
      <xs:all>
        <xs:element name="first" type="xs:string"/>
        <xs:element name="surname" type="xs:string"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="relationships">
    <xs:complexType>
      <xs:attribute name="superior" type="xs:IDREF"/>
      <xs:attribute name="inferior" type="xs:IDREFS"/>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Fig. 1. An example of an XSD of employees I

e.g., **string**, **integer**, **date** etc., as well as user-defined data types derived from existing simple types using **simpleType** construct. Complex data types enable one to specify both content models of elements and their sets of attributes. The content models can involve ordered sequences (**sequence**), choices (**choice**), unordered sequences (**all**), groups of elements (**group**) or their allowable combinations. Similarly, they enable one to derive new complex types from existing simple (**simpleContent**) or complex types (**complexContent**). Elements simply join simple/complex types with respective element names and, similarly, attributes join simple types with attribute names. And, finally, groups of elements and attributes enable one to globally mark selected schema fragments and exploit them repeatedly in various parts using so-called *references*. In general, basic constructs are present in almost all XSDs.

The set of *advanced* constructs involves type substitutability and substitution groups, identity constraints (**unique**, **key**, **keyref**) as well as related simple data types (**ID**, **IDREF**, **IDREFS**) and assertions (**assert**, **report**). Type substitutability and substitution groups enable one to change data types or allowed location of elements. Identity constraints enable one to restrict allowed values of elements/attributes to unique/key values within a specified area and to specify references to them. Similarly, assertions specify additional conditions that the

values of elements/attributes need to satisfy, i.e. they can be considered as an extension of simple types.

The set of *auxiliary* constructs involves wildcards (**any**, **anyAttribute**), external schemas (**include**, **import**, **redefine**), notations (**notation**) and annotations (**annotation**). Wildcards and external schemas combine data from various XML schemas. Notations bear additional information for superior applications. And annotations can be considered as a kind of advanced comments. Consequently, since these constructs do not have a key impact on schema structure or semantics, we will not deal with them in the rest of the text.

3.1 Structural Equivalence

As it is obvious from the above overview, there are sets of XML Schema constructs that enable one to generate XSDs that have different structure but are *structurally equivalent*.

Definition 1. Let S_x and S_y be two XSD fragments. Let $I(S) = \{D \text{ s.t. } D \text{ is an XML document fragment valid against } S\}$. Then S_x and S_y are structurally equivalent, $S_x \sim S_y$, if $I(S_x) = I(S_y)$.

Consequently, having a set X of all XSD constructs, we can specify the quotient set $X/\sim$ of X by $\sim$ and respective equivalence classes – see Table 1.

Table 1. XSD equivalence classes of $X/\sim$

Class	Constructs	Canonical representative
C_{ST}	globally defined simple type, locally defined simple type	locally defined simple type
C_{CT}	globally defined complex type, locally defined complex type	locally defined complex type
C_{El}	referenced element, locally defined element	locally defined element
C_{At}	referenced attribute, locally defined attribute, attribute referenced via an attribute group	locally defined attribute
C_{ElGr}	content model referenced via an element group, locally defined content model	locally defined content model
C_{Seq}	unordered sequence of elements $e_1, e_2, \dots, e_l$, choice of all possible ordered sequences of $e_1, e_2, \dots, e_l$	choice of all possible ordered sequences of $e_1, e_2, \dots, e_l$
C_{CTDer}	derived complex type, newly defined complex type	newly defined complex type
C_{SubSk}	elements in a substitution group G , choice of elements in G	choice of elements in G
C_{Sub}	data types $M_1, M_2, \dots, M_k$ derived from type M , choice of content models defined in $M_1, M_2, \dots, M_k, M$	choice of content models defined in $M_1, M_2, \dots, M_k, M$

Classes C_{ST} and C_{CT} specify that there is no difference if a simple or a complex type is defined globally or locally as depicted in Figure 2.

<pre>... <xs:attribute name="holiday"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:enumeration value="yes"/> <xs:enumeration value="no"/> </xs:restriction> </xs:simpleType> </xs:attribute> <xs:element name="name"> <xs:complexType> <xs:all> <xs:element name="first" type="xs:string"/> <xs:element name="surname" type="xs:string"/> </xs:all> </xs:complexType> </xs:element> ...</pre>	<pre>... <xs:attribute name="holiday" type="typeHoliday"/> <xs:simpleType name="typeHoliday"> <xs:restriction base="xs:string"> <xs:enumeration value="yes"/> <xs:enumeration value="no"/> </xs:restriction> </xs:simpleType> <xs:element name="name" type="typeName"/> <xs:complexType name="typeName"> <xs:all> <xs:element name="first" type="xs:string"/> <xs:element name="surname" type="xs:string"/> </xs:all> </xs:complexType> ...</pre>

Fig. 2. Locally and globally defined data types

Similarly, classes C_{El} and C_{At} determine that locally defined elements/attributes and globally defined referenced elements/attributes are equivalent as depicted in Figure 3.

<pre>... <xs:complexType name="typePerson"> <xs:sequence> <xs:element name="name" type="xs:string"/> <xs:element name="email" type="xs:string"/> </xs:sequence> <xs:attribute name="id" type="xs:ID"/> <xs:attribute name="note" type="xs:string"/> </xs:complexType> ...</pre>	<pre>... <xs:complexType name="typePerson"> <xs:sequence> <xs:element name="name" type="xs:string"/> <xs:element ref="email"/> </xs:sequence> <xs:attribute ref="id"/> <xs:attribute name="note" type="xs:string"/> </xs:complexType> <xs:attribute name="id" type="xs:ID"/> <xs:element name="email" type="xs:string"/> ...</pre>

Fig. 3. Locally and globally defined elements and attributes

In addition, C_{At} determines that also attributes referenced via attribute groups are equivalent to all other types of attribute specifications. And a similar meaning has also C_{ElGr} class for content models referenced via groups or defined locally. Both situations are depicted in Figure 4.

Class C_{Seq} expresses the equivalence between an unordered sequence of elements $e_1, e_2, \dots, e_l$ and a choice of its all possible ordered permutations as depicted in Figure 5.

Class C_{Inh} determines equivalence between a complex type derived from an existing one or a complex type that is defined newly as depicted in Figure 6.

Class C_{SubSk} expresses that the mechanism of substitution groups is equivalent to the choice of respective elements, i.e. having elements e_1 and e_2 that are in substitution group of element e_3 , it means that everywhere where e_3 occurs, also elements e_1 and e_2 can occur. The only exception is if e_3 is denoted as *abstract*

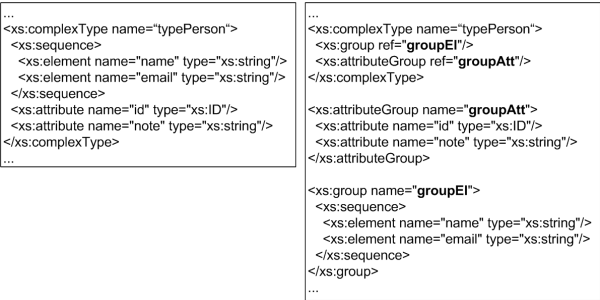


Fig. 4. Element and attribute groups

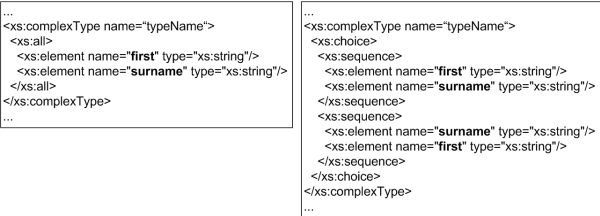


Fig. 5. Unordered and ordered sequences

(using attribute `abstract="true"`) and, hence, it must be always substituted. The structure of a substitution group can be also influenced via attributes `final` and `block` that disable substitution for a particular element anywhere it occurs or only at particular positions. An example of respective equivalent schemas is depicted in Figure 7, where we assume that types `typeBook` and `typeJournal` are derived from `typePublication`.

Similarly, class C_{Sub} expresses the fact that having an element e with type M , using the attribute `xsi:type` we can substitute M with any of data types $M_1, M_2, \dots, M_k$ derived from M . Consequently, the content model of e is equivalent to choice of content models defined in $M_1, M_2, \dots, M_k, M$. The only exception is when the type substitutability is blocked using the `block` attribute. An example of the equivalent schemas is depicted in Figure 8.

Each of the remaining XML Schema constructs not mentioned in Table 1 forms a single class. We will denote these classes as $C_1, C_2, \dots, C_n$.

3.2 Semantic Equivalence

Apart from XSD constructs that restrict the allowed structure of XML data, we can find also constructs that express various semantic constraints. They involve identity constrains and simple data types `ID` and `IDREF(S)`. (Note that `ID`, `IDREF(S)` can be expressed using `key` and `keyref`.) The idea of semantic similarity is based on the following observation: A `keyref` construct refers to a particular part of the XSD – e.g. having an XSD containing a list of books

<pre> ... <xs:complexType name="typePerson1"> <xs:sequence> <xs:element name="name" type="xs:string"/> </xs:sequence> <xs:attribute name="id" type="xs:ID"/> </xs:complexType> <xs:complexType name="typePerson2"> <xs:complexContent> <xs:extension base="typePerson1"> <xs:sequence> <xs:element name="email" type="xs:string"/> </xs:sequence> <xs:attribute name="note" type="xs:string"/> </xs:extension> </xs:complexContent> </xs:complexType> ... </pre>	<pre> ... <xs:complexType name="typePerson1"> <xs:sequence> <xs:element name="name" type="xs:string"/> </xs:sequence> <xs:attribute name="id" type="xs:ID"/> </xs:complexType> <xs:complexType name="typePerson2"> <xs:sequence> <xs:element name="name" type="xs:string"/> <xs:element name="email" type="xs:string"/> </xs:sequence> <xs:attribute name="id" type="xs:ID"/> <xs:attribute name="note" type="xs:string"/> </xs:complexType> ... </pre>

Fig. 6. Derived complex types

<pre> ... <xs:element name="publication" type="typePublication"/> <xs:element name="book" type="typeBook" substitutionGroup="publication"/> <xs:element name="journal" type="typeJournal" substitutionGroup="publication"/> <xs:element name="library"> <xs:complexType> <xs:sequence> <xs:element ref="publication" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element> ... </pre>	<pre> ... <xs:element name="library"> <xs:complexType> <xs:sequence> <xs:choice maxOccurs="unbounded"> <xs:element ref="publication"/> <xs:element ref="book"/> <xs:element ref="journal"/> </xs:choice> </xs:sequence> </xs:complexType> </xs:element> ... </pre>

Fig. 7. Substitution groups

and a list of authors, each author can refer to his best book. And this situation described in a semantically equivalent manner occurs when the referenced fragment, i.e. the element describing the best book, is directly present within element author. Hence, these constructs enable one to generate XSDs that have different structure but are *semantically equivalent*.

Definition 2. Let S_x and S_y be two XSD fragments. Then S_x and S_y are semantically equivalent, $S_x \approx S_y$, if they abstract the same reality.

Having a set X of all XSD constructs, we can specify the quotient set $X/\approx$ of X by $\approx$ and respective equivalence classes – see Table 2. Classes C'_{IdRef} and C'_{KeyRef} express the fact that both **IDREF(S)** and **keyref** constructs, i.e. references to schema fragments, are semantically equivalent to the situation when we directly copy the referenced schema fragments to the referencing positions. An example of the equivalent schemas is depicted in Figure 9.

Since every **key/keyref** constraint must contain one reference (**selector**) to a set of elements and at least one reference (**field**) to their subelements (descendants) and/or attributes expressed in the following grammar [2, 11]:

```

Selector ::= PathS ( ' | ' PathS ) *
Field    ::= PathF ( ' | ' PathF ) *
PathS    ::= ( ' . / ' ) ? Step ( ' / ' Step ) *

```

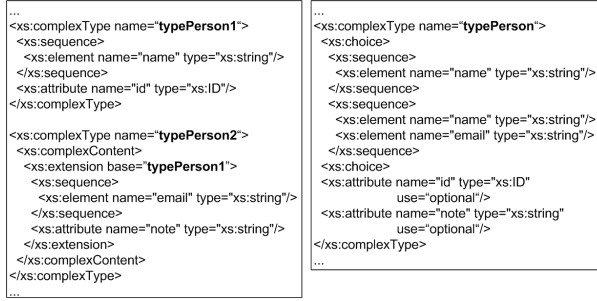


Fig. 8. Type substitutability

Table 2. XSD equivalence classes of $X/\approx$

Class	Constructs	Canonical representative
C'_{IdRef}	locally defined schema fragment, schema fragment referenced via IDREF attribute	locally defined schema fragment
C'_{KeyRef}	locally defined schema fragment, schema fragment referenced via keyref element	locally defined schema fragment

PathF ::= ('./')? (Step '/')* (Step | '@' NameTest)
 Step ::= '.' | NameTest
 NameTest ::= QName | '*' | NCName ':' '*'

the referenced fragments can be always easily copied to particular positions.

Similar to the previous case, each of the remaining XML Schema constructs not mentioned in Table 2 forms a single class. We will denote these classes as $C'_1, C'_2, \dots, C'_m$.

Each of the previously defined classes of $\sim$ or $\approx$ equivalence can be represented using any of its elements. Since we want to simplify the specification of XSD for the purpose of analysis of its similarity, we have selected respective *canonical representatives* listed in Tables 1 and 2 as well. They enable one to simplify the structure of the XSD only to core constructs. (Note that since $C_1, C_2, \dots, C_n$ and $C'_1, C'_2, \dots, C'_m$ are singletons, the canonical representatives are obvious.)

4 Similarity Evaluation

The proposed algorithm is based mainly on the work presented in [9] which focuses on expressing similarity of XML documents D_x and D_y using tree edit distance. The main contribution of the algorithm is in introducing two new edit operations *InsertTree* and *DeleteTree* which allow manipulating more complex structures than only a single node. But, repeated structures can be found in an XSD as well, if it contains *shared* fragments or *recursive* elements.

<pre> ... <xs:element name="person"> <xs:complexType> <xs:sequence> <xs:element name="name" type="xs:string"/> </xs:sequence> <xs:attribute name="id" type="xs:ID"/> </xs:complexType> </xs:element> <xs:element name="relationships"> <xs:complexType> <xs:attribute name="inferior" type="xs:IDREFS"/> </xs:complexType> </xs:element> ... </pre>	<pre> ... <xs:element name="relationships"> <xs:complexType> <xs:sequence> <xs:element name="personInferior" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element name="name" type="xs:string"/> </xs:sequence> <xs:attribute name="id" type="xs:ID"/> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> ... </pre>

Fig. 9. Identity constraints

On the other hand, contrary to XML documents that can be modeled as trees, XSDs can, in general, form general cyclic graphs. Hence, procedures for computing edit distance of trees need to be utilized to XSD graphs. In addition, not only the structural, but also the semantic aspect is very important. Therefore, we will also concern both semantic equivalence of XSD fragments as well as semantic similarity of element/attribute names.

The whole method can be divided into three parts depicted in Algorithm 1.

Algorithm 1. Main body of the algorithm

Input: XSDs S_x and S_y

Output: Edit distance between S_x and S_y

1. $T_x = \text{ParseXSD}(S_x)$;
 2. $T_y = \text{ParseXSD}(S_y)$;
 3. $\text{Cost}_{\text{Graft}} = \text{ComputeCost}(T_y)$;
 4. $\text{Cost}_{\text{Prune}} = \text{ComputeCost}(T_x)$;
 5. **return** EditDistance($T_x, T_y, \text{Cost}_{\text{Graft}}, \text{Cost}_{\text{Prune}}$);
-

Firstly, the input XSDs S_x and S_y are parsed (line 1 and 2) and their tree representations are constructed. Next, costs for tree inserting (line 3) and tree deleting (line 4) are computed. And in the final step (line 5) we compute the resulting edit distance, i.e. similarity, using classical dynamic programming.

4.1 XSD Tree Construction

The key operation of our approach is tree representation of the given XSDs. However, since the structure of an XSD can be quite complex, we firstly normalize and simplify it.

Normalization of XSDs. Firstly, we normalize the given XSDs using the equivalence classes. In the first step we exploit structural equivalence $\sim$ and we iteratively replace each non-canonical construct (naturally except for the root element) with the respective canonical representative until there can be found

any. At the same time, for each element v of the schema (i.e. XSD construct) we keep the set $v_{eq\sim}$ of classes it originally belonged to.

In the second step we exploit semantic equivalence $\approx$ and we again replace each non-canonical construct with its canonical representative and we construct sets $v_{eq\approx}$. Now the resulting schema involves elements, attributes, operators **choice** and **sequence**, intervals of allowed occurrences, simple types and assertions.

Simplification of XSDs. Next we simplify the remaining content models. For this purpose we can use various transformation rules. Probably the biggest set was defined in [10] for DTD constructs, but these simplifications are for our purpose too strong. Hence, we use only a subset of them as depicted in Figures 10 and 11. They are expressed for DTD constructs, where “|” represents **choice**, “,” represents **sequence**, “?” represents interval $[0, 1]$, “+” represents intervals $[v_{low}, v_{up}]$, where $v_{low} > 0$ and $v_{up} > 1$, “*” represents intervals $[v_{low}, v_{up}]$, where $v_{low} \geq 0$ and $v_{up} > 1$ and empty operator represents interval $[1, 1]$.

I-a)	$(e_1 e_2)^* \rightarrow e_1^*, e_2^*$
I-b)	$(e_1, e_2)^* \rightarrow e_1^*, e_2^*$
I-c)	$(e_1, e_2)? \rightarrow e_1?, e_2?$
I-d)	$(e_1, e_2)^+ \rightarrow e_1^+, e_2^+$
I-e)	$(e_1 e_2)^+ \rightarrow e_1?, e_2?$

Fig. 10. Flattening rules

II-a)	$e_1^{++} \rightarrow e_1^+$	II-b)	$e_1^{**} \rightarrow e_1^*$
II-c)	$e_1^{*?} \rightarrow e_1^*$	II-d)	$e_1^{?*} \rightarrow e_1^*$
II-e)	$e_1^{++} \rightarrow e_1^*$	II-f)	$e_1^{*+} \rightarrow e_1^*$
II-g)	$e_1^{?+} \rightarrow e_1^*$	II-h)	$e_1^{+?} \rightarrow e_1^*$
II-i)	$e_1^{??} \rightarrow e_1^?$		

Fig. 11. Simplification rules

The rules enable one to convert all element definitions so that each cardinality constraint operator is connected to a single element. The second purpose is to avoid usage of **choice** construct. Note that some of the rules do not produce equivalent XML schemes and cause a kind of information loss. But this aspect is common for all existing XML schema similarity measures – it seems that the full generality of the regular expressions cannot be captured easily.

XSD Tree. Having a normalized and simplified XSD, its tree representation is defined as follows:

Definition 3. An XSD tree is an ordered tree $T = (V, E)$, where

1. V is a set of nodes of the form $v = (v_{Type}, v_{Name}, v_{Cardinality}, v_{eq\sim}, v_{eq\approx})$, where v_{Type} is the type of a node (i.e. attribute, element or particular simple data type), v_{Name} is the name of an element or an attribute, $v_{Cardinality}$ is the interval $[v_{low}, v_{up}]$ of allowed occurrence of v , $v_{eq\sim}$ is the set of classes of $\sim v$ belongs to and $v_{eq\approx}$ is the set of classes of $\approx v$ belongs to,
2. $E \subseteq V \times V$ is a set of edges representing relationships between elements and their attributes or subelements.

An example of tree representation of XSD in Figure 1 (after normalization and simplification) is depicted in Figure 12.

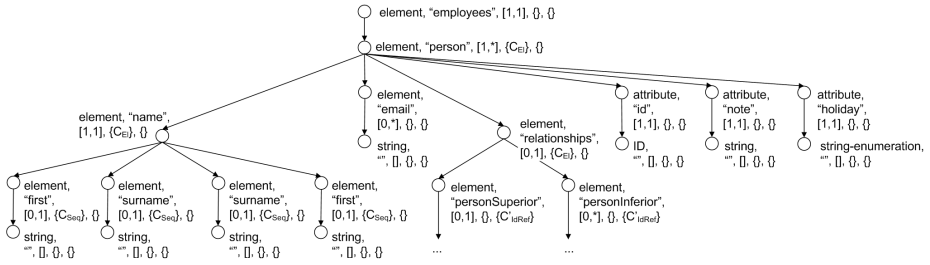


Fig. 12. An example of an XSD tree

Shared and Recursive Elements. As we have mentioned, the structure of an XSD does not have to be purely tree-like. There can occur both shared elements which invoke undirected cycles and recursive elements which invoke directed cycles. The shared elements are eliminated in XSD normalization using canonical representatives, where all globally defined schema fragments are replaced with their locally defined copy. But, in case of recursive elements we cannot repeat the same idea since the recursion would invoke infinitely deep tree branches. However, in this case we exploit the observation of an analysis of real-world XML data [7] that the amount of recursive inclusions is on average less than 10. So we approximate the infinite amount with the constant one. Naturally, this is a kind of information loss, but based on the knowledge of real-world data.

4.2 Tree Edit Operations

Having the above described tree representation of an XSD, we can now easily utilize the tree edit algorithm proposed in [9]. For a given tree T with a root node r of degree t and its first-level subtrees $T_1, T_2, \dots, T_t$, the tree edit operations are defined formally as follows:

Definition 4. *Substitution $_T(r_{new})$ is a node substitution operation applied to T that yields the tree T' with root node r_{new} and first-level subtrees $T_1, \dots, T_t$.*

Definition 5. *Given a node x with degree 0, Insert $_T(x, i)$ is a node insertion operation applied to T at i that yields the new tree T' with root node r and first-level subtrees $T_1, \dots, T_i, x, T_{i+1}, \dots, T_t$.*

Definition 6. *If the first-level subtree T_i is a leaf node, Delete $_T(T_i)$ is a delete node operation applied to T at i that yields the tree T' with root node r and first-level subtrees $T_1, \dots, T_{i-1}, T_{i+1}, \dots, T_t$.*

Definition 7. *Given a tree T_x , InsertTree $_T(T_x, i)$ is an insert tree operation applied to T at i that yields the tree T' with root node r and first-level subtrees $T_1, \dots, T_i, T_x, T_{i+1}, \dots, T_t$.*

Definition 8. *DeleteTree $_T(T_i)$ is a delete tree operation applied to T at i that yields the tree T' with root node r and first-level subtrees $T_1, \dots, T_{i-1}, T_{i+1}, \dots, T_t$.*

Transformation of a source tree T_x to a destination tree T_y can be done using a number of sequences of the operations. But, we can only deal with so-called *allowable* sequences, i.e. the relevant ones. For the purpose of our approach we only need to modify the original definition as follows:

Definition 9. *A sequence of edit operations transforming a source tree T_x to a destination tree T_y is allowable if it satisfies the following two conditions:*

1. *A tree T may be inserted only if tree similar to T already occurs in T_x . A tree T may be deleted only if tree similar to T occurs in T_y .*
2. *A tree that has been inserted via the *InsertTree* operation may not subsequently have additional nodes inserted. A tree that has been deleted via the *DeleteTree* operation may not previously have had nodes deleted.*

While the original definition requires exactly the same nodes and trees, we relax the requirement only to similar ones. The exact meaning of the similarity is explained in the following text and enables one to combine the tree edit distance with other approaches. Also note that each of the edit operations is associated with a non-negative cost.

4.3 Costs of Inserting and Deleting Trees

Inserting (deleting) a subtree T_i can be done with a single operation *InsertTree* (*DeleteTree*) or with a combination of *InsertTree* (*DeleteTree*) and *Insert* (*Delete*) operations. To find the optimal variant the algorithm uses pre-computed cost for inserting T_i , $Cost_{Graft}(T_i)$ and deleting tree T_i , $Cost_{Prune}(T_i)$. The procedure can be divided into two parts: In the first part *ContainedIn* list is created for each subtree of T_i . In the second part $Cost_{Graft}$ and $Cost_{Prune}$ are computed for T_i . The procedure is described in [9], but in our approach it is modified to involve similarity of elements/attributes and their respective parameters.

Similarity of Elements/Attributes. Similarity of two elements/attributes v and v' can be evaluated using various criteria. Since the structural similarity is solved via the edit distance, we focus on semantic and syntactic similarity of element/attribute names, cardinality-constraint similarity, structural/semantic similarity of schema fragments and similarity of simple data types.

Semantic similarity of element/attribute names is a score that reflects the semantic relation between the meanings of two words. We exploit procedure described in [5] which determines ontology similarity between two words v_{Name} and v'_{Name} by comparing v_{Name} with synonyms of v'_{Name} .

Syntactic similarity of element/attribute names is determined by computing the edit distance between v_{Name} and v'_{Name} . For our purpose the classical Levenshtein algorithm [6] is used that determines the edit distance of two strings using inserting, deleting or replacing single characters.

Similarity of cardinality constraints is determined by similarity of intervals $v_{Cardinality} = [v_{low}, v_{up}]$ and $v'_{Cardinality} = [v'_{low}, v'_{up}]$. It is defined as follows:

$$\begin{aligned}
CardSim(v, v') &= 0 && ; (v_{up} < v'_{low}) \vee (v'_{up} < v_{low}) \\
&= 1 && ; v_{up}, v'_{up} = \infty \wedge v_{low} = v'_{low} \\
&= 0.9 && ; v_{up}, v'_{up} = \infty \wedge v_{low} \neq v'_{low} \\
&= 0.6 && ; v_{up} = \infty \vee v'_{up} = \infty \\
&= \frac{\min(v_{up}, v'_{up}) - \max(v_{low}, v'_{low})}{\max(v_{up}, v'_{up}) - \min(v_{low}, v'_{low})} && ; \text{otherwise}
\end{aligned}$$

Structural/semantic similarity of schema fragments rooted at v and v' is determined by the similarity of sets $v_{eq\sim}$, $v'_{eq\sim}$ and $v_{eq\approx}$, $v'_{eq\approx}$ as follows:

$$\begin{aligned}
StrFragSim(v, v') &= 1 && ; v_{eq\sim}, v'_{eq\sim} = \emptyset \\
&= \frac{|v_{eq\sim} \cap v'_{eq\sim}|}{|v_{eq\sim} \cup v'_{eq\sim}|} && ; \text{otherwise} \\
SemFragSim(v, v') &= 1 && ; v_{eq\approx}, v'_{eq\approx} = \emptyset \\
&= \frac{|v_{eq\approx} \cap v'_{eq\approx}|}{|v_{eq\approx} \cup v'_{eq\approx}|} && ; \text{otherwise}
\end{aligned}$$

And, finally, *similarity of data types* is determined by similarity of simple types v_{Type} and v'_{Type} . It is specified by *type compatibility matrix* that determines similarity of distinct simple types. For instance, similarity of **string** and **normalizedString** is 0.9, whereas similarity of **string** and **positiveInteger** is 0.5. Similarly, the table involves similarity of restrictions of simple types specified either via derivation of data types or assertions as well as similarity between element and attribute nodes. (We omit the whole table for the paper length.)

The overall similarity, $Sim(v, v')$ is computed as follows:

$$\begin{aligned}
Sim(v, v') &= Max(SemanticSim(v, v'), SyntacticSim(v, v')) \times \alpha_1 \\
&\quad + CardSim(v, v') \times \alpha_2 \\
&\quad + StrFragSim(v, v') \times \alpha_3 \\
&\quad + SemFragSim(v, v') \times \alpha_4 \\
&\quad + DataTypeSim(v, v') \times \alpha_5
\end{aligned}$$

where $\sum_{i=1}^5 \alpha_i = 1$ and $\forall i : \alpha_i \geq 0$.

Construction of ContainedIn Lists. The procedure for determining element/attribute similarity is used for creating *ContainedIn* lists which are then used for computing $Cost_{Graft}$ and $Cost_{Prune}$. The list is created for each node of the destination tree T_y and contains pointers to similar nodes in the source tree T_x . The procedure for creating *ContainedIn* lists is shown in Algorithm 2.

Since creating of lists from leaves and continues to root, there is recursive calling of procedure at line 2. At line 4 we find all similar nodes of r in tree T_x and add them to a temporary list. If r is a leaf node, the *ContainedIn* list is created. For a non-leaf node we have to filter the list with lists of its descendants (line 6). At this step each descendant of r has to be found at corresponding position in descendants of nodes in the created *ContainedIn* list. More precisely, let $u \in r_{ContainedIn}$, $children_u$ is the set of u descendants and v is a child of r . Then $v_{ContainedIn} \cap children_u \neq \emptyset$, otherwise u is removed from $r_{ContainedIn}$.

Costs of Inserting Trees. When the *ContainedIn* list with corresponding nodes is created for node r , the cost for inserting the tree rooted at r can be assigned.

Algorithm 2. CreateContainedInLists(T_x, r)

Input: tree T_x , root r of tree T_y **Output:** *ContainedIn* lists for all nodes in tree T_y

1. **for all** *child* of r **do**
 2. CreateContainedInLists($T_x, child$);
 3. **end for**
 4. $r_{ContainedIn} = \text{FindSimilarNodes}(T_x, r)$;
 5. **for all** *child* of r **do**
 6. $r_{ContainedIn} = \text{FilterLists}(r_{ContainedIn}, child_{ContainedIn})$;
 7. **end for**
 8. Sort($r_{ContainedIn}$);
-

The procedure is shown in Algorithm 3. The *forall* loop computes sum sum_0 for inserting node r and all its subtrees. If *InsertTree* operation can be applied (*ContainedIn* list of r is not empty), sum_1 is computed for this operation at line 8. The minimum of these costs is finally denoted as $Cost_{Graft}$ for node r .

Algorithm 3. ComputeCost(r)

Input: root r of tree T_y **Output:** $Cost_{Graft}$ for tree T_y

1. $sum_0 = 1$;
 2. **for all** *child* of r **do**
 3. ComputeCost(*child*);
 4. $sum_0 += Cost_{Graft}(child)$;
 5. **end for**
 6. $sum_1 = \infty$;
 7. **if** $r_{ContainedIn}$ is not empty **then**
 8. $sum_1 = \text{ComputeInsertTreeCost}(r)$;
 9. **end if**
 10. $Cost_{Graft}(r) = \text{Min}(sum_0, sum_1)$;
-

Costs of Deleting Trees. Since the rules for deleting a subtree from the source tree T_x are the same as rules for inserting a subtree into the destination tree T_y , costs for deleting trees are obtained by the same procedures. We only switch tree T_x with T_y in procedures *CreateContainedInLists* and *ComputeCost*.

4.4 Computing Edit Distance

The last part of the algorithm, i.e. computing the edit distance, is based on dynamic programming. At this step the procedure decides which of the operations defined in Section 4.2 will be applied for each node to transform source tree T_x to destination tree T_y . This part of algorithm does not have to be modified for XSDs so the original procedure presented in [9] is used.

4.5 Complexity

In [9] it was proven that the complexity of transforming tree T_x into tree T_y is $O(|T_x||T_y|)$. In our method we have to consider procedures for constructing XSD trees and mainly for evaluating similarity. Constructing an XSD tree can be done in $O(|T_x|)$ for tree T_x . Complexity of similarity evaluation depends on procedures *SemanticSim*, *SyntacticSim*, *CardSim*, *StrFragSim*, *SemFragSim* and *DataTypeSim*. Syntactic similarity is computed for each pair of elements in T_x and T_y , so its complexity is $O(|T_x||T_y||\omega|)$, where ω is maximum length of an element/attribute label. Similarity of cardinality, similarity of simple types and structural/semantic similarity of schema fragments is also computed for each pair of elements, however, it is an operation with constant complexity, i.e. their complexity is $O(|T_x||T_y|)$. Complexity of finding semantic similarity depends on the size of the thesaurus and on the number of iterations of searching synonyms. Since it is reasonable to search synonyms only for a few steps, the overall complexity is $O(|T_x||T_y||\Sigma|)$, where Σ is the set of words in the thesaurus. And it also determines the overall complexity of the algorithm.

5 Experiments

For the purpose of experimental evaluation of the proposal we have created next two synthetic XSDs that are from various points of view more or less similar to the XSD depicted in Figure 1. They are depicted in Figures 13 and 14.

At first glance the XSD II is structurally highly different from the original XSD (denoted as XSD I). But, under a closer investigation, we can see that the difference is only within classes of $\sim$ equivalence. On the other hand, XSD III differs in more aspects, such as, e.g., simple types, allowed occurrences, globally/locally defined data types, exploitation of groups, element/attribute names, attributes vs. elements with simple types etc.

As we can see in Table 3 which depicts the results in case we set $\alpha_3 = \alpha_4 = 0$, i.e. we ignore the information on original constructs of XML Schema, the similarity of XSD I and XSD II is 1.0, because they are represented using identical XSD trees. Similarity between XSD I vs. XSD III and XSD II vs. XSD III are for the same reason equivalent, though naturally lower.

If we set $\alpha_3 \neq 0$ (according to our experiments it should be > 0.2 to influence the algorithm), the resulting similarity is influenced by the difference between the used XML Schema constructs. The results are depicted in Table 4, where we can see more precise results. In particular, the similarity of XSD I and II is

Table 3. Similarity for $\alpha_3 = \alpha_4 = 0$

	XSD I	XSD II	XSD III
XSD I	1.00	1.00	0.82
XSD II	1.00	1.00	0.82
XSD III	0.82	0.82	1.00

Table 4. Similarity for $\alpha_3 \neq 0$

	XSD I	XSD II	XSD III
XSD I	1.00	0.89	0.66
XSD II	0.89	1.00	0.70
XSD III	0.66	0.70	1.00

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="typeHoliday">
    <xs:restriction base="xs:string">
      <xs:enumeration value="yes"/>
      <xs:enumeration value="no"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="typeEmail">
    <xs:restriction base="xs:string">
    </xs:simpleType>

  <xs:group name="groupContact">
    <xs:sequence>
      <xs:element ref="name"/>
      <xs:element name="email" type="typeEmail"
        minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="relationships" minOccurs="0"
        maxOccurs="1"/>
    </xs:sequence>
  </xs:group>

  <xs:element name="employees">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="person" minOccurs="1"
          maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:element name="person">
    <xs:complexType>
      <xs:group ref="groupContact"/>
      <xs:attribute name="id" type="xs:ID" use="required"/>
      <xs:attribute name="note" type="xs:string"/>
      <xs:attribute name="holiday" type="typeHoliday"/>
    </xs:complexType>
  </xs:element>

  <xs:element name="name">
    <xs:complexType>
      <xs:all>
        <xs:element name="first" type="xs:string"/>
        <xs:element name="surname" type="xs:string"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="relationships">
    <xs:complexType>
      <xs:attribute name="superior" type="xs:IDREF"/>
      <xs:attribute name="inferior" type="xs:IDREFS"/>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Fig. 13. An example of an XSD of employees II

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="typeVacation">
    <xs:restriction base="xs:string">
      <xs:enumeration value="yes"/>
      <xs:enumeration value="no"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="typeEmail">
    <xs:restriction base="xs:string">
      <xs:maxLength value="256"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:group name="groupContact">
    <xs:sequence>
      <xs:element ref="name"/>
      <xs:element name="email" type="typeEmail"
        minOccurs="1" maxOccurs="unbounded"/>
      <xs:element ref="connections" minOccurs="0"
        maxOccurs="1"/>
      <xs:element name="note" type="xs:string"
        minOccurs="0" maxOccurs="1"/>
    </xs:sequence>
  </xs:group>

  <xs:element name="employees">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="subject" minOccurs="1"
          maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:element name="subject">
    <xs:complexType>
      <xs:group ref="groupContact"/>
      <xs:attribute name="id" type="xs:ID" use="required"/>
      <xs:attribute name="vacation" type="typeVacation"/>
    </xs:complexType>
  </xs:element>

  <xs:element name="name">
    <xs:complexType>
      <xs:all>
        <xs:element name="first" type="xs:string"/>
        <xs:element name="surname" type="xs:string"/>
      </xs:all>
    </xs:complexType>
  </xs:element>

  <xs:element name="connections">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="subject" minOccurs="0"
          maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Fig. 14. An example of an XSD of employees III

naturally $\neq 1.0$, and similarity of XSD II and III is higher due to the respective higher structural similarity of constructs.

On the other hand, if we set $\alpha_4 \neq 0$ and $\alpha_3 = 0$, i.e. we are interested in semantic similarity of schema fragments, the results have the same trend as results in Table 3, because we again omit structural similarity of XSD constructs,

but in this case the semantic similarity of schema fragments **relationships** and **connections** is high.

As we have mentioned in Section 4.5, the most time consuming operation of the approach which determines the overall complexity of the algorithm is searching the thesaurus. Hence, in the last test we try to omit evaluation of *SemanticSim*. If we consider the first situation, i.e. when $\alpha_3 = \alpha_4 = 0$, it influences similarity with XSD III (which drops to 0.33), whereas similarity of XSD I and II remains the same because the respective element/attribute names are the same. The results in case $\alpha_3 \neq 0$ are depicted in Table 5. As we can see, the similarity of XSD I and II remains the same again, whereas the other values are much lower.

Table 5. Similarity without *SemanticSim*

	XSD I	XSD II	XSD III
XSD I	1.00	0.89	0.24
XSD II	0.89	1.00	0.255
XSD III	0.24	0.255	1.00

In general, the experiments show that various parameters of the similarity measure can highly influence the results. On the other hand, we cannot simply analyze all possible aspects, since some applications may not be interested, e.g., in semantic similarity of used element/attribute names or the “syntactic sugar” (i.e. structurally equivalent constructs) XML Schema involves. Consequently, a reasonable approach should enable one to exploit various aspects as well as temporarily omit the irrelevant ones.

6 Conclusion

The aim of this paper was a proposal of an algorithm for evaluating similarity of XML Schema constructs which enable one to specify the structure and semantics of XML data more precisely. For this purpose we have defined structural and semantic equivalence of XSD constructs and we have proposed similarity measure based on classical edit distance strategy that enables one to analyze their structure more precisely and to involve additional similarity aspects. In particular, we have exploited the proposed equivalence classes and semantic similarity of element/attribute names.

In our future work we will focus mainly on further improvements of our approach. We will deal with other edit operations (e.g. moving a node or adding/deleting a non-leaf node), improvements of efficiency of supplemental algorithms, especially the semantic similarity, and on problems related to reasonable setting of involved weights. We will also deal with more elaborate experimental testing. In particular, we will focus on implementing a simulator that would provide distinct XSDs.

Acknowledgement

This work was supported in part by the National Programme of Research (Information Society Project 1ET100300419).

References

1. Bertino, E., Guerrini, G., Mesiti, M.: A Matching Algorithm for Measuring the Structural Similarity between an XML Document and a DTD and its Applications. *Inf. Syst.* 29(1), 23–46 (2004)
2. Biron, P.V., Malhotra, A.: XML Schema Part 2: Datatypes, 2nd edn. W3C (2004)
3. Bray, T., Paoli, J., Sperberg-McQueen, C.M., Maler, E., Yergeau, F.: Extensible Markup Language (XML) 1.0, 4th edn. W3C (2006)
4. Do, H.H., Rahm, E.: COMA – A System for Flexible Combination of Schema Matching Approaches. In: VLDB 2002: Proc. of the 28th Int. Conf. on Very Large Data Bases, Hong Kong, China, pp. 610–621. Morgan Kaufmann, San Francisco (2002)
5. Lee, M.L., Yang, L.H., Hsu, W., Yang, X.: XClust: Clustering XML Schemas for Effective Integration. In: CIKM 2002: Proc. of the 11th Int. Conf. on Information and Knowledge Management, New York, NY, USA, pp. 292–299. ACM, New York (2002)
6. Levenshtein, V.I.: Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady* 10, 707 (1966)
7. Mlynkova, I., Toman, K., Pokorny, J.: Statistical Analysis of Real XML Data Collections. In: COMAD 2006: Proc. of the 13th Int. Conf. on Management of Data, New Delhi, India, pp. 20–31. Tata McGraw-Hill Publishing, New York (2006)
8. Ng, P.K.L., Ng, V.T.Y.: Structural Similarity between XML Documents and DTDs. In: ICCS 2003: Proc. of the Int. Conf. on Computational Science, pp. 412–421. Springer, Heidelberg (2003)
9. Nierman, A., Jagadish, H.V.: Evaluating Structural Similarity in XML Documents. In: WebDB 2002: Proc. of the 5th Int. Workshop on the Web and Databases, Madison, Wisconsin, USA, pp. 61–66 (2002)
10. Shanmugasundaram, J., Tufte, K., Zhang, C., He, G., DeWitt, D.J., Naughton, J.F.: Relational Databases for Querying XML Documents: Limitations and Opportunities. In: VLDB 1999: Proc. of 25th Int. Conf. on Very Large Data Bases, pp. 302–314. Morgan Kaufmann, San Francisco (1999)
11. Thompson, H.S., Beech, D., Maloney, M., Mendelsohn, N.: XML Schema Part 1: Structures, 2nd edn. W3C (2004)
12. Zhang, Z., Li, R., Cao, S., Zhu, Y.: Similarity Metric for XML Documents. In: FGWM 2003: Proc. of Workshop on Knowledge and Experience Management, Karlsruhe, Germany (2003)

Design, Implementation and Evaluation of a New Semantic Similarity Metric Combining Features and Intrinsic Information Content

Giuseppe Pirr ¹ and Nuno Seco²

¹ D.E.I.S, University of Calabria, Italy

`gpirro@deis.unical.it`

² DEI-CISUC, University of Coimbra, Portugal

`nseco@dei.uc.pt`

Abstract. In many research fields such as Psychology, Linguistics, Cognitive Science, Biomedicine, and Artificial Intelligence, computing semantic similarity between words is an important issue. In this paper we present a new semantic similarity metric that exploits some notions of the early work done using a feature based theory of similarity, and translates it into the information theoretic domain which leverages the notion of Information Content (IC). In particular, the proposed metric exploits the notion of *intrinsic* IC which quantifies IC values by scrutinizing how concepts are arranged in an ontological structure. In order to evaluate this metric, we conducted an on line experiment asking the community of researchers to rank a list of 65 word pairs. The experiment’s web setup allowed to collect 101 similarity ratings, and to differentiate native and non-native English speakers. Such a large and diverse dataset enables to confidently evaluate similarity metrics by correlating them with human assessments. Experimental evaluations using WordNet indicate that our metric, coupled with the notion of intrinsic IC, yields results above the state of the art. Moreover, the intrinsic IC formulation also improves the accuracy of other IC based metrics. We implemented our metric and several others in the Java WordNet Similarity Library.

Keywords: Semantic Similarity, Intrinsic Information Content, Feature Based Similarity, Java WordNet Similarity Library.

1 Introduction

Assessing semantic similarity between words is a central issue in many research fields such as Psychology, Linguistics, Cognitive Science, Biomedicine, and Artificial Intelligence. Semantic similarity can be exploited to improve accuracy of current Information Retrieval techniques (e.g., [6, 9]), to discover mapping between ontology entities [17], to validate or repair mappings [13], to perform word-sense disambiguation [19]. Recently, Li and colleagues in [11] have proposed a methodology to compute similarity between short sentences through semantic similarity. Semantic similarity has found its way also in the context of Peer to

Peer networks (e.g., [4]). In particular, assuming a shared taxonomy among the peers to which they can annotate their content, semantic similarity is exploited to infer similarity among peers by computing similarity among their representative concepts in the shared taxonomy, this way, the more two peers are similar the more efficient is to route messages toward them. In [21] are discussed several applications of similarity in Artificial Intelligence. In the biomedical domain there exist some applications to compute semantic similarity between concepts of ontologies such as MeSH or Gene (e.g., [16]). However, despite the numerous practical applications of semantic similarity, it is important pointing out its theoretical underpinning in Cognitive Science and Psychology where several investigations (e.g., [24]) and theories (e.g., [14, 29]) have been proposed.

As a matter of fact, semantic similarity is relevant in many research areas and therefore, designing accurate methods is mandatory for improving the "performance" of the bulk of applications relying on it. Basically, similarity or distance methods (e.g., [1]) aim at assessing a score between a pair of words by exploiting some information sources. These can be search engines (e.g., [2, 3]) or a well-defined semantic network such as WordNet [15] or MeSH (<http://www.nlm.nih.gov/mesh/>). To date, several approaches to assess similarity have been proposed ([7] provide an exhaustive list of references) which can be classified on the basis of the source of information they exploit. There are ontology-based approaches (e.g., [18]), information-theoretic approaches which exploit the notion of Information Content (IC) (e.g., [8, 12, 20]), hybrid approaches (e.g., [10, 22, 25]) just to cite a few of them.

In this paper, our purpose is to systematically design, evaluate and implement a new similarity metric. This metric has not to be derived empirically but has to be justified by a theoretical underpinning (i.e., a theory of semantic similarity). The contributions of this paper can be summarized as follows:

1. We propose a new similarity metric which exploits some of the early work on the feature based theory of semantic similarity proposed by Tversky [29], and projects it into the information theoretic domain which has attained impressive results. As our results will show, this metric coupled with the notion of *intrinsic* Information Content [27] outperforms current implementations on different datasets.
2. We performed a similarity experiment to collect human similarity ratings to evaluate similarity metrics. In particular, we used the 65 word pairs dataset originally proposed by Rubenstein and Goodenough (R&G) [23]. Note that even if similar experiments have been carried out during the years (e.g., [15, 20]), none of these considered the whole R&G set of 65 word pairs. As we will discuss, the number of participants in our experiment is significantly higher than that of other experiments and hence we hope to provide a more robust and reliable evaluation tool. Moreover, by correlating our ratings with those collected by R&G we investigate a possible upper-bound for results that we can expect from computational methods.
3. We evaluated the proposed metric on different datasets and analyzed its structure to identify commonalities and differences w.r.t the state of the art.

Moreover, we evaluated the impact of the *intrinsic* IC formulation on our and other IC based metrics.

4. We implemented our metric and several others in the Java WordNet Similarity Library (JWSL). JWSL, to the best of our knowledge, is the only tool written in Java devoted to compute similarity in WordNet. JWSL, by exploiting an ad-hoc index, allows to speed up the similarity computation without requiring the WordNet software to be installed.

The remainder of this paper is organized as follows. Section 2 provides some background information regarding WordNet and popular similarity metrics. Section 3 presents our similarity metric and the intuitions that motivated its origin. In Section 4 we explain how the new dataset was created and compare it with previously used datasets. Section 5 uses the new dataset to analyze and compare several similarity metrics, by correlating them to the human assessments. Moreover, here we evaluate the impact of the *intrinsic* IC formulation. In this section we also propose a new upper bound on the degree of correlation that may be obtained using computational approaches and briefly introduce the JWSL. Finally, Section 6 concludes the paper.

2 WordNet and Similarity Metrics

WordNet is a light-weight lexical ontology where concepts are connected to each other by well-defined types of relations. It is intended to model the human lexicon, and took psycholinguistic findings into account during its design [14]. We call it a light-weight ontology because it is heavily grounded on its taxonomic structure that employs the IS-A inheritance relation and lexical ontology because it contains both linguistic and ontological information. In WordNet concepts are referred to by different words; for example if we want to refer to the concept expressed by "someone deranged and possibly dangerous" we could use any of the words contained in the set $\{crazy, loony, looney, weirdo\}$. So in a given context we can say that the words in the above set are synonyms. Hence, a synset (Synonym Set), the term adopted by the founders of WordNet, represents the underlying lexical concept. Each concept contains a gloss that expresses its semantics by means of a textual description and a list of words that can be used to refer to it. There are several types of relations used to connect the different types of synsets. Some of these define inheritance (IS-A) relations (hypernymy/hyponymy), other part-of relations (holonymy/meronymy). The antonymy relation is used to state that a noun is the opposite of another. The relations *instance of* and *has instance* have been introduced in WordNet 3.0. However, note that the hypernymy/hyponymy relations constitute 65% of the relations connecting noun synsets. The prototypical definition of a noun consists of its immediate superordinate followed by a relative clause that describes how this instance differs from all other instances. For example, *Fortified Wine* is distinguished from *Wine* because "... alcohol (usually grape brandy)" has been added just as the gloss mentions. This type of model is usually said to employ a

differential theory of meaning, where each subordinate differentiates itself from its super ordinate.

2.1 Similarity Metrics on WordNet

Similarity metrics between concepts can be divided into four general, and not necessarily disjoint, categories [30]: Ontology Based Approaches, Corpus Based Approaches, Information Theoretic and Dictionary based approaches. In this paper we will focus on popular metrics that may use WordNet as their main knowledge resource and that belong to either the information theoretic or ontology based category. A complete survey of existing metrics is out of the scope of this paper (for a list of related references refer to [7]).

Information theoretic approaches usually employ the notion of Information Content (IC), which can be considered a measure that quantifies the amount of information a concept expresses. Previous information theoretic approaches [8, 12, 20] obtain the needed IC values by statistically analyzing corpora. They associate probabilities to each concept in the taxonomy based on word occurrences in a given corpus. These probabilities are cumulative as we go up the taxonomy from specific concepts to more abstract ones. This means that every occurrence of a noun in the corpus is also counted as an occurrence of each taxonomic class containing it. The IC value is obtained by considering negative the log likelihood:

$$IC(c) = -\log p(c) . \quad (1)$$

where c is a concept in WordNet and $p(c)$ is the probability of encountering c in a given corpus. It should be noted that this method ensures that IC is monotonically decreasing as we move from the leaves of the taxonomy to its roots. Resnik [20] was the first to consider the use of this formula, which stems from the work of Shannon [28], for the purpose of semantic similarity judgments. The basic intuition behind the use of the negative likelihood is that the more probable a concept is of appearing then the less information it conveys, in other words, infrequent words are more informative than frequent ones. Knowing the IC values for each concept we may then calculate the similarity between two given concepts. According to Resnik, similarity depends on the amount of information two concepts have in common, this shared information is given by the Most Specific Common Abstraction (*msca*) that subsumes both concepts. In order to find a quantitative value of shared information we must first discover the *msca*, if one does not exist then the two concepts are maximally dissimilar, otherwise the shared information is equal to the IC value of their *msca*. Resnik's formula is modeled as follows:

$$sim_{res}(c_1, c_2) = \max_{c \in S(c_1, c_2)} IC(c) . \quad (2)$$

where $S(c_1, c_2)$ is the set of concepts that subsume c_1 and c_2 . This formulation of similarity is actually very similar to the one proposed by Tversky [29] but using a set theoretic framework. Following Resnik's first work two other distinguishable metrics were postulated, that of Jiang and Conrath [8] and the work of Lin

[12]. Both metrics used the notion of IC and calculated it in the same manner proposed by Resnik. Both Lin's and Jiang's formulations correct a problem with Resnik's similarity metric; if one were to calculate $sim_{res}(c_1, c_1)$ one would not obtain the maximal similarity value of 1, but instead the value given by $IC(c_1)$. Moreover, with this approach any two pairs of concepts having the same $msca$ have exactly the same semantic similarity. For example, $sim_{res}(horse, plant) = sim_{res}(animal, plant)$ because in each case the $msca$ is *Living Thing*. According to Lin "The similarity between c_1 and c_2 is measured by the ratio between the amount of information needed to state the commonality of c_1 and c_2 and the information needed to fully describe what c_1 and c_2 are". Formally this formula is given in the following equation:

$$sim_{Lin}(c_1, c_2) = \frac{2sim_{res}(c_1, c_2)}{IC(c_1) + IC(c_2)}. \quad (3)$$

The Jiang et al. metric is a semantic distance measure, but as shown in [26] it can be transformed to a similarity metric yielding:

$$sim_{J\&C}(c_1, c_2) = 1 - \frac{IC(c_1) + IC(c_2) - 2sim_{res}(c_1, c_2)}{2}. \quad (4)$$

Regarding the ontology based approaches we review two noteworthy approaches, one of Rada et al. [18] and the one of Hirst et al. [5]. The first is also referred to as a depth based approach and the second as a path based approach. The Rada metric is similar to the Resnik metric in that it also computes the $msca$ between two concepts, but instead of considering the IC as the value of similarity, it considers the number of links that were needed to attain the $msca$. Obviously, the less number of links separating the concepts the more similar they are. The approach of Hirst et al.¹ is similar to the previous but instead they use all types of relations in WordNet coupled with rules that restrict the way concepts are transversed. Nonetheless, the intuition is the same; the number of links separating two concepts is inversely proportional to the degree of similarity. Finally, an approach combining structural semantic information in a nonlinear model is that proposed by Li et al. [10]. The authors empirically defined a similarity measure that uses shortest path length, depth and local density in a taxonomy. The next equation reflects their metric:

$$sim_{Li}(c_1, c_2) = \begin{cases} e^{-\alpha l \frac{e^{\beta h} - e^{-\beta h}}{e^{\beta h} + e^{-\beta h}}} & \text{if } c_1 \neq c_2 \\ 1 & \text{if } c_1 = c_2 \end{cases} \quad (5)$$

In equation 5, l is the length of the shortest path between c_1 and c_2 in the graph spanned by the IS-A relation, h is the level in the tree of the $msca$ from c_1 and c_2 . The parameters α and β . represent the contribution of the shortest path length l and depth h . The optimal values for these parameters, determined experimentally, are: $\alpha = 0.2$ and $\beta = 0.6$ as discussed in [10].

¹ This approach actually measures relatedness, but since similarity is a special case of relatedness (see [26]) we consider it in our study.

3 The Pirró and Seco Similarity Metric

In this section we introduce our new similarity metric which is conceptually similar to the previous ones, but is founded on the feature-based theory of similarity posed by Tversky [29]. We argue that this theory fits nicely into the information theoretic domain, and obtains results that improve the current state of the art. Moreover, to avoid the problem of corpus-dependence of IC based metrics we exploit the method discussed in Section 3.1. The argumentation presented here follows from the work conducted in [26].

Tversky presented an abstract model of similarity that takes into account the features that are common to two concepts and also the differentiating features specific to each. More specifically, the similarity of a concept c_1 to a concept c_2 is a function of the features common to c_1 and c_2 , those in c_1 but not in c_2 and those in c_2 but not in c_1 . Admitting a function $\psi(c)$ that yields the set of features relevant to c , he proposed the following similarity function:

$$sim_{tvr}(c_1, c_2) = \alpha F(\Psi(c_1) \cap \Psi(c_2)) - \beta F(\Psi(c_1)/\Psi(c_2)) - \gamma F(\Psi(c_2)/\Psi(c_1)) . \quad (6)$$

where F is some function that reflects the salience of a set of features, and α , β and γ are parameters that provide for differences in focus on the different components. According to Tversky, similarity is not symmetric, that is, $sim_{tvr}(c_1, c_2) \neq sim_{tvr}(c_2, c_1)$ because subjects tend to focus more on one object than on the other depending on the way the comparison experiment has been laid out. Obviously, the above formulation is not framed in information theoretic terms. Nonetheless, we argue that a parallel may be established that will lead to a new similarity function. Resnik considered the $msca$ of two concepts c_1 and c_2 as reflecting the information these concepts share, which is exactly what is intended with the intersection of features from c_1 and c_2 (i.e., $\Psi(c_1) \cap \Psi(c_2)$). Now, remembering that function F quantifies the salience of a set of features, then we postulate that we may find that quantification in the form of information content. The above reasoning will lead us to the following analogy represented in the following equation:

$$\begin{aligned} sim_{res}(c_1, c_2) &= IC(msca(c_1, c_2)) \approx F(\Psi(c_1) \cap \Psi(c_2)) \\ &= 1F(\Psi(c_1) \cap \Psi(c_2)) - 0F(\Psi(c_1)/\Psi(c_2)) - 0F(\Psi(c_2)/\Psi(c_1)) . \end{aligned} \quad (7)$$

Since the $msca$ is the only parameter taken into account we may say that his formulation is a special case of equation 6 where $\beta=\gamma=0$. The above discussion lends itself to the proposal of an information theoretic counterpart of equation 7 that can be formalized as:

$$\begin{aligned} sim_{tvr}'(c_1, c_2) &= IC(msca(c_1, c_2)) - (IC(c_1) - IC(msca(c_1, c_2))) - (IC(c_2) - IC(msca(c_1, c_2))) \\ &= 3IC(msca(c_1, c_2)) - IC(c_1) - IC(c_2) \end{aligned} \quad (8)$$

A careful analysis of equation 8 shows that this metric suffers from the same problem as Resnik's metric. When computing the similarity between identical

concepts the output yields the information content value of their *msca* and not the value corresponding to maximum similarity. In order to overcome this limitation we assign the value of 1 if the two concepts are the same, hence yielding the similarity metric that can be formalized as follows:

$$sim_{P\&S}(c_1, c_2) = \begin{cases} sim_{tvr'} & \text{if } c_1 \neq c_2 \\ 1 & \text{if } c_1 = c_2 \end{cases} \quad (9)$$

Note that in equation 9 we use $sim_{tvr'}$ which is the information theoretic counterpart of Tversky's set theoretic formulation. This new formulation will be dubbed as the $sim_{P\&S}$ metric in the rest of the paper. At this point a possible drawback related to IC-metrics remains to be solved: how to obtain IC values in a more direct and corpus-independent way ? We address this problem in the next section.

3.1 Intrinsic Information Content

As pointed out before, similarity metrics grounded on IC obtain IC values for concepts by statistically analyzing large corpora and associating a probability to each concept in the taxonomy based on its occurrences within the considered corpora. From a practical point of view, this approach has two main drawbacks: (i) it is time consuming and (ii) it heavily depends on the type of corpora considered. Research toward mitigating these drawbacks has been proposed by Seco et al. [27]. Here, values of IC of concepts rest on the assumption that the taxonomic structure of WordNet is organized in a "meaningful and structured way", where concepts with many hyponyms convey less information than concepts that are leaves, that is, the more hyponyms a concept has the less information it expresses. Hence, the IC for a concept c is defined as:

$$IC(c) = 1 - \frac{\log(hypo(c) + 1)}{\log(max_{wn})}. \quad (10)$$

where the function *hypo* returns the number of hyponyms of a given concept c . Note that concepts that represent leaves in the taxonomy will have an IC of one, since they do not have hyponyms. The value of 1 states that a concept is maximally expressed and cannot be further differentiated. Moreover max_{wn} is a constant that indicates the total number of concepts in the WordNet noun taxonomy. This definition of IC will be exploited in the P&S similarity metric thus enabling to obtain IC values in a corpus independent way. In Section 5.4 we show how the intrinsic IC improves the accuracy of all IC based metrics.

4 The Pirró and Seco Similarity Experiment

In order to assess the quality of a computational method to determine similarity between words, that is, its accuracy, a natural way is to compare its behavior w.r.t human judgments. The more a method approaches human similarity

judgment the more accurate it is. In evaluating the different methodologies two datasets are commonly used, those of Rubenstein and Goodenough (R&G in the following) and Miller and Charles (M&C in the following). R&G [23] in 1965 performed a similarity experiment by providing 51 human subjects, all native English speakers, with 65 word pairs and asking them to assess similarity between word pairs on a scale from 0 ("semantically unrelated") to 4 ("highly synonymous"). M&C [15], 25 years later, repeated the R&G experiment by only considering a subset of 30 word pairs from the original 65, and involving 38 undergraduate students (all native English speakers). In this case humans were also asked to rate similarity between pairs of words on a scale from 0 to 4. Although the M&C experiment was carried out 25 years later, the correlation between the two sets of human ratings is 0.97 which is a very remarkable value considering the diachronic nature of languages. Resnik [20] on his turn in 1995 replicated the M&C experiment by involving 10 computer science graduate students and post-doc (all native English speakers) obtaining a correlation of 0.96, also in this case a high value.

The results of these experiments point out that human knowledge about semantic similarity between words is remarkably stable over years (25 and 30 years later the R&G, for the M&C and Resnik experiment respectively). Moreover, they also point out how the usage of human ratings could be a reliable reference to compare computational methods with. However, researchers tend to focus on the results of the M&C experiment to evaluate similarity metrics and, to the best of our knowledge, no systematic replicas of the entire R&G experiment have been performed. Therefore, we argue that it would be valuable to perform a "new" similarity experiment in order to obtain a baseline for comparison with the entire R&G dataset.

4.1 Experiment Setup

We replicate the R&G experiment (naming it Pirró and Seco, P&S in the following) but one step closer to the 21st century, the century of the Internet and global information exchange. In particular, we performed the experiment on the Internet by advertising it in some of the most famous computer science mailing lists (e.g., DBWORLD, CORPORA, LINGUIST) with the aim to involve as many people as possible. Each participant, after a registration process on the similarity experiment website² could take part in the experiment. In the web site were provided all the instructions to correctly perform the experiment. The similarity scores along with the emails provided by the participants have been stored in a relational database for subsequent analysis. As one can imagine, and as our results confirmed, the participants were mostly graduate students, researches and professors. Note that we also opened the experiment to non native English speakers. As said above, in the era of globalization more and more people speak English thus participating in the creation and spreading of new forms of interpreting terms. Furthermore, semantic relations among words are affected

² All the details along with extensive evaluations are downloadable at the JWSL website <http://grid.deis.unical.it/similarity>.

by language evolution that, on its turn, is affected by the presence of a larger number of speakers of a particular language. Our objective is to investigate if and how the presence of non native speakers affects similarity judgments. Among the participants, about 70% are native American English speakers, 30% British English speakers while non native speakers are for the most part European. Table 1 provides some information about the experiment. As can be noted, even if we collected 121 similarity ratings we discarded some of them for the reasons explained in the next section.

Table 1. Information about the *P&S* experiment

Start of the experiment	07/15/2007
Result considered until	04/15/2008
Overall number of similarity judgments collected	121
Number of similarity judgments considered in the gold standard	101
Number of similarity judgments provided by native English speakers	76
Number of similarity judgments provided by non native English speakers	25

4.2 Elaborating the Collected Similarity Ratings

In order to design a systematic experiment and consider its results reliable, an a posteriori analysis of its results is required. In our case, this analysis is particularly important for ratings provided by non native speakers since the group of non native speakers could be quite large and heterogeneous, ranging from near-native speakers over very fluent speakers to speakers with only rudimentary knowledge of English. In order to check the quality of the ratings provided by the participants, we calculate, for each participant, a rating coefficient (i.e., *C*) defined as follows:

$$C = \sum_{i=1}^{65} |C_i - avg_i| . \tag{11}$$

In particular, for each word pairs the distance between the score provided by the participant and the average score provided by the others is measured. The distance values for all the 65 pairs are then summed up. Once computing all the coefficients *C* we could discard the participants that present values of *C* differing too much from the average. Fig. 1 represents the *C* values for all the 121 participants. As can be noted, most of the *C* coefficients lie between 30 and 40. However, ratings provided by some participants (and then *C* coefficients) clearly differers from the average. The ratings provided by these participants have been discarded. In particular, by observing the results provided in Fig. 1 it can be noted that the anomalous ratings were for the most part given by non native speakers (about 90%). Table 2 provides an overall view of the different similarity experiments. Note that even if we collected 121 similarity ratings, we only considered 101 as reliable. We collected a larger number of similarity ratings than R&G, M&C and Resnik experiments and about 30% of participants in our

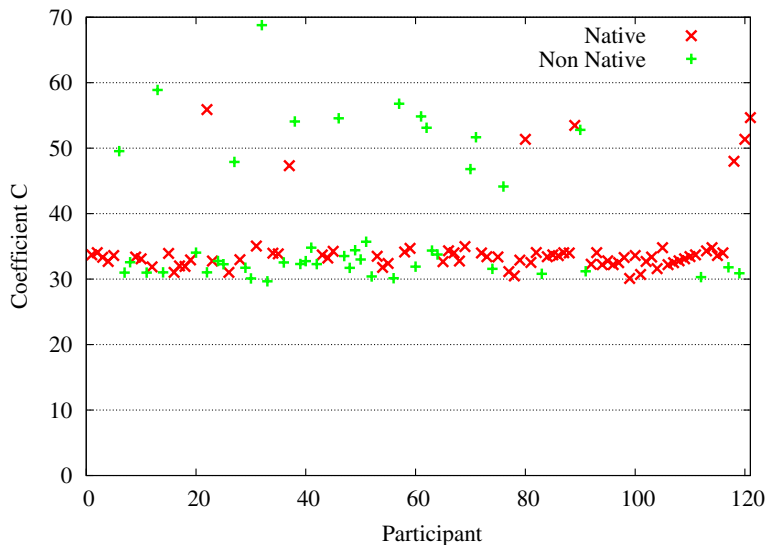


Fig. 1. Values of the coefficient C for the participants to the P&S experiment

Table 2. Overall view of the different similarity experiments

Experiment	Year	Number of pairs	Number of participants
R&G	1965	65	51 (all native speakers)
M&C	1991	30	38 (all native speakers)
Resnik	1995	30	10 (all native speakers)
P&S	2008	65	101(76 native speakers and 25 non native)

experiment are (reliable) non native English speakers. Moreover, differently from M&C and Resnik we performed the experiment by considering the whole initial R&G dataset.

4.3 Comparison among Experiments

We split the collected similarity judgments in two sets. The first set ($S_{M\&C}$ in the following) contains the judgments for the 28 word pairs in the M&C experiment. We consider only 28 pairs of the initial 30 used by M&C since due to a word missing in WordNet it is only possible to obtain computational rating for 28 word pairs. The second set ($S_{R\&G}$ in the following) contains the 65 word pairs in the R&G dataset. In particular, this latter dataset is used to define a possible upper-bound for computational methods to assess semantic similarity. Note that the word pairs in M&C, extracted from the original R&G dataset, are chosen in a way that they range from "highly synonymous" (e.g., *car-automobile*) to "semantically unrelated" (i.e., *cord-smile*). In order to have a more accurate view of the values of the ratings provided by the different experiments and investigate

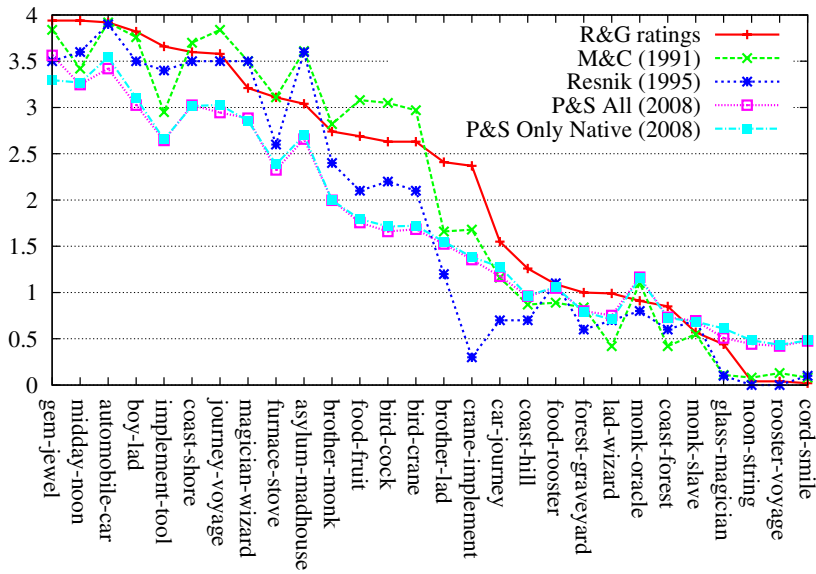


Fig. 2. Human ratings collected by the different experiments

the regularity of the decreasing similarity trend demanded by R&G, we considered the similarity ratings of the four experiments as virtually connected, thus obtaining the representation in Fig. 2. As can be observed in Fig. 2, the decreasing trend of the R&G judgments is quite regular whereas that of M&C is quite irregular due to some pairs of concepts that are judged more similar/dissimilar by participants to the M&C experiment.

For instance, the pairs *implement-tool*, *asylum-madhouse*, and *brother-lad* are evident alterations of the decreasing trend of the R&G rating curve. The Resnik rating curve seems to be the most irregular, in particular, the pairs *furnace-stove*, *asylum-madhouse* and *crane-implement* are evident singular points. The P&S rating curves considering native speakers and all speakers also present some singular points (e.g., *implement-tool* and *asylum-madhouse*).

As a final comparison, in Tables 3 and 4 the Pearson correlation coefficient among the different experiments are reported. For sake of space we do not report the scores obtained for the whole $S_{R\&G}$ dataset. As can be noted, the correlation values obtained by our experiment are high. In particular, the correlation values considering only native ($P\&S_{nat}$) and all the participants ($P\&S_{full}$) are

Table 3. Correlation on $S_{M\&C}$

	$P\&S_{full}$	$P\&S_{nat}$
R&G(1965)	0.961	0.964
M&C(1991)	0.951	0.955
Resnik(1995)	0.970	0.972

Table 4. Correlation on $S_{R\&G}$

	$P\&S_{full}$	$P\&S_{nat}$
R&G(1965)	0.972	0.971

almost the same. Therefore, we argue that results of the P&S experiment can be adopted as a reliable basis for comparing similarity metrics with. Moreover, since the number of judgments collected is larger than that collected by previous experiments and the presence of non native speakers does not affect the similarity judgments we hope to provide a more reliable and robust evaluation tool.

4.4 Inter-annotator Agreement and Correlation between Groups of Participants

To complete the elaboration of the collected results, we computed two additional parameters: (i) the inter-subject agreement also known as kappa-statistic and (ii) the correlation between ratings provided by native and non native speakers. Considering the $S_{M\&C}$ the kappa-statistic obtained is 0.82 which symbolizes the agreement among participants in rating the word pairs. On the same set, the correlation between the average judgments of native and non native speakers is 0.97, which is a very high value. Considering the $S_{R\&G}$ the kappa-statistic obtained is 0.81 while the correlation between the average judgments of non native and native speakers in this case is 0.98. Finally, note that the experiments involved a different number of participants (51 for R&G, 30 for M&C, 10 for Resnik and 101 for P&S).

5 Evaluation and Implementation of the P&S Metric

In this section, to substantiate the investigation that led to the definition of the P&S metric we evaluate and compare it w.r.t the state of the art. In performing this evaluation we consider the results of the P&S experiment on the $S_{M\&C}$ and $S_{R\&G}$ datasets. All the evaluations have been performed using WordNet 3.0.

In our evaluation, instead of reporting for each metric the results obtained in a tabular form we represent them as shown in Fig. 3. This way, we can further discuss and characterize in more details the peculiarities, analogies and differences of the different metrics. However, to have an overall view of the outcome of our evaluation and compare the different metrics we calculated, for each metric, the Pearson correlation coefficient between its results and human judgments (see Tables 5 and 6).

Table 5. Correlation on $S_{M\&C}$

	P&S (2008)	
	$P\&S_{full}$	$P\&S_{nat}$
Length	0.611	0.602
Depth	0.841	0.839
Resnik	0.854	0.842
Lin	0.875	0.871
J&C	0.884	0.883
Li	0.911	0.904
P&S	0.912	0.908

Table 6. Correlation on $S_{R\&G}$

	P&S (2008)	
	$P\&S_{full}$	$P\&S_{nat}$
Length	0.587	0.578
Depth	0.807	0.805
Resnik	0.877	0.869
Lin	0.892	0.888
J&C	0.878	0.877
Li	0.900	0.897
P&S	0.908	0.905

The similarity values for the Length and Depth metrics are obtained by considering the shortest path between the two words to be compared and the depth of their subsumer respectively. For the metrics based on IC and the P&S metric the values of IC are obtained by the method described in Section 3.1. Moreover, for the Li metric the similarity results are those reported in [10].

5.1 Discussion

From the values reported in Tables 5 and 6 emerges that edge counting approaches reach the lowest correlation with human ratings. That is mainly due to the fact that path lengths and depth approaches are appropriate only when the values of path and depth have a "consistent interpretation". This is not the case of WordNet, since concepts higher in the hierarchy are more general than those lower in the hierarchy. Therefore, a path of length one between two general concepts can suggest a larger semantic leap whereas one between two specific concepts may not (e.g., *Entity—Psychological Feature* and *Canine—Dog*). Resnik's metric, which only considers the IC of the *msca* in assessing semantic similarity, obtained the lowest value of correlation among the IC metrics using $S_{M\&C}$. The Lin and J&C metrics, which also consider the IC of the two words to be compared, obtained higher values of correlation on the same dataset. Note that the Li metric which combines the depth of the *msca* and the length of the path between two concepts to be compared obtained a remarkable value of correlation even if it relies on two coefficients (i.e., α and β) whose optimal values have been experimentally determined as described in [10]. The J&C metric combines an IC formulation of semantic similarity with edge counting. The P&S metric obtained the higher value of correlation on the $S_{M\&C}$ dataset.

On the second dataset, that is $S_{R\&G}$, the correlation values obtained by the different metrics slightly change. Even in this case, the Length metric obtains the poorest correlation. Resnik's metric obtained a correlation comparable to that obtained by the J&C metric. The Lin metric obtained better results. The Li metric, evaluated by considering the optimal parameter determined by authors in [10] obtained a better correlation. However, the P&S metric remains the most correlated w.r.t human judgments also in this dataset. Correlation results reported in Tables 5 and 6 show that the presence of non native speakers barely affects the values of correlation of the different metrics.

5.2 Commonalities and Differences among Metrics

In order to have a deeper insight into the structure of the different metrics, we represent their results as shown in Fig. 3. Here, it can be recognized the different nature of edge-counting (i.e., Length, Depth), IC-based and Li's multi-source metrics. In particular, edge-counting metrics give discrete results as output (i.e., integer values). For the Length metric, a lower value of path length corresponds to a higher similarity value between words. For instance the first three pairs (i.e., *gem-jewel*, *midday-noon* and *automobile-car*) have a length equal to zero which is due to the fact that these word pairs belong to the same WordNet synset respectively. On the other side, word pairs as *noon-string* and *rooster-voyage*

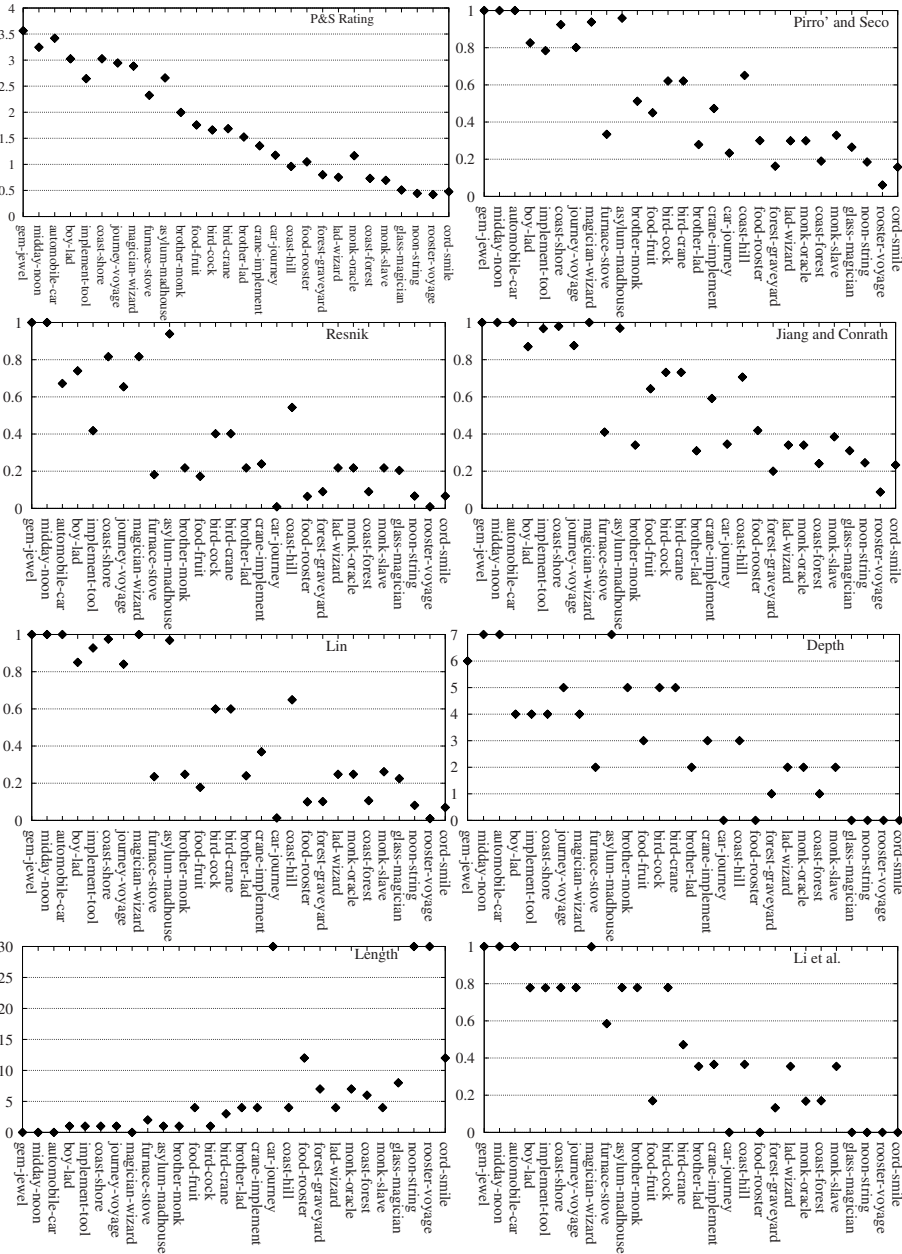


Fig. 3. Results and ratings considering the M&C dataset. Vertical axis represents the similarity value while horizontal axis word pairs

have a relatively high distance which means that the words in the two pairs are not similar. A potential anomaly could be represented by the pair *car-journey*

which gets a length of 30, the maximum value. The two words, even if generally related as a car can be the means to do a journey, are not considered similar. That is because similarity is a special case of relatedness and only considers the relations of hypernymy/hyponymy defined in WordNet which is exactly what the Length metric does. For the Depth metric, a number of "similarity levels" can be recognized (in Fig. 3 for instance, it can be noted that there are 3 ratings in the level 7, 5 in the level 2 and 6 in the level 0). This metric, differently from that of Resnik takes into account the depth of the *msca* thus allowing more specific concepts to be generally judged more similar than more abstract one. Note that this metric obtained a correlation about 30% better than the Length metric.

A more interesting discussion can be done for the IC based metrics. In particular, the Resnik and Lin metrics present two similar regions, one in the center identified by the pairs *bird-cock* and *bird-crane* (translated by 0.2) and the other comprising all the pairs from *car-journey* to *cord-smile*. Note that when the two words to be compared are leaves, according to the intrinsic IC formulation described in equation 10, they have IC equals to 1 and therefore equation 5 turns into equation 2. A similar condition holds for the transformed Jiang and Conrath metric in equation 4. The P&S metric when c_1 and c_2 are leaves gives as result $sim_{P\&S}(c_1, c_2) = 3IC(msca) - 2$. In this case, if the *msca* is high in the taxonomy (it receives a low IC) the metric returns a lower similarity value than when the *msca* is low. A similar area can be recognized between the J&C and P&S metric (i.e., from the pairs *forest-graveyard* to *cord-smile*). In this area generally, the J&C obtains higher similarity scores. However, according to the original intent of R&G to chose word pairs from very similar to less similar, the P&S metric seems to better respect this trend in this case. Finally, the Li metric has a very similar region (comprising the pairs from *car-journey* to *cord-smile*) to the Depth metric. The word pairs in this region are rated equally due to the fact that the Li metric exploits the value of Depth and when this is zero, according to equation 5 the similarity value returned by the Li metric is zero.

In summary, the results of these experiments demonstrate that our intuition to consider the original formulation of IC provided by Resnik, to some extent, a special case of the formulation given by Tversky is consistent. Moreover, the metric (i.e., Li) that obtained results comparable to the P&S metric has been empirically designed and relies on two parameters to be adjusted.

5.3 Some Considerations on the P&S Metric

By scrutinizing equation 9 that defines the P&S metric a couple of observations arise. The first is related to the intrinsic IC formulation: if the number of hyponyms of a concept changes, the similarity between a pair of concepts will change as stated in equation 10. Note that this formulation of IC takes into account the strengthens of links: links (hypernym/hyponym) higher in the ontology are not as strong as those closer to the bottom. Moreover, links that leaf nodes have with their immediate hypernym are the strongest (have the smallest semantic leap between them). Therefore, if we add a hyponym to a concept we are weakening the relation between the concept and its immediate hypernym hence

weakening the relation between the pair being compared. Here, the underlying assumption is that the ontology is organized in a "meaningful and principled way", and if there is need to reorganize the ontology then we should accept that similarity values change. The second consideration is related to the branch of equation (9) $sim_{P\&S}(c_1, c_1) = 1$. We added this branch to solve the problem of the Resnik metric i.e., $sim_{P\&S}(c_1, c_1) \neq 1$. Moreover, our evaluation show that such a function yields results that correlate better with human judgments.

5.4 Impact of Intrinsic Information Content

In this section we evaluate the impact of the intrinsic IC formulation on the IC metrics. Fig. 4 shows the results of this evaluation. For sake of space we do not report the scores obtained by considering the two IC formulations. As can be noted, the correlation is improved for each metric. In particular, a notable improvement is reached by the J&C (about 40%) and P&S metrics (about 15%). In the light of these results we can conclude that the intrinsic IC formulation is an effective and convenient way to compute IC values.

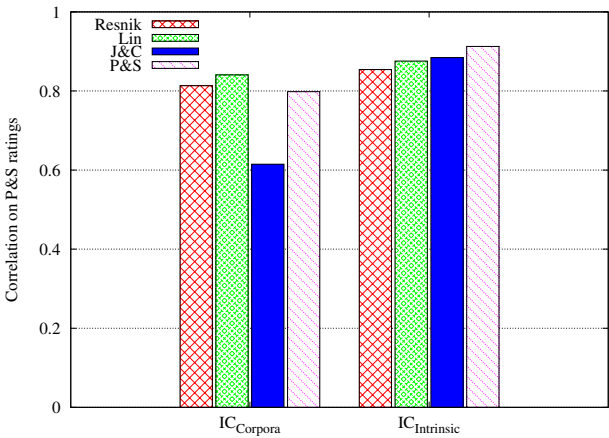


Fig. 4. Impact of the Intrinsic IC formulation

5.5 New Challenges for Researchers

The results obtained by some metrics in our experiments are very close to human judgments. At this point a question arises: how much we can expect from a computational method for assessing semantic similarity? Resnik in [20] took into account the correlation between experiments in order to obtain a possible upper bound. Resnik obtained a value of correlation w.r.t M&C experiment of 0.9583 while the inter-annotator agreement obtained was 0.9015. This latter result has been considered for many years as the theoretical upper bound. However, we agree with what was observed in [10] and propose to consider as upper bound not the inter-annotator agreement but the correlation between the rating of the

different experiments. This is because semantic similarity should be considered as a collective property of groups of peoples (i.e., all the participant to the experiment) rather than considering them individually as done by Resnik with the inter-subject agreement. Moreover, since we replicated the R&G experiment on all the 65 word pairs dataset we can correlate our results with those obtained by R&G. Hence, we propose to set as new hypothetical upper bound the value of correlation between the R&G and P&S ratings, that is, 0.972. This latter consideration provides new challenges for researches. In fact, even if the metric we presented obtains a correlation value of 0.908 using this dataset, this value is far from the new hypothetical upper bound.

5.6 The Java WordNet Similarity Library (JWSL)

The P&S metric has been included in the Java WordNet Similarity Library (JWSL). The main features of JWSL (available at <http://grid.deis.unical.it/similarity>) can be summarized as follows: (i) it exploits a Lucene (<http://lucene.apache.org>) index including the whole WordNet structure and does not require the WordNet software to be installed; (ii) it is written in Java. To the best of our knowledge, the most similar tool to JWSL is the WordNet::Similarity (<http://wn-similarity.sourceforge.net>). However, this valuable tool, is a web-based application (written in Perl). Another Java library, the JWNL (<http://wordnet.sourceforge.net>) only provides access to WordNet and does not implement similarity metrics; (iii) it implements several similarity metrics and allows to implement new ones.

6 Concluding Remarks and Future Work

This paper presented a new similarity metric combining the feature based and information theoretic theories of similarity. This metric, as shown by experimental evaluation, outperforms the state of the art. Another contribution has been the similarity experiment we performed in order to build a reference basis for comparing the different metrics. Finally, we implemented our metric and several others in the JWSL. As future work we aim at investigating the domain-independence of our approach by evaluating it on other ontologies (e.g., MeSH) and perform a computational complexity analysis of the different metrics.

References

1. Budanitsky, H.G.A.: Semantic distance in WordNet: an Experimental Application Oriented Evaluation of Five Measures. In: Proc. of NACCL 2001, pp. 29–34 (2001)
2. Cilibrasi, R.L., Vitanyi, P.M.B.: The Google Similarity Distance. IEEE TKDE 19(3), 370–383 (2007)
3. Danushka, B., Yutaka, M., Mitsuru, I.: Measuring Semantic Similarity between Words Using Web Search Engines. In: Proc. of WWW 2007, pp. 757–766 (2007)
4. Hai, C., Hanhua, J.: Semrex: Efficient Search in Semantic Overlay for Literature Retrieval. FGCS 24(6), 475–488 (2008)

5. Hirst, G., St-Onge, D.: WordNet: An Electronic Lexical Database. In: *Lexical Chains as Representations of Context for the Detection and Correction of Malapropisms*. MIT Press, Cambridge (1998)
6. Hliaoutakis, A., Varelakis, G., Voutsakis, E., Petrakis, E.G.M., Milios, E.E.: Information retrieval by Semantic Similarity. *Int. J. SWIS* 2(3), 55–73 (2006)
7. Janowicz, K.: Semantic Similarity Blog, <http://www.similarity-blog.de/>
8. Jiang, J., Conrath, D.: Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. In: *Proc. ROCLING X* (1997)
9. Lee, J., Kim, M., Lee, Y.: Information Retrieval Based on Conceptual Distance in is-a Hierarchies. *Journal of Documentation* 49, 188–207 (1993)
10. Li, Y., Bandar, A., McLean, D.: An Approach for Measuring Semantic Similarity between Words Using Multiple Information Sources. *IEEE TKDE* 15(4), 871–882 (2003)
11. Li, Y., McLean, D., Bandar, Z., O'Shea, J., Crockett, K.: Sentence Similarity based on Semantic Nets and Corpus Statistics. *IEEE TKDE* 18(8), 1138–1150 (2006)
12. Lin, D.: An Information-Theoretic Definition of Similarity. In: *Proc. of Conf. on Machine Learning*, pp. 296–304 (1998)
13. Meilicke, C., Stuckenschmidt, H., Tamilin, A.: Repairing Ontology Mappings. In: *Proc. of AAAI 2007*, pp. 1408–1413 (2007)
14. Miller, G.: Wordnet an On-Line Lexical Database. *International Journal of Lexicography* 3(4), 235–312 (1990)
15. Miller, G., Charles, W.: Contextual Correlates of Semantic Similarity. *Language and Cognitive Processes* 6, 1–28 (1991)
16. Pedersen, T., Pakhomov, S.V.S., Patwardhan, S., Chute, C.G.: Measures of Semantic Similarity and Relatedness in the Biomedical Domain. *Journal of Biomedical Informatics* 40(3), 288–299 (2007)
17. Pirró, G., Ruffolo, M., Talia, D.: SECCO: On Building Semantic Links in Peer to Peer Networks. *Journal on Data Semantics XII* (to appear, 2008)
18. Rada, R., Mili, H., Bicknell, M., Blettner, E.: Development and Application of a Metric on Semantic Nets. *IEEE Transactions on Systems, Man, and Cybernetics* 19, 17–30 (1989)
19. Ravi, S., Rada, M.: Unsupervised Graph-Based Word Sense Disambiguation Using Measures of Word Semantic Similarity. In: *Proc. of ICSC 2007* (2007)
20. Resnik, P.: Information Content to Evaluate Semantic Similarity in a Taxonomy. In: *Proc. of IJCAI 1995*, pp. 448–453 (1995)
21. Rissland, E.L.: Ai and Similarity. *IEEE Intelligent Systems* 21, 39–49 (2006)
22. Rodriguez, M., Egenhofer, M.: Determining Semantic Similarity among Entity Classes from Different Ontologies. *IEEE TKDE* 15(2), 442–456 (2003)
23. Rubenstein, H., Goodenough, J.B.: Contextual Correlates of Synonymy. *CACM* 8(10), 627–633 (1965)
24. Schaeffer, B., Wallace, R.: Semantic Similarity and the Comparison of Word Meanings. *J. Experimental Psychology* 82, 343–346 (1969)
25. Schwering, A.: Hybrid Model for Semantic Similarity Measurement. In: *Proc. of ODBASE 2005*, pp. 1449–1465 (2005)
26. Seco, N.: Computational Models of Similarity in Lexical Ontologies. Master's thesis, University College Dublin (2005)
27. Seco, N., Veale, T., Hayes, J.: An Intrinsic Information Content Metric for Semantic Similarity in WordNet. In: *Proc. of ECAI 2004*, pp. 1089–1090 (2004)
28. Shannon, C.: A Mathematical Theory of Communication. *Bell System Technical Journal* 27, 379–423 (1948)
29. Tversky, A.: Features of similarity. *Psychological Review* 84(2), 327–352 (1977)
30. Zavaracky, A.: Glossary-Based Semantic Similarity in the WordNet Ontology. Master's thesis, University College Dublin (2003)

Weighted Ontology for Semantic Search

Anna Formica, Michele Missikoff, Elaheh Pourabbas, and Francesco Taglino

Istituto di Analisi dei Sistemi ed Informatica “Antonio Ruberti”
Consiglio Nazionale delle Ricerche, Viale Manzoni 30
00185 Rome, Italy
{anna.formica,michele.missikoff,elaheh.pourabbas,
francesco.taglino}@iasi.cnr.it

Abstract. This paper presents a method, *SemSim*, for the semantic search and retrieval of digital resources (DRs) that have been previously annotated. The annotation is performed by using a set of characterizing concepts, referred to as *features*, selected from a reference ontology. The proposed semantic search method requires that the features in the ontology are weighted. The weight represents the probability that a resource is annotated with the associated feature. The *SemSim* method operates in three stages. In the first stage, the similarity between concepts (*consim*) is computed by using their weights. In the second stage, the concept weights are used to derive the semantic similarity (*semsim*) between a user request and the DRs. In the last stage, the answer is returned in the form of a ranked list. An experiment aimed at assessing the proposed method and a comparison against a few among the most popular competing solutions is given.

Keywords: Similarity Reasoning, Reference Ontology, Information content, Digital Resources.

1 Introduction

Similarity reasoning is a very challenging research area. After some decades of research, there is an enormous corpus of scientific results available, but still there is not a single solution that is clearly emerging, capable of outperforming all the others. Probably, it will never be the case, due to the great variety of problems requiring similarity reasoning, and situations in which such problems arise. There is a nice example, reported in [12], concerning the perceived similarity between the elements of the triple: (*pig*, *pick-up*, *donkey*). At first, one would assert a higher similarity between the two terms that represent living entities: *pig* and *donkey*, since the term *pick-up* denotes a mechanical artefact. However, if we change perspective, considering a situation where we need to transport, say, potatoes, then *pick-up* and *donkey* are the most alike. Such divergent outcomes derive from a shift of perspective, and therefore a change in what are the relevant characteristics taken into consideration to determine the similarity. An effective similarity reasoner will be endowed with multiple methods and strategies, and the capacity of analyzing a situation to determine the most promising method.

1.1 Semantic Searching

One of the primary uses of similarity reasoning is in method processing a user request to retrieve a set of (digital) resources from a given repository (or, more in general, from the Web). Such resources can be the actual target of the retrieval process (textual or multimedia documents, images, Web services, business process diagrams, etc.) or digital surrogates of non-digital objects (people, organizations, cars, furniture, hotels, etc.). In our work, we are only dealing with digital surrogates, i.e., semantic annotations¹ that we will refer to as *feature vectors*². Therefore, even if we intend to search objects of the first category, for instance digitalized documents, we are not going to consider them directly, in the search phase, e.g., by applying Natural Language or Information Retrieval techniques. Our search method is based on feature vectors, that we assume have been preliminary built (with some *feature extraction* techniques³) and made accessible.

Searching over features vectors instead of target resources has several advantages:

- feature vectors are homogeneous structures, independent of the nature of the resources they are associated with; therefore,
- the semantic search method can be unified for different kinds of resources (e.g., text, photo, video, etc.), once they have been properly annotated; therefore it is possible to search different repositories of a different nature by using the same method; then,
- the retrieved results can be reported in an homogeneous form, and ranked in a unique list, even if the concrete forms of the retrieved resources are very different.

Beside feature vectors, a second characterization of the proposed search method is represented by the use of a *weighted reference ontology (WRO)*, containing the relevant concepts in the given domain.

Here, we wish to recall the definition of an ontology, taken from the OMG Ontology Definition Metamodel [4]:

"An *ontology* defines the common *terms* and *concepts* (meaning) used to describe and represent an area of knowledge. An ontology can range in expressivity from a *Taxonomy* (knowledge with minimal hierarchy or a parent/child structure), to a *Thesaurus* (words and synonyms), to a *Conceptual Model* (with more complex knowledge), to a *Logical Theory* (with very rich, complex, consistent and meaningful knowledge)."

In our case, we restrict our view of the ontology to a taxonomy of concepts. This simplified view, as anticipated, is then enriched with a weight associated with each concept. Intuitively, the weight of a concept represents its *featuring power*, i.e., how

¹ Semantic annotation is a very active research area [25] whose description goes beyond the scope of the paper.

² A *feature vector* is an n-dimensional vector of (numerical) features that represent some object... [it is obtained as a] reduced representation of the key characteristics of the object (see http://en.wikipedia.org/wiki/Feature_vector).

³ Feature extraction is an important topic that will not be addressed in this paper for lack of space [24].

selective is such a concept in characterizing the resources of our universe. A high weight corresponds to a low selectivity level, i.e., many resources are characterized by the concept. Conversely, a low weight corresponds to a high selectivity, and therefore its use in a request will return less instances. In accordance with the Information Theory [23], a concept weight will be used to determine its (relative) information content.

The WRO is an important element of our proposal, since we restrict the elements of a feature vector to the terms that represent concepts in the ontology. For this reason, we will refer to the former as: *Ontology-based Feature Vector (OFV)*. The same is true for a user request that takes the form of a *Request Vector (RV)*.

The *SemSim* method proposed in this paper supports a user wishing to retrieve digital resources from a given UDR (Universe of Digital Resources). An UDR can be a document repository maintained by one enterprise, or can be a shared, distributed content repository hosted in different organizations belonging to a Virtual Enterprise, or even it can be the Web as a whole. When searching, the user indicates a set of desired features (in the form of a request vector rv) expecting to have in return a set of resources (partially) satisfying such features. Similarly to Google search, the output of *SemSim* is a ranked list of resources, sorted according to their similarity to the rv . The semantic search method is mainly based on the notion of similarity of ontology-based feature vectors, and therefore on the similarity of the concepts that compose the two structures. Similarity reasoning is a challenging job.

The following list reports the primary dimensions that can be considered in performing similarity reasoning:

- *Terminological*, if the concepts are characterized by a set of terms (e.g., WordNet synset, user generated keywords, etc.);
- *Linguistic*, determined contrasting the textual descriptions (if available) of the concepts;
- *Structural*, when we consider the information structures (e.g., attributes and associations) of the contrasted concepts;
- *Taxonomic*, when the similarity is determined by the hierarchal organization of concepts in the ontology;
- *Extensional*, when the similarity is derived considering the instances of the contrasted concepts;
- *Operational*, based on operations associated to the contrasted concepts.

An effective similarity reasoning system should be able to take into consideration more than one dimension from the above list. In this paper, we will present a similarity reasoning method, *SemSim*, that operates along the taxonomic and extensional dimensions. In fact, a central issue is the weighting of concepts in the ontology that is based on both the position in taxonomy and extension of each concept, seen as the set of annotated DRs. To maintain a precise count of the annotated resources is a difficult job, especially in a dynamic domain. Therefore, we propose a probabilistic approach, where the weight of a concept represents the probability that a resource in the domain is characterized by that concept.

1.2 Promising Application Domains for Semantic Search

This work has started in the context of a large industrial conglomerate (Finmeccanica) that develops large engineering systems: from air traffic control (ATC) to integrated

civil protection networks. Each project consists of thousands of parts, devices, apparatuses, subsystems, and interconnected systems. In previously accomplished projects, an incredible wealth of knowledge has been accumulated in the form of blueprints, design drafts, data sheets, CAD/CAM files, test cases and measures, technical notes, installation manuals, troubleshooting plans, etc. All these documents are stored in digital forms, but are placed in different sites, different formats, created with different software tools; therefore they are not easy, for an interested user, to be identified and retrieved. When a new project starts, it is extremely useful (and cost saving) to have the possibility to effectively access the wealth of knowledge produced by previous projects. To this end, the availability of a global search engine, based on semantic technologies, is particularly promising. This is the application context in which the SemSim method has been initially developed.

Another application domain is represented by the tourism industry. Here instead of having a single large industrial conglomerate (with a closed UDR), we have a network of SMEs (e.g., providing transportation, accommodation, food, cultural services, natural parks access) able to provide an integrated offer for a tourist⁴. In the tourism domain we have again a large variety of digital resources, available on different web sites, in different locations. Here the variety of formats is less important (essentially we have web documents), but the fragmentation and the possibility of retrieving documents on the basis of their semantic content is equally important. Moreover, this domain is more open, dynamic, and less regulated than the previous one. Since tourism is more intuitive, we drew from it the running example used in this paper.

The rest of the paper is organized as follows. Section 2 is dedicated to the related work, while Section 3 presents the basic notions and the structures used in SemSim. In Section 4, the definition of a Weighted Reference Ontology is given and a running example is introduced. In Section 5, the proposed SemSim method for evaluating the semantic similarity of Ontology-based Feature Vectors is presented. In Section 6, we present an assessment of the SemSim method, contrasting it against a few other methods. Finally, Section 7 concludes the paper with indications of future work.

2 Related Work

In the vast literature available (see for instance [1,5,17,18]), we will focus on the proposals tightly related to our approach. We wish to recap that the focus of our work is on the method to compute the similarity between concept vectors. To this end, we need to build a two stages method: firstly computing the pair-wise concept similarity, and then deriving the similarity of two vectors of concepts. Thus, we adopted a technique based on the information content [22], which has been successively refined by Lin in [15]. The Lin's approach shows a higher correlation with human judgement than other methods, such as the *edge-counting* approach [21] and Wu-Palmer [26].

We need to emphasise that we deal with specialised domains (e.g., systems engineering, tourism, etc.), requiring specialised domain ontologies. The large majority of existing proposals make use of WordNet. This is a lexical ontology that is generic

⁴ We exclude the big tour operators and hotel chains, to address the constellation of SMEs and small to micro tourism services providers.

(i.e., not focused on a specific domain) and, furthermore, contains only simple terms, no multi-word terms are reported (e.g., terms such as “power supply” or “farm house” are not available in WordNet). Therefore, our approach is different from all other proposals that use any generic ontology.

SemSim is based on an ontology with weighted concepts. In [6] there is an interesting proposal that makes use of an ontology enriched with a typical Natural Language Processing method, based on *term frequency* and *inverse document frequency (tf-idf)*. With respect to this work, our proposal abstracts from the linguistic domain during the annotation phase, allowing therefore for a pure semantic approach. Furthermore, in weighting the terms connected to the elements of the ontology, [6] relies on a rigid approach, i.e., it proposes 5 relevance levels that correspond to 5 constant values: *direct(1.0)*, *strong(0.7)*, *normal(0.4)*, *weak(0.2)*, *irrelevant(0.0)*. In our method, the weights, and the relationships among concepts, are not discrete and take any value between 0 and 1.

The work presented in [13] shares some similarity with our approach. It proposes a bottom up method that, starting from the weight associated with concept nodes, determines the concept similarity by building vectors of weights. Therefore, the objective is the similarity of concepts that depends on the topology of the ontology and the position of concepts therein. However, our scope is wider: similarity of concepts (*consim*) is just a step of a more comprehensive method aimed at determining the similarity of two concept vectors (*semsim*). We could have selected the method proposed in [13] for the first phase of our work (concept similarity), but it was not completely convincing, since its assessment is based on the well known Miller and Charles experiment [19] that, being based on WordNet, is not conceived for specialized domains.

In [14], a similarity measure between words is defined, where each word is associated with a concept in a given ISA hierarchy. The proposed measure essentially combines path length between words, depth of word subsumer in the hierarchy, and local semantic density of the words. However, similar to [13], the authors evaluate their method using Miller and Charles experiment that was conceived for general domains and is not appropriate for specialized applications.

Other research results concern the similarity between two sets (or vectors) of concepts. In the literature the *Dice* [9,16] and *Jaccard* [11] methods are often adopted in order to compare vectors of concepts. However, in both Dice and Jaccard concept similarity is computed by using exact match, with 0 or 1 response. Therefore, the matchmaking of two concept vectors is based on their intersection, without considering the positioning of the concepts in the ontology. More recent works (see [2]) introduce the ontology, hence proposing a more elaborated concept matching. Our proposal is based on a more refined semantic matchmaking, since the match of two concepts is based on their shared information content, and the vectors similarity is based on the optimal concepts coupling.

[3] introduces two new algorithms for computing the semantic distance/similarity between sets of concepts belonging to the same ontology. They are based on an extension of the Dijkstra algorithm⁵ to search for the shortest path in a graph. With respect to our approach, here the similarity is based on the distance between concepts rather than the

⁵ http://en.wikipedia.org/wiki/Dijkstra's_algorithm

information content carried by each concept. Furthermore, the similarity between two sets of concepts is computed by considering the similarity between each concept from a set and all the concepts from the other. Finally, the similarity between adjacent concepts is supposed to be decided at design-time by the ontology developer and consequently introduces a certain degree of rigidity and bias on the results.

This brief overview has mainly the goal of positioning our work with respect to the most relevant results in the literature. As shown, in the various cases, our approach is either more focused (i.e., for specialised domains) or more elaborated (e.g., considering the information content of concepts with respect to the UDR and their positioning in the ontology). But we know that more elaborated solutions may not perform better than simpler ones. For this reason, we decided to conduct an experiment to assess the results of the SemSim method against a few among the most promising competitors. These results are reported in Section 6.

3 Basic Definitions

In this section we introduce the basic notions and the structures used in the SemSim method. Summarising, SemSim is based on the following structures:

- a Universe of Digital Resources (UDR) over which the search is performed;
- a Weighted Reference Ontology (WRO);
- a Semantic Annotation Repository (SAR) containing the ontology-based feature vectors (OFVs), one for each digital resource in UDR;
- a Request Vector (RV);
- a Ranked Solution Vector (RSV), subset of the UDR resources, whose OFVs are similar to the RV, filtered by a given threshold.

In this section, we provide a formal account of the structures that are used in the SemSim method.

Definition 1. *Universe of Digital Resources (UDR).* The UDR is the totality of the digital resources that are semantically annotated.

Definition 2. *Ontology.* An *Ontology* is a formal, explicit specification of a shared conceptualization [10]. In our work we address a simplified notion of ontology, *Ont*, that focuses on a set of concepts organized according to a specialization hierarchy. In particular, *Ont* is a *tree* structure defined by the pair:

$$Ont = \langle C, H \rangle$$

where C is a set of concepts and H is the set of pairs of concepts of C that are in subsumption (*subs*) relation:

$$H = \{(c_i, c_j) \in C \times C \mid subs(c_i, c_j)\}$$

Since we assume that *Ont* is a tree, given two concepts $c_i, c_j \in C$, the *least upper bound* of c_i, c_j , $lub(c_i, c_j)$, is always defined in C . It represents the less abstract concept of the ontology that subsumes both c_i and c_j .

Definition 3. *Weighted Reference Ontology (WRO).* Given an ontology $Ont = \langle C, H \rangle$, a WRO is a pair:

$$WRO = \langle Ont, w \rangle$$

where w is a function defined over C , such that given a concept $c \in C$, $w(c)$ is a rational number in the interval $[0..1]$ standing for a weight associated with the concept c in the ontology Ont .

Definition 4. *Ontology Feature Vector (OFV).* Given an ontology $Ont = \langle C, H \rangle$ and a digital resource $dr_i \in UDR$, an OFV associated with dr , ofv_{dr} , is a set of ontology concepts defined as follows:

$$ofv_i = (c_{i,1}, \dots, c_{i,m}) \text{ where } c_{i,j} \in C, j = 1 \dots m$$

To actually link a concept to the resources that it characterises, it is necessary to introduce the notion of a *featured extension*.

Definition 5. *Featured Extension.* Given an ontology $Ont = \langle C, H \rangle$, and a concept (feature) $c \in C$, the *featured extension* of c is defined according to the extension function F_{Ont} as follows:

$$F_{Ont}(c) = \{dr \in UDR \mid c \in ofv_{dr}\}$$

Therefore, given a feature c , $F_{Ont}(c)$ provides all the digital resources in UDR whose OFVs contain c , i.e., all the digital resources that are annotated by the feature c .

Definition 6. *Similarly Featured Extension.* Given an ontology $Ont = \langle C, H \rangle$, and a concept (feature) $c \in C$, the *similarly featured extension* of c is defined according to the extension function S_{Ont} as follows:

$$S_{Ont}(c) = \{dr \in UDR \mid \exists c' \in ofv_{dr} \text{ consim}(c, c') > k \}$$

where *consim* is the concept similarity that will be formally introduced in Section 5, and k is a threshold suitably defined according to the cases. In this paper we assumed $k=0.3$. Therefore, given a feature c , $S_{Ont}(c)$ provides all the digital resources in UDR whose OFVs contain a feature c' whose similarity with c is higher than a fixed threshold. In other words, it provides all the digital resources that are annotated by a feature similar to that required, up to a threshold.

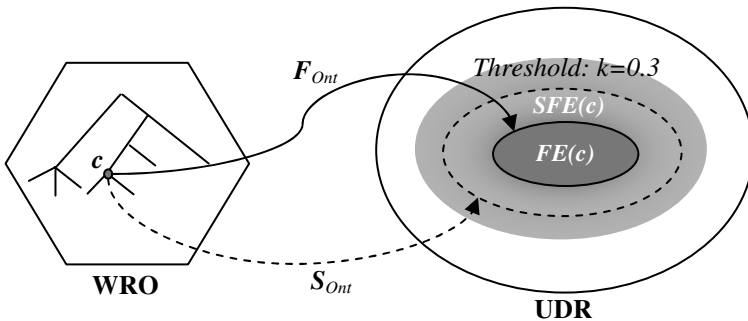


Fig. 1. Relationship between a concept c and its extensions

Figure 1 visually depicts the relationship among a concept c and its extensions: Featured Extension ($FE(c)$ represented by the inner set) and Similarly Featured Extension ($SFE(c)$ represented by the dash-bordered set).

Definition 7. Semantics of an OFV. Given a repository UDR annotated with OFVs, the semantics of an OFV, ofv , is defined according to the extension function E_{ofv} as follows:

$$E_{ofv}(ofv) = \bigcap_{i=1, \dots, m} F_{Ont}(c_i)$$

where $F_{Ont}(c_i)$ is the *featured extension* of the concept c_i , for $i = 1 \dots m$. Therefore, $E_{ofv}(ofv)$ provides all the digital resources in UDR characterized by the features in the OFV.

Definition 8. Semantics of a Request Vector. Given an ontology $Ont = \langle C, H \rangle$ and a Request Vector rv :

$$rv = (c_1, \dots, c_n)$$

where $c_i \in C$ for $i = 1 \dots n$, the semantics of rv is defined according to the extension function E_{RV} as follows:

$$E_{RV}(rv) = \bigcup_{i=1, \dots, n} (F_{Ont}(c_i) \cup S_{Ont}(c_i))$$

where $F_{Ont}(c_i)$ and $S_{Ont}(c_i)$ are respectively the *featured extension* and the *similarly featured extension* of the concept c_i for $i = 1 \dots n$. Therefore, $E_{RV}(rv)$ provides all the digital resources in UDR whose OFVs contain at least one feature of rv , or one feature that is similar to at least one feature of rv .

Definition 9. Ranked Solution Vector. Given an ontology $Ont = \langle C, H \rangle$ and a Request Vector rv , the Ranked Solution Vector associated with rv , $RSV(rv)$, is defined as follows:

$$RSV(rv) = \{(dr, semsim) \mid dr \in E_{RV}(rv) \text{ AND } semsim(dr, rv) > h\}$$

where $semsim(dr, rv)$ is the semantic similarity between dr and rv that will be introduced in Section 5, and h is a threshold suitably defined according to the cases. Therefore, the Ranked Solution Vector of a Request Vector provides all the digital resources of UDR whose similarity with the Request Vector is higher than the given threshold.

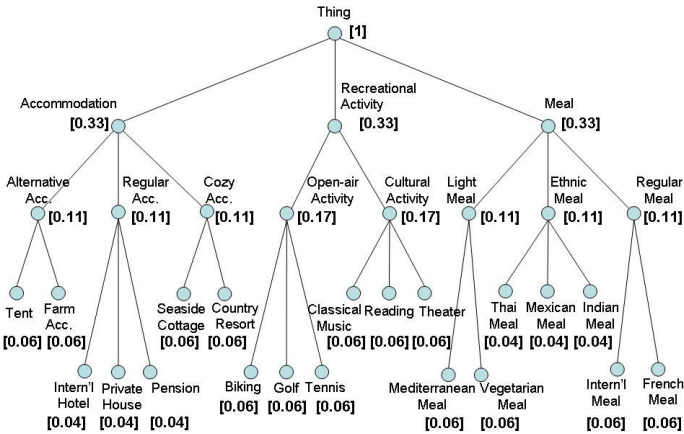
4 Weight Assignment in the WRO

Prior to addressing the method to associate weights with the concepts in the ontology, we introduce our example drawn from the tourism domain. In the example we assume to have a dozen of hotels that accepted to annotate their leaflets by using a common reference ontology. Each annotation is therefore an OFV, as reported in Table 1.

Table 1. OFV-based annotation of Digital Resources

ofv ₁ = (InternationalHotel, Golf, InternationalMeal, Theatre)
ofv ₂ = (Pension, FrenchMeal, Biking, Reading)
ofv ₃ = (CountryResort, MediterraneanMeal, Tennis)
ofv ₄ = (CozyAccommodation, ClassicalMusic, InternationalMeal)
ofv ₅ = (InternationalHotel, ThaiMeal, IndianMeal, ClassicalMusic)
ofv ₆ = (CountryResort, LightMeal, ClassicalMusic)
ofv ₇ = (SeasideCottage, EthnicMeal, CulturalActivity)
ofv ₈ = (CountryResort, VegetarianMeal, CulturalActivity)
ofv ₉ = (SeasideCottage, MediterraneanMeal, Golf, Biking)
ofv ₁₀ = (RegularAccommodation, RegularMeal, Biking)
ofv ₁₁ = (SeasideCottage, VegetarianMeal, Tennis)
ofv ₁₂ = (SeasideCottage, VegetarianMeal)

The above Semantic Annotation Repository (SAR) has been built starting with the WRO reported in Figure 2.

**Fig. 2.** Concept weights as uniform probabilistic distribution

Our approach in weighting is based on the probability distribution along the hierarchy of concepts starting from the root, namely *Thing*, that stands for the most abstract concept, whose weight $w_p(\text{Thing})$ is equal to 1. Here we adopt a uniform probabilistic distribution, therefore, given a number n of children ($c_i, i=1\dots n$) of this top concept, the probability of each child is $w_p(c_i)=1/n$. Accordingly, for any other concept c , $w_p(c)$ is equal to the probability of the father of c , divided by the number of the children of the father of c (i.e., the fan-out). In Figure 2, an example of the probabilistic distribution over concepts of our ISA hierarchy is illustrated. For instance, let us consider the concept *LightMeal*, where $w_p(\text{LightMeal}) = 1/9$, since $w_p(\text{Meal})=1/3$ and *Meal* has 3 sub-concepts.

In the next section, we will show how the set of OFVs will be used to perform a semantic search of the hotels, starting from a vector of desired features indicated by the user.

5 The SemSim Method for Semantic Search and Retrieval

Consider the following user request:

"I would like to stay in a seaside hotel, where I can have vegetarian food, play tennis, and attend sessions of classical music in the evening".

It can be formulated according to the request feature vector notation as follows:

$rv = (SeasideCottage, VegetarianMeal, Tennis, ClassicalMusic)$

Once the rv has been specified, the SemSim method is able to evaluate the semantic similarity (*semsim*) among the rv and each available OFV. As already mentioned, in order to compute the *semsim* between two feature vectors, it is necessary first to compute the similarity (*consim*) between pairs of concepts.

5.1 Computing Concept Similarity: *consim*

Given a WRO, the notion of *consim* relies on the probabilistic approach defined by Lin [15], which is based on the notion of information content. According to the standard argumentation of information theory, the *information content* of a concept c is defined as $-\log w_p(c)$, therefore, as the weight of a concept increases the informativeness decreases, hence, the more abstract a concept the lower its information content.

Given two concepts c_i and c_j , their similarity, $consim(c_i, c_j)$, is defined as the maximum information content shared by the concepts divided by the information contents of the two concepts [11]. Note that, since we assumed that the ontology is a tree, the least upper bound of c_i and c_j , $lub(c_i, c_j)$, is always defined and provides the maximum information content shared by the concepts in the taxonomy. Formally, we have:

$$consim(c_i, c_j) = 2 \log w_p(lub(c_i, c_j)) / (\log w_p(c_i) + \log w_p(c_j))$$

For instance, considering the pair of concepts *Biking* and *Tennis* of the WRO shown in Figure 2, the *consim* is defined as follows:

$$consim(Biking, Tennis) = 2 \log w_p(Open-airActivity) / (\log w_p(Biking) + \log w_p(Tennis)) = 0.63$$

since, according to Figure 2, *Open-airActivity* is the *lub* of *Biking* and *Tennis* and therefore provides the maximum information content shared by the comparing concepts.

5.2 Computing Semantic Similarity Degree: *semsim*

In this section we show how we derive the semantic similarity of two vectors, rv and ofv , by using the *consim* function. In principle, we need to start from the cartesian product of the vectors:

$$rv \otimes ofv = \{ (c_i, c_j) \}$$

where: $i = 1..n$, $j = 1..m$, $n = |rv|$, $m = |ofv|$, $c_i \in rv$, and $c_j \in ofv$.

For each pair we can derive the concept similarity *consim*, as seen in the previous section. However, we do not need to consider all possible pairs, since in many cases the check is meaningless (e.g., contrasting a vegetarian meal with a classical music concert). Hence, we aim at restricting our analysis considering only the pairs that exhibit a higher affinity. Furthermore, we adopted the exclusive match philosophy (sometimes named *wedding* approach) where once a pair of concepts has been successfully matched, concepts do not participate in any other pair. In other words, assuming *rv* and *ofv* represent a set of boys and a set of girls respectively, we analyze all possible sets of marriages, when polygamy is not allowed. Our solution, for the computation of the semantic similarity, *semsim(rv, ofv)*, makes use of the method based on the *maximum weighted bipartite matching* problem in bipartite graphs [7,8].

Essentially, the method aims at the identification of the sets of pairs of concepts of the two vectors that maximizes the sum of *consim*.

$$semsim(rv, ofv) = \max(\sum consim(c_i, c_j)) / \min(n, m)$$

In particular, the method that we adopted to solve this problem is based on the well-known Hungarian Algorithm [20].

For instance, in the case of *rv* and *ofv*₁ of our running example, the following set of pairs of concepts has the maximum *consim* sum:

{(SeasideCottage, InternationalHotel),
(VegetarianMeal, InternationalMeal)
(ClassicalMusic, Theater),
(Tennis, Golf)}

since:

consim(SeasideCottage, InternationalHotel)= 0.36
consim(VegetarianMeal, InternationalMeal)= 0.38
consim(ClassicalMusic, Theater)=0.62
consim(Tennis, Golf)=0.62

and any other pairing will lead to a smaller sum. Therefore:

$$semsim(rv, ofv_1) = (0.36 + 0.38 + 0.62 + 0.62) / 4 = 0.49$$

where the sum of *consim* has been normalized according to the minimal cardinality of the contrasted vectors (in this case 4 for both).

6 SemSim Assessment

In this section we present some preliminary results on the assessment of the proposed SemSim method. The assessment is based on the correlation of *semsim* with human judgment (HJ). Essentially, we contrasted the results of our method with those obtained by a selected group of 20 people. We asked them to express their judgement (on a scale of 0 to 3) on the similarity among the *rv*, and the set of resources at hand, i.e., the hotels H_i , $i = 1 \dots 12$, annotated with the *ofv*_{*i*} shown in Table 1. In Table 2, the human judgment (whose values have been normalized) and *semsim* scores are illustrated (see second and third column).

Table 2. Results of the comparison among human judgment, SemSim and some representative similarity methods

<i>Feature Vectors</i>	<i>HJ</i>	<i>SemSim</i>	<i>Dice</i>	<i>Jaccard</i>	<i>Salton's Cosine</i>	<i>Weighted Sum</i>
<i>ofv₁</i>	0.60	0.49	0.00	0.00	0.00	0.00
<i>ofv₂</i>	0.60	0.49	0.00	0.00	0.00	0.00
<i>ofv₃</i>	0.67	0.63	0.29	0.17	0.08	0.29
<i>ofv₄</i>	0.60	0.56	0.29	0.17	0.08	0.43
<i>ofv₅</i>	0.59	0.43	0.25	0.14	0.06	0.25
<i>ofv₆</i>	0.80	0.66	0.29	0.17	0.08	0.43
<i>ofv₇</i>	0.60	0.55	0.29	0.17	0.08	0.43
<i>ofv₈</i>	0.67	0.63	0.29	0.17	0.08	0.43
<i>ofv₉</i>	0.67	0.69	0.25	0.14	0.06	0.25
<i>ofv₁₀</i>	0.36	0.37	0.00	0.00	0.00	0.00
<i>ofv₁₁</i>	0.82	0.75	0.86	0.75	0.25	0.86
<i>ofv₁₂</i>	0.71	0.50	0.67	0.50	0.25	0.67
<i>Correlation with HJ</i>	1.00	0.82	0.70	0.67	0.66	0.72

Furthermore, we compared SemSim with some representative similarity methods proposed in the literature: Dice, Jaccard, Salton's Cosine[16] and the Weighted Sum defined in [2]. For the sake of simplicity, we recall their formulas below, where *X* and *Y* represent the *rv* and an *ofv*, respectively.

$$2 \frac{|X \cap Y|}{|X| + |Y|}$$

Dice' coefficient

$$\frac{|X \cap Y|}{|X \cup Y|}$$

Jaccard's coefficient

$$\frac{|X \cap Y|}{|X| \times |Y|}$$

Salton's Cosine coefficient

$$2 \frac{\sum Aff(X_i, Y_j)}{|X| + |Y|}$$

Weighted Sum function,
where Aff(*X_i*, *Y_j*), the affinity b/w *X_i* and *Y_j*, is
1 if *X_i* = *Y_j*
0.5 if *X_i* is a broader or narrower concept of *Y_j*
0 otherwise

The experiment has shown that SemSim yields a higher correlation with human judgement (0.82) with respect to other representative methods.

In order to improve readability, the results given in Table 2 are also illustrated in Figure 3.

6.1 The Ranked Solution Vector

In Table 3, the lists of the DRs, obtained by human judgement and SemSim, are shown. They are ordered according to decreasing semantic similarity degrees with *rv*.

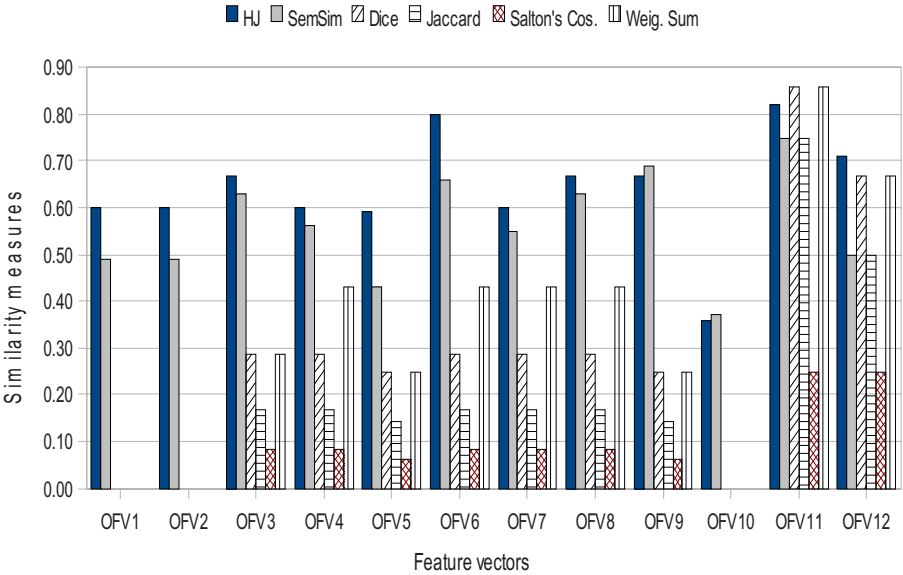


Fig. 3. Illustration of the comparison shown in Table 2

Table 3. Ranked Solution Vectors for HJ and SemSim

Human Judgment (HJ)		SemSim	
Ranked Resources	Values	Ranked Resources	Values
H6	0.83	H11	0.75
H11	0.78	H9	0.69
H3, H8, H9	0.75	H6	0.66
H12	0.70	H3, H8	0.63
H1, H2, H4, H7	0.67	H4	0.56
H5	0.63	H7	0.55
H10	0.37	H12	0.50
		H1, H2	0.49
		H5	0.43
		H10	0.37

In this table, the horizontal line separates DRs according to a threshold, here fixed to 0.55. Thus, the Ranked Solution Vector (RSV) of our running example is defined by the DRs above the horizontal line.

Analyzing Table 3, we are able to show the effectiveness of our method according to the precision and recall values. As usual, *precision* is obtained dividing the number of discovered valid resources (the intersection between the first and the third columns of Table 3) by the total number of discovered resources (third column of Table 3), while *recall* is computed dividing the number of discovered valid resources by the total number of valid resources (first column of Table 3), up to a threshold. Finally, the F-measure, which is two times the product of precision and recall divided by their sum, is also given.

In our case, assuming the threshold equal to 0.55, we have:

Precision	Recall	F-measure
1	0.64	0.78

Note that, in general, selecting higher thresholds results in higher precision values, while selecting lower thresholds leads to higher recall values.

7 Conclusions and Future Work

The problem of achieving new generations of search engines capable of exploiting the emerging semantic technologies is attracting much attention today. It is a widely shared opinion that we need to perform a “quantum leap” and achieve new generation search engines that exploit the semantic content of a resource when performing a search and retrieval task. In this paper, we presented the SemSim method that goes in this direction. Our method is innovative since it is based on the possibility of annotating each DR with a vector of characterizing features (OFV), selected from the concepts of an ontology. Our method is based on three key elements: (i) a Weighted Reference Ontology, where each concept in the ISA hierarchy is weighted using a probabilistic distribution approach; (ii) the use of the Lin method to determine the similarity between concepts (i.e., *consim*) in the Ontology; (iii) the use of the Hungarian Algorithm to compute the similarity degree between a *rv* and an *ofv*.

The SemSim method has been implemented and a number of tests have been carried out that show its high correlation with human judgment.

In the future we intend first of all to carry out extensive experiments to acquire a better understanding of the characteristics of our method. A further direction is represented by the possibility of associating a weight with the elements of the request vector, allowing the user to specify a scale of importance on the desired features. This is the first requirement that emerged from the participants when they performed human test: not all the required features are equivalent, users would like to indicate what are the important (or even mandatory) features with respect to other features for which a compromise is acceptable.

Another line of activities will concern the WRO and the method to assign weights to concepts. Currently, the weights are defined according to a uniform distribution of probability. We wish to explore the behaviour of the SemSim method in presence of a skewed probability distribution that may be useful in many cases.

References

1. Alani, H., Brewster, C.: Ontology ranking based on the Analysis of Concept Structures. In: K-CAP 2005, Banff, Alberta, Canada (2005)
2. Castano, S., De Antonellis, V., Fugini, M.G., Pernici, B.: Conceptual Schema Analysis: Techniques and Applications. *ACM Transactions on Databases Systems* 23(3), 286–333 (1998)
3. Cordi, V., Lombardi, P., Martelli, M., Mascardi, V.: An Ontology-Based Similarity between Sets of Concepts. In: *Proceeding of WOA 2005*, pp. 16–21 (2005)
4. DSTC, IBM, Sandpiper Software; Ontology Definition Metamodel, Revised submission to OMG (2005), <http://www.omg.org/docs/ad/05-01-01.pdf>

5. Euzenat, J., Shvaiko, P.: *Ontology Matching*. Springer, Heidelberg (2007)
6. Fang, W.-D., Zhang, L., Wang, Y.-X., Dong, S.-B.: Towards a Semantic Search Engine Based on Ontologies. In: *Proc. of 4th Int'l Conference on Machine Learning*, Guangzhou (2005)
7. Formica, A.: Concept similarity by evaluating Information Contents and Feature Vectors: a combined approach. *Communications of the ACM (CACM)* (to appear, 2008)
8. Formica, A., Missikoff, M.: Concept Similarity in SymOntos: an Enterprise Ontology Management Tool. *Computer Journal* 45(6), 583–594 (2002)
9. Frakes, W.B., Baeza-Yates, R.: *Information Retrieval, Data Structure and Algorithms*. Prentice Hall, Englewood Cliffs (1992)
10. Gruber, T.R.: A translation approach to portable ontologies. *Knowledge Acquisition* 5(2), 199–220 (1993)
11. Jaccard, P.: *Bulletin del la Société Vaudoise des Sciences. Naturelles* 37, 241–272 (1901)
12. Kasahara, K., Matsuzawa, K., Ishikawa, T., Kawaoka, T.: Viewpoint-Based Measurement of Semantic Similarity between Words. In: *Proceedings of the Fifth International Workshop on Artificial Intelligence and Statistics*, pp. 292–302. Fort Lauderdale (1995)
13. Kim, J.W., Candan, K.S.: CP/CV: Concept Similarity Mining without Frequency Information from Domain Describing Taxonomies. In: *Proc. of CIKM 2006 Conference* (2006)
14. Li, Y., Bandar, Z.A., McLean, D.: An Approach fro Measuring Semantic Similarity between Words Using Multiple Information Sources. *IEEE Transactions on Knowledge and Data Engineering* 15(4), 871–882 (2003)
15. Lin, D.: An Information-Theoretic Definition of Similarity. In: Shavlik, J.W. (ed.) *Proc. of 15th the International Conference on Machine Learning*, Madison, Wisconsin, USA, pp. 296–304. Morgan Kaufmann, San Francisco (1998)
16. Maarek, Y.S., Berry, D.M., Kaiser, G.E.: An Information Retrieval Approach For Automatically Constructing Software Libraries. *IEEE Transactions on Software Engineering* 17(8), 800–813 (1991)
17. Madhavan, J., Halevy, A.Y.: Composing Mappings among Data Sources. In: *VLDB 2003*, pp. 572–583 (2003)
18. Maguitman, A.G., Menczer, F., Roinestad, H., Vespignani, A.: Algorithmic Detection of Semantic Similarity. In: *Proc. of WWW 2005 Conference*, Chiba, Japan (May 2005)
19. Miller, G.A., Charles, W.G.: Contextual correlates of semantic similarity. *Language and Cognitive Processes* 6(1), 1–28 (1991)
20. Munkres, J.: Algorithms for the Assignment and Transportation Problems. *Journal of the Society of Industrial and Applied Mathematics* 5(1), 32–38 (1957)
21. Rada, L., Mili, V., Bicknell, E., Bletter, M.: Development and application of a metric on semantic nets. *IEEE Transaction on systems. Man, and Cybernetics* 19(1), 17–30 (1989)
22. Resnik, P.: Using information content to evaluate semantic similarity in a taxonomy. In: *Proc. of IJCAI* (1995)
23. Shannon, C.E.: A Mathematical Theory of Communication. *The Bell System Technical Journal* 27, 379–423, 623–656 (1948)
24. Velardi, P., et al.: TermExtractor: a Web Application to Learn the Common Terminology of Interest Groups and Research Communities. In: *9th Conf. on Terminology and Artificial Intelligence TIA 2007*, Sophia Antinopolis (2007)
25. Velardi, P., Navigli, R., Cuchiarelli, A., Neri, F.: Evaluation of ontolearn, a methodology for automatic population of domain ontologies. In: Buitelaar, P., Cimiano, P., Magnini, B. (eds.) *Ontology Learning from Text: Methods, Evaluation and Applications*. IOS Press, Amsterdam (2005)
26. Wu, Z., Palmer, M.: Verb semantics and lexicon selection. In: *The 32nd Annual Meeting of the Association for Computational Linguistics*, Las Cruces, New, Mexico, pp. 133–138 (1994)

RDF Snippets for Semantic Web Search Engines

Xi Bai^{1,2}, Renaud Delbru¹, and Giovanni Tummarello¹

¹ Digital Enterprise Research Institute,
National University of Ireland, Galway, Ireland

² College of Computer Science and Technology,
Jilin University, Changchun, China

{xi.bai,renaud.delbru,giovanni.tummarello}@deri.org

Abstract. Recently, there has been interest in ranking the resources and generating corresponding expressive descriptions from the Semantic Web. This paper proposes an approach for automatically generating snippets from RDF documents and assisting users in better understanding the content of RDF documents returned by Semantic Web search engines. A heuristic method for discovering topics, based on the occurrences of RDF nodes and the URIs of original RDF documents, is presented and experimented in this paper. In order to make the snippets more understandable, two strategies are proposed and used for ranking the topic-related statements and the query-related statements respectively. Finally, the conclusion is drawn based on the discussion about the performances of our topic discovery and the whole snippet generation approaches on a test dataset provided by Sindice.

1 Introduction

With the development of the Semantic Web, a large amount of Resource Description Framework (RDF) documents have been published on the Web. Therefore, finding out the most helpful RDF documents effectively and efficiently becomes one of the hottest problems in the Semantic Web community. Due to the structure of RDF documents and their uncontrolled growth, it is hard for non-technician users to understand or decipher the information coded in RDF syntax. When users search the RDF documents with some specific topics, the results (i.e., RDF documents links returned by SWSE [7] or by the previous version of Sindice [8]) are not clear enough for users to recognize what exactly the RDF documents refer to. Therefore, expressive resource descriptions should be generated, which will give users the clues and assist them in recognizing the content of the searched results. Several publications [1][2][3][4][5][6] have already contributed to the verbalization of domain ontologies using natural language processing (NLP), clustering techniques, or analysis of URLs using language models, but little work has been done to discover the topics and generate snippets from RDF documents. RDF statements ranking is also very important within the process of generating snippets, but it still faces difficulties and has not been further researched.

In this paper, we propose an approach for summarizing and ranking the content of RDF documents. We use a heuristic method based on the occurrences of

Crawler is in charge of crawling RDF documents from the Web. *RDF Parser* parses each document into a set of RDF statements, which are then saved into the triple repository. *Reasoner* is in charge of using the original RDF documents and their imported ontologies to obtain extra RDF triples. Then the original triples, the inferred triples and the triples in the imported ontologies will be indexed. *Query-Related Nodes Generator* and *Topic Node Generator* take the charge of using heuristic methods to find out the query-related nodes and the topic node respectively. *Snippet Generator* takes the RDF nodes as the input and based on our two ranking algorithms, pre-generates the summarized content which is then saved into the XML documents. According to the users' preferences and predefined XSL templates, *Data Formatter* finally transforms the XML documents into the snippets regarded as the summarizations and returns them to the users. If users want to get the detail information about the resources contained in the generated snippets, the *URI Analyzer* is capable of acquiring their corresponding URIs of the resources appointed by users and then query *Index* again to recursively generate snippets. If the required resources have not been indexed, *Crawler* will be reactivated to crawl the missed resources. According to the framework, the finally generated snippet is related to both the URL of the original RDF document and the user's query. Therefore, we use the cache to optimize the generation process.

3 Preprocessing of RDF Documents

An RDF document contains the resources whose types are possibly not declared inside. Users will be puzzled, if the majority of the resources appears without their types in the final snippet. Therefore, we first supplement the content of original RDF documents by adding the imported ontologies. We also involve the inference to find out the Inverse Functional Property (IFP) which will assist us in looking for the topic of an RDF document. In [9], the reasoner based on OWL "ter Horst" [10] is used for finding out IFPs. Here, we also use it to fulfill the inference tasks. As shown in Figure 1, after the retrieving and the inference processes, the original RDF document, the ontologies it used and the inferred information will be indexed in N-Triple format.

4 Topic Generation

The RDF documents can be retrieved from the Web in various ways and our RDF dataset is established using Sindice, a scalable online service, which crawls the documents from the Semantic Web and indexes the resources encountered in each source [8]. Before the snippet generation, *RDF Parser* parses them into RDF statements and generates the RDF graphs. Then the subject and the object in each statement are recognized as RDF nodes and each two nodes are connected by a property. Our first step for generating snippets is to figure out what topic a specific document is mainly referring to. In other words, we should find out the topic node from the RDF graph. The RDF document created based

on *linked data* [11] (e.g., the document from DBpedia [12]) usually contains property *p:primaryTopic* and under this circumstance, the value of this property will be recognized as the topic node. However, when the RDF document does not contain this kind of properties, we should find a generic method to solve this problem. Usually, the centrality of a node in an undirected graph is calculated by counting the number of connections it has. Since RDF graph is a directed graph, a simple and intuitive method for finding out the central node is counting the sum of the in-degree and the out-degree for each node. In [13], Xiang et al. compared five measurements used for automatically summarizing ontologies and the experiments showed that the weighted in-degree centrality measures have the best performance. However, for the case that the target RDF documents not only contain ontologies but also contain a large number of RDF individuals, this measurement usually does not work effectively. Moreover, according to the definition of the inverse property, each property can have its own inverse property. Therefore, for each RDF node, its in-degree and out-degree should be equivalently important. We can learn this well through the following example.

Suppose there is an RDF document mainly referring to person P_A and its corresponding RDF graph is described in Figure 2. In this figure, we can see that person P_A knows person P_B , person P_C and person P_D . Moreover, P_A , P_B , P_C , P_D and P_E are all working at the same firm. Apparently, the occurrence of the node indicating *Firm* is larger than that of the node indicating P_A . However, we can not roughly draw the conclusion that node *Firm* is the topic node of this graph. Actually, this document is mainly referring to person P_A . Therefore, just based on the in-degrees and the out-degrees of nodes, we can not find out the topic node accurately. Here, we present our heuristic method that makes good use of the original URLs of RDF documents to find out the topic nodes effectively. Based on our large number of observations, more than 90% of RDF documents have the topic-related information residing in their URLs. For instance, URL http://dblp.l3s.de/d2r/resource/authors/Tim_Berners-Lee contains the name of the topic-related person *Tim Berners Lee*. Therefore, instead of selecting one node with the max occurrence, we choose several topic-node candidates and compare their URIs to the original URLs of the RDF documents. We regard the candidate with the max similarity as the topic node. Since URIs and URLs are strings and each string can be recognized as a set of characters or digits, we calculate the similarity between string α and string β using the following formula:

$$similarity(\alpha, \beta) = \frac{|\alpha \cap \beta|}{\min(|\alpha|, |\beta|)}$$

Here, $|\alpha|$ and $|\beta|$ denote the length of string α and string β respectively. In other words, the similarity between α and β is the percentage the length of their longest common substring accounts for of the length of the relatively shorter string. In order to alleviate the influence of the nodes with the high similarity but the low occurrence on the performance of our topic-node selection, we give a method to exclude this kind of nodes. Firstly, we find out the node with the max occurrence. Secondly, we calculate the percentage the occurrence of each node accounts for

of this max occurrence. Thirdly, we compare the percentages with a predefined threshold and take off the nodes whose corresponding percentages are lower than this threshold. Finally, we get a set of candidate nodes and then calculate the similarities between each of their URIs and the original URL of the RDF document. The candidate node with the largest similarity will be regarded as the topic node. Indeed, it is not uncommon that an RDF document contains more than one topic node, especially when this document contains a large number of triples. Under this circumstance, taking the efficiency into account, our method will still return one topic node as the clue to users.

We further generate the snippet that contains the topic node. We first calculate the occurrence of each property connected with this node. There are two types of properties: object properties and datatype properties [15]. Suppose the topic node is the subject of the statement, our algorithm will return the types of the corresponding objects using diverse ways for different cases based on the reasoning technique described in Section 3. For the case that the property is an object property, we first try to find out the type of the object defined in this document. If we find it, it is then regarded as the type of the object; otherwise, we further check if the indexed content, which is composed of the triples in the original RDF document, the inferred triples and the triples from the ontologies, has already contained this type definition or other new RDF documents will be retrieved and indexed if necessary. Likewise, for the case that the topic node is the object of the statement, we can also use the above method to find out the type of the corresponding subject.

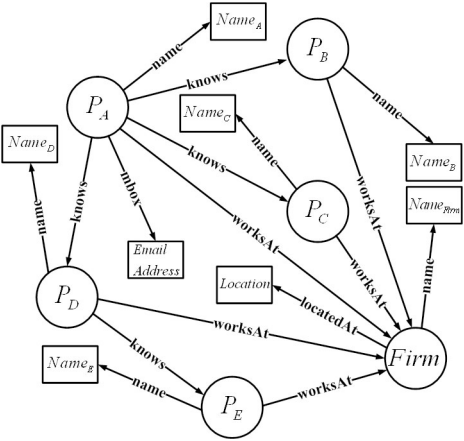


Fig. 2. Graph of an RDF document

5 RDF Statements Ranking Algorithms

In this section, we propose two algorithms for ranking the topic-related statements and the query-related statements respectively.

5.1 Ranking of Topic-Related Statements

Usually, the topic-related statements occupy too many lines in the final snippet. We should find out an automatic method for selecting the most important statements and display them at the top of the snippet. In this subsection, we propose an algorithm for ranking the topic-related statements based on the priorities of the properties, as shown in Algorithm 1.

Algorithm 1. Topic-related statements ranking algorithm

Input: Unranked snippet S and property table T

Output: Snippet S' after ranking

```

begin
    Set all the values in Display column to "SHOW";
    for each row  $r \in T$  do
        add the corresponding URI of  $r$  to pro_uri;
        if  $S$  does not contain pro_uri then
            Set the Display value of the current property to "EXCLUDED";
            continue;
        end
        add the values of the Exclusive column to an array ex_pro_uris and add
        the values of the Relative column to an array re_pro_uris;
        for each string  $epu \in ex\_pro\_uris$  do
            Set the Display value of  $epu$  to "EXCLUDED";
        end
        for each string  $rpu \in re\_pro\_uris$  do
            if the Display value of  $rpu$  / "EXCLUDED" then
                set the Display value of  $rpu$  to "SHOW";
                revise the priority of  $rpu$  based on the current property;
            end
        end
    end
    add the URIs whose corresponding Display value are "SHOW" in  $T$  to an
    array pro_uris;
    sort the values in pro_uris ascendingly;
    for each string  $spu \in pro\_uris$  do
        get the statements containing  $spu$  and add them to  $S'$ ;
    end
end

```

We collect a variety of current widely used RDF schemas and finally find that different properties have different importance degrees in each RDF schema. For instance, in the FOAF [16] file, property *foaf:name* is more important than other properties. Therefore, we set different priorities to different properties. The more important the property is, the higher its corresponding priority is. Here, we also consider the relationships between properties. Generally speaking, there are two types of relationships: *correlative* and *exclusive*. If a specific property appears in the snippet and another one should also appear, these two properties are *correlative*. On the other hand, if a specific property appears and another one

should be hidden in the snippet, these two properties are *exclusive*. For instance, in the FOAF file, *foaf:surname* and *foaf:family_name* are correlative properties. However, *foaf:name* is exclusive to *foaf:surname* and *foaf:family_name* since users do not expect to see duplicated information in the final length-limited snippet. For each property, we define its priorities, correlative properties and exclusive properties, and finally save them into a table.

5.2 Ranking of Query-Related Statements

As a matter of fact, users usually care about the RDF documents mainly referring to the individuals which are associated with the users’ inputs. Therefore, if the topic of a specific RDF document apparently has nothing to do with the user’s input, we should tell him/her the relations between them. Here, we propose a method for ranking the statements in the query-related section. The user’s input is not a node but a string with various formations. For instance, if a user expects to look for the information about *Tim Berners Lee*, he or she may input “TIM BERNERS LEE” or “Tim Berners-Lee”. Here, we use Lucene¹ to split the query phrase into separate words and uniform the user’s input and then select the nodes associated with these inputted words.

Table 1. Statement sequence after ranking

Rank	Statements
1	Query-Related Node + Predicate + Topic Node
2	Topic Node + Predicate + Query-Related Node
3	Query-Related Node + Predicate + Un-Topic Node
4	Un-Topic Node + Predicate + Query-Related Node
5	Topic Node + Predicate + Un-Query-Related Node
6	Un-Query-Related Node + Predicate + Topic Node

We also give the criteria for deciding whether a node is a query-related one or not. If a node denotes an individual and its URI contains the words the user has inputted, the node is a query-related node. If the node denotes a literal (e.g., string) and it contains the words the user inputted, the node is a query-related node. Otherwise, it is not a query-related node at all. After the selection, we list all the statements containing the topic node or the query-related nodes according to the sequence described in Table 1. It is also notable that the statements which have been displayed in the topic-related section will not be displayed in the query-related section any more.

6 Descriptions Generation

An RDF statement contains the subject, the predicate and the object. Sometimes it is not readable since the predicate is usually personally defined for short. In

¹ <http://jakarta.apache.org/lucene>

Algorithm 2. Description generation algorithm

Input: RDF statements *Stats* and pre-indexed content *Con_{indexed}***Output:** discription *Sents* in the snippet

```

begin
  for each statement  $s \in Stats$  do
    create strings sub_part, pred_part and obj_part;
    add the subject of s to Resource subject;
    add the predicate of s to Property predicate;
    add the object of s to RDFNode object;
    if subject's type / null then
      | sub_part = subject's type name + subject's identifier;
    end
    else
      | if subject has not been indexed in Conindexed then
        | retrieve and index the RDF document with the URI of subject;
      end
      else
        | find the type of the subject in the index;
        | sub_part = subject's type name + subject's identifier;
      end
    end
    split the name of predicate and add them to pred_part;
    if object is an instance of Resource then
      generate Resource obj_resource as the copy of object;
      if obj_resource is an individual && obj_resource's type / null then
        | obj_part = object's type name + object's identifier;
      end
      else
        | if object has not been indexed in Conindexed then
          | retrieve and index the RDF document with the URI of object;
        end
        else
          | find the type of the object in the index;
          | obj_part = object's type name + object's identifier;
        end
      end
    end
    add sub_part+“ $\sqcup$ ”+pred_part+“ $\sqcup$ ”+obj_part to Sents;
  end
end

```

this section, we give our method for generating more understandable descriptions from RDF statements by splitting the predicate in a reasonable way.

Based on the analysis gave by Xiantang and Chris [17], we classify the labels of the predicates into two sorts: single-word predicates and multi-word predicates. For the former case, there is no need to split the label further; for the latter case, traditionally, each word will be separated by a specific symbol such as the underline, the horizontal line and so on. Here, we use regular expressions to generate a predicate pattern for splitting the predicates. It is also possible that a predicate does not contain any separators but its first letter is in upper case. Then we split the predicate using the uppercase letters as the separators. We also return the types of the subject and the object to make the final description more understandable. However, maybe the processed RDF document does not contain the type declaration of each individual or literal. Based on the method described in Section 4, we use current indexed content or retrieve another new RDF document to find out the type if necessary. The algorithm for generating description is depicted in Algorithm 2. Usually, the URIs of the resources are long strings and we should not return them directly to users since some URIs do not end with their names (e.g., URIs of anonymous resources). This kind of URIs will make users puzzled if they appear in the final snippets. Here, we use the reasoning techniques described in Section 3 again to find the missed types resources.

7 Personalized Snippet Generation

Considering that users may come from different language regions, in this section, we introduce our method for internationalizing the generated snippets. Based on our snippet generation framework, each snippet will finally fall into 2 parts: the topic-related section and the query-related section. The topic-related section is composed of the topic node and the statements that take the topic node as the subject or the object. The query-related section is composed of all the statements which are displayed according to the sequence described in Table 1. For each of the most popular languages around the world, we create the corresponding template for the unchangeable content in the snippets and use variables to represent the changeable content. For instance, in the topic-related section of the English template, the unchangeable content is “This RDF Document mostly talks about the”. Finally, we replace the variables with the content generated by our predefined multi-language template. It is also notable that Sindice will display the snippets in a specific language the user chooses at the top of the RDF document lists.

Moreover, since the users’ requirements for finally displayed snippets are usually different, we generate the personalized snippets according to their predefined preferences. For instance, users can set the configuration about how many lines should be finally displayed. In our approach, we manage the data and the formation of the snippets separately. Those will be stored in the XML file and the XSLT [18] file respectively.

8 Experiments and Performance Evaluation

Our snippet generation approach has been carried out on a PC with an Intel Core 2 Duo 2.0GHz CPU and 2G of RAM using Java and Ruby on Rails (ROR). The data set is provided by Sindice which currently indexes over 26.69 millions RDF documents. Suppose the inputted querying phrase is “Tim Berners Lee”, Figure 3 shows the snapshot of the generated snippets. From the snippet belonging to RDF document with the URL http://dblp.l3s.de/d2r/resource/authors/Tim_Berners-Lee, we can see that this document is mainly referring to *Tim Berners-Lee*, which is consistent with the querying phrase. Below the topic description, more information about this topic are further displayed based on the predefined priorities of properties. We also use “see all” button to shorten the length of each line. By pushing this button, users can get the whole version of the snippet. Since the querying phrase exactly matches the topic node, the query-related section does not exist in the final snippet of this document.

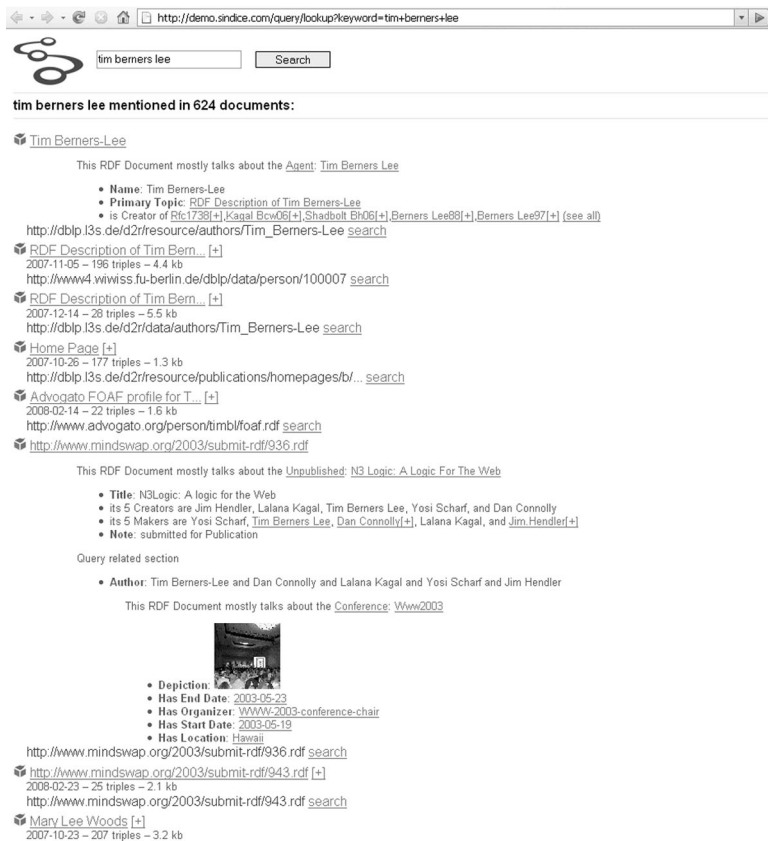


Fig. 3. Snapshot of generated snippets

From the snippet of another RDF document with the URL *http://www.minds-wap.org/2003/submit-rdf/936.rdf*, we can see that the extracted topic node with the type *Unpublished* actually denotes an article, which is apparently not associated with the inputted querying phrase. In this case, the query-related section will be generated according to the sequence described in Subsection 5.2. From this section, we can see that *Tim Berners-Lee* is one of the authors of this article. So users do get extra and helpful information from this section. Moreover, users can recursively get further snippets of the resources which exist in the current snippet by pushing the buttons “[+]” behind the resources.

In order to experiment with our topic node generation algorithm, we input 54 different querying phrases into Sindice using *keyword search*, covering people, locations, Web techniques, marvelous spectacles, organizations and sports, and get totally 1152 indexed RDF documents. These documents contain a variety of RDF formations (e.g., FOAF and SIOC [19]) coming from diverse sources (e.g., Wikipedia [20] and DBpedia). Firstly, we use the max in-degree method, the max out-degree method and the simplex occurrence method to look for the topic node, respectively. Secondly, we use our max occurrence method associated with the original URL to generate the topic node again. Then we ask six domain experts to manually select the topic nodes from the indexed RDF documents with the help of the Tabulator². Assuming the manually-found topic nodes are all correct and reasonable, we compare our method with the aforementioned methods. Finally, we find that the max in-degree method has the lowest accuracy and the two occurrence-based methods both work better than the max out-degree method. Moreover, our original-URL-based method works better when the RDF documents contain a large number of RDF triples. Out of the above 1152 RDF documents, 1072 correct topic nodes are finally generated using our heuristic topic discovery method and the accuracy rate is 93.1%.

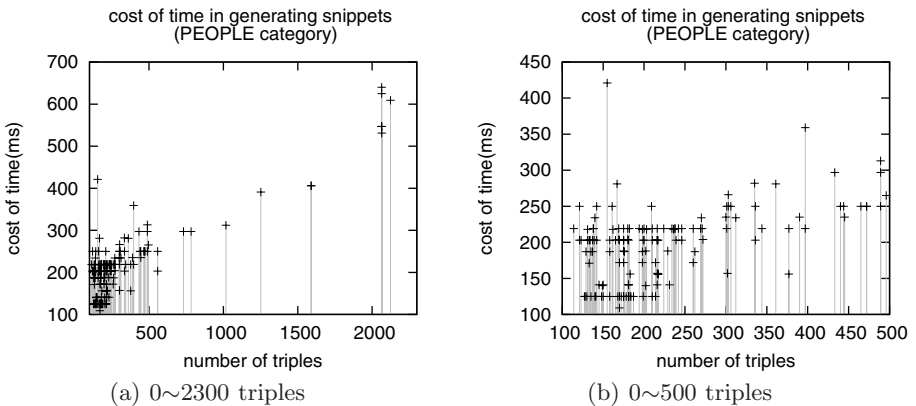


Fig. 4. Cost of time in generating snippets from *People* category

² <http://www.w3.org/2005/ajar/tab>

We also evaluate the performance of our whole snippet generation approach. Within the generation process, for each RDF document, we record the cost of time and the number of triples contained by this document. Figure 4 describes the cost of time in generating snippets from the *People* category. According to Subfigure (a), the cost of time increases with the increasing of the number of triples apparently. Actually, the number of the RDF documents containing less than 500 triples accounts for 89.7% of the total documents. From Subfigure (b), just taking the RDF documents contain less than 500 triples into the consideration, we can see that the time of cost is not associated with the number of triples any more, especially for the documents containing less than 250 triples. Likewise, for other categories, we calculate the percentage the number of less-than-500-triples documents accounts for of the total number of documents and the average costs of time in generating snippets from less-than-500-triples document and all the documents, respectively. The results are shown in Table 2. From this table, we can see that the average cost of time for less-than-500-triples documents is less than 0.2s. So for most of the searched RDF documents, our approach can return their snippets quickly.

Table 2. Experiment results

category	percentage _{<500}	average $COT_{<500}$	average COT_{total}
People	89.7%	198.2 <i>ms</i>	616.1 <i>ms</i>
WebTech	100.0%	186.9 <i>ms</i>	186.9 <i>ms</i>
Organization	99.3%	183.7 <i>ms</i>	201.1 <i>ms</i>
Spectacle	86.3%	198.4 <i>ms</i>	466.3 <i>ms</i>
Location	98.8%	187.9 <i>ms</i>	188.9 <i>ms</i>
Sport	100.0%	193.2 <i>ms</i>	193.2 <i>ms</i>
Average	94.8%	191.0 <i>ms</i>	353.3 <i>ms</i>

9 Related Works

The motivation for summarizing the ontologies is that the current formations for storing ontologies are difficult for non-technician users to understand. Some work for verbalizing ontologies have been done recently.

Wilcock described the on-going work on an ontology verbalizer which combines the Semantic Web techniques with natural language generation and text-to-speech in [1]. Since this verbalization is mainly based on XSLT, it can not deal with all the kinds of RDF documents but focuses on some specific kinds. Kalina et al. proposed a method of automatically generating reports from domain ontologies encoded in OWL using MIAKT generator [2], which takes the medical ontology, RDF description of the case and the MIAKT natural language generation lexicon as the input. Daniel et al. gave an algorithm for using a part-of-speech (POS) tagger, a fast and simple application, to produce concise, accurate natural language paraphrases for OWL concepts [3]. Shumao et al. proposed a Model Driven Integration Architecture (MDIA) to integrate rigorous model specifications and generate context-aware application either semi-automatically

or automatically [4]. Chris et al. resented the evidence that natural language words are used in complex ways in current ontologies and gave their own work using natural language generation to present parts of ontologies [5]. Based on their analysis, Gunther et al. generated a proposal for linguistically determined label generation which benefits the process of mapping OWL concepts to natural language patterns [6].

Fresnel [14] is a display vocabulary for specifying how RDF graphs are presented. However, it is tedious and time consuming since users are required to know how to create the lenses and the formats, which are both the important components for generating the descriptions. In [13], the experiments show that the weighted in-degree centrality measures have the best performance in finding out the most content. However, this measurement usually does not work effectively when the target RDF documents not only contain ontologies but also contain a large number of RDF individuals. To the best of our knowledge, topic-node discovery and RDF statements ranking have not been further researched and still faces difficulties nowadays. Based on our work, users can conveniently get the snippets without any necessary domain knowledge.

10 Conclusions and Future Work

This paper proposes an automatic approach for generating snippets from RDF documents. This approach has been already implemented and integrated into the Sindice demo website³ for users to test our algorithms. Our heuristic topic-discovery method can find out the topic node efficiently and effectively, according to the occurrences of RDF nodes and the original URLs of the RDF documents. Moreover, two ranking algorithms, topic-related-statements ranking and query-related-statements ranking, are presented in order to make the generated snippets more understandable. The use case of our approach is also given in the end and the experimental results indicate the superiority and high efficiency of our approach.

Our long-term goal is to bring certain simple but effective natural language processing techniques into the snippet generation process to improve the readability of the snippets. Besides, some RDF documents probably have multiple topics and it seems more reasonable to list all the most promising topic nodes. So we need a way to rank all the possible topic nodes using a novelty method that occupies relatively less computing resources.

Acknowledgement

We acknowledge the support of the OKKAM ICT Project, Grant Agreement No. 215032. We would like to thank Gabriele Renzi for his work on the maintenance of the snippet generation demo and Michele Catasta, Richard Cyganiak, Holger Stenzhorn and Adam Westerski for their valuable suggestions and efforts on experiments.

³ <http://beta0.sindice.com>

References

1. Wilcock, G.: Talking OWLs: towards an ontology verbalizer. In: Proceedings of Human Language Technology for the Semantic Web and Web Service Workshop at the International Semantic Web Conference, pp. 109–112 (2003)
2. Bontcheva, K., Wilks, Y.: Automatic report generation from ontologies: the MI-AKT approach. In: Proceedings of the International Conference on Applications of Natural Language to Information System, pp. 324–335 (2004)
3. Hewlett, D., Kalyanpur, A., Kolovski, V., Wiener, C.H.: Effective natural language paraphrasing of ontologies on the Semantic Web. In: Proceedings of the End User Semantic Web Interaction Workshop at the International Semantic Web Conference (2005)
4. Ou, S., Georgalas, N., Azmoodeh, M., Yang, K., Sun, X.: A model driven integration architecture for ontology-based context modeling and context-aware application development. In: Rensink, A., Warmer, J. (eds.) ECMDA-FA 2006. LNCS, vol. 4066, pp. 188–197. Springer, Heidelberg (2006)
5. Mellish, C., Sun, X.: The Semantic Web as a linguistic resource: opportunities for natural language generation. *Knowledge Based Systems* 19(5), 298–303 (2006)
6. Fliedl, G., Kop, C., Vöhringer, J.: From OWL class and property labels to human understandable natural language. In: Kedad, Z., Lammari, N., Métails, E., Meziane, F., Rezgui, Y. (eds.) NLDB 2007. LNCS, vol. 4592, pp. 156–167. Springer, Heidelberg (2007)
7. Harth, A., Hogan, A., Delbru, R., Umbrich, J., O’Riain, S., Decker, S.: SWSE: answers before links! In: Proceedings of the 6th Semantic Web Challenge (2007)
8. Tummarello, G., Delbru, R., Oren, E.: Sindice.com: weaving the open linked data. In: Aberer, K., Choi, K.-S., Noy, N., Allemang, D., Lee, K.-I., Nixon, L., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P. (eds.) ASWC 2007 and ISWC 2007. LNCS, vol. 4825, pp. 552–565. Springer, Heidelberg (2007)
9. Oren, E., Delbru, R., Catasta, M., Cyganiak, R., Stenzhorn, H., Tummarello, G.: Sindice.com: a document-oriented lookup index for open linked data. *International Journal of Metadata, Semantics and Ontologies* 3(1) (2008)
10. ter Horst, H.J.: Combining RDF and part of OWL with rules: semantics, decidability, complexity. In: Gil, Y., Motta, E., Benjamins, V.R., Musen, M.A. (eds.) ISWC 2005. LNCS, vol. 3729, pp. 668–684. Springer, Heidelberg (2005)
11. Berners-Lee, T.: Linked data (2006), <http://www.w3.org/DesignIssues/LinkedData.html>
12. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: DBpedia: a nucleus for a Web of open data. In: Aberer, K., Choi, K.-S., Noy, N., Allemang, D., Lee, K.-I., Nixon, L., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P. (eds.) ASWC 2007 and ISWC 2007. LNCS, vol. 4825, pp. 722–735. Springer, Heidelberg (2007)
13. Zhang, X., Cheng, G., Qu, Y.: Ontology summarization based on RDF sentence graph. In: Proceedings of the 16th International Conference on World Wide Web, pp. 707–716. ACM Press, New York (2007)
14. Lee, R.: Introduction to Fresnel: an RDF display vocabulary (2006), <http://dig.csail.mit.edu/2006/Talks/0724-fresnel/>
15. Manola, F., Miller, E.: RDF primer (2004), <http://www.w3.org/TR/REC-rdf-syntax/>

16. Brickley, D., Miller, L.: FOAF Specification (2007), <http://xmlns.com/foaf/0.1>
17. Sun, X., Mellish, C.: An experiment on “free generation” from single RDF triples. In: Proceedings of the 11th European Workshop on Natural Language Generation, pp. 105–108 (2007)
18. Clark, J.: XSL Transformations (XSLT) (1999), <http://www.w3.org/TR/xslt>
19. Berrueta, D., Brickley, D., Decker, S., Fernández, S., Görn, C., Harth, A., Heath, T., Idehen, K., Kjernsmo, K., Miles, A., Passant, A., Polleres, A., Polo, L.: SIOC core ontology specification (2007), <http://rdfs.org/sioc/spec/>
20. Völkel, M., Krötzsch, M., Vrandečić, D., Haller, H., Studer, R.: Semantic Wikipedia. In: Proceedings of the 15th International Conference on World Wide Web, pp. 585–594. ACM Press, New York (2006)

Empirical Insights on a Value of Ontology Quality in Ontology-Driven Web Search

Darijus Strasunskas¹ and Stein L. Tomassen²

¹ Dept. of Industrial Economics and Technology Management

`darijus.strasunskas@iot.ntnu.no`

² Dept. of Computer and Information Science,

Norwegian University of Science and Technology,

NO-7491 Trondheim, Norway

`stein.l.tomassen@idi.ntnu.no`

Abstract. Nowadays ontologies are often used to improve search applications. Quality of ontology plays an important role in these applications. An important body of work exists in both information retrieval evaluation and ontology quality assessment areas. However, there is a lack of task- and scenario-based quality assessment methods. In this paper we discuss a framework to assess fitness of ontology for use in ontology-driven Web search. We define metrics for ontology fitness to particular search tasks and metrics for ontology capability to enhance recall and precision. Further, we discuss results of a preliminary experiment showing applicability of the proposed framework and a value of ontology quality in ontology-driven Web search.

1 Introduction

In this article we investigate the application of ontology to enhance search tasks. Since information quality is critical for organizations, ontologies are being applied in a number of ontology-based information retrieval systems [10], [11], [14] in order to improve the performance of information retrieval (IR) systems.

The literature reports on improvement of search using ontology-based information retrieval (ObIR) tools (e.g., [1], [14]), as well indicates that inexperienced users find ontology helpful in comprehending domain, familiarizing themselves with the terminology and formulating queries [6], [14]. In such cases, visualization of an ontology is a certain quality of ObIR systems, but that concerns graphical user interface, not ontology itself. In addition, it was found that linguistic enhancements (inclusion of synonyms) close the gap between ontology concepts and document text [1], [6], and enable the ontology to perform better for queries that are required to find only a small number of documents. However, there is no systematic investigation on what ontology features enhance or impair search performance.

From another hand, the ontology's ability to capture the content of the universe of discourse at the appropriate level of granularity and precision and offer the application understandable correct information are important features that

are addressed in many ontology quality frameworks (e.g., [2], [9], [15], [23]). However, ontologies are not fixed specifications but always depend on the context of use. There are many different criteria proposed for ontology evaluation (e.g. [15]), but in order “*to be meaningful and relevant, criteria need to be connected to scenarios of use, and these scenarios to be explained and further analyzed need to be connected to activity models*” [16]. Therefore, the evaluation of the ontology also needs to take into account usage scenarios as well as the behavior of the application.

Web search could be characterized by having a focus on retrieving documents rather than browsing knowledge or answering a question. Typically, subclass hierarchies are considered to be sufficient for document retrieval and any further ontology specification (properties and axioms) are required only for knowledge browsing and question answering [19], [24]. In general, ObIR could be characterized either as built on top of a knowledge base or built on top of a vector-space machine (i.e. a conventional search engine). Correspondingly, they have different target, the former has a target to answer questions and browse the knowledge, while the latter is focused on improving large-scale search results and is more important in transition from current Web to the Semantic Web. Therefore, we focus on ontology application in Web search, but consider a whole spectrum of ontology elements. Here we show that ontology quality improvement (by specifying equivalent and disjoint classes, adding instances and properties) can significantly improve Web search results.

The objective of this paper is to analyze the role of ontology quality in ontology-driven Web search applications. This objective is achieved by presenting a framework to assess ontology quality w.r.t. information needs and search tasks. Then we present a preliminary experiment analyzing how the quality of an ontology affects the results of the search. Results of the experiment provides an initial validation of the proposed framework.

The rest of the paper is structured as follows. First we briefly review related work that comes from two main areas, ontology-based information retrieval and ontology quality. Second, we present a revised framework for Evaluation of Ontology Quality for Search (EvOQS) [32]. Then we elaborate on an initial experiment and evaluation of ontology quality aspects and their role in search performance. The main results are presented and discussed. Finally, we conclude the paper and outline future work.

2 Related Work

An increasing number of recent information retrieval systems make use of ontologies to help the users clarify their information needs and expand users’ queries. In this section we provide an overview of related work. First, we analyze information needs and typical search scenarios. Second, we summarize ontology-based information retrieval (ObIR) methods, taking a closer look at what role ontologies play in the methods proposed. Third, we provide a synopsis of state-of-the-art in ontology quality evaluation.

2.1 Information Needs and Search Strategies

There are many studies of users' information needs, their search strategies and behavior (e.g. [3], [21]) resulting in different classification of search strategies. For instance, Guha *et al.* [18] distinguish two different kinds of search, namely, *navigational search* and *research search*. Navigational search is defined as the one where user provides a phrase or keywords and expects to find them in the documents, i.e. the user is using a search engine to navigate to a particular document. While in the research search the user provides a phrase or keywords which are intended to denote object or phenomena about which the user wants to gather information, i.e. the user is trying to locate a collection of documents which will provide required information [18].

Similarly, Rose & Levinson [29] report on three top-level categories of search goals, namely, *navigational*, *informational* and *resource*. Where navigational and informational search goals correspond to the ones identified in [18]. While the last search goal categorizes searches dealing with finding and obtaining a resource, not information, available on Web. Informational and resource search goals are further subdivided into sub-categories [29], such as *locate*, *advice*, *download*, *interact*, etc.

With the emergent Semantic Web there is envisioned a shift in IR from retrieval of appropriate Web pages to answering questions without extraneous information [24]. This being separate and important area in information retrieval and knowledge management that requires robust ontology quality, reasoning and fine-grained annotation of documents. However, a precise question answering is the most ambitious information retrieval task, but still inevitable and required feature of Web search. Therefore, we consider a fact-finding search being able to partially substitute question answering on the Web. For this reason, we adopt a classification of search tasks into the following categories: *fact-finding*, *exploratory*, and *comprehensive* search tasks [3]. In fact-finding, a precise set of results is important, while the amount of retrieved documents is less important. In exploratory search task, the user wants to obtain a general understanding about the search topic, consequently, high precision of the result set is not necessarily the most important thing, nor is high level of recall [3]. Finally, a concern of comprehensive search task is to find as many documents as possible on a given topic, therefore the recall and precision should be as high as possible.

2.2 Ontology-Based Information Retrieval

The basic assumption of ObIR systems is as follows. If a person is interested in information about B, it is likely that she will find information about A interesting, given that A and B are closely related terms/concepts in an ontology. (I.e. ObIR exploits semantic relationships.) In the simplest way, user's query is expanded by hypernyms (superclasses)- i.e. generalization [4] or hyponyms (subclasses) - i.e. focalization (more detailed knowledge) [4] or other related concepts (e.g., sibling concept and other neighborhood concepts). In this way an ontology is used in the process of enriching queries (cf. [5], [11]). There an ontology typically serves as thesaurus containing synonyms, hypernyms/hyponyms.

There are also more sophisticated approaches to ObIR, which we classify as follows.

Knowledge base based ObIR. These approaches use reasoning mechanisms and ontology querying languages to retrieve instances from a knowledge base. There, documents are treated either as instances or are annotated using ontology instances [22], [26], [28], [30], i.e. their focus is on retrieving instances rather than documents. Main disadvantages of these approaches are as follows, use of formal ontology querying languages straitens their adoption by inexperienced users; requires annotation of web resources - the process is tedious and results may be misused by the content providers for the purpose of giving the documents a misleading higher ranking by the search engines [33]. These characteristics make the knowledge base based approaches problematic for a large-scale Web search.

Integrated with vector space model. These approaches combine ObIR with already traditional vector space model. Some start with semantic querying using ontology query languages (e.g. SPARQL, RDQL, OWL-QL) and use resulting instances to retrieve relevant documents using the vector space model [10], [22], [25], [35]. Where Castells *et al.* [10] use weighted annotation when associating documents with ontology instances. The weights are based on the frequency of occurrence of the instances in each document, i.e. term frequency. Nagypal [25] combines ontology usage with the vector space model by extending a non-ontological query. There ontologies are used to disambiguate queries. Simple text search is run on the concepts' labels and users are asked to choose the proper term interpretation.

2.3 Evaluation of Ontology Quality

An important body of work exists in ontology quality assessment area (e.g., [9], [15], [23]). Most of them aim at defining a generic quality evaluation framework and, therefore, do not take into account specific application of ontologies. For instance, the Ontometric [23] methodology defines Reference Ontology that consists of metrics to evaluate ontology, methodology, language and tool (used to develop ontology) - 117 metrics in total. The OntoQA framework [34] is proposed to evaluate ontologies and knowledge bases. There metrics are divided into two categories: schema metrics and instance metrics. The first category of metrics evaluates ontology design and its potential for rich knowledge representation. The second category evaluates the effective usage of the ontology to represent the knowledge modeled in ontology.

Analysis of the literature shows that ontologies are typically examined according to five aspects: syntax, vocabulary, structure, population of classes and usage statistics. Where *evaluation of syntax* checks whether an ontology is syntactically correct. This quality aspect is the most important in any ontology-based application, since syntactic correctness is a prerequisite to be able to process an ontology. Syntactic quality is a central quality aspect in most quality frameworks (e.g., [9], [23]).

Cohesion to domain and vocabulary. Congruence between an ontology and a domain is another important aspect in ontology quality evaluation. There ontology

concepts (including taxonomical relations and properties) are checked against terminology used in the domain. In the OntoKhoj approach [27] ontologies are classified into a directory of topics by extracting textual data from the ontology (i.e. names of concepts and relations). Similarly, Brewster *et al.* [7] extracted a set of relevant domain-specific terms from documents. The amount of overlap between the domain-specific terms and the terms appearing in the ontology is then used to measure the fit between the ontology and the corpus. Similar lexical approach is taken in EvaLexon [31] where recall/precision type metrics are used to evaluate how well ontology triples were extracted from a corpus. Burton-Jones *et al.* [9] define a metric called *accuracy* that is measured as a percentage of false statements in an ontology.

Structural evaluation. Structural evaluation deals with assessment of taxonomical relations vs. other semantic relations, i.e. the ratio of **IsA** relationships and other semantic relationships in an ontology is evaluated. Presence of various semantic relationships would identify the richness of ontology. In OntoSelect [8] a metric, called *structure*, is used. The value of the structure measure is simply the number of properties relative to the number of classes in the ontology. Similarly, *density measure* defined in [2] indicates how well a given concept is defined in the ontology. While *relationship richness* [34] reflects the diversity of relations and placement of relations in the ontology.

Population of classes. This quality aspect is based on instance related metrics. Tartir *et al.* [34] define *class richness* that measures how instances are distributed across classes. The amount of classes having instances is compared with the overall number of classes. *Average population* [34] indicates the number of instances compared to the number of classes. It is used to determine how well a knowledge base has been populated.

Usage statistics and metadata. Evaluation of this aspect focuses on the level of annotation of ontologies, i.e. the metadata about an ontology and its elements. There are defined three basic levels of usability profiling in [15] as follows. *Recognition annotations* take care of user-satisfaction, provenance and version information; *efficiency annotations* deal with application-history information; and the last level is about *organizational-design information*. Burton-Jones *et al.* [9] define similar metrics, namely, *relevance* assesses the amount of statements that involve syntactic features marked as useful or acceptable to the user/agent; *history* accounts for how many times a particular ontology has been accessed relatively to other ontologies. Furthermore, the Swoogle approach [13] ranks retrieved ontologies based on references between them. Analogical metric to Swoogle's is defined in [9] and is called *authority* - i.e. how many other ontologies use concepts from this ontology. Hartmann *et al.* [20] extend the above discussed approaches to ontology metadata by proposing a systematic vocabulary for ontology metadata (OMV) and presenting its application in the Oyster P2P system for exchanging ontology metadata among communities. d'Aquin *et al.* [12] present the Watson system that can be considered an advanced version of Swoogle by extending knowledge characterization beyond **import** link between ontologies.

Table 1. Summary of existing approaches to ontology evaluation

Quality framework	Syntax evaluation	Domain cohesion	Structural evaluation	Population of classes	Usage statistics
AKTiveRank [2]		X	X		
OntoClean [17]			X		
OntoKhoj [27]		X			X
Ontometric [23]	X				
OntoQA [34]			X	X	
OntoSelect [8]			X		
oQval [15]		X			X
Semiotic metrics [9]	X	X			X
Swoogle [13]					X
<i>Other</i>		[7],[31]			

Table 1 summarizes ontology evaluation approaches with respect to the five aspects discussed above. In summary, cohesion to domain terminology, measured as a direct match of the vocabulary used to denote concepts in the ontology with a terminology used in text corpora, has positive impact on overall ObIR performance. Lexical fit allows better adoption of an ontology, both from user and document collection perspectives. However, that is not vital for every single approach to ObIR. For instance, an approach by Tomassen [35] aligns terminologies of a document collection with the concepts of an ontology by the help of a feature vector constructed for each of the concepts. Evaluation of a structural aspect determines the richness of ontology, and, therefore, is important for KB and vector-space model based ObIR.

Consequently, some of the above discussed metrics and criteria are applicable and feasible to assess capability of ontologies to enhance information retrieval. However, there is a lack of a systematic framework to assess fitness of ontologies for a particular search strategy and/or ObIR approach. Adequate optimality criteria should be selected to enable quality estimation of ObIR. These measures should be related to the users' information needs.

3 A Framework for Evaluation of Ontology Value in Search Applications

In this section we present the EvOQS (Evaluation of Ontology Quality for Searching) framework including functional steps and assessment criteria as defined in Figure 1. It consists of three steps as follows.

Step 1. Generic quality evaluation. This initial step concerns filtering out poor quality (i.e. syntactically incorrect) and irrelevant ontologies. More detail account on this step is provided in subsection 3.1.

Step 2. Search task fitness. This step concerns evaluation of ontology fitness for a particular search task. Typical search tasks were discussed in section 2.1. For instance, ratio of taxonomic vs. non-taxonomic relationships is

① <i>Generic quality evaluation</i>		
Syntactical correctness		Domain fitness
② <i>Search task fitness</i>		
Fact-finding	Exploratory	Comprehensive
③ <i>Search enhancement capability</i>		
Recall Enhancement		Precision Enhancement

Fig. 1. The EvOQS framework for ontology fitness in information retrieval

important when selecting an appropriate ontology for exploratory and comprehensive search tasks. For more detail the reader is referred to subsection 3.2.

Step 3. Search enhancement capability. This final step in our framework concerns evaluating vocabulary of ontologies. Here we account for availability of internal lexical resources in ontologies, i.e. presence of specified synonyms, alternative labels that might potentially be used for a query expansion. The step is supplementary to the second step and is designated for further selection of an ontology based on desired enhancement of search performance. More detail account on this step is given in subsection 3.3.

3.1 Generic Quality Evaluation

This step evaluates syntactic correctness and domain fitness. For the syntactic correctness we define a trivial measure (Eq. 1).

$$SC = \lambda \frac{1}{|E|}. \quad (1)$$

Where, E is the number of error messages generated by a parser, and $\lambda \in \Lambda$ and Λ is a set of OWL sub-language¹ preference weights, i.e. $\Lambda = \{0.0; 0.5; 1.0\}$. For instance, based on a particular implementation of ObIR, OWL DL might be a preferable ontology language, though OWL Lite would be a second choice. Correspondingly, an ontology in OWL DL would be given a preference weight $\lambda=1.0$; OWL Lite, $\lambda=0.5$; and OWL FULL, $\lambda=0.0$. Furthermore, these coefficients can be related to a particular search task. For instance, an ontology specification in a form of subject hierarchy/taxonomy is enough to support an exploratory search task (for more details, see next subsection), therefore, an ontology specified in OWL Lite is appropriate for this task. While for the domain fitness sub-step we adopt the AKTiveRank algorithm [2], discussed in subsection 2.3.

3.2 Search Task Fitness

We have identified three typical search tasks in section 2.1. Here we discuss what ontology features are needed to support these tasks.

¹ <http://www.w3.org/TR/2004/REC-owl-guide-20040210/#OwlVarieties>

Fact-finding. Here, high precision can be achieved by using precise terms or phrases in a query, and typically, by formulating a query consisting of several terms. In order to enhance results in fact-finding search task, provided concepts need to be extended by their instances. Consequently, concepts, their instances and properties are essential here.

Exploratory search. Here, the user may find topic-related documents by extending simple keyword-based search with subclass concepts.

Comprehensive search. In order to cover broader-topic hypernyms, sibling concepts and semantic relationships are included in the query (in addition to hyponyms), to cover the most important aspects of the search topic.

In Table 2 we summarize ontology support necessary to support search tasks as discussed.

Table 2. Search tasks and ontology support

Search tasks	Ontology support	OWL constructs
Fact-finding	Concepts, their instances, object and datatype properties (at instance level)	<code>owl:Class</code> , <code>rdfs:subClassOf</code> , <code>owl:Thing</code> , <code>owl:ObjectProperty</code> , <code>owl:DatatypeProperty</code> , <code>owl:FunctionalProperty</code>
Exploratory	Sub-concepts	<code>rdf:subClass</code>
Comprehensive	Super- and sub-concepts, sibling-concepts, object properties	<code>owl:Class</code> , <code>rdfs:subClassOf</code> , <code>owl:ObjectProperty</code>

Based on above discussion we define metrics to measure ontology fitness for a particular search task. The metrics are defined for a cluster of concepts (i.e. a fragment of ontology). Values computed for a cluster can be used to assess a particular query (i.e. concepts used to formulate a query), or a notion of cluster can be extended to a whole ontology. Consequently, evaluation of an ontology would reveal a general fitness of an ontology for a particular search task. While computed metrics for a cluster would allow analyzing an ontology-based query more rigid with regards to its impact on search results. In other words, ontology evaluation should be used to pre-select existing ontologies, while a cluster (query) evaluation is useful for a thorough analysis of search results.

We define a cluster being a set of concepts of interest (e.g., used in query to specify information needs). In evaluation of the cluster we investigate the level of domain knowledge specified about a concept in the cluster, i.e. direct relationships (object and datatype properties, super- and sub-class relations) and associated instances. Namely, corresponding to Table 2, we define a coefficient for cluster's fact finding fitness (FFF):

$$FFF_{cl} = \alpha \frac{|I_{cl}|}{|C_{cl}|} + \beta \frac{|OP_{cl}| + |DP_{cl}|}{|C_{cl}|}. \quad (2)$$

Where, I is the number of instances associated with concepts in a cluster (cl), OP and DP are OWL constructs `owl:ObjectProperty` and `owl:DatatypeProperty`, correspondingly. Here α, β are adjustment weights. Their purpose is discussed later.

Fitness of ontology for exploratory search task is defined as an arithmetic average of subclass concepts associated with the concepts in a cluster under evaluation. An exploratory search task fitness (EXF) is defined in Eq. 3.

$$EXF_{cl} = \frac{|SubC_{cl}|}{|C_{cl}|}. \quad (3)$$

Where, $SubC$ is the number of subclasses specified for concepts in a cluster (cl) or, eventually, defined in an ontology.

Finally, ontology fitness for comprehensive search task is defined by the comprehensive search task fitness (COF) coefficient in Eq. 4.

$$COF_{cl} = \frac{\beta|OP_{cl}| + \alpha(|SupC_{cl}| + |SubC_{cl}| + |SibC_{cl}|)}{|C_{cl}|}. \quad (4)$$

Where, C is the number of concepts in a cluster (cl), as above. OP is the number of object properties for the concepts in the cluster, and $SupC$, $SubC$ and $SibC$ are amount of super-, sub- and sibling concepts for a particular concept, respectively.

3.3 Search Enhancement Capability

In order to improve the result of search, query expansion is typically used, where a query is refined to improve both, recall and precision. Table 3 summarizes a role of main ontology elements (and corresponding OWL constructs) in query expansion. Our aim is to define metrics to assess capability of ontologies to provide lexical resources for enhancement of precision and recall. As it was mentioned above, ontology lexicon improves recall. Ontology lexicon is a set of lexical entries for the concepts of ontology (synonyms). Each concept is represented by one or more lexical entries that are extracted from the concept name, and synonyms specified by the `rdfs:label` construct.

Table 3. OWL language constructs relevance for IR performance

Search enhancement	Ontology support	OWL constructs
Precision	related concepts	<code>owl:intersectionOf</code> , <code>owl:unionOf</code>
	disjoint concepts (<i>to be used with boolean operator NOT</i>)	<code>owl:complementOf</code> , <code>owl:disjointWith</code>
	properties	<code>owl:ObjectProperty</code> , <code>owl:DatatypeProperty</code> , <code>rdfs:subPropertyOf</code>
	instances (<i>w/ boolean operator NOT</i>)	<code>owl:differentFrom</code>
Recall	instances	<code>owl:sameAs</code>
	synonyms	<code>owl:equivalentClass</code> , <code>rdfs:label</code>
	related concepts	<code>owl:intersectionOf</code> , <code>owl:unionOf</code>

Therefore, we define a recall enhancement capability (REC) that shows average amount of synonyms and related terms specified for concepts in a cluster (or ontology) (see Eq. 5).

$$REC_{cl} = \alpha \frac{|L_{cl}| + |eC_{cl}|}{|C_{cl}|} + \beta \frac{|uO_{cl}| + |iO_{cl}|}{|C_{cl}|}. \quad (5)$$

Where, $L=\text{rdfs:label}$, $eC=\text{owl:equivalentClass} + \text{owl:sameAs}$, $iO=\text{owl:intersectionOf}$, $C=\text{owl:Class}$, $uO=\text{owl:unionOf}$, and α, β are adjustment weights.

A precision enhancement capability (PEC) is defined based on OWL constructs provided in Table 3 as follows (see Eq. 6).

$$PEC_{cl} = \alpha \frac{|cO_{cl}| + |dW_{cl}| + |uO_{cl}| + |iO_{cl}|}{|C_{cl}|} + \beta \frac{|OP_{cl}| + |DP_{cl}|}{|C_{cl}|}. \quad (6)$$

Where, $iO=\text{owl:intersectionOf}$, $cO=\text{owl:complementOf}$, $uO=\text{owl:unionOf}$, $C=\text{owl:Class}$, $dW=\text{owl:disjointWith}$, and α, β are adjustment weights.

However, applicability of the above defined metrics depends a lot on a particular implementation of ObIR. Therefore, we include adjustment weights² ($\alpha + \beta=1$) to tailor metrics (by specifying preferable OWL constructs) to a particular implementation. For instance, there are ObIR systems exploiting only subclass hierarchy and not taking processing properties, then α value could be set to 1 with $\beta=0$. Furthermore, all coefficients are normalized to fall into range [0..1].

A prototype of the EvOQS framework has been implemented in Java using the OWL API³. The prototype has been used to compute metrics in an experiment described next.

4 Experiment

For the assessment of ontology quality role in an ontology-driven search application we have conducted an experiment with four different ontologies (different domains) and two different versions of each of the ontologies. The experimental settings are detailed as follows.

4.1 Web Information Search Tool

For the experiment the WebOdIR system⁴ (see figure 2) was used that is an ontology-driven information retrieval system for the Web [35]. An advantage with WebOdIR is that the users can specify one or more concepts being related to a domain of interest when formulating a query and hence bringing the query closer to the real intention of the user's query. In addition, it is possible to specify

² See, equations 2, 4, 5 and 6.

³ <http://owlapi.sourceforge.net/>

⁴ WebOdIR prototype, <http://129.241.110.220>.

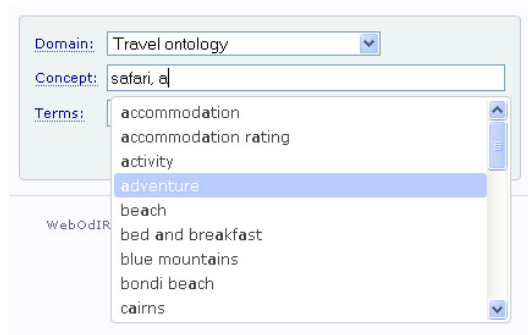


Fig. 2. A user interface of the first WebOdIR prototype

a set of keywords to narrow the search even further. The user interface was also meant to be as simple and familiar as possible, consequently the interface is similar to many of the interfaces of commonly used search engines found on the Web. As backend search engine the prototype used the Yahoo! Web Search API⁵.

WebOdIR uses ontologies extensively, both when constructing a feature vector and in the search process. One or more ontologies specify sets of concepts for the domains of interest. Next, each concept is extended with a feature vector (fv) that adapts the concept to the terminology used in a particular domain (a document collection). The process of constructing feature vectors constitutes three main steps, which is done offline and prior to the search process. The aim of the first step is to do some preparation to optimize the construction of the feature vectors for each of the ontology concepts and mainly includes ranking of every concept in accordance to perceived relevancy to the ontology. In the next two steps, the highest ranked concept is processed first and then later used to construct the feature vector of the next ranked concept and so forth. The aim of the second step is to extract sets of candidate terms being relevant to each concept. Before the terms can be extracted, a query is formulated and submitted to the underlying search engine for each concept. The queries are formulated by considering how the concepts relate to each neighboring concepts. A result of this is a set of retrieved documents for each concept. Next, the documents are clustered to group those documents having high similarity. For each cluster a set of candidate terms are extracted. At this stage the candidate terms are not necessarily relevant to the domain of interest defined by the ontology. Consequently, the aim of the third and last step is to identify those candidate terms being relevant to the current ontology. The similarity with the neighboring concepts' clusters for each cluster of the concepts is calculated. In this process different weighting is used to differentiate on the importance of the relation types. Finally, the cluster for each concept with the highest similarity is selected (assumed most relevant to the domain of interest specified by the ontology) and next used when creating

⁵ Yahoo! Developer Network, <http://developer.yahoo.com/search/web/>.

feature vectors (for more details see [35]). In the current implementation, both concepts and instances are available for users to express their queries.

In a search process an ontology is firstly used to help a user formulating a query. Then the specified query concepts are used by the system to formulate one or more new queries. These new queries are next sent to the underlying search engine. Currently Yahoo! and Nutch are supported, but any search engine can be used. The new query is based on how the current concept relates to other neighboring concepts (e.g. for an exploratory search the hypernyms of the concept are typically included). Finally, the concepts are used to re-rank and filter out those documents retrieved by the underlying search engine considered to be of no or little relevance to the current domain.

4.2 Experiment Settings and Materials

The participants in our experiment were mainly 4th year students from Dept. of Computer and Information Science. There were 21 subjects that participated; they were offered payment for used time after full completion of the experiment. The experiment consisted of two parts. The first part included formulating search queries for both WebOdIR and Yahoo!. The participants were presented four domains with two topics of interest for each domain (see Table 4). They had to formulate in total 16 queries, eight to be submitted to WebOdIR and eight to Yahoo!. The participants were divided into two groups that used different ontologies for the same domain. The first group used the original ontology, while the second group used an enhanced version of the original ontology. The enhanced ontologies contained more relations and/or instances to see if this would influence the search results. Different feature vectors were generated as the result of modifications in ontologies. Group 1 contained 10 participants, while group 2 had 11 participants. In total, users executed 81 queries using the original ontologies and 92 queries using the modified ontologies, and 152 queries were simple keyword based executed directly to Yahoo!. However, in this paper we focus on search performance analysis using different quality of ontologies, therefore only ontology-based search is analyzed further. The keen reader is directed to [35] for comparison of Yahoo! and WebOdIR.

The participants needed to mark each of top 10 retrieved documents according to perceived relevance. The relevance score for each query has been calculated using the following equation [6]:

$$Score_q = \frac{1}{2} \sum_{i=1}^{10} (P_{Di} \times P_{Pi}). \quad (7)$$

where P_{Di} is an individual score for document D_i , and P_{Pi} - the weighting factor for position P_i . Score for document is as follows: -1 for trash; 0 for non-relevant or duplicate; 1 - related; and 2 - good document. Document ranking position has weights as follows: 1st - 20; 2nd - 15; 3rd - 13; 4th - 11; 5th - 9; 6th & 7th - 8; 8th & 9th - 6; 10th - 4. Consequently, the final score falls into a range [-50, 100].

Table 4. Search topics (domain) and tasks

Search topic id	Information needs and task description
<i>Food & Wine domain</i> (http://www.w3.org/2001/sw/WebOnt/guide-src/wine.owl and integrated with http://www.w3.org/2001/sw/WebOnt/guide-src/food.owl)	
1.	Explorative search task. Imagine that you are going to prepare a dinner for tonight. You plan to make beef curry and would like some wine to drink with this meal. Find out what grapes are used for suitable wines to this meal.
2.	Fact-Finding search task. Imagine that you are going to prepare a dessert as well. The main component of this dessert is chocolate but also contains some sweet fruits. You would like to find the perfect dessert wine but don't know which, try to find it.
<i>Travel domain</i> (http://protege.cim3.net/file/pub/ontologies/travel/travel.owl)	
3.	Comprehensive search task. Imagine that you are going on a vacation and would like to try a safari. You don't know yet which country or what kind of safaris you would like. Try to get an overview of the kind of safaris that are available.
4.	Fact-Finding search task. Suppose that you would like to see leopards and have decided to go on a leopard safari but don't know where. Explore the possibilities for a leopard safari.
<i>Animal domain</i> (http://nlp.shef.ac.uk/abraxas/ontologies/animals.owl)	
5.	Explorative search task. Imagine that you should write an article about jaguars but don't know very much about jaguars. Try to find some facts about jaguars.
6.	Comprehensive search task. Imagine that you would also like to write an article about jaguars and leopards and similar kind of cats. Try to get an overview of the cat family.
<i>Autos domain</i> (http://gaia.isti.cnr.it/~straccia/download/teaching/SI/2006/Autos.owl)	
7.	Fact-Finding search task. Imagine that you have heard that the neighbor has bought a new car of the brand Saturn. Further, imagine that you have never heard of this brand before. Try to find some facts about this brand.
8.	Comprehensive search task. Suppose your neighbor has recently bought a beautiful new car. Therefore, you would like to impress your neighbor as well buy getting a bigger car, an SUV. However, you do not know much about cars; try to get an overview of what SUVs are.

The relevance score substitutes a conventional precision metric. We have decided to focus on precision instead of recall since we targeted Web search, where precision (i.e. relevant documents at top positions) is more important than recall. Consequently, we focus to validate the metrics defined in the second and partially the third steps of the EvOQS framework during this experiment. The first step (generic quality evaluation) has been conducted manually when preparing for the experiment and, furthermore, is outside of the scope of this paper.

5 Results

All four ontologies were modified by adding instances (all ontologies), specifying additional object properties (travel, animal and wine ontologies) and introducing equivalent classes (animal and autos ontologies). Difference in ontology fitness metrics and precision enhancement capability is displayed in Table 5^{6,7,8}. Consequently, comparing relevance scores for the original ontologies vs. the modified ones, we have found an improvement in mean score that equals to 10.6% (overall mean relevance for original ontologies score 42.1 vs. 46.6 for modified ontologies), see Table 6 and Figure 3 for comparison per search topic.

Table 5. Normalized values of the EvOQS metrics for search queries

Topic ID	1		2		3		4		5		6		7		8	
Task cat	Explor.		FactFind.		Compreh.		FactFind.		Explor.		Compreh.		FactFind.		Compreh.	
Ontology version	v.1	v.2	v.1	v.2	v.1	v.2	v.1	v.2	v.1	v.2	v.1	v.2	v.1	v.2	v.1	v.2
FFF/EXF/COF	0.36	0.65	0.16	0.29	0.39	0.29	0.00	0.31	0.08	0.58	0.52	1.00	0.56	0.60	0.72	0.70
diff.		0.28		0.12		-0.10		0.31		0.50		0.48		0.04		-0.02
PEC	0.29	0.53	0.12	0.18	0.03	0.00	0.00	0.00	0.07	0.46	0.00	0.20	1.00	0.28	0.33	0.29
diff.		0.24		0.06		-0.03		0.00		0.40		0.20		-0.72		-0.05

As we can see from Figure 3, the changes in ontology has resulted in difference of the corresponding metrics. Decrease of metrics value is attributed to the heterogeneous queries specified by users (i.e. different concepts used). Increase/decrease of ontology quality (i.e., quality of a concept clusters used in search) had a corresponding effect on the search results, with exception of topic 4. This can also be attributed to a variance between user perception of what is a relevant document, since no explicit instance "leopard safari" has been added, just indirect instances as "Africa big 5 safari" etc., therefore some results could have been perceived as irrelevant. Consequently, this caused bigger variance between participants, i.e. using ontology version 1 mean score was 71.4, st.dev. 14.7, variation coefficient 21%, while usage of ontology version 2 resulted in a mean score of 63.0, st.dev 25.2 and coefficient of variance 40%. When comparing the relevance score w.r.t. search task types, the least increase in search performance has been observed in a comprehensive search tasks (topics 3, 6 & 8). There performance has decreased on topics 3 and 8, and only because of dramatical increase of result in topic 6, the overall improvement has been achieved. This can be attributed to the significantly shorter concept-based queries used by participants in Group 2, i.e. 17% and 39% shorter queries, respectively in the topic 3 and 8.

In general, inclusion of more instances and object properties has improved the mean relevance score of fact-finding search tasks, while addition of disjoint and

⁶ 4th row in the table contains values of a fitness measure corresponding to the search task.
⁷ Values are normalized and computed as an average value of concept-based search queries.
⁸ α and β values have been set to 0.5.

Table 6. Average scores depending on ontology version

	v.1	v.2	Diff. (%)
Animals	19.4	38.0	96.6%
Autos	32.9	33.7	2.2%
Travel	71.8	65.2	-9.1%
Wine&Food	42.9	51.8	20.6%
Overall	42.1	46.6	10.7%

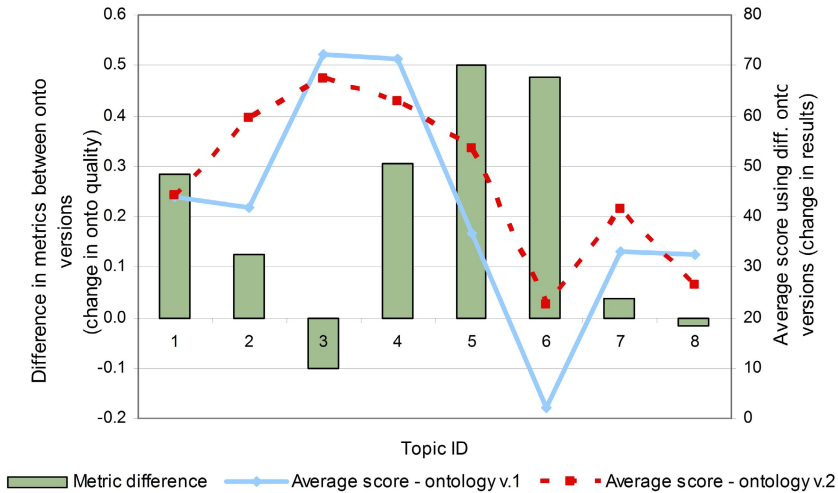
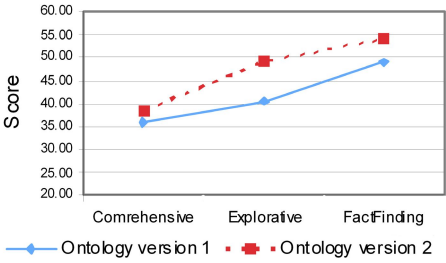


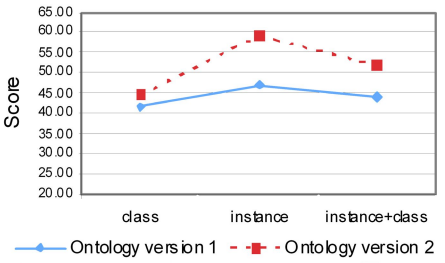
Fig. 3. Comparison of changes in ontology quality and search performance

equivalent concepts resulted in better performance of exploratory and comprehensive tasks (see Figure 4 a)). Further, the a) chart can be related to the b) chart that shows the performance based on what type ontological information has been used in queries, i.e. instance or concept. Since the specified individuals in ontologies had concrete object and datatype properties, that resulted in better document selection. Chart c) visualizes results based on amount of concepts/instances used in queries. The concepts in the modified ontologies contained more precise knowledge specified about them (e.g. disjoint subclass relations), that helped to better discriminate the retrieved documents and improve the mean of relevance scores.

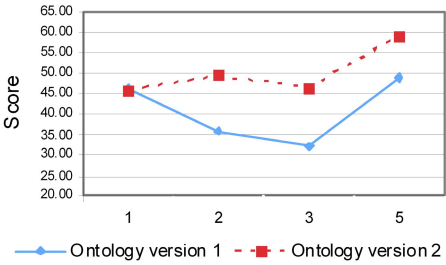
Further analysis has been conducted using an average path length, measured as a distance between the concepts specified in queries (counting edges between concepts), with the purpose to measure semantic distance between concepts provided in queries and its impact on the result. Here the hypothesis is that closer located concepts have overlapping feature vectors. From Figure 4 d) we can see that "broader" clusters (i.e. longer distance between concepts) better discriminated the retrieved documents. This suggests future improvement of a feature vector construction algorithm including even outer (indirect) neighbors of a concept when computing its feature vector.



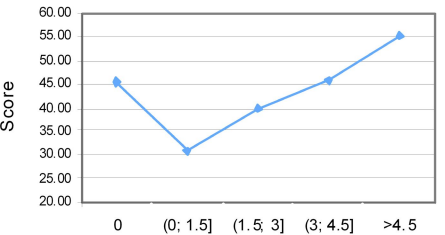
a) Mean score per search task type



b) Mean score per concept-based query type



c) Mean score per amount of concepts in query



d) Mean score based on path length among concepts specified in queries

Fig. 4. Main results of ontology quality impact on search performance

6 Conclusions and Future Work

Ontologies are intensively used to improve information retrieval. However, the outputs of the application and its performance in a given task, might be better or worse partly depending on ontologies used. Therefore, evaluation criteria need to be connected to scenarios of use with a purpose to enhance particular search tasks. In this article we have proposed the EvOQS framework to assess ontology fitness and capability to improve ontology-based search. The framework consists of three functional steps that guide in selecting appropriate ontology for a particular search task. In summary, first step filters out syntactically incorrect and irrelevant ontologies. Second step classifies ontologies according their fitness for a particular search task. Whereas the last step classifies ontologies based on their characteristics to enhance recall and precision. The framework is meant to be used either to evaluate a general fitness of an ontology for a particular search task and its capability to enhance search performance, or assess a particular query (i.e. concepts used to formulate a query) in order to analyze the search results.

In this article we have discussed preliminary results of an experiment showing how different ontology quality aspects can improve ontology-driven Web search

performance. The results proof the applicability of the proposed framework. However, the results of the experiment need to be further analyzed. Furthermore a more controlled experiment should be conducted, since in the current experiment we have allowed the participants to interpret the task and freely construct the query and once again to interpret the relevance of retrieved documents. Consequently, we have found a significant variance between users' assessments. Therefore, the scope of the experiment should be extended. Moreover, the experiment relied on just one ontology-based search system [35]. However, in order to formally validate efficacy of the proposed metrics, more semantic search systems should be included in the future experiments.

Acknowledgments. This work is partially financed by the IO-Evaluation project, financed by the iO (the Center for Integrated Operations in the Petroleum Industry, <http://www.ntnu.no/iocenter>), and by the IIP project (Integrated Information Platform for reservoir and subsea production systems), financed by Norwegian Research Council (NFR # 163457/S30).

References

1. Aitken, S., Reid, S.: Evaluation of an ontology-based information retrieval tool. In: Gomez-Perez, A., et al. (eds.) Workshop on the Applications of Ontologies and Problem-Solving Methods, ECAI 2000, Berlin (2000)
2. Alani, H., Brewster, C., Shadbolt, N.: Ranking ontologies with AKTiveRank. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L.M. (eds.) ISWC 2006. LNCS, vol. 4273, pp. 1–15. Springer, Heidelberg (2006)
3. Aula, A.: Query formulation in Web information search. In: Proceedings of IADIS International Conference WWW/Internet, IADIS, pp. 403–410 (2003)
4. Bonino, D., Corno, F., Farinetti, L., Bosca, A.: Ontology driven semantic search. WSEAS Transaction on Information Science and Application 1(6), 1597–1605 (2004)
5. Braga, R., Werner, C., Mattoso, M.: Using ontologies for domain information retrieval. In: Proceedings of the 11th International Workshop on Database and Expert Systems Applications, pp. 836–840. IEEE Computer Society, Los Alamitos (2000)
6. Brasethvik, T.: Conceptual modelling for domain specific document description and retrieval- An approach to semantic document modelling. Ph.D thesis, NTNU, Trondheim, Norway (2004)
7. Brewster, C., Alani, H., Dasmahapatra, S., Wilks, Y.: Data driven ontology evaluation. In: International Conference on Language Resources and Evaluation, Lisbon, Portugal (2004)
8. Buitelaar, P., Eigner, T., Declerck, T.: OntoSelect: A dynamic ontology library with support for ontology selection. In: Proceedings of the Demo Session at ISWC 2004, Hiroshima, Japan (2004)
9. Burton-Jones, A., Storey, V., Sugumaran, V., Ahluwalia, P.: A semiotic metrics suite for assessing the quality of ontologies. Data and Knowledge Engineering 55(1), 84–102 (2005)

10. Castells, P., Fernandez, M., Vallet, D.: An adaptation of the vector-space model for ontology-based information retrieval. *IEEE Transactions on Knowledge and Data Engineering* 19(2), 261–272 (2007)
11. Ciorascu, C., Ciorascu, I., Stoffel, K.: knOWler - ontological support for information retrieval systems. In: *SIGIR 2003 Conference, Workshop on Semantic Web*, Toronto, Canada (2003)
12. d'Aquin, M., Baldassarre, C., Gridinoc, L., Sabou, M., Angeletou, S., Motta, E.: Watson: Supporting next generation semantic web applications. In: *Proceedings of the IADIS International Conference WWW/Internet* (2007)
13. Ding, L., Finin, T., Joshi, A., Pan, R., Cost, R.S., Peng, Y., Reddivari, P., Doshi, V., Sachs, J.: Swoogle: A search and metadata engine for the Semantic WSeb. In: *Proceedings of CIKM 2004*, pp. 652–659. ACM Press, New York (2004)
14. Suomela, S., Kekalainen, J.: Ontology as a search-tool: A study of real user's query formulation with and without conceptual support. In: Losada, D., Fernandez-Luna, J. (eds.) *ECIR 2005. LNCS*, vol. 3408, pp. 315–329. Springer, Heidelberg (2005)
15. Gangemi, A., Catenacci, C., Ciaramita, M., Lehmann, J.: Modelling ontology evaluation and validation. In: Sure, Y., Domingue, J. (eds.) *ESWC 2006. LNCS*, vol. 4011, pp. 140–154. Springer, Heidelberg (2006)
16. Giboin, A., Gandon, F., Corby, O., Dieng, R.: Assessment of ontology-based tools: a step towards systemizing the scenario approach. In: *EON 2002 workshop* (2002)
17. Guarino, N., Welty, C.: An Overview of OntoClean. In: *Handbook on Ontologies*, pp. 151–172. Springer, Heidelberg (2004)
18. Guha, R., McCool, R., Miller, E.: Semantic search. In: *Proceedings of WWW 2003*, pp. 700–709. ACM Press, New York (2003)
19. Gulla, J., Borch, H., Ingvaldsen, J.: Ontology learning for search applications. In: Meersman, R., Tari, Z. (eds.) *OTM 2007, Part I. LNCS*, vol. 4803, pp. 1050–1062. Springer, Heidelberg (2007)
20. Hartmann, J., Palma, R., Sure, Y., Suarez-Figueroa, M., Haase, P., Gómez-Pérez, A., Studer, R.: Ontology metadata vocabulary and applications. In: Meersman, R., et al. (eds.) *OTM-WS 2005. LNCS*, vol. 3762, pp. 906–915. Springer, Heidelberg (2005)
21. Jansen, B., Spink, A., Saracevic, T.: Real life, real users, and real needs: A study and analysis of user queries on the Web. *Information Processing and Management* 36(2), 207–227 (2000)
22. Kiryakov, A., Popov, B., Terziev, I., Manov, D., Ognyanoff, D.: Semantic annotation, indexing, and retrieval. *Journal of Web Semantics* 2(1), 49–79 (2004)
23. Lozano-Tello, A., Gomez-Perez, A.: Ontometric: A method to choose appropriate ontology. *Journal of Database Management* 15(2), 1–18 (2004)
24. McGuinness, D.: Question answering on the Semantic Web. *IEEE Intelligent Systems* 19(1), 82–85 (2004)
25. Nagypal, G.: Possibly imperfect ontologies for effective information retrieval. Ph.D thesis, University of Karlsruhe (2007)
26. Paralic, J., Kostial, I.: Ontology-based information retrieval. In: *Proceedings of the 14th International Conference on Information and Intelligent systems (IIS 2003)*, Varazdin, Croatia, pp. 23–28 (2003)
27. Patel, C., Supekar, K., Lee, Y., Park, E.: OntoKhoj: A semantic web portal for ontology searching, ranking and classification. In: *Proceedings of the Workshop on Web Information and Data Management*, pp. 58–61. ACM Press, New York (2003)
28. Rocha, C., Schwabe, D., de Aragao, M.: A hybrid approach for searching in the Semantic Web. In: *Proceedings of WWW 2004*, pp. 374–383. ACM Press, New York (2004)

29. Rose, D., Levinson, D.: Understanding user goals in web search. In: Proceedings of WWW 2004, pp. 13–19. ACM Press, New York (2004)
30. Song, J.F., Zhang, W.M., Xiao, W., Li, G.H., Xu, Z.N.: Ontology-based information retrieval model for the Semantic Web. In: EEE 2005, pp. 152–155. IEEE Computer Society, Los Alamitos (2005)
31. Spyns, P., Reinberger, M.L.: Lexically evaluating ontology triples generated automatically from texts. In: Gómez-Pérez, A., Euzenat, J. (eds.) ESWC 2005. LNCS, vol. 3532, pp. 563–577. Springer, Heidelberg (2005)
32. Strasunskas, D., Tomassen, S.L.: Web search tailored ontology evaluation framework. In: Chang, K.C.-C., Wang, W., Chen, L., Ellis, C.A., Hsu, C.-H., Tsoi, A.C., Wang, H. (eds.) APWeb/WAIM 2007. LNCS, vol. 4537, pp. 372–383. Springer, Heidelberg (2007)
33. Sullivan, D.: Death of a meta tag. Search Engine Watch (2002)
34. Tartir, S., Arpinar, I., Moore, M., Sheth, A., Aleman-Meza, B.: OntoQA: Metric-based ontology quality analysis. In: IEEE Workshop on Knowledge Acquisition from Distributed, Autonomous, Semantically Heterogeneous Data and Knowledge Sources, Houston, TX, USA, pp. 45–53. IEEE Computer Society, Los Alamitos (2005)
35. Tomassen, S.L.: Searching with document space adapted ontologies. In: Lytras, M., et al. (eds.) WSKS 2008. LNCS (LNAI), vol. 5288, pp. 513–522. Springer, Heidelberg (2008)

Magic Rewritings for Efficiently Processing Reactivity on Web Ontologies

Elsa Liliana Tovar^{1,2} and María-Esther Vidal¹

¹ Universidad Simón Bolívar
Caracas, Venezuela

{`etovar,mvidal`}@`ldc.usb.ve`

² Universidad de Carabobo
Valencia, Venezuela
`eltovar@uc.edu.ve`

Abstract. In this paper, we describe an approach aiming at enriching the Semantic Web with active information. We propose **ACTION**, an **ACTI**ve **ON**tology formalism to express reactive behavior. In **ACTION**, events are categorized as concepts of an ontology and, in conjunction with classes, properties and instances, are considered during the query answering and reasoning tasks. We hypothesize that **ACTION** provides a more expressive solution to the problem of representing and querying active knowledge than existing ECA-based approaches. However, this expressivity power can negatively impact on the complexity of the query processing and reasoning tasks because the number of derived data depends on the number and relationships of the events. The main source of complexity is produced because the number of the derived facts is polynomial with respect to the size of the events, and the same evaluations may be fired by different events. To overcome this problem, we propose optimization strategies to identify Magic Set rewritings where the number of duplicate evaluations is minimized. We present the query rewriting technique called Intersection of Magic Rewritings (IMR), which is based on Magic Sets rewritings that annotate the minimal set of rules that need to be evaluated to process reactive behavior on an ontology. We have conducted an experimental study and have observed that the proposed strategies are able to speed up the tasks of reasoning and query evaluation in two orders of magnitude for small ontologies, and in four orders of magnitude for medium and large ontologies, with respect to the bottom-up strategy.

Keywords: Ontology, Active Knowledge, Magic Set rewritings.

1 Introduction

Behavior of reactive data expresses changes of the data values when events occur and data satisfy specific conditions. Active databases [13] were defined to provide a framework for uniformly addressing issues related to the reactive behavior of the data once users interactions or events are triggered. In this context,

events are the typical transactions over data such as insert, delete, and update primitives. The most widely used approach to process reactive behavior is the execution of active rules which have the syntactical structure and the semantics of the event-condition-action rules (ECA rules paradigm). ECA rules model the actions to be taken when an event is fired and a condition is satisfied. Additionally, the ECA rules have been used with other aims in the database area, for example, to exchange data among peer databases [12] and to change the perspective of an active database system augmented with rules, to a database driven information system that offers rules-based services [10]. In the context of the Semantic Web, existing formalisms only allow the representation of static properties, i.e., they express information about data and meta-data that do not react when events occur; classes and properties of RDF/RDFS [18] and OWL [14] for example. These formalisms do not allow expressing reactive behavior of the data. On the other hand, the simple and intuitive semantics of the ECA rules have been naturally used in e-commerce and e-services as well as for representing reactive behavior processing on XML and RDF [2,3,4,11,14] repositories. Active knowledge is encoded by using ECA rules and the events that fire these rules are considered transactional data. Active knowledge, classes and roles are treated independently. Although these approaches solve the problem of representing certain attributes of objects in a particular domain, in the real world an object is characterized by a diversity of attributes. For example, it may be described through its structure, its behavior, and the way it reacts when an event occurs in its realm, i.e., its reactive behavior. Such characterization combines a static dimension with an active dimension of the data and is interesting, since in some cases, on-the-fly dynamic processing for information on the Semantic Web with respect to new data events is needed. Thus a reactive processing capability is necessary. In this paper, we present the active ontology formalism called **ACTION**, aiming at enriching the Semantic Web with active specification to express reactive behavior. In addition, we propose the *aOWL* language as an extension of OWL Lite [14], which augments OWL Lite with a set of operators to express reactive behavior. However, the enrichment of expressivity power can negatively impact the complexity of reasoning tasks and answering query because the number of the derived facts is polynomial with respect to the number of fired events [20], and the same evaluations may be triggered by different events. Therefore, we additionally propose an optimization strategy named Intersection of Magic Sets Rewritings (IMR). IMR processes a set of events that affect active properties of a particular ontological concept. It is based on Magic Sets rewritings and annotates the minimal set of rules that need to be evaluated to generate all the facts derivable from the fired events. Thus, the number of duplicate evaluations is minimized. This paper comprises seven additional sections. In section 2, related works are briefly described. In section 3, a comparative example that illustrates the importance of representing active knowledge in the Semantic Web is described. In section 4, the **ACTION** approach formalism, how a language as OWL Lite can be extended to represent reactive behavior is illustrated; also, a general algorithm to processing reactive

behavior is presented. In section 5, the optimization strategy IMR for efficiently processing reactivity on **ACTION** ontologies is defined. In section 6 the results of our experimental study are reported. Finally, in section 7, conclusions and future work are pointed out.

2 Related Work

Active databases [13] were first in managing reactive data. In order to achieve reactive behavior, traditional database systems integrated an efficient handling of data management to create and execute ECA rules. Events were viewed as any primitive for database state changes, e.g., data transactions. The syntactic structure of ECA rules impacted software architecture to process reactive behavior. Thus, relational active systems were basically comprised of an event manager to detect any operation of insert, delete, or update against the database, a query manager to process a boolean condition on the data affected by the event, and an action manager to execute some actions defined in terms of database operations such as new inserts, deletes, or updates. The use of triggers based on the SQL standard and the establishment of triggers activation policies were the main features of these proposals. A Dynamic Model was presented in [8], in order to capture data semantics related to the object evolution in the conceptual model. The Dynamic Model allows the definition of active rules that can be attached to classes, but are separated from the class definition. Active rules are implemented as ECA rules. Although the semantics of reactive behavior, codified through these rules, are part of the conceptual design, the Dynamic Model only establishes a direct link between a class and an active rule. Semantic Web formalisms are presented to expressing structure and behavior but not reactive behavior. In [2], active rules are defined in XSLT [23] and Lorel [1] is used to query these XML documents; ECA rules are used to implement e-commerce services enabling the generation and the manipulation of the contents of an XML repository as a reaction to the changes occurring in the documents. Moreover, the possibility for modeling a negotiation between services by means of conflict resolution policies among rules is offered in [6]. In [5], Active XQuery is presented as an active language for any kind of XML repository. Such language emulates the SQL3 trigger definition and the SQL3 active rules execution model. An ECA rule language where the trigger execution is atomic (i.e., a single trigger is fired once all the changes are made in the document) is presented in [15]. In [11] a proposal to processing data changes over time is presented. This approach consists of a logical framework for representing activities, states, and time within an ontology in first order logic and reasoning regarding occurrence of actions by means of intelligent agent; although this approach categorizes static and active knowledge at the same level it does not augment the expressive power of the formalism with the reactive behavior represented in the active knowledge. Recently, an ontology-based information integration approach [22] was presented to distributed sources. In that work, a traditional ontology is used to express the information of frequent changes of metadata parts and overlapping pieces

of information in distributed and heterogeneous sources. That approach was not intended to express reactive data behavior. Finally, in order to improve the functionality of the Semantic Web, several extensions of languages have been proposed. In [17] pRDF is defined as a version of RDF to express probabilistic knowledge about data. In [9] mRDF is defined as a version of RDF to represent RDF data depending on the context; with mRDF, dimensions state different contexts or worlds where certain parts of a RDF graph hold. Although, a variety of approaches have appeared, no proposal referring to incorporating reactivity within characterization of the data has been made so far. To the best of our knowledge, there is no known approach to express the events as concepts. The events have always been categorized as transactions over data and all research conducted in this area deals with the use of ECA rules to encode reactivity. Since the ECA rules are procedures separated from the data definition, the traditional reasoning tasks from the ontological languages cannot be used to manage reactive behavior, i.e., the active knowledge encoded in ECA rules is not used in conjunction with static data to infer new knowledge in order to manage reactivity.

3 Motivating Example

Consider the example presented in [15], which illustrates learning objects (LOs) by using data and metadata available on the Web and personal metadata about users of these LOs. Books are described as *title*, *ISBN*, *creators*, and *reviewers*, while users of LOs are modeled by using *identification*, *name*, *subjects of interest*, and their *last reviewers*. Users can be notified about the last review submitted for books in subjects in which they are interested; this information is used to automatically update their personal metadata. For example, the personal information related to a user called Johnny Mnemonic, establishes that he is interested in all LOs on the subject Computer Science. The book *Data on the Web* appears in the metadata of Johnny Mnemonic. In order to achieve reactive behavior on the repository, an ECA rule is defined. Each time a new review is appended to *Data on the Web*, e.g., when the insert event is fired, an ECA rule is triggered causing the replacement (sequence of one or more actions) of the previous review by the new review within the personal metadata of Johnny Mnemonic (specified by a Boolean condition). The ECA rules language defined in [15] is also used for RDF. But even in this case, it is impossible to use metadata related to users or LOs to process reactive behavior. For example, Johnny Mnemonic is an instance of the class *Subscribed_User* and his personal information is updated because a new review is appended to *Data on the Web*. However, the updates of the information of the rest of the users, who also belong to the class *Subscribed_User*, and who may also be interested in that book, cannot be derived using this rule. Now, consider this repository as also containing information concerning scientific reviewers, i.e., personal metadata related to known scientists who have made an evaluation to enhance the information of the LOs. This information is expressed in the class *Scientific_Reviewer* and in its properties. Each time a new review of

a scientist is appended to the information of a LO, his personal identification must be appended to the metadata related to the reviewers. Under the ECA rule paradigm, a new rule must be defined in order to process this new reactivity. The semantics encoded within a previous rule (related to the replacement of the last review) is not at all related to the semantics represented by the rule associated with the scientific reviewer. The event of the insertion of a new review that triggers both rules is the same, but the boolean conditions and the actions of these rules are different. In addition, the executions of every rule are independent from each other. In general, the relationship between these rules is operational in nature. It consists in the execution of actions of a rule that may, in turn, trigger further rules. In this case, the reactive behavior processing proceeds until no other rule can be triggered. However, unexpected interactions between rules may cause non-termination, i.e., an endless-loop behavior. This disadvantage of ECA rules is significant as it becomes more critical with the increasing complexity of the reactive functionality of an application. Several static rules analysis techniques exist in order to avoid non-termination of execution rule sets [2]. Extending the use of ECA rules to processing reactivity on data and metadata annotated by ontologies is not different and has the same disadvantage, i.e., the information about classes and properties cannot be used in conjunction with the reactive knowledge, encoded within ECA rules, to infer new reactive knowledge. To overcome this limitation, we propose an alternative processing scenario and an active ontology formalism to model the information required to represent reactivity. Consider an active ontology that comprises concepts related to LOs, *Subscribed_User* and *Scientific_Reviewer* and concepts related to the events that modify data of *Subscribed_User* and *Scientific_Reviewer*. In this active ontology if we can define the constructors *isEvent* and *isSubEventOf*, then we can state *isEvent(new-review)* and *isEvent(new-scientific-review)*, in order to express that *new-review* and *new-scientific-review* are events. We can also state *isSubEventOf(new-scientific-review, new-review)* in order to express that the event *new-scientific-review* is a *sub-event* of the event *new-review*. Moreover, this active ontology allows us to state that the event *new-review* is related to the property *last_review* of the class *Subscribed_User*, and the event *new-scientific-review* is related to the class *Scientific_Reviewer*. In addition, in an ontological framework, we can define an axiom that states when an event and all its *super-events* occur, i.e., we can express that the relation *isSubEventOf* is transitive. The computation of the transitive closure of the relation *isSubEventOf* can be expressed by the computation of the predicate *areSubEvents*, as follows:

```
areSupEvents(F,E):- isSubEventOf(E,F).
areSupEvents(F,E):- isSubEventOf(G,F),
                     areSupEvents(G,E).
```

Then, if the event *new-scientific-review* occurs, new personal information of this scientist must be associated with *Scientific_Reviewer* and it is automatically derived that the information of the review must also be associated with the corresponding instances of the class *Subscribed_User*. The set of *super-events* related

to *new-scientific-review* is derived using this inference mechanism. When the sequence of events to execute is derived by means of the inference, no analysis technique to process the events is necessary. The non-termination problem does not exist because the cardinality of the transitive closure over *isSubEventOf* is finite. The relationships between events are defined by means of metadata (i.e., event taxonomy) within the ontology (i.e., the active ontology), and the processing of reactivity is defined by using the axioms that state the semantics of the formalism. Thus, considering events as concepts gives the possibility to express a global vision over collective data changes. In this paper, we propose an active ontology formalism called **ACTION**, to model static and active knowledge. All this knowledge is considered during query and reasoning tasks processing. Additionally, we propose optimization strategies based on Magic Sets rewritings for efficiently processing the reactivity expressed by using **ACTION** ontologies. By doing so, the inference capacity of the existing formalisms is augmented.

4 Our Approach

4.1 ACTION: An ACTIVE ONtology Formalism

The ACTION ontology to express static and active knowledge is defined as follows:

Definition 1 (ACTION Ontology Knowledge Base). *An active ontology knowledge base is a 7-tuple $Oa = \langle C, E, Ps, Pa, F, fr, I \rangle$, where:*

- *C*: a set of classes or basic data types.
- *E*: a set of events.
- *Ps*: a set of static properties; each property corresponds to a function from $C \cup E$ to $C \cup E$.
- *Pa*: a set of active properties; each property corresponds to a function from C to C .
- *F*: a set of predicates representing instances of the classes, properties and events.
- *fr* a function, s.t., $fr : F \times Pa \times E \rightarrow F$; *fr* defines the reactive behavior in *Oa*.
- *l*: a set of axioms that describe the properties of the built-in properties provided in *Ps* and *Pa*.

The set of static properties *Ps* can be comprised of properties that induce a hierarchy of events, or a hierarchy of classes. Thus, for example, by using the **ACTION** ontology, we could represent that an event *new-scientific-review* is a sub-event of the event *new-review*, by using the fact *isSubEventOf*(*new-scientific-review new-review*) in the *F* of the **ACTION** ontology. On the other hand, there will be a deductive rule indicating that the built-in predicate *isSubEventOf* is transitive, in the set of axioms *I* of the **ACTION** ontology. In [16], optimization and evaluation techniques applied to ontology deductive bases have been developed to perform ontology query and reasoning tasks efficiently. We extend

that approach with meta-level predicates to represent the reactive behavior of the data. The reactive behavior is characterized by extending some definitions taken from [16]. Thus, we provide an efficient framework to implement active ontologies. We represent each Oa , as follows:

Definition 2 (Active Deductive Ontology Base). *Given an ACTION ontology $Oa = \langle C, E, Ps, Pa, F, fr, I \rangle$, an Active Ontology Base for Oa , $ADOB$ is a pair $\langle AEO, AIO \rangle$, where:*

- *AEO corresponds to an Active Extensional Ontology base composed of meta-level predicates that represent the knowledge explicitly represented in the sets C, E, Ps, Pa , and F ;*
- *AIO is an Active Intensional Ontology base comprised of deductive rules that define the semantics of the knowledge represented in AEO and modeled by the axioms in I .*

Each active deductive base $ADOB$ is comprised of meta-level predicates, e.g., the built-in predicate $isEvent(E)$ where E is a name event, the built-in predicate $isSubEventOf(E1, E2)$ defines that the event name $E1$ is a sub-event of the event name $E2$, and the intensional meta-level predicate $areSupEvents(E2, E1)$ is specified by a deductive rule that defines the transitive closure of the predicate $isSubEventOf(E1, E2)$.

On the other hand, the predicate $activeProperty(AP, T, D, R)$ defines an active property AP in terms of its type T (not the same as $rdf : type$), domain D and range R ; and the predicate $reactiveBehavior(AP, E1, BE, V)$, specifies the reactive behavior of an active property AP that takes the value V when an event $E1$ occurs and the Boolean expression BE holds. BE is a Boolean expression over the properties in Ps and Pa . As usual, an active ontology query is a rule $q : Q(X) \leftarrow \exists Y B(X, Y)$ where B is a conjunction of predicates in the sets AEO and AIO . A valuation μ is a function $\mu : Vars \rightarrow D$ where $Vars$ is a set of variables and D is a set of constants. Given a meta-level predicate R , the valuation μ is a valid instantiation of R , if and only if, $\mu(R)$ evaluates true in the minimal model of $ADOB$. No free variables exist in the ontology and our approach is based in the Closed World assumption. The model-theoric semantics for $ADOB$ is presented in [20].

4.2 aOWL: Extending OWL Lite with Reactivity

In order to illustrate how OWL can be extended with the ability to express reactive behavior, we have extended OWL Lite with the set of the following built-in constructors: *isEvent*, *isSubEventOf*, *activeProperty*, *reactiveBehavior*, *exclusiveStatement*, *simultaneousStatement* and *orderedStatement* (different kinds of active properties according to the amount and the sequence of their values). We name this dialect of OWL, *aOWL* (activeOWL). By using *aOWL*, we can represent statements such as *new-scientific-review* is a *sub-event* of the event *new-review*, and the *latest-reviewer* is an active property that changes its value when the event *new-review* occurs. The *aOWL* language inherits all axioms

from the fragment of OWL that can be modeled as deductive database [16], and extends them by using the active axioms. The abstract syntax and the definition of some of the *aOWL* constructors in the form of meta-level *ADOB* predicates are shown in Table 1.

Table 1. Abstract Syntax *aOWL* and *ADOB* meta-level predicates

Abstract Syntax <i>aOWL</i>	Built-in Predicates AEO
<code>isEvent(e)</code>	<code>isEvent(E)</code>
<code>isSubEventOf(e1 e2)</code>	<code>isSubEventOf(E1,E2).</code>
<code>activeProperty(ap type(rd) range(rr))</code>	<code>activeProperty(AP,TYPE,RD,RR)</code>
<code>reactiveBehavior(ap isEvent(e,bc) value v)</code>	<code>reactiveBehavior(AP,E,BC,V).</code>
<code>exclusiveStatement(i value (ap v)</code>	<code>exclusiveStatement (I,AP,V).</code>
<code>simultaneousStatement (i value (ap v)</code>	<code>simultaneousStatement (I,AP,V).</code>
<code>orderedStatement (i value(ap (v,ti))</code>	<code>orderedStatement (I,AP,V,Ti).</code>
<i>aOWL</i> Axioms	Built-ins Predicates AIO
<code>if isSubEventOf(E2,E1) and</code>	<code>areSupEvents(E1,E2) :-isSubEventOf(E2,E1).</code>
<code>isSubEventOf(E3,E2) then</code>	<code>areSupEvents(E1,E2) :-isSubEventOf(E2,E3),</code>
<code>isSubEventOf(E3,E1).</code>	<code>areSupEvents(E1,E3).</code>

4.3 The Reactive Behavior Processing

We present a general algorithm to process reactive behavior triggered by an Event E occurring on concept C . This algorithm, given the minimal model MM of the ontology Oa , determines the individuals of C affected by E and its super-events. Figure 1 outlines Algorithm 1; it is based on the following assumptions: a) if an active property AP is affected by an event E then all the super-properties of AP are also affected, b) if an event E affects some active property AP when the property P has the value V , then E affects AP when any sub-property of P has the value V . To accomplish this, the following predicates are evaluated: *areSupEvents*(F, E), *areSubProperties*(AP, PA), *areStatements*($Ii, PA, V1$), *areReactiveBehavior*($PA, F, P, BC, V2$).

Active properties can be of different types and they change according to this characteristic. If the type of an active property AP is *Exclusive*, the predicate *exclusiveStatement*($Ii, AP, V1$) is replaced by *exclusiveStatement*($Ii, AP, V2$). If AP is *Ordered* a new time stamp (T_i) is assigned in order to indicate time when AP takes the value $V2$ and the predicates *orderedStatement*($Ii, AP, V1, T_{i-1}$) and *orderedStatement*($Ii, AP, V2, T_i$) coexist. Finally, if an AP is *Simultaneous* it means that AP simultaneously has two values $V1$ and $V2$, and the predicates *simultaneousStatement*($Ii, AP, V1$) and *simultaneousStatement*($Ii, AP, V2$) coexist. The time complexity of Algorithm 1 is bound by the time complexity of the transitive closure [7]. Thus, the complexity of Algorithm 1 is $O(n^3 * M)$, where n is the number of predicates *isSubEventOf*, and M is the number of active property predicates to be changed. On the other hand, the number of derived facts polynomially depends on the number and relationships of the events and the same evaluations may be fired by different events. Thus, this enrichment of expressivity can negatively impact the complexity of the reasoning task implemented by Algorithm 1. Details in [20].

Algorithm 1

Input: Oa : an active ontology modeled as $ADOB = \langle AEO, AIO \rangle$.
 (E, C) : the event E affects concept C , where $E, C \in Oa$.

Output: Oa : resulting ontology after processing reactive behavior triggered by event E occurring on C .

Method:

Let MM the minimal model of Oa .

Let IND the set of individuals $I \in C$, where $areIndividuals(I, C) \in MM$

1. **for each** $Ii \in IND$, such as:
 - $areStatements(Ii, PA, V1) \in MM$.
 - $areSubProperties(AP, PA) \in MM$.
 - $areSupEvents(F, E) \in MM$.
 - $activeProperty(PA, T, D, R) \in MM$.
 - $areReactiveBehavior(PA, F, P, BC, V2) \in MM$.
 - $areSubProperties(P, P) \in MM$.
 - (a) **if** $areStatements(Ii, P, BC) \in MM$ and $activeProperty(PA, T, D, R) \in MM$ **then**:
 - Case** $T = \text{Exclusive}$:
 $Oa' = (Oa - \{exclusiveStatement(Ii, AP, V1)\}) \cup \{exclusiveStatement(Ii, AP, V2)\}$.
 - Case** $T = \text{Ordered}$:
 $Oa' = Oa \cup \{orderedStatement(Ii, AP, V2, Ti)\}$, where
 $Ti = Ti-1 + 1$, and, $orderedStatement(Ii, AP, V1, Ti-1) \in MM$.
 - Case** $T = \text{Simultaneous}$:
 $Oa' = Oa \cup \{simultaneousStatement(Ii, AP, V2)\}$.
2. **Return** Oa

Fig. 1. Algorithm 1

5 Magic Rewriting for Processing Reactivity

The Magic Set approach is based on rewriting a logic program so that bottom-up fix-point evaluation of the magic program avoids derivation of irrelevant facts. The basic idea of Magic Set is to emulate top-down sideways passing of bindings by using rules to be executed in a bottom-up fashion. This notion is very important to prune computations of derivation trees for recursive predicates. The efficiency of Algorithm 1 can be improved by rewriting recursive predicates as magic rules, i.e., by means of rewriting predicates such as $areSupEvents(F, E)$, $areIndividuals(I, C)$, $areSubProperties(AP, PA)$, that require the computation of the transitive closure. Given the constants e and c (when event E affects C) in the input reactive goal G , the unification between G and a rule head H causes some of the variables in H 's head to be bound to these constants. Given bindings for variables of predicates by means of sideways information passing, we can

solve the predicates with these bindings and thus obtain bindings for some of its other variables. These new bindings can be passed to other predicates such as *areStatements(Ii, PA, V1)*, *areReactiveBehavior(PA, F, P, BC, V2)* to restrict the computation for these predicates.

Example 1. Consider the program to compute the super-events of an event *E*:

```
areSupEvents(F,E):- isSubEventOf(E,F).
areSupEvents(F,E):- isSubEventOf(G,F), areSupEvents(G,E).
```

The magic rewriting of above program for the query: *areSupEvents(F, e1)* is:

```
magic-areSupEvents-fb(e1).
magic-areSupEvents-fb(G):- isSubEventOf(G,E), magic-areSupEvents-fb(E).
areSupEvents-fb(F,E):- isSubEventOf(E,G).
areSupEvents-fb(F,E):- magic-areSupEvents-fb(E),
                        isSubEventOf(E,G),
                        areSupEvents-fb(F,G).
```

We can rewrite the program to process reactivity of an event occurring on *C* by means of the classic Magic Set algorithm [3]. In this manner, the irrelevant facts are avoided in contrast to bottom-up strategy. However, this approach suffers from the drawback that every time an event is fired, the program to process reactivity must be rewritten for that event, and given a set of events, many facts may be evaluated repeatedly. The proposed strategy Intersection of Magic Rewritings (IMR) minimizes the number of duplicate evaluations for a set of events. Consider the running example, using the traditional Magic Sets rewritings, the following two magic programs are generated when the events *e1* and *e2* are simultaneously fired:

For query *q1: areSupEvents(F, e1)* from Example 1 the following program *P₁* is created:

```
P1: magic-areSupEvents-fb(e1).
    magic-areSupEvents-fb(G):- isSubEventOf(E,G),
                                magic-areSupEvents-fb(E).
    areSupEvents-fb(F,E):- isSubEventOf(E,F).
    areSupEvents-fb(F,E):- magic-areSupEvents-fb(E),
                            isSubEventOf(E,G),
                            areSupEvents-fb(F,G).
```

On the other hand, for the query *q2 : areSupEvents(F, e2)* the program *P₂* is produced:

```
P2: magic-areSupEvents-fb(e2).
    magic-areSupEvents-fb(G):- isSubEventOf(E,G),
                                magic-areSupEvents-fb(E).
    areSupEvents-fb(F,E):- isSubEventOf(E,F).
    areSupEvents-fb(F,E):- magic-areSupEvents-fb(E),
                            isSubEventOf(E,G),
                            areSupEvents-fb(F,G).
```

The only difference between the programs P_1 and P_2 is the fragment of magic predicates: *magic-areSupEvents-fb(e1)* and *magic-areSupEvents-fb(e2)*.

The rest of the rules are the same in both programs. We name the set of rules belonging to P_1 and P_2 , the intersection of the magic rewritings of P_1 and P_2 . If we construct a program merging magic predicates *magic-areSupEvents-fb(e1)* and *magic-areSupEvents-fb(e2)*, and the intersection of the magic rewritings of P_1 and P_2 , we succeed in inferring the super-events of $e1$ and $e2$ with a smaller number of magic rules to be evaluated.

Example 2. The following magic program P_3 computes super-events of the set $\{e1, e2\}$: For query $q3$: *areSupEvents-fb(F, E)*

```

P3: magic-areSupEvents-fb(e1).
    magic-areSupEvents-fb(e2).
    magic-areSupEvents-fb(G):- isSubEventOf(E, G),
                                magic-areSupEvents-fb(E).
    areSupEvents-fb(F, E):- isSubEventOf(E, F).
    areSupEvents-fb(F, E):- magic-areSupEvents-fb(E),
                            isSubEventOf(E, G),
                            areSupEvents-fb(F, G).

```

5.1 Intersection of Magic Sets Rewritings

Generalizing from Example 2, we propose the following strategy to identify the intersection of the magic rules between two Magic Sets Rewriting programs:

Definition 3 (Intersection of Magic Sets Rewritings). *Given two magic programs P_1 and P_2 , where:*

- P_1 is a magic rewriting of the program P for a query $q_1(C_1, X_1)$, where C_1 is the vector of the arguments bound to constants in P_1 , X_1 is the vector of the free arguments in P_1 , and $q_1(C_1, X_1)$ is a query that generates the adornment A_1 for P_1 .
- P_2 is a magic rewriting of the program P for the query $q_2(C_2, X_2)$, where C_2 is the vector of the arguments bound to constants in P_2 , X_2 is the vector of the free arguments in P_2 , and $q_2(C_2, X_2)$ is a query that generates the adornment A_2 for P_2 , such that:
 - C_1 and C_2 are different.
 - A_1 and A_2 are the same.

the Intersection of Magic Rewritings strategy consists in generating a new magic program P_3 that has two components:

- *The set of magic rules shared by P_1 and P_2 .*
- *The set of magic predicates where exists a magic predicate, of the form $magic\_p(c)$, for each $c \in C_1 \cup C_2$.*

The adornment for each rule or predicate of P_3 is A_1 (or A_2). The adornment for a n -ary predicate is a string a of length n on the alphabet $\{b, f\}$ where b stands for bound and f stands for free [4]. The adornment of Example 1 for $areSupEvents(F, E)$ is fb because the query is $areSupEvents(F, e1)$. We rewrite all recursive predicates required to process reactivity (the magic rules generation) and we create the magic predicates according to the set of events that appear in the reactive goal. Essentially, this strategy seeks to capture the work that must be done to process reactivity for a set of events.

5.2 A Naive vs. An Efficient Evaluation Approach

Even though the IMR strategy computes only the facts relevant to the set of events that appear in the reactive goal, some computations are repeated. On one hand, when an active property AP is affected by the event E , all the super-properties of AP are also affected.

Consider the predicate $isSubProperty(headOf, worksFor)$ that indicates that the property $headOf$ is a sub-property of $worksFor$, and the predicate $exclusiveStatement(fullProfessor1, headOf, dept1)$ that indicates that $fullProfessor1$ is related to $dept1$ by the property $headOf$. A reasoner must infer that $fullProfessor1$ is also related to $dept1$ by the property $worksFor$. If $headOf$ is an active property and is affected by an event E , then the property $worksFor$ is also affected. Thus, the predicates $exclusiveStatement(fullProfessor1, headOf, dept1)$, $exclusiveStatement(fullProfessor1, worksFor, dept1)$ must be changed. On the other hand, when an event E affects some active property AP because the property P has the value V , then E affects AP when any sub-property of P has the value V .

Consider the predicates $isSubProperty(belongsTo, memberOf)$ and $reactiveBehavior(takesCourse, eventE, memberOf, univ1, gradCourse1)$ that indicate that the active property $takesCourse$ must have the value $gradCourse$ when an event E occurs and the value of the property $memberOf$ is $univ1$. Provided that the property $belongsTo$ is a sub-property of $memberOf$, then the property $takesCourse$ must also have the value $gradCourse1$ when an event E occurs and the property $belongsTo$ is $univ1$.

In the naive approach, reactivity processing is made for each event and its super-events- from the set of events. In doing so, it computes the recursive predicates several times because: a) two or more events can have the same super-events and the reactivity processing for each super-event can be repeated, b) the active properties can be affected by means of different events and the predicate $isSubPropertyOf$ is computed repeatedly for a particular active property, and c) the Boolean conditions associated with the reactive behavior of an active property depend on different properties (static properties), and the computation of the recursive predicate $isSubPropertyOf$ must be done repeatedly for a particular property. We present an optimization strategy for avoiding repeating the computation of the magic rules for the same event, active property, or individual. Based on the reactive goal, the central idea consists in generating in one step the magic predicates required for: a) all events fired, b) all active properties that are affected for each event, and c) all individuals that hold the boolean

conditions associated for each event. Given a reactive goal that indicates the set of events and the class C for reactive processing, the algorithm is as follows:

1. The materialization of the transitive closure of *isSupEventOf* is computed for each event - of the set of events - that appears in the reactive goal, and a new set of events that is called SE is generated.
2. For each event e of SE :
 - (a) Generate predicates: *magic_areSupEvents_fb*(e), *magic_areReactiveBehavior_fbfff*.
 - (b) Active properties and boolean conditions are retrieved from executing the magic rule *areReactiveBehavior*, which is restricted by the magic predicate *magic_areReactiveBehavior_fbfff*. We call SAP the set of active properties and BC the set of the property-value pairs (P, V) associated with event e .
3. For each active property ap of SAP , generate predicate *magic_areActiveProperties_bf*(ap). Execute the magic rule *areActiveProperties_bf*(AP, T).
4. Finally, for each bc of BC , generate predicate *magic_areIndividuals_fbbb*(C, P, V) to restrict the individuals of the class C affected by the fired events.

6 Experimental Study

In this section we present the results of our experimental study. We report on the evaluation time and on the number of derived facts. We compare the bottom-up evaluation of the IMR rewritings to the bottom-up evaluation of the input program and the program rewritten by using traditional Magic Sets techniques.

6.1 Experimental Design

Datasets. The experimental study was conducted with Lehigh University Benchmark (LUBM) [19], which is considered the *de facto* standard when it comes to reasoning with large ontologies [21]. We have extended the instance generator LUBM to insert active properties and events. The original data schema, named TBOX, comprises 43 classes and 32 properties referring to the university domain. The instance generator LUBM uses class and property subsets to generate documents that comprise the ABOX. Given that the generator LUBM uses eleven properties (that we consider static properties), we insert eleven active properties and eleven events into TBOX. Additionally, eleven new static properties to construct boolean conditions associated with events were appended to the TBOX. The *univ_num* parameter (number of universities to generate) of the generator program is used to construct the different ABOX sizes. Once the instances of the ABOX are generated as *aOWL* documents, this information is translated into meta-level predicates of ADOB by means of prologs DCG (Definite Clauses Grammars).

To compare IMR with bottom-up, we consider a dataset with three kinds of ABOX: small (information of one university), medium (five universities) and large (ten universities). Ten different reactive goals (queries) were posed for

each kind of ABOX, each of them evaluated using bottom-up evaluation and IMR. To compare IMR with classic Magic Sets evaluation (CMS), we consider a second dataset with bigger ABOX sizes: small (information of one university), medium (ten universities) and large (twenty universities). Thirty reactive goals were posed for each ABOX; each of them evaluated using the CMS evaluation and IMR. In Table 2, the generated ABOX are described in terms of the number of classes, static and active properties.

Table 2. DataSet Description

#Univ	#Class-Inst	# Prop-Inst	#ActProp-Inst	MB
1	15393	55528	11802	5.7
5	91408	325429	69441	33.5
10	184086	652868	139438	67.3
20	414194	1305736	278876	151.0

Hardware and Software. The experiments were evaluated on a Solaris SUN machine with Sparcv9 1281 MHz processor and 16GB of RAM. The proposed algorithms have been implemented in SWI-Prolog, Version 5.6.54.

Metrics. We report on the metrics **TNDF** and **Time** defined as follows:

- **Total Number of Derived Facts (TNDF):** the cost of the tasks of reasoning and query evaluation are measured in terms of the number of derived facts needed for the reactivity processing. The **TNDF** that results by means of bottom-up is the size of the minimal model.
- **Time:** measures the time in seconds required to achieve reactive processing.

6.2 Results

First, we considered the query (Event11,GraduateCourse). Figure 2 shows compares the performance of the bottom-up evaluation of the input program that represents the reactive processing of this query versus the bottom-up evaluation of the IMR rewriting of the same program. Figure 2 reports on **TNDF** and **Time** for both strategies. We observed that IMR strategy is able to accelerate the tasks of reasoning and query evaluation in two orders of magnitude for small ontologies, and in four orders of magnitude for medium and large ontologies.

Figure 3 compares the bottom-up evaluation of Magic Sets rewritings of the programs that represent the thirty queries studied versus the IMR rewritings of the same programs. Figure 3 reports on **TNDF** and **Time** for both strategies. We observed that IMR always needs to infer fewer **TNDF** to process the reactivity than the classic Magic Sets strategy. The reason is that the strategy proposed avoids duplicate inferences when it processes a set of events. By contrast, the classic Magic Sets strategy makes all inferences for each event of the set. The difference between both strategies is in one order of magnitude

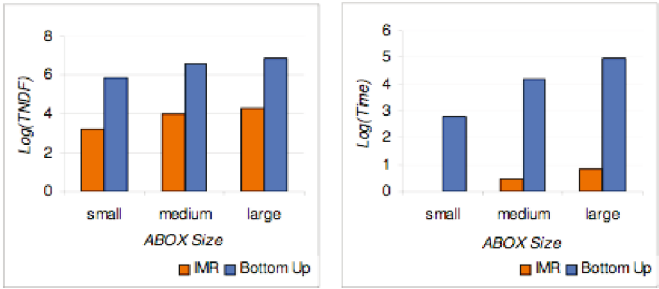


Fig. 2. TNDF and Time for IMR and Bottom-Up on three ABOX sizes for query (Event11,GraduateCourse)

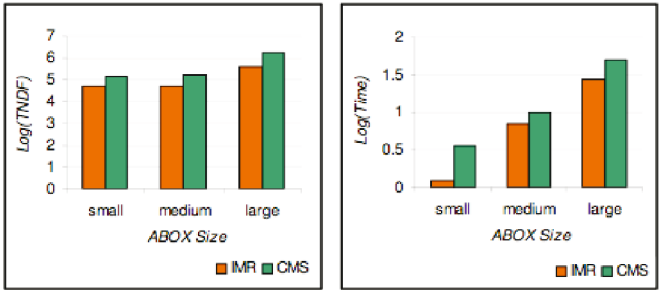


Fig. 3. Averaged TNDF and Time for IMR and Bottom-Up on three ABOX sizes for thirty queries

Table 3. Detailed TNDF for IMR and CMS strategies

TNDF						
	Small ABOX		Medium ABOX		Large ABOX	
Strategy	M	S	M	S	M	S
IMR	48394.43	25851.70	48731.27	23336.16	382784.23	152014.89
CMS	147676.52	68224.84	172394.04	90128.4	1720143.27	605794.02
Md ($M_{CMS} - M_{IMR}$)	99282.09		123662.77		1337359.05	
Sd ($SCMS - SIMR$)	51085.00		70347.87		551684.93	
CI (90%)	[80991.13, 117573.03]		[95248.65,152076.89]		[1134965.91,1539752.17]	
t-test (paired, two-tailed)	<i>p-value</i> =4.27E-09		<i>p-value</i> =2.78E-09		<i>p-value</i> =1.95E-10	
Time						
	Small ABOX		Medium ABOX		Large ABOX	
Strategy	M	S	M	S	M	S
IMR	1.24	1.12	7.00	5.33	27.65	10.50
CMS	3.62	2.34	10.00	5.37	51.35	21.00
Md ($M_{CMS} - M_{IMR}$)	2.38		3.00		23.70	
Sd ($SCMS - SIMR$)	1.70		1.05		15.26	
CI (90%)	[1.84, 2.92]		[2.61,3.39]		[18.23,29.16]	
t-test (paired, two-tailed)	<i>p-value</i> =3.25E-08		<i>p-value</i> =2.8E-11		<i>p-value</i> =1.89E-07	

for all sizes of ontologies. As it was expected, the Time required to achieve the reactivity processing by the IMR program is also less.

Table 3 illustrates detailed results for **TNDF** and **Time** for the considered ABOX sizes. We calculate the means of both metrics and their standard deviation. We completed a hypothesis test by means of a Students t-test (paired samples and two-tailed). The null hypothesis was that the **TNDF** mean of IMR (MIMR) and the **TNDF**'s mean of CMS (MCMS) are the same (similar null hypothesis for **Time** was considered). The *p-value* for all cases shows that there is statistical evidence to reject the null hypothesis and to conclude that the differences between IMR and CMS are highly significant. A 90% confidence interval for difference between means (MCMS - MIMR, for both **TNDF** and **Time**) was completed. Table 3 shows that with 90% confidence IMR reduces the total number of derived facts in at least 80,991 facts for small ABOX, 95,249 facts for medium ABOX, and 1,134,966 facts for large ABOX. The acceleration of IMR with respect to CMS is at least two seconds for small ABOX, three seconds for medium ABOX, and eighteen seconds for large ABOX. These observations further support that the **ACTION** formalism contributes a more semantic expressivity to the reactive behavior of the data in Semantic Web ontologies, and that the strategies proposed to process that reactivity are efficient.

7 Conclusions and Future Work

The ability of active ontologies to represent events as first-class concepts allows us to use the inference power to manage reactive behavior. This feature opens a new scenario to control the reactivity in the Semantic Web. We show that active ontologies provide a simpler and more expressive solution to the problem of representing and querying active knowledge. We have shown that the expressivity of our formalism negatively impacts the complexity of the query answering and reasoning tasks. To overcome this problem, we proposed optimization strategies to identify Magic Set rewritings where the number of duplicate evaluations is minimized. In the future, we will extend SPARQL to represent queries that capture the active knowledge represented in **ACTION** ontologies.

References

1. Abiteboul, S., Quass, D., McHugh, J., Widom, J., Wiener, J.: The Lorel Query Language for Semistructured Data. International Journal on Digital Libraries 1 (April 1997)
2. Bailey, J., Poulouvasilis, A., Wood, P.: Analysis and optimization for event-condition-action rules on XML. Computer Networks (2002)
3. Bancilhon, F., Maier, D., Sagiv, Y., Ullman, J.: Magic sets and other strange ways to implement logic programs (extended abstract). Symposium on Principles of Database Systems. In: Proceedings of the fifth ACM SIGACT-SIGMOD, 1985 symposium on Principles of database systems, Cambridge, Massachusetts, USA, pp. 1–15 (1985)

4. Beeri, C., Ramakrishnan, R.: Symposium on Principles of Database Systems. In: Proceedings of the sixth ACM SIGACT-SIGMOD-SIGART 1987 symposium on Principles of database systems, San Diego California, USA (1987)
5. Bonifati, A., et al.: Active XQuery. In: Proc. of the IEEE (ICDE) (2002)
6. Bonifati, A., et al.: Active rules for XML: A new paradigm for e-services. *Vldb Journal* 10 (2001)
7. Cohen, E.: Estimating the size of the transitive closure in linear time. In: 35th Annual Symposium on Foundations of Computer Science, pp. 190–200. IEEE, Los Alamitos (1994)
8. Foster, I., Voeckler, J., Wilde, M., Zhao, Y.: Chimera: A virtual data system for representing, querying and automating data derivation. In: Proceedings of Global and Peer-to-Peer Computing on Large Scale Distributed Systems Workshop (May 1995)
9. Gergatsoulis, M., Lilis, P.: Multidimensional RDF. In: Meersman, R., Tari, Z. (eds.) OTM 2005. LNCS, vol. 3761, pp. 1188–1205. Springer, Heidelberg (2005)
10. Goldin, D., Srinivasa, S., Srikanti, V.: Active Databases as Information Systems, cse.uconn.edu/~dqg/papers/ideas04.pdf
11. Gruninger, M., Fox, M.: An Activity for Enterprise Modeling, www.eil.utoronto.ca/enterprise-modelling/papers/gruninger-wetice94-act.pdf
12. Kantere, V., Kiringa, I., Mylopoulos, J., Kementsitsidis, A., Arenas, A.: Coordinating peer databases using ECA rules. *DBISP2P* (2003)
13. Morgenstern, M.: Active databases as a Paradigm for Enhanced Computing Environments. In: Proc. of Intl. Conf. on Very Large Data Bases, pp. 34–42 (1983)
14. OWL Recommendation W3C, <http://www.w3.org/TR/owl-features>
15. Papamarkos, G., Poulouvasilis, A., Wood, P.: Event-Condition-Action Rule Languages for the Semantic Web. In: Proc. WWW 2002, Hawaii, USA (2002)
16. Ruckhaus, E., Ruiz, E., Vidal, M.E.: Query Optimization in The Semantic Web. In: International Workshop on Applications of Logic Programming in the Semantic Web and Semantic Web Services, Seattle, USA (2006)
17. Udea, O., Subrahmanian, V.S., Majkic, Z.: Probabilistic RDF. In: Proceedings of the IEEE International Conference on Information Reuse and Integration (IRI), pp. 172–177 (2006)
18. RDF Semantics, W3C Working Draft, Edit. Patrick Hayes (October 1, 2003)
19. SWAT Projects- the Lehigh University Benchmark (LUBM), <http://swat.cse.lehigh.edu/projects/lubm/>
20. Tovar, E., Vidal, M.E.: Technical report Tovar-Vidal-1-2007: Events as Concept within Ontology (2007), <http://mg.facyt.uc.edu.ve/etovar/TR-tovar-vidal-1-2007.pdf>
21. Weithaner, T., Liebig, T., Luther, M., Bahm, S.: What's Wrong with OWL Benchmarks? In: Proceedings of the Second International Workshop on Scalable Semantic Web Knowledge Base Systems SSWS 2006, Athens, GA, USA (November 2006)
22. Xing, W., Corcho, O., Goble, C., Dikaiakos, M.: Active Ontology: An Information Integration Approach for Dynamic Information Sources. Poster at 4th European Semantic Web Conference (2007)
23. XSLT Specification. W3C Recommendation (November 1999), <http://www.w3.org/TR/WD-xsl>

Mediating and Analyzing Social Data

Ying Ding¹, Ioan Toma², Sin-Jae Kang³, Zhixiong Zhang⁴, and Michael Fried²

¹ Indiana University, 1320 E 10th, Bloomington, IN 47405 USA
dingying@indiana.edu

² University of Innsbruck, Innsbruck, Austria
{ioan.toma,michael.fried}@sti2.at

³ Daegu University, Daegu, South Korea
sjkang@daegu.ac.kr

⁴ Library of Chinese Academy of Sciences, Beijing, China
zhangzx@mail.las.ac.cn

Abstract. Web 2.0 is turning current Web into social platform for knowing people and sharing information. The Web is strongly socially linked than ever. This paper takes major social tagging systems as examples, namely delicious, flickr and youtube, to analyze the social phenomena in the Social Web in order to identify the way of mediating and linking social data. A simple Upper Tag Ontology (UTO) is proposed to integrate different social tagging data and mediate and link with other related social metadata.

Keywords: Social Tagging, data mediation, Social Web, ontology.

1 Introduction

Web 2.0 is turning current Web into social platform for knowing people and sharing information. The Web is strongly socially linked than ever. The term “Social Web” was introduced in 1998 by Peter Hoshka [1] who tried to stress the social medium function of the Web. From Wikipedia, the Social Web is defined as an open global distributed data sharing network which links people, organizations and concepts. Current Web 2.0 is the main stream of the Social Web which provides platform and technologies (such as wiki, blog, tag, RSS feed, etc.) for online collaboration and communication.

The online publishing in Web 2.0 made everything so easy that anyone who can write or type can publish their data to the Web. This revolution significantly stimulates the amount of normal users to get involved to the Web communication; those of them are just teenagers or old people. One of the new ways of adding data to the current Web is tagging which reflects community effort on organizing and sharing information. Tagging is a kind of adding keywords through typed hyperlinks. Now the web is changing from hyperlinked documents to typed hyperlinked data web.

As from current Web 2.0, we already evident human-created metadata (such as tags) which are growing daily on the Web. This trend will further lead to more similar metadata as well as metadata generated from Semantic Web community which is

ontologically explicitly defined, for example, FOAF (metadata for friends), SKOS (metadata for taxonomies), DOAP (metadata for project), RSS (metadata for news), SIOC (metadata for social networks), Dublin Core (metadata for documents), GEO (metadata for geographic coordinates), GeneOnt (metadata for human genes), microformat (metadata for Social Web) and so on.

Furthermore, machine can also start to contribute data to the Web as machine can generate data automatically based on pre-defined ontologies. Those metadata and data are not isolated but interlinked. Based on four principles of linking open data proposed by Tim Berners-Lee, more and more linked semantic data are available (see Link Open Data initiative¹). Those kind of linking is mainly through owl:sameAs or foaf:knows to link different concepts or instances. We call those links semantic links. These powerful semantic links will weave the current Web to its future. The future Web is the Web of semantically linked semantic data.

This paper takes major social tagging systems as examples, namely delicious, flickr and youtube, to analyze the social phenomena in the Social Web in order to identify the way of mediating and linking social data. The main contributions of our work include:

- Modeling social tagging data according to proposed Upper Tag Ontology (UTO).
- Linking UTO with other related social metadata (such as FOAF, DC, SIOC, SKOS, etc.)
- Crawling tag data from major social tagging systems and integrating them according to UTO.
- Clustering crawled tagging data.

According to above, this paper is organized as follows. Section 2 gives the detailed description of how to model social tagging data, how to link them with related social semantics, how to crawl social tagging data and how to analyze tagging data via clustering. Section 3 discusses the related work. Section 4 concludes the paper and presents some future work.

2 Social Tagging

Tag is a keyword used to categorize online objects. The goal of tagging is to make a body of information increasingly easier to search, discover, share and navigate over time. Social tagging is not simply just tagging, tags are social metadata generated from collective intelligence. The consensus of tags forms social semantics which are called folksonomies. It is bottom-up approach and reflects collective agreement. It speaks the same language as the users and makes the things easier to find.

2.1 Modelling Social Tagging Data

We can tag bookmarks (del.icio.us), photos (flickr), videos (YouTube), books (LibraryThing), Music (Last.fm), citations (CiteULike), blogs (Technorati), etc. Tag

¹ <http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData>

is nothing special than a typed hyperlink. We can use “rel” attribute to create typed hyperlink. There are many social networks providing tagging services, here we take three major social tagging systems, namely delicious, flickr, and youtube, to analyze their social tagging behavior. Based on this analysis, we propose Upper Tag Ontology (UTO) which is originated from Tag Ontology proposed by Tom Gruber [2]. In his tag ontology, he proposed five key concepts which are object, tag, tagger, source and vote. Here in UTO, we add another three concepts: comment, date and tagging. Because most of the social networks contain information about comments for the tags or objects, these provide extra information for us to better understand the meaning of the tags or objects. Date is another important concept for us as it depicts the evolution of the tags and tagging behavior. It can also help us to unveil the hidden social changes inside a social network. The tagging concept plays a role to interlink all these core concepts together. Itself does not have real meaning. Furthermore, we add *has_relatedTag* relationship to tag concept itself. More details about modeling social tagging data were discussed in [3].

Let O be UTO ontology,

$$O = (C, \mathfrak{R}) \quad (1)$$

Where $C = \{c_i, i \in N\}$ is a finite set of concepts.

$\mathfrak{R} = \{(c_i, c_k), i, k \in N\}$ is a finite set of relations established among concepts in C .

In UTO,

$$C = \left\{ \begin{array}{l} \text{Tag, Tagging, Object, Tagger, Source,} \\ \text{Date, Comment, Vote} \end{array} \right\},$$

$$\mathfrak{R} = \left\{ \begin{array}{l} \text{has\_relatedTag, has\_tag, has\_object,} \\ \text{has\_source, has\_date, has\_creator,} \\ \text{has\_comment, has\_vote} \end{array} \right\}$$

Figure 1 presents the concepts and relations of UTO. As we see, UTO is a very small and simple ontology with 8 concepts and 8 relationships (see Table 1 and Table 2). The tagging concept acts as a virtual connection among different concepts in UTO. It does not have real meaning rather than the function of linking some core concepts. For instance, it is hard to tell whether the date is for tag or the tagging behavior, or comment can be viewed as being added to tag or to object directly. So most of the relations in UTO are defined as transitive so that comment can be connected to object via tagging or to tag via tagging.

According to formula (1), when $r \in \mathfrak{R}$, $i \in I$ (I is the instances of ontology O), $h, j, k \in N$

r' is the inverse relation of r , when $i_j, i_k \in I$, then $r(i_j) = i_k \Rightarrow r'(i_k) = i_j$

r is transitive, when $i_h, i_j, i_k \in I$, then $r(i_h) = i_j, r(i_j) = i_k \Rightarrow r(i_h) = i_k$

r is symmetric, when $i_j, i_k \in I$, then $r(i_j) = i_k \Leftrightarrow r(i_k) = i_j$

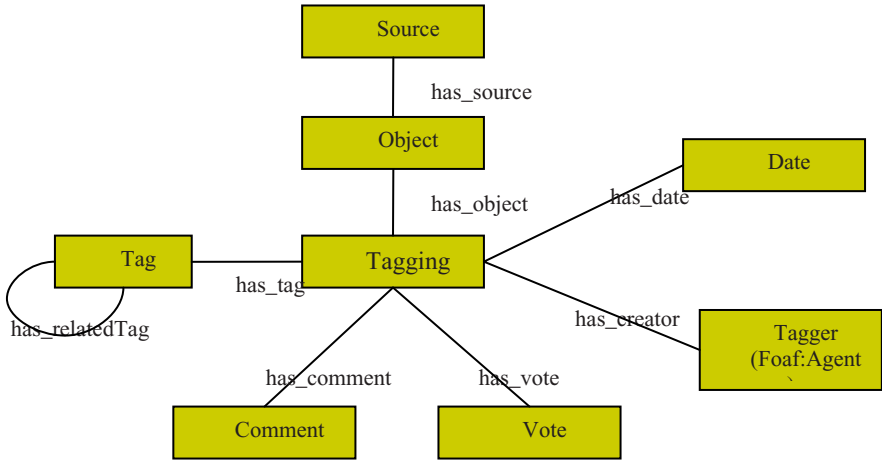


Fig. 1. Upper Tag Ontology (UTO)

Table 1. Concepts in UTO

Concept	Synonyms	Description	Value Type	Instance
Tagging		Tagging is the concept which is created to link other concepts. It, itself, does not have any real meaning.	string	e.g., tagging
Tag	keyword	Tag is the keyword which users add to object	string	e.g., design, web2.0, instructional_design, tutorials
Tagger	user	Tagger is the user who tags object	string	e.g., sborrelli
Object	Online object	Object is the thing which tagger is tagging. It can be bookmarks (URLs), photos, videos, musics, books, slides, etc.	string	e.g., www.commoncraft.com/show
Source	Social network	Source is the place where the object is hosted. It can be del.icio.us, flickr, youtube, etc.	string	e.g., del.icio.us
Comment	note	Comment is what the tagger adds to the object or tag during the tagging.	string	e.g., The CommonCraft Show 1 Common Craft – Social Design for the Web.
Date	time	Date is the time stamp of the tagging behavior. Format is “Mmm JJ”.	date	e.g., Jun 07
Vote	favorite	Tagging can be viewed as voting. Vote can be the number of different taggers tagging this bookmark (del.icio.us), a photo been favored (flickr), or a video been voted (youtube)	integer	e.g., 103 (there are 103 taggers tagged this bookmark)

Table 2. Relations in UTO

Relation	Domain	Range	Cardinality	OWL Type	Math properties	Inverse relation
has_tag	Tagging	Tag	N	Object Property	Transitive	is_tag_of
has_relatedTag	Tag	Tag	N	Object Property	Transitive, Symmetric	--
has_creator	Tagging	Tagger	1	Object Property	--	is_creator_of
has_object	Tagging	Object	1	Object Property	--	is_object_of
has_date	Tagging	Date	1	Object Property	--	--
has_source	Object	Source	N	Object Property	--	is_source_of
has_comment	Tagging	Comment	N	Object Property	--	is_comment_of
has_vote	Tagging	Vote	N	Object Property	--	is_vote_of

UTO is different comparing to folksonomy which focuses on the meaning of tags. With the basic ontology design idea of “making it easy and simple to use”, UTO is designed to capture the structure of the social tagging behavior rather than the topic or meaning of the tags. It aims to model the structure of the tagging data in order to integrate different tagging data and link them with existing social metadata.

2.2 Linking Social Data

As mentioned previously, data should be interlinked. Link is changing from normal hyperlink in Web 1.0, to typed hyperlink in Web 2.0, till semantic link in web 3.0. First of all, we try to link documents, therefore we have linked online documents as Web 1.0. Then, we are adding more metadata to those documents and turning unstructured information into structured information. Later on, we should semantically link those structured information so as to form so called Web 3.0 or Semantic Web. Social tagging plays an important role here by not only structuring information but also linking structured data.

Table 3 shows the alignment between UTO and other social metadata, such as FOAF, DC, SIOC and SKOS. Here we try to make the alignment as simple as possible because the complicated alignment may generate problems or double the complicity of application. So here we focus mainly on class mapping with the consideration of equal and sub-class mapping. For instance, “Tagger” concept equals to foaf:Person, sioc:User, dc:Contributor and dc:Creator; it is the subclass of foaf:Agent, foaf:Group, foaf:Organization and sioc:Usergroup. “Tag” concept equals to skos:Concept; it is subclass of dc:Subject and skos:Subject. “Object” concept is superclass of foaf:Document, foaf:Image, sioc:Post, sioc:Item, dc:Text and dc:Image.

Table 3. Different ontology alignment with UTO

UTO	FOAF	SIOC	DC	SKOS
Tagging	--	--	--	--
Tag	--	--	$\subseteq$ Subject	= Concept $\subseteq$ Subject
Tagger	= Person $\subseteq$ Agent $\subseteq$ Group $\subseteq$ Organization	= User $\subseteq$ Usergroup	= Contributor = Creator	--
Object	$\supseteq$ Document $\supseteq$ Image	$\supseteq$ Post $\supseteq$ Item	$\supseteq$ Text $\supseteq$ Image	--
Source	--	$\subseteq$ Community	= Source	--
Comment	--	--	$\subseteq$ Description	--
Date	--	--	= Date	--
Vote	--	--	--	
has_relatedTag	--	--	--	$\supseteq$ narrower $\supseteq$ broader $\supseteq$ related

Notes: according to formula (1), $c_i, c_j \in C$, $c_i \subseteq c_j \Leftrightarrow c_i$ is the sub-class of c_j , while, $c_i \supseteq c_j \Leftrightarrow c_i$ is the super-class of c_j , while $c_i = c_j \Leftrightarrow c_i$ equals to c_j . The same is valid for relationship.

“has_relatedTag” relationship is the super-property of skos:narrower, skos:broader and skos:related.

Aligning UTO with other existing social semantics enables easy data integration, mash-ups different semantics and interlinks structured data. Based on these integrated data, we can perform tag search across multiple sites, applications, sources, hosts and mine relations (associations) cross different platforms and applications. For instance, we can do the following queries: finding friends of Stefan who tagged “spicy-Chinese-food” by aligning FOAF with UTO; finding different blogs, wikis, or discussion groups which Stefan or his friends join and discuss the topic on “spicy-Chinese-food” by aligning FOAF, SOIC with UTO, etc. Associations among tag, tagger and objects can be mined as well. For instance, we can mine the social network relations of taggers through foaf:knows by aligning FOAF with UTO; we can mine the relation or association of tags through skos:broader, skos:narrower or skos:related; we can use co-occurrence technologies to mine the association among tags, taggers and objects, etc.

2.3 Crawling Social Tagging Data

Social Tagging crawler (in short ST crawler) is a developed multi-crawler designed for crawling major social tagging systems including del.icio.us, flickr and youtube

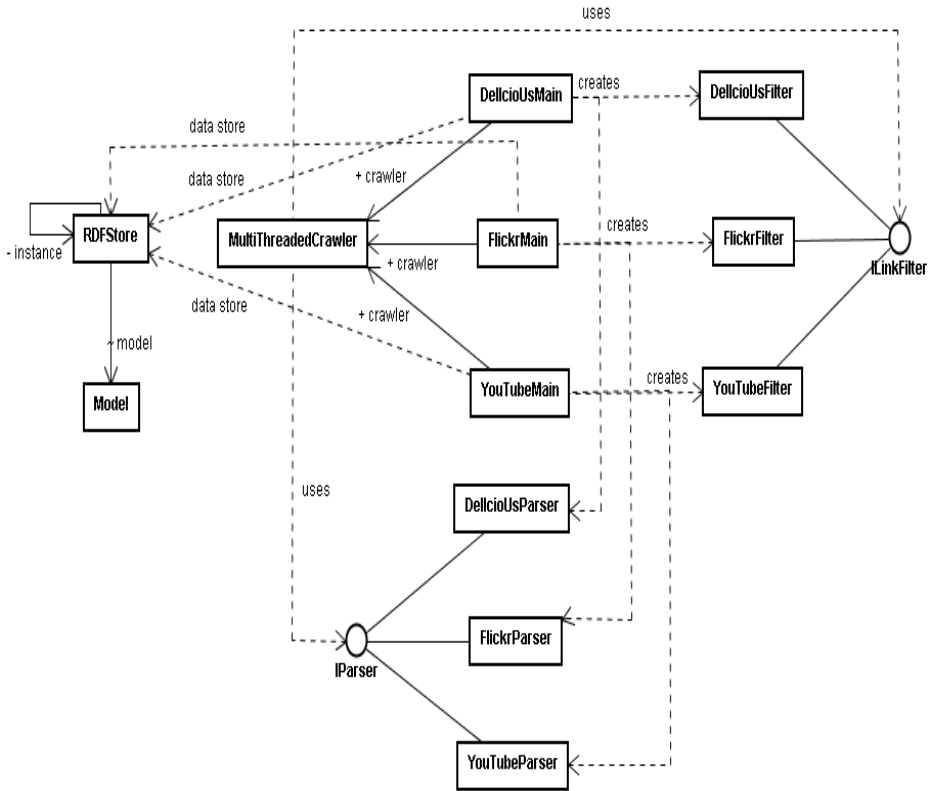


Fig. 2. Class diagram overview of the ST crawler

[4]. This crawler is based on the “Smart and Simple Webcrawler”² and UTO. Figure 2 shows the detailed class diagrams of the crawler.

The ST crawler is written in Java with Eclipse IDE 3.2 on Windows XP and Ubuntu 6.04. Data has been cleaned up using linux batch commands. ST crawler can start from one or a list of links. There are two crawling models:

- **Max Iterations:** Crawling a web site through a limited number of links. It needs a small memory footprint and CPU usage.
- **Max Depth:** A simple graph model parser without recording incoming and outgoing links. It uses filter to limit the links to be crawled.

Finally, ST crawler has crawled social tagging data from delicious, flickr and youtube and modelled them according to UTO. These data are represented in RDF triples and stored in Jena. In the summer of 2007, we use ST crawler to crawl tagging data from these three websites. After one-week crawling, the crawled output contains several RDF files with a complete file size of 2.10GB. In detail:

² <https://crawler.dev.java.net/>

- 16 del.icio.us data files at a size of 1.64GB
- 3 flickr data files at a size of 233MB
- 3 youtube data files at a size of 234MB

2.4 Clustering Social Tagging Data

Based on above crawled data, we took the 1.64GB tagging data crawled from delicious as one sample to analyze social feature of its community. The crawled tagging data from delicious contains 462,733 taggers, 404,388 tags and 483,564 bookmarks. All these tag data are represented in RDF and stored in Jena. We took the tag data as they are and did perform data cleaning (for instance, stemming and checking with WordNet). By querying these data, we got the top 20 highly ranked tags and top 20 highly ranked bookmarks during that time (see Table 4).

Table 4. Top 20 highly ranked tags and bookmarks in del.icio.us

Rank	Tag	Tag Frequency	Bookmark	Bookmark Frequency
1	blog	141,871	en.wikipedia.org	26,745
2	system	120,673	www.youtube.com	14,990
3	design	109,249	community.livejournal.com	6,594
4	software	87,719	www.google.com	6,376
5	programming	83,665	www.w3.org	6,193
6	tool	83,461	news.bbc.co.uk	5,718
7	reference	74,602	www.flickr.com	5,645
8	web	70,538	java.sun.com	5,538
9	video	65,226	www.nytimes.com	5,222
10	music	61,246	www.microsoft.com	5,219
11	art	57,970	lifel hacker.com	5,207
12	linux	47,965	www-128.ibm.com	4,569
13	tutorial	41,844	www.codeproject.com	4,429
14	java	40,780	www.wired.com	4,269
15	news	40,652	video.google.com	4,261
16	game	39,391	www.techcrunch.com	3,818
17	free	39,006	www.bbc.co.uk	3,318
18	development	37,914	www.readwriteweb.com	3,159
19	business	35,272	blogs.msdn.com	3,121
20	internet	34,580	msdn2.microsoft.com	2,950

It seems that blog topic dominates del.icio.us. Most of taggers are IT guru as system, design, software, programming, tool are ranked very high. Web and Internet are evergreen topics among the community. People like to share music, video, news, game which are popular topics in social web. People like things for “free” (as free is ranked as 17th). Highly ranked bookmarks include major social networks (youtube, livejournal, wikipedia, flickr), major news (BBC, New York Times), major computer giants (Microsoft, Google, IBM, Sun) which show the social impact of these websites.

Table 5. Tag clusters in del.icio.us

Cluster	Tags
1	ajax, c, code, development, html, java, library, net, python, rails, rudy
2	dictionary, English, language, literature, writing
3	comic, entertainment, film, forum, japan, Japanese, movie, radio, streaming, television, tv
4	calculator, conversion, convert, converter, currency, euro, exchange
5	account, bank, banking, bill, consumer, credit, deal, doctor, financial, healthcare, insurance, loan, medical, medicare, medicine, savings
6	air, apartment, building, cleaning, do, fire, guide, house, housing, move, rental, safety, studio
7	Black, blue, brown, fairy, flower, gratis, leather, line, neo, pink, red, skull, stripes, style, Sweden, Swedish, vintage, white, yellow
8	culture, history, philosophy, politics, religion
9	astronomy, earth, geography, german, map, nasa, space, world
10	font, illustration, inspiration, portfolio, typography

We conduct clustering analysis based on the same data set by using X-Means algorithm. X-Means is an unsupervised clustering algorithm which one can set minimum and maximum number of clusters while training [5]. Table 5 presents some interesting clusters from our analysis.

Cluster 1 contains 11 tags and is about programming languages. Cluster 2 has 5 tags with the topics around natural language and dictionary. Cluster 3 has 11 tags and is talking about entertainment, movie, video and radio. Cluster 4 contains 7 tags on currency conversion. Cluster 5 contains 16 tags on banking and insurance. Cluster 6 contains 13 tags on housing. Cluster 7 contains 19 tags on color. Cluster 8 on culture, Cluster 9 on geography and Cluster 10 on portfolio. Although we cannot rank clusters, comparing with Table 4 top 20 highly ranked tags, we can find out that programming languages and entertainment (video, film, movie, news and radio) are both reflected in Table 4 and Table 5. Furthermore, we can draw some interesting conclusions from Table 4 and Table 5:

- Taggers like to use adjectives (such as color) as tags to categorize their bookmarks.
- When tagging bookmarks related to currency conversion, housing and banking, taggers tend to use quite similar tags (see Cluster 4, Cluster 5 and Cluster 6)
- Two major topics in delicious are programming and entertainment. This also means that the main user groups in delicious contain users who are interested in programming and users who are interested in entertainment.

3 Related Works

In 2005, Tom Gruber proposed the idea of using ontology to model tagging data. His idea has been further formalized and published in 2007 [2]. His tag ontology contains tagging (object, tag, tagger, source, + or -). He introduced vote to tag ontology and uses it for collaborative filtering. UTO contains more concepts and relations

comparing to his tag ontology, such as date, source, comment, etc. Furthermore, UTO also focuses on integration with other existing social metadata in order to achieve data integration. UTO is based on Gruber's idea and goes a bit further on ontology alignment and data integration.

SCOT³ (Social Semantic Cloud of Tags) Ontology semantically represents the structure and semantics of a collection of tags and to represent social networks among users based on the tags. The core concepts of SCOT include Tagcloud and Tag. SCOT uses URI mechanism as unique tag namespace to link tag and resource. SCOT ontology is based on and linked to SIOC, FOAF and SKOS. It uses SIOC concepts to describe site information and relationships among site-resources. It uses FOAF concepts to represent a human or machine agent. It uses SKOS to characterize the relations between tags. While UTO does not care much of tagcloud and it is defined in such a way which can be further aligned with many other social metadata, such as DC, microformat, etc.

Holygoat Tag Ontology⁴ models the relationship between an agent, an arbitrary resource and one or more tags. Taggers are linked to foaf:agents. Taggings reify the n-ary relationship between tagger, tag, resource and data. This ontology also links itself to RSS and dc, such as rss:item, rss:category, rss:pubDate, rss:link and dc:subject by using rdfs:subClassOf or rdfs:subPropertyOf. Based on these, they can perform some simple subsumption inference. This approach goes a bit deep to semantic web by utilizing ontology reasoning and inference. UTO aims to keep things simple and easy to use therefore ontology reasoning and inference is not considered at this stage.

MOAT Ontology⁵ is a lightweight ontology to represent how different meanings can be related to a tag. It focuses on providing unique identifier to tag which associated semantic meaning to the tag. It is based on Holygoat Tag Ontology to define tag object. MOAT assumes that there exists a unique relationship between a tag and a label that a tag can have a unique MOAT identifier in the semantic web. UTO cares more about the structure of the tagging behavior rather than the meaning of the tags. But provide unique identifier to tag is always a helpful and important issue to social tagging and furthermore to web in general.

4 Conclusion and Future Work

The current Web has experienced tremendous changes to connect information, knowledge, people and intelligence. There are a couple of existing efforts trying to bring the Web to its next generation. The Semantic Web is one of the efforts embedded significantly in academic artificial intelligence area. It has the long-term vision to make the Web as the global brain of human and machine by representing data in machine understandable way and automating the mediation of data and services. Meanwhile, Web 2.0 represented Social Web has successfully motivated users to share information and collaborate each other directly via the Web [6].

³ <http://scot-project.org/>

⁴ <http://www.holygoat.co.uk/projects/tags/>

⁵ <http://moat-project.org/ontology>

Web 2.0 is not completely different from the Semantic Web [7]. As Sir Tim Berners-Lee mentioned “the Semantic Web is an extension of the current Web in which information is given well-defined meaning, better enabling computers and people to work in cooperation⁶”. Web 2.0 not only extends the communication dimensions (publishing, commenting and arguing) but also tries to add extra contextual information (we can call it “social metadata”) to the current Web data in a social and informal way (e.g. tagging, bookmarking and annotating). The power of the Semantic Web lies in the potential for interoperability through some well-defined metadata in machine understandable way and logic reasoning support [8]. Module and layer design principle in the Semantic Web (e.g. ontologies, languages and services) paves the way for reuse and intelligent search with more granularity and relevance [9]. Web 2.0 provides scalable community-powered information sharing platform, while the Semantic Web adds valuable machine understandable metadata to enable efficient and automatic way of heterogeneous information sharing and cross-portal communication and collaboration [10].

This paper takes social tagging systems as examples and aims to identify some pragmatic ways of utilizing Semantic Web and Social Web phenomena to structure unstructured information. A simple Upper Tag Ontology (UTO) is proposed to integrate social tagging data from different social networks and mediate with other related social metadata so that data are interlinked. Furthermore, the broader way of data mediation (mediate different ontological concepts or relationships) can be established based on community driven methods with the consideration of instances and contextual information. It has the following important features:

- *Community driven mediation based on collective intelligence:* Ontology mediation is one of the hardest problems in the Semantic Web which is mainly achieved formally and manually. These kinds of approaches can be hardly adopted by the Web due to the scalability issue. Social Web changes the current Web into a community platform where ordinary users participate daily for communication and collaboration. This social synergy can be used for data mediation as mediation itself is a kind of activity supporting communication and collaboration. Community driven mediation based on social collective intelligence can be an appropriate approach for data mediation. Furthermore social web services can provide further support for browsing and querying mediated data.
- *Instance-based metadata mediation:* There are already some existing researches on instance-based metadata mediation from the Semantic Web and database area. But they are more focusing on the formal transformation problem between schema and instances. Ideas on how to advance the data mining techniques to mediate metadata based on instances and contextual information around the data and metadata can be further explored. Especially, due to the Social Web effect, social involvement of the users should be significantly considered during the process and should be integrated into the approach.
- *Efficient mashing-up of Social Web services and metadata semantics:* In its current state, the Web is often described as being in the Lego phase, with all

⁶ <http://www.w3.org/2001/sw/EO/points>

of its different parts capable of connecting to one another. Properly mashing-up social services can assist the mediation process and further enable the browsing and querying of the mediated data.

Social aspect of the Web indeed influences fundamentally the usage and sharing of the web information. The Web relies on people serving useful content, linking them and providing trust and feedback. The massive participation of the web users has significantly increased the heterogeneity of the Web. On the other hand, it has created the additional way for data integration, namely integration by collective intelligence. By tagging and sharing data, intuitively they also enrich the contextual information of the concepts and relations. Here we take social tagging systems as examples to identify some pragmatic ways of utilizing Semantic Web and Social Web phenomena to realize data mediation and integration. A simple Upper Tag Ontology (UTO) is proposed to integrate different social tagging data and mediate with other related social metadata. In the future, we would like to put some efforts to mine some associations among these tagging data in order to portray tagging behavior in current social networks. We can also build up recommender systems based on these associations. Furthermore, some efficient statistical methods can be identified to extract mediation rules based on instances and contextual information.

References

1. Hoschka, P.: CSCW research at GMD-FIT: From basic groupware to the Social Web. *ACM SIGGROUP Bulletin* 19(2), 5–9 (1998)
2. Gruber, T.: Ontology of Folksonomy: A Mash-up of Apples and Oranges. *International Journal on Semantic Web & Information Systems* 3(2) (2007), <http://tomgruber.org/writing/ontology-of-folksonomy.htm>
3. Ding, Y., Toma, I., Kang, S., Fried, M., Yan, Z.: Data Mediation and Interoperation in Social Web: Modeling, Crawling and Integrating Social Tagging Data. In: *WWW2008 Workshop on Social Web Search and Mining (SWSM 2008)*, April 21–25, Beijing, China (2008)
4. Fried, M.: Social Tagging Wrapper. Bachelor Thesis, Institute of Computer Sciences, University of Innsbruck, Austria (2007)
5. Pelleg, D., Moore, A.W.: X-means: Extending K-means with Efficient Estimation of the Number of Clusters. In: *Seventeenth International Conference on Machine Learning*, pp. 727–734 (2000)
6. Hinchcliffe, D.: The State of Web 2.0. *Web Services Journal* (2006), http://web2.wsj2.com/the_state_of_web_20.htm
7. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. *Scientific American* 284(5), 34–43 (2001)
8. Antoniou, G., van Harmelen, F.: *A Semantic Web Primer*. MIT Press, Cambridge (2004)
9. Gomez-Perez, A., Fernandez-Lopez, M., Corcho, O.: *Ontological Engineering: Advanced Information and Knowledge Processing*. Springer, Heidelberg (2003)
10. Mika, P.: *Social Networks and the Semantic Web*. Springer, Heidelberg (2007)

Conceptual Synopses of Semantics in Social Networks Sharing Structured Data

Verena Kantere¹, Maria-Eirini Politou², and Timos Sellis²

¹ Ecole Polytechnique Fédérale de Lausanne
verena.kantere@epfl.ch

² School of Electr. and Comp. Engineering,
National Technical University of Athens
{politou,timos}@dbnet.ece.ntua.gr

Abstract. We are interested in the problem of data sharing in overlay networks that have a social structure, i.e. participants are linked and exchange data with others w.r.t. the similarity of their data semantics. In this paper we propose a methodology to produce conceptual synopses for the semantics that are encapsulated in the schemas of relational data that are shared in a social network. These synopses are constructed solely based on semantics that can be deduced from schemas themselves with some optional additional conceptual clarifications. The produced synopses represent in a concentrated way the current semantics. Existing or new participants can refer to these synopses in order to determine their interest in the network. We present a methodology that employs the conceptual synopsis for the construction of a mediating schema. These can be used as global interfaces for sharing of information in the social network. Furthermore, we extend our methodology in order to compress the conceptual synopsis such that infrequent concepts are eliminated and the respective inferred global schema encapsulates the most popular semantics of the social network.

1 Introduction

Social networks are structures that map semantic relations of the members to overlay links. In such a network, linkage usually follows the unstructured model, i.e. members are connected to the most similar others and are aware of a small part of the network.

We are interested in social networks that share structured data, i.e. data that adhere to a schema; Our focus is the relational model, since it is the most commonly-used one in practice to represent the structured data. Linked, or else, *acquainted* members of such networks create and maintain sets of mappings between the schemas that their data conform to. These mappings are necessary in order for the acquaintees to understand each other, not only in terms of semantics, but also in terms of data structure; these mappings offer a way to organize and compromise their intra data-sharing [1, 5].

In a broad network which hosts a set of social groups, prospective members need guidance in order to select the groups they desire to participate. Thus, they would benefit from information that is related to member ids or names, but, more essentially, to the content that is shared. Members of such networks need additional assistance in order to match their data to the data of members with similar interest. Actually, they would

benefit even more from summarizations of semantics, if they could infer from them a mapping scheme for their own data to other shared data.

A global conceptual synopsis and, furthermore, a respective mediating schema gives the opportunity to the participants of the social network to get answers that adhere better to the semantics of their queries for data, since query loss of information due to successive query rewriting is avoided [7, 14]. Beyond this, the conceptual synopsis enables joining members to obtain an overall idea, make an “educated guess” about the semantics of the data shared in this social network. Moreover, the respective mediating schema can be used for the creation of direct mappings that facilitate the data exchange with the total of participants.

A practical problem related to conceptual synopses of semantics is to limit its size such that it contains only the important semantics. This is vital in order to prevent users of the synopses to be lost in or misled by semantics that are actually of subordinate significance, in their effort to understand the nature of the respective social network.

In this paper we deal with the problem of creating a conceptual synopsis for the semantics of a social network employing solely the available schema and mapping information, as well as any optional conceptual clarifications that may be held by the network members. We aim at the minimization of human involvement in this process, as well as to offer tools for conceptual representation that are, on one hand, intuitive and can be manually used in a straightforward manner to express basic human rationale, and, on the other, capable of representing semantics that can be inferred from schemas and mappings. We explore a methodology that allows the deduction of schema and mappings semantics and their unification with additional optional manually expressed clarifications on them. This methodology creates a conceptual synopsis of the respective semantics. We employ the conceptual synopsis in order to construct a global schema that represents adequately the semantics of the respective social network. These can be used as mediating interfaces for sharing of information in the social network.

Furthermore, we consider the practical problem of refining the complete conceptual synopses in order to maintain only the dominant semantics. We solve this problem by proposing a methodology for the compression of the synopses that tracks infrequent semantics, that are also of limited interest to the members, and eliminates them.

Finally, we study thoroughly the quality of the global schemas produced with our methodology experimenting on two use cases.

After briefly discussing related work in section 2, in section 3 we formalize the problem. Section 4 describes the methodology for the deduction of the conceptual synopsis of a social network and section 5 presents the construction of the respective global schema emphasizing on compressed synopses. Section 6 summarizes the experimental study and section 7 concludes this paper.

2 Related Work

The problem of semantic schema merging is generally related to the problems of schema or ontology matching and integration. The recent survey in [13] approaches in a unified way all these problems, since they are basically dealing with schema-based matching. A survey of ontology mapping techniques is presented in [6]. The authors focus on the

current state of the art in ontology matching. They review recent approaches, techniques and tools. Once appropriate mappings between two ontologies have been established, either manually, semi-automatically or automatically, these mappings can be used to merge the two ontologies or to translate elements from one ontology to the other. Examples of tools for ontology merging are OntoMerge [4] and PROMPT [10]. However, creating and maintaining a merged ontology incurs a significant overhead. Moreover, a translation service for OWL ontologies is presented in [8]. The translation relies on a provided mapping between the vocabularies of the two ontologies.

Schema matching is a fundamental issue in the database field, from database integration and warehousing to the newly proposed P2P data management systems. As discussed in [12], most approaches to this problem are semi-automatic, in that they assume human tuning of parameters and final refinement of the results. This is also the case in some recent P2P data management approaches (e.g., [3, 11]). Generally, schema matching [12] and integration [2] are operations that adhere to schema structure in a strict way. Thus, most of the effort is concentrated in detecting and compromising contradictory dependencies and constraints.

Ontology matching/integration is a very similar problem to schema matching/integration. As discussed in [9, 13], both ontologies and schemas provide a vocabulary of terms with a constrained meaning. Yet ontologies and schemas differ in the declaration of semantics: on one hand ontologies specify strict semantics and on the other hand schemas do not specify almost at all explicit semantics. Because of this vital difference and of different aims and usage, ontology matching/integration has to follow a strict semantics structure, whereas schema matching/integration has to obey to strict structural semantic-less constraints. Moreover, different aims lead schema matching/integration to adhere to structural similarity that may or not encompass some similarity of semantics and ontology matching/integration to the opposite. Our work is an effort to complement these approaches by filling their gap. Our focus is the semantics that can be deduced from schemas without being restrained by the schema structure. Instead of making the overly strict assumption that these semantics adhere to an a priori full-fledged ontology, we consider the existence of optional basic clarifications on semantics.

3 Problem Definition

We consider a flat network of nodes (i.e. without super-nodes) that share data stored in a relational DBMS and, thus, that comply to a relational schema. The latter is a set of relations and each relation a set of attributes. The only internal constraints of a schema are foreign key constraints. Pairs of nodes of the social network maintain schema mappings in order to be able to share data. As assumed in other related works [1, 5, 7], these mappings are actually bidirectional inclusion dependencies that match a query on the one schema to a query on the other. Furthermore, each acquaintance may be enhanced with some additional optional clarifications on concept matching using a set of available types of concept correspondences. We would like to deduce the semantics of such a social network employing only the available meta-information on the shared data, i.e. schemas, mappings and correspondences. We want to form this semantics into conceptual synopses that can be used for the better understanding of the participants'

(existing and new-coming) requirements and interests on data and for the fulfillment of them, by constructing a global mediating abstract (i.e. not to be populated) schema. In the following we describe in a formal manner the assumptions of this problem and the characteristics of the pursued solution.

A social network is a pair $(\mathcal{N}, \mathcal{M}, C)$. Here, $\mathcal{N} = (S, \mathcal{L})$ is an undirected graph, where $S = \{S_1, \dots, S_n\}$ is a set of nodes, $\mathcal{L} = \{(S_i, S_j) \mid S_i, S_j \in S\}$ is a set of acquaintances; each acquaintance (S_i, S_j) is associated with a set $\mathcal{M}_{ij} \in \mathcal{M}$ of mappings and a set of correspondences $C_{ij} \in C$.

Each $S \in \mathcal{S}$ is a relational schema, i.e. is a nonempty, finite set $\{R_1[A_1], \dots, R_n[A_n]\}$, where $R_i[A_i]$, $i = 1, \dots, n$ denotes a relation R_i over an ordered set A_i of attributes. An instance of a relation $I_{R[A]}$ is a (possibly empty) finite set of tuples $\langle t_1, \dots, t_m \rangle$ where t_i , for $i = 1, \dots, m$, is an ordered set of constants c with $|t_i| = |A|$. A set $K(R_i) \subseteq A$ constitutes a key of R .

We assume the existence of a countable finite set of words $\mathcal{D}$ that constitutes the domain of the social network. This means that for each $S \in \mathcal{S}$, for each $R \in S$ the name of R , denoted as $name(R)$ takes value from $\mathcal{D}$, i.e. $name(R) \in \mathcal{D}$. In the same way, for each $A \in R$, $name(A) \in \mathcal{D}$ and for each $c \in t \in I_{R[A]}$, $name(c) \in \mathcal{D}$. Each member of $\mathcal{D}$ constitutes a distinct and possibly non-unique concept. Thus, the function $name(x) = y : \{x \mid x \in S.R, S.R.A, I_{S.R[A]}.t\} \mapsto \mathcal{D}$, gives the concept that corresponds to a schema element or a data value ($S.R$ denotes the R relation of schema S , $S.R.A$ denotes the A attribute of the R relation of schema S).

Considering two schemas S and a S' , a mapping between them $M(S, S')$ is the set $\{Equ_M(S, S'), Cond_M(S, S')\}$, where the set of equivalences of concepts $Equ_M(S, S') = \{name(R.A) = name(R'.A') \mid R.A \in S, R'.A' \in S'\}$ holds under the set of conditions $Cond_M(S, S') = \{R_1.A = R_2.B \text{ or } R_1.A = const \mid R_1, R_2 \in S \text{ or } R_1, R_2 \in S'\}$; $const$ is a data value.

Figure 1 depicts part of a social network that consists of two universities. The figure shows part of their schemas and some mappings that have been created in order

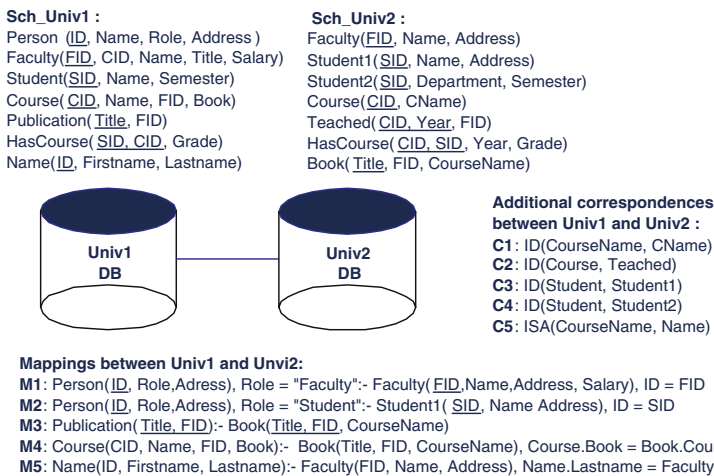


Fig. 1. Parts of the schemas of two universities that collaborate through a social network

to enable schema understanding and data sharing between them. Underlined attribute names refer to attributes that are part of the key of the respective relation. Each mapping corresponds to a conjunctive query on one schema to a conjunctive query on the other.

Beyond mappings, we consider that two acquainted nodes can also declare conceptual correspondences in order to optionally clarify or specify some conceptual relation between two schema elements, i.e. relations, attributes, or attribute values.

A conceptual correspondence CC is a directed relationship between the concepts that correspond to two schema elements E_1, E_2 (i.e. a relation R , an attribute A , or an attribute value c). The concept of a schema element E is denoted as $name(E)$. Thus a conceptual correspondence CC between E_1, E_2 , is declared as $CC(name(E_1), name(E_2))$. Note that the two schema elements do not have to be of the same type. Such a correspondence can be of 4 types: $CC \in \{ISA, ID, HASA, REL\}$. Especially the type ID is bidirectional, i.e. $ID(name(E_1), name(E_2)) \Leftrightarrow ID(name(E_2), name(E_1))$. The interpretations of the correspondence types are pretty straightforward and, thus, very easy to be used by administrators in order to declare some conceptual relations between the concepts of the schema elements E_1 and E_2 , i.e. $name(E_1)$ and $name(E_2)$, respectively:

- $ISA(name(E_1), name(E_2))$: $name(E_1)$ is a specialization of $name(E_2)$
- $ID(name(E_1), name(E_2))$: $name(E_1)$ is identical with $name(E_2)$ and vice versa.
- $HASA(name(E_1), name(E_2))$: $name(E_2)$ is part of $name(E_1)$
- $REL(name(E_1), name(E_2))$: $name(E_1)$ is in generally related by an unspecified manner to $name(E_2)$

The ID type of correspondence means that the two members are different textual interpretations of exactly the same concept. The REL type of correspondence is associative. This type can be used by the administrator if she wants to declare that two concepts are related but she does not (a) want to specialize it to one of the other three types, (b) does not know if this relation can be specialized by another type, or (c) believes that this is a kind of relationship that cannot be represented by the other three types.

The correspondence types are not equally strong. The hierarchy of the four types described above is $(ID \succ ISA \succ HASA \succ REL)$, where $cc_j \succ cc_k$ means that cc_j is stronger than cc_k . This means that if there are more than one correspondence links between two schema elements, then the strongest one obliterates the rest.

In Figure 1 some examples of optional correspondences are shown. These can be very easily and intuitively formed in addition to the mappings for the schemas of the two universities by their administrators, as clarifications. For simplicity, in the examples we omit the function $name(\cdot)$ and we denote corresponding concepts and schema elements with the same symbol.

A *conceptual synopsis* of a social network $(\mathcal{N}, \mathcal{M}, C)$ is represented by a directed labeled graph $CG = (V, E)$, where each vertex $v \in V$ is a distinct concept and each edge $e \in E$ is a correspondence. Specifically, each vertex $v \in V$ corresponds to one or more schema elements of the nodes participating in the social network, i.e. $v = \{name(x) \mid x \in S.R, S.R.A, I_{S.R[A]}, S \in \mathcal{S}\}$; also, each edge $e \in E$ corresponds to an element of C that includes correspondences that have been explicitly expressed and added to C at the point of acquaintance creations, or correspondences that are deduced in some way from the mappings $\mathcal{M}$. Note that a conceptual synopsis can summarize all or some of the semantics of the social network.

A global schema GS of a social network is a relational schema that is coherent with the respective conceptual synopsis represented by CG . This means that each concept and each correspondence in CG is represented in a lossless way in GS , such that we can use GS in order to reconstruct CG . GS can be employed as a mediating schema for data sharing in the social network.

The conceptual synopsis is a flatter and simpler version of an elaborated ontology; yet it is still very expressive since it allows any kind of four essential kinds of relationship between any two nodes. The conceptual synopsis is an intuitive description of a set of concepts and can be easily constructed by given simple concept correspondences.

In the following sections we will describe algorithms that can construct the conceptual synopsis of a social network by employing existing concept correspondences and deducing correspondences from the schema mappings. Moreover, we will discuss how a conceptual synopsis can be compressed in order to summarize the most frequent concepts. Finally we will present the algorithm for the construction of the global schema that corresponds to a conceptual synopsis.

4 Creation of Conceptual Synopses

In this section we describe the steps for the creation of a conceptual synopsis that represents the complete semantics of a social network. Briefly, a conceptual synopsis is created for each individual schema that participates in the social network. These synopses are merged in a serialized way (according to existing acquaintances) employing predefined concept correspondences as well as correspondences that are deduced from the existing schema mappings. Finally the merged conceptual synopsis is refined. The algorithm is summarized in Figure 2 and described in detail in the following.

Creation of the conceptual synopsis

Input: Two relational schemas S_1 and S_2 , a set of mappings $\mathcal{M}_{12}$, a set of additional concept correspondences C_{12} and an existing conceptual synopsis $CG = \{V, E\}$

Output: A global conceptual synopsis $CG' = \{V', E'\}$

Initialization: $CG' = CG$

Step1: Represent S_1 and S_2 as a conceptual synopsis, CG_1 and CG_2 , respectively.

Step2: For each mapping $M \in \mathcal{M}$:

- extract the conceptual correspondences C
- add these correspondences to the existing ones: $C \cup C$

Step3: Merge the conceptual synopses CG_1 , CG_2 with CG'

Step4: Refine the CG' by:

- adding the correspondences in C
- removing subsumed correspondences

Step5: Return CG' .

Fig. 2. Algorithm for the creation of the global conceptual synopsis

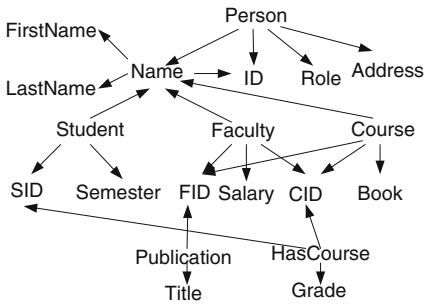


Fig. 3. Conceptual synopsis from Univ1

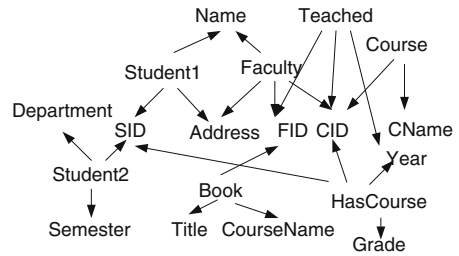


Fig. 4. Conceptual synopsis from Univ2

4.1 Creating the Conceptual Synopsis of a Schema

We use the schema of each node that participates in the social network in order to deduce a relevant conceptual synopsis. This synopsis represents in a concise and intuitive manner the domain for which this node stores and shares data. In order to produce the conceptual synopsis of a schema we create one vertex for each relation in the schema. Formally, for a schema $S = \{R_1, \dots, R_m\}$ of m relations $CG_S = (V_S, E_S)$ of a schema S , we instantiate the set of vertices as $V_S = \{V_1, \dots, V_m\}$, such that $V_i = \text{name}(R_i)$, for $i = 1, \dots, m$. For each relation $R_i = \{A_{i1}, \dots, A_{in}\}$ we add to V_S respective vertices for all the attributes A_{ij} , for $j = 1, \dots, n$. The set of edges E_S is instantiated such that there is one entry E_i in the set for each pair of vertices (V_i, V_{ij}) where V_i is the respective vertex of relation R_i and V_{ij} is the respective vertex of the attribute A_{ij} of R_i . Assuming that foreign key constraints between the i^{th} and k^{th} relations are actually represented as sets of equivalences: $A_{ij} \equiv A_{km}$, for some j and some m in the arity of the respective relations, the conceptual synopsis is connected¹. Also, note that duplicates in V_S are collapsed, in order for the synopsis to represent a concept uniquely. Hence, after the elimination of the duplicates, each vertex in V_S has a unique name, which is the name of the concept that it represents.

Figures 3 and 4 show the conceptual synopses that are created from the schemas of the two universities of Figure 1. For simplicity, the graphs in these and the following figures do not label the HASA correspondences.

4.2 Merging Conceptual Schemas

A set of conceptual schemas are merged sequentially, employing the mappings that they hold in pairs. In order to merge two conceptual synopses there are two coarse steps: (a) first, we add edges that connect semantically the graphs, and (b) second, we collapse vertices with the same name, i.e. vertices that represent the same concept. In order to add inter-graph edges, we employ the knowledge we may have about the semantics interrelations of the schemas that are the origins of these graphs. These interrelations

¹ Note that the conceptual synopsis may be a non-connected graph. This occurs in the rare case that the relations of a schema do not have any foreign key constraints.

Extracted correspondences from mappings between Univ1 and Univ2 :

C6: ISA (Faculty, Person)	C9: ISA(Book, Publication/Book)
C7: ISA(Student, Person)	C10: REL(Course, CourseName)
C8: ISA(Publication, Publication/Book)	C11: REL(Faculty, LastName)

Fig. 5. Conceptual correspondences extracted from the mappings

are denoted by situations such as: identical relation or attribute names, identical relation keys, value conditions on attributes, etc. Depending on the presence of these situations conceptual correspondences between schema elements can be deduced. Such rules can be derived from basic and intuitive rationale as well as from studies of use cases. We have concluded with the following set of rules that guide the procedure of the deduction of concept correspondences from the mappings between two schemas.

For a mapping M between two schemas S_1, S_2 :

- If there is a relation $R_1 \in S_1$ and a relation $R_2 \in S_2$ for which $name(R_1) = name(R_2)$ and the share the same key: $\forall A_1 \in K(R_1), \exists A_2 \in K(R_2)$, s.t. $name(A_1) = name(A_2)$, and vice versa, then the respective vertices are joined with an *ID* correspondence.
- If there is a relation $R_1 \in S_1$ and a relation $R_2 \in S_2$ that share all their attributes, i.e. $R_1(A_1, \dots, A_k)$ and $R_2(A_1, \dots, A_k)$, where $name(R_1.A_i) = name(R_2.A_i)$, for $i = 1, \dots, k$, then the respective vertices of R_1 and R_2 are joined with an *ID* correspondence.
- If there is a relation $R_1 \in S_1$ and a relation $R_2 \in S_2$ that share the same key and there is a value condition on one of them, e.g. $R_1.A_j < \text{constant} >$, then a *ISA*(V_2, V_1) correspondence is added for V_2, V_1 which are the corresponding vertices of R_2, R_1 , respectively.
- If there is a correspondence between two attributes of the two involved schemas: a relation attribute $A_1 \in R_1 \in S_1$ corresponded in the mapping with a relation attribute $A_2 \in R_2 \in S_2$ and $name(R_1) = name(R_2)$, then we add *REL*(V_1, V_2), where V_1, V_2 are the corresponding vertices of R_1 and A_2 , respectively.
- If there is a relation $R_1 \in S_1$ and a relation $R_2 \in S_2$ that share the same key then we add a new vertex V and we add the correspondences *ISA*(V_1, V) and *ISA*(V_2, V), where V_1, V_2 are the corresponding vertices of R_1 and R_2 , respectively.

Figure 5 shows the correspondences that can be deduced from the schema mappings of Figure 1 employing the described set of rules. Using these correspondences, as well as the optional correspondences defined by the administrators (see Figure 1), the conceptual synopses of the two university schemas (see Figures 3 and 4) can be merged. Figure 6 shows the first step of merging, where only *ID* correspondences have been processed. We remind that *HASA* correspondences are not labeled.

4.3 Refining the Global Conceptual Synopsis

After the global conceptual synopsis is produced, it is often the case that there are redundant edges between pairs of vertices of the graph. Thus, the latter is refined so that it contains only one edge between each pair of vertices. The following simple steps are taken:

- merging of vertices that are linked with an *ID* correspondence
- eliminating all subsumed correspondences
- eliminating $ISA(V_1, V_k)$ correspondences, if there exist also the correspondences $ISA(V_i, V_{i+1})$, for $i = 1, \dots, k - 1$
- substituting $HASA(V_i, V_j)$ correspondences, if there is a set of correspondences $ISA(V_i, V_{i+1})$, for $i = 1, \dots, k - 1$ with the correspondence $HASA(V_k, V_j)$

It is evident that the role of *ID* correspondences in the synopsis is associative, since they actually denote that there is a duplication of a concept. In order to simplify the picture of the synopsis, we eliminate the *ID* correspondences; these can be entered in an accompanying dictionary, which can be referenced later by joining members of the social network. Moreover, it is often the case that after the merging of individual conceptual synopses, there are multi-linked vertices. Hence, we eliminate all the edges between two vertices except one: the one with is over the others in the hierarchy of correspondences. Also, we eliminate redundant *ISA* correspondences. Finally, we substitute *HASA* correspondences to specialized (through *ISA* ones) with the *HASA* correspondences towards the most generalized respective concepts. Figure 7 shows the refined merged conceptual synopsis for the schemas of the two universities of Figure 1.

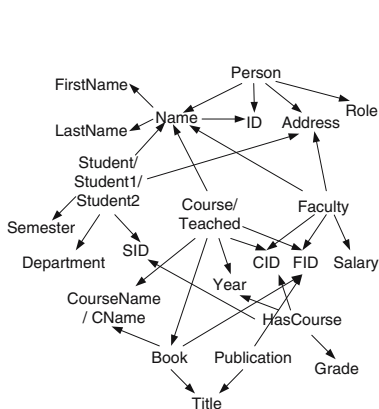


Fig. 6. Merged conceptual synopsis after adding *ID* correspondences

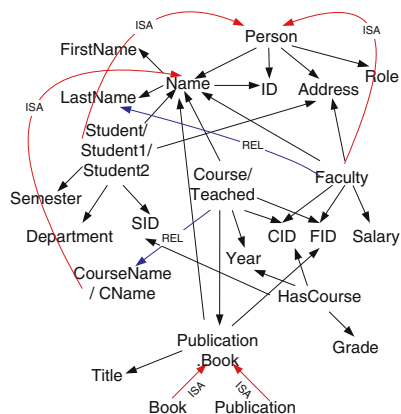


Fig. 7. Final global synopsis

5 Creation of the Global Schema

The conceptual synopsis can be employed in order to produce a global abstract schema that can be used as a mediator for the data sharing of the members of the social network. In the following we discuss how this global schemas can be constructed from the complete conceptual synopsis, but, also, from compressed versions of the latter. The complete conceptual synopsis encapsulates the total of the concepts that are included in the semantics of all the participants in the social network. Yet, it is often the case that the individual semantics of each member, comprise also concepts that are only of local interest; such concepts are not representative of the common network semantics and, therefore, it is suitable to eliminate them from the respective conceptual synopsis.

GlobalSchemaExtraction

Input: A conceptual synopsis $CG = \{V, E\}$

Output: The global schema $GS = \{R_i\}$, for some integer i

Initialization: $GS = \emptyset$

Step1: For each vertex $V_j \in V$ that has outgoing edges of type *HASA* or *ISA* create a new respective relation R_j , i.e. $name(R_j) = V_j$, and add this to GS .

Step2: For each created relation R_j , insert an attribute A_{jk} for each respective concept $V_{jk} \in V$, i.e. $name(A_{jk}) = V_{jk}$, that has an incoming *HASA* correspondence from V_j .

Step3: For each relation R_j that corresponds to a concept V_j that has no outgoing *ISA* edges, determine the subset of the attributes that constitute the key.

Step4: For each relation R_j that corresponds to a concept V_j that has outgoing *ISA* edges, define the key of the relation to be the set of the following attributes:

- add to R_j the attributes that are part of the key of each relation R_k that corresponds to a concept V_k for which there is an edge $ISA(V_k, V_j)$; make these part of the key.
- optionally add more existing attributes of R_j to the key.

Step5: For each relation R_j that corresponds to a concept V_j that has no outgoing *ISA* edges, optionally add more existing attributes of R_j to the key.

Step6: If there are two relations R_i, R_j and there is an attribute of the first, A_{ik} , such that both A_{ik} and R_j correspond to the same concept in V , then:

- if there are no edges $REL(V_j, V_{in})$ such that there is also $ISA(V_{in}, V_i)$, where $V_i = name(A_i)$, substitute A_{jk} with the key of relation R_i ,
- else, substitute A_{jk} with the attributes A_{in} that correspond to the concept V_{in} .

Step7: Return GS .

Fig. 8. Construction of the global schema extraction from the conceptual synopsis

5.1 Complete: Keeping All Concepts

The general algorithm that produces a global mediating schema involving all the concepts of a conceptual synopsis is presented in Figure 8. Summarizing, the algorithm creates a relation in the global schema for each concept of the conceptual synopsis that comprises other concepts (outgoing *HASA* correspondences) or are specializations of other concepts (outgoing *ISA* correspondences). Relations that are created from specialized concepts inherit as foreign keys the keys of relations that correspond to the most generalized concepts (respective concepts with no outgoing *ISA* correspondences). We comment here that in order to produce the keys of the relations, we must have some knowledge about at least the basic concepts that are deduced from relation keys in the participating individual schemas². Otherwise keys have to be selected either randomly or based on some heuristic (e.g. number of incoming/outgoing and type of edges); yet, such a method cannot guarantee to produce the most rationally selected keys in the

² We omit a full discussion on the determination of keys due to lack of space. We note that knowledge of the eligibility of concepts to produce keys must formally be encoded in the structure of the conceptual synopsis. However, for the sake of simplicity, we have omitted this formality in this paper.

Sch_Global_wo_compression:

Person (ID, Role, Address)
 Faculty(FID,ID Lastname, CID, Salary)
 Student(SID, ID, Semester, Department)
 Course(CID, CourseName, FID, Title, Year)
 Publication.Book(Title, FID, ID)
 HasCourse(SID, CID, Grade, Year)
 Name(ID, Firstname, Lastname)
 Book(Title)
 Publication(Title)

Sch_Global_w_compression_1:

Person (ID, Role, Address)
 Faculty(FID,ID Lastname, CID, Salary)
 Student(SID, ID, Semester, Department)
 Course(CID, CourseName, FID, Title, Year)
 Publication.Book(Title, FID, ID)
 HasCourse(SID, CID, Grade, Year)
 Name(ID, Firstname, Lastname)

Sch_Global_w_compression_2:

Faculty(FID,ID Role, Address, Lastname, CID, Salary)
 Student(SID, ID, Role, Address, Semester, Department)
 Course(CID, CourseName, FID, Title, Year)
 Publication.Book(Title, FID, ID)
 HasCourse(SID, CID, Grade, Year)
 Name(ID, Firstname, Lastname)

Fig. 9. Global schema construction from the global conceptual synopsis

general case. Finally, *REL* correspondences are checked in order to specialize the concept representation in the global schema by suitably replacing some relation attributes. Figure 9 presents the global schema constructed from the conceptual synopsis of Figure 7.

5.2 Compressed: Eliminating Infrequent Concepts

Sometimes a global conceptual synopsis is very large since it comprises not only concepts that are frequent among the participants in the social network, but also the seldom ones that interest only very few participants. Hence, there is a need for an algorithm than can produce a global schema that includes only the most frequent, and, therefore, most popular concepts. In order to achieve this, we propose the compression of the conceptual synopsis so that infrequent concepts are eliminated. Then, the summarized global schema is constructed with the algorithm of Figure 8 from the compressed conceptual synopsis. The compression of the latter is performed with the algorithm shown in Figure 10.

The algorithm is guided by the coarse rationale that a global schema is preferred to include fewer relations with more attributes, rather than more relations with fewer attributes. The reason is that this global schema is intended to be used as a mediator with which existing or new participants will have to create schema mappings. The latter are easier to be constructed if there is not much need for joins between relations. Nevertheless, the global schema is not purposed to be populated; thus, there is no fear that the few relations with many attributes will be filled with sparse tuples. Therefore the algorithm chooses to eliminate concepts that do not have outgoing *HASA* but may have incoming *ISA* correspondences. Overall, the algorithm is guided by the logic that elimination of concepts that are specializations or that are multi-linked should be avoided, since this would cause permanent loss of semantics and, as a side-effect, additional loss of more semantics due to probable disconnection of the graph. The algorithm terminates when a pre-specified limit of compression has been reached. This limit refers to the size of the

Conceptual_Synopsis_Compression

Input: A global conceptual synopsis $CG = \{V, E\}$

Output: A compressed conceptual synopsis $CG' = \{V', E'\}$

Initialization: $CG' = CG$

Step1: Concept elimination of is guided by the following set of rules, checked in ascending order. Concept V_i is removed if:

Rule1: There are no outgoing *ISA* edges from V_i and no incoming *HASA*.

Rule2: There are the fewest outgoing *ISA* edges from V_i and no incoming *HASA*.

Rule3: There are no outgoing *ISA* edges from V_i and there are the fewest incoming *HASA*.

Rule3: There are the fewest outgoing *ISA* edges from V_i and the fewest incoming *HASA*.

Rule4: There are the fewest outgoing *ISA* edges from V_i , and the fewest incoming and outgoing *HASA*.

Rule5: There are the fewest outgoing *HASA* correspondences.

Step2: Concepts V_j that had an outgoing *ISA* edge to an eliminated concept V_i , inherit the latter's *HASA* edges.

Step3: Check if the required limit of compression is reached. If no, goto Step1.

Step4: Return CG' .

Fig. 10. Compression of the conceptual synopsis

C12: ID(CourseName, Name)

M6: Book(Title, FID, CourseName):-Name(ID, Firstname, Lastname) , Book.CourseName = Name.ID

Fig. 11. Solving the problem of misleading declarations about social network semantics

conceptual synopsis and can be expressed either in terms of storage requirements or in terms of the size of the global schema to be constructed from the synopsis. We prefer the second of the two and, specifically, we determine the schema size in terms of the number of included relations, since these are the principal schema features. Figure 9 shows the global schemas after compressing the conceptual synopsis twice and three times.

5.3 Problem Limitations

After presenting our approach for the creation of conceptual synopses and respective mediating schemas, we briefly comment on the natural limitations imposed by the assumptions of the problem. First, the quality of the conceptual synopsis, and therefore, of the mediating schema depends in a straightforward manner on the quality of the mappings and the correctness of the additional conceptual correspondences. Ideally, complete mappings and consistent correspondences between acquainted members can lead to the creation of a representative conceptual synopsis. However, the lack of an a priori default agreement on concept matching, makes it impossible to guarantee the creation of infallible conceptual synopses. For example, observe the additional correspondence *C5* in Figure 1, which denotes that “CourseName” is a special kind of “Name”. If the latter is indeed used in a very broad manner, then this estimation is correct;

however, if “Name” turns out to refer only to people’s names, then this estimation is wrong. Wrong or controversial concept matching estimations can be compromised with concept correspondences that are deduced from mappings and subsume the first. Moreover, the *REL* correspondence type can indicate special usage of concepts and can lead to a more correct schema construction. For example, assume that, additionally to the correspondences and mappings of Figure 1, there is correspondence *C12* and mapping *M6* in Figure 11. Thus, *C12* subsumes *C5*, and, from *M6*, the correspondence *C13* : *REL(Book, CourseName)* is deduced. The latter leads to a refinement of the produced respective relation in the global schema: *Publication.Book(Title, FID, CourseName)*.

Mistaken or inconsistent estimations on concept matching are possible and even unavoidable as the semantics of the social network refer to a broader domain of life. Naturally, broad concepts that are used with several meanings (such as the concept “Name”) are certain to provoke confusion of semantics. Yet, social networks that target a more specific domain of knowledge, e.g. domain of a specific science, profession, sport, etc, are more eligible to use our proposed approach to the problem of conceptual synopsis in the lack of a global default ontology.

6 Experimental Study

In this section we present a summary of the experimental study that we have conducted in order to measure the efficiency of our technique in creating conceptual synopses and global schemas for social networks.

Experimental setup. In all the series of experiments that we have performed we have measured the similarity of the individual schemas of the participant databases in the social group with the global schema that is constructed from the deduced respective conceptual synopsis. The schema similarity comprises three partial similarity metrics; average similarity of (a) relations, (attributes), and (c) relation keys. Due to lack of space we present results only for the metrics (a) and (b). We note that we do not employ an overall schema similarity metric, since we believe that similarity of individual schema features is more informative.

We have conducted three groups of experiments. The first group studies the similarity of the global schema that is constructed from the complete conceptual synopsis, with the individual participating schemas. The second group studies the similarity of the global schema after compression of the conceptual synopsis, with the individual participating schemas. Finally, the third group of experiments studies the role of compression on the major schema features.

Experimental data. The experimental study has been performed for individual schemas of databases that participate in two social networks: (a) a network of hospitals and, (b) a network of universities. Specifically, for each one of these two domains, we have created a big pool of related concepts; we have given the latter to people with good knowledge of the database field and we have asked them to produce a relevant original schema with names of schema features or even data values that come from the respective pool of concepts. After collecting these original schemas, we have artificially produced additional new schema groups in order achieve schema similarities with values approximate

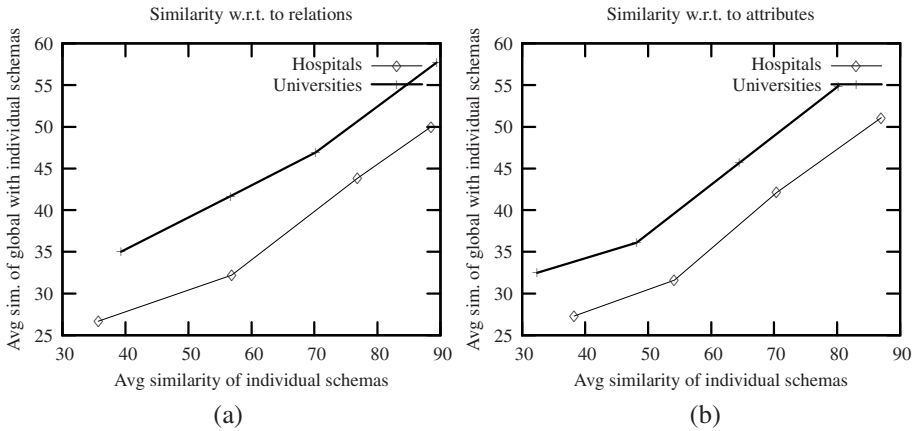


Fig. 12. Results for global schema constructed from the full conceptual synopsis

to the ones required by our experiments. The total of 100 (50 for the domain of hospitals and 50 for that of the universities) individual schemas is the input of our technique; the intermediate output of the latter are the conceptual synopses and the final output are the global schemas. It is interesting to note that the similarity of the original schemas w.r.t. to the relations is compatible with the respective similarity of keys. Overall, we followed this compatibility for the altered schemas.

6.1 Results for Similarity of Global and Individual Schemas

Figures 12(a),(b) show the average similarity of the global schema (inferred from the complete global conceptual synopsis) with the individual schemas, versus the average similarity of the individual schemas. Figure 12(a) shows results on the similarity of relations and 12(b) on the similarity of attributes. Both show smooth increasing average similarity of global with individual schemas, as the similarity of the latter increases. This is good since it indicates that the global schema encapsulates the overall semantics of the social network and the more coherent this semantics is, the more of it can be found in the global schema. However, the figures show that the similarity of the global with individual schemas is slightly more influenced by the similarity of the relations than the attributes of the individual schemas. Rationally, relations are considered to be more dominant schema elements and more determinant for the semantics of the schema (for example the lack of a relation influences more the schema semantics than the lack of an attribute, even if the lack of the relation does not entail the lack of attributes, too). Finally, the gradient of the synopses, (which is more abrupt as similarity increases), shows that as the individual schemas are more similar, the global schema naturally turns out to be overall more similar to all of them.

6.2 Results for Similarity of Compressed Global Schema and Individual Schemas

Figures 13(a), (b) show the average similarity of the compressed global schema (i.e. the global schema that is constructed after compression of the global conceptual synopsis)

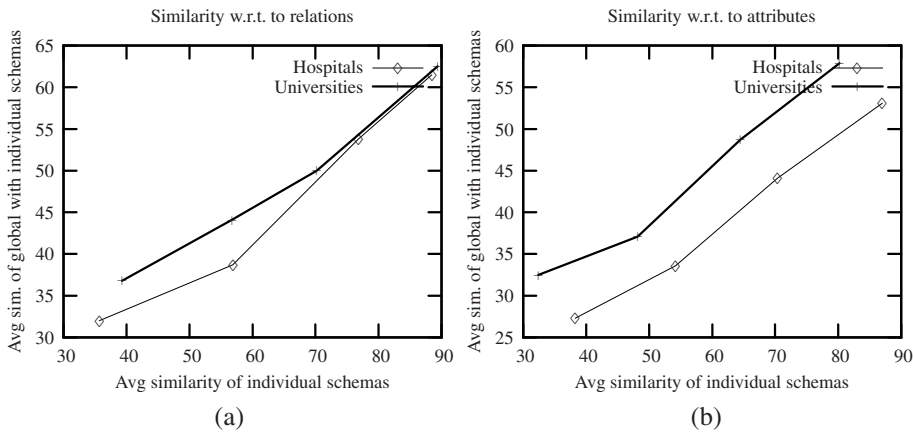


Fig. 13. Results for global schema constructed from the compressed conceptual synopses

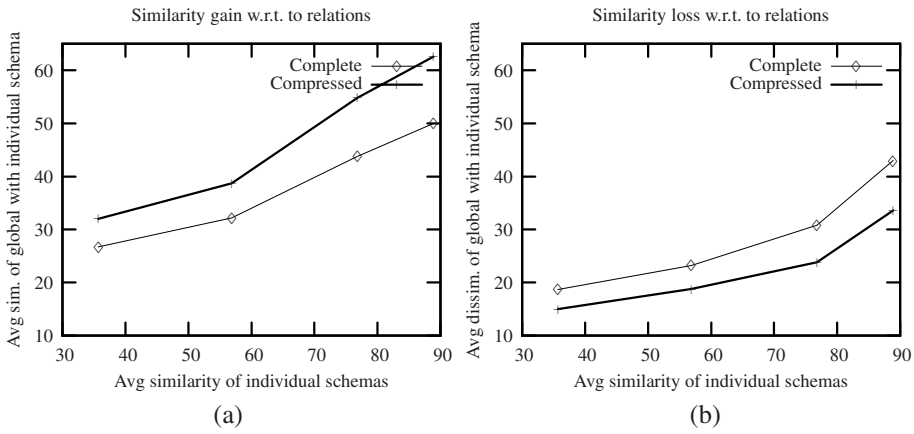


Fig. 14. Results for gain and loss in similarity with the individual schemas w.r.t. relations for complete and compressed global schema

versus the overall similarity of the individual schemas. Note that the degree of compression is constant in this set of experiments. The results are analogous to those of the respective Figures 12(a), (b). Again, it is verified that the role of relations is more critical to schema similarity than the role of attributes and, that, as the overall similarity of the individual schemas increases, their similarity with the compressed global schema increases with a faster rate. Moreover, a comparison between Figures 12, 13 reveals that the compressed schema is slightly more similar in general with the individual schemas than the respective complete global schema. We explore this interesting result more:

As indicated by Figure 13(a), the elimination of relations in the compressed global schema increases the overall similarity of it with the individual schemas. This means that the removed relations are indeed rare ones. As the similarity of individual schemas increases, this effect is even more obvious, since the rate of similarity increment

becomes bigger (this is very evident for the social network of hospitals). Figure 13(b) shows that the similarity of attributes is not so much affected by the compression, since the latter does not influence a lot the attributes.

Figure 14 shows the results for experiments on the gain and loss in similarity of the compressed and the complete schema with the individual schemas focusing on relations. The compressed schema differs from the complete schema in that some relations of the latter are not there in the first.

The gain in similarity is actually derived from the elimination of respective dissimilarity of the global compressed schema with the individual ones, due to the elimination of infrequent schema elements. The relations that are eliminated from the compressed schema, are still there in some individual schemas. This causes increment of the dissimilarity of the first with the latter, and constitutes actually the loss in similarity between them.

Figure 14(a) shows that the compression w.r.t. relations not only increases the similarity of the global schema with the individual ones, but also increases the rate of similarity increment. This means that there is substantial gain in similarity w.r.t. relations for the compressed vs. the complete global schema. Moreover, Figure 14(b) shows that the compression w.r.t. relations decreases also the dissimilarity of the global schema with the individual ones, which verifies that the eliminated relations are rare in the individual schemas. Naturally, the dissimilarity increases as the overall similarity of individual schemas increases, for both the complete and the compressed schema; yet, the rate of dissimilarity increment, and, therefore, the loss in similarity w.r.t. relations, is smaller for the compressed than the complete global schema.

This result is coherent with the intuitive rationale that the importance of schema features, (relations, in the case of these experiments), to the semantics of the social network is proportional to their frequency among the individual participating schemas. Going a step further and taking into consideration the experimental results, this means that the quality in terms of representative semantics of the global schema depends on the similarity of the individual schemas: the more similar they are, the better the complete and the worse the compressed global schema is. Hence, as final outcome, we form the following proposition:

For social networks with very similar participants the complete conceptual synopsis is necessary for the construction of the global schema, as each concept is valuable to the semantics of the network; whereas, for those with dissimilar participants, compression of the conceptual synopsis is a better option, as it tends to maintain the most important/frequent semantics and disregard the rest.

6.3 Results for Similarity for Different Degrees of Global Schema Compression

Figures 15(a), (b) show the results for experiments about various degrees of compression on the conceptual synopsis. These experiments are performed on schemas with 5 relations; Figure 15(a) shows results for the complete global schema as well as for 3 degrees of compression, where each degree corresponds to the elimination of one relation. Thus, the compression reaches up to 60% of the average size of the individual schemas in terms of relations. The average similarity of the global schema with

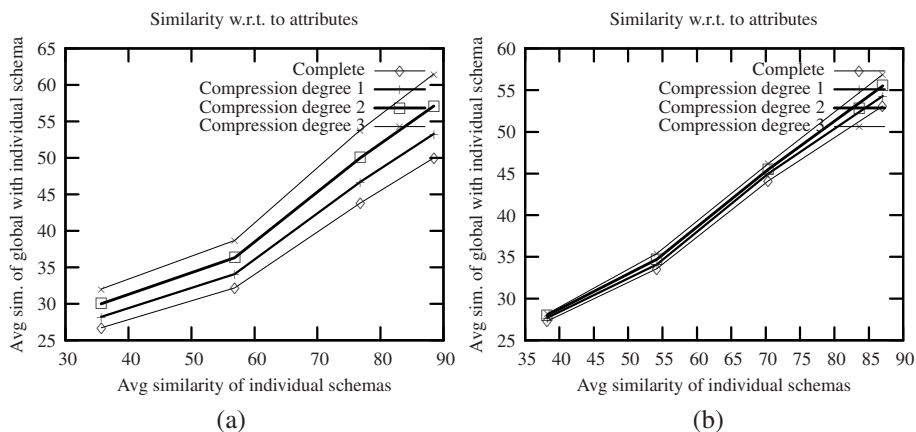


Fig. 15. Results for similarity of the compressed global schema with the individual ones for several compression degrees

the individual schemas increases with compression even up to 60%, even for schemas that are quite similar. With reference to previous experiments on gain and loss of similarity, this means that gain is bigger than loss, even for high degrees of compression w.r.t. relations. Figure 15(b) shows similar the results for compression w.r.t. attributes. In these experiments compression degrees refer to elimination of attributes ³: degree i means that one attribute is eliminated. Naturally, the elimination of an attribute does not have a big good or bad impact to the average similarity of the global schema with the individual ones, although as this similarity increases, this impact becomes greater.

7 Conclusions

We tackle the problem of creating a conceptual synopsis for the semantics of a social network that shares relational data. We focus on networks that base their communication in schema mappings and that may hold some clarifications about conceptual matching. We propose a methodology for the deduction of conceptual correspondences from the schema mappings and integration of them in a refined way so that a synopsis that represents concept interrelations is produced. Using this synopsis we create a global mediating schema. We elaborate on the problem of producing a schema that maintains the most popular concepts of the social network and eliminates the concepts of limited interest. Finally, we perform an experimental study on the quality of the global schema.

References

1. Arenas, M., Kantere, V., Kementsietsidis, A., Kiringa, I., Miller, R.J., Mylopoulos, J.: The hyperion project: from data integration to data coordination. *SIGMOD Record* 32(3), 53–58 (2003)

³ Of course, compression of attributes is achieved after elimination of relations has reached the maximum according to the compression rules: i.e. all the relations that are derived from ISA correspondences in the global conceptual synopsis are eliminated.

2. Batini, C., Lenzerini, M., Navathe, S.B.: A comparative analysis of methodologies for database schema integration. *ACM Comput. Surv.* 18(4), 323–364 (1986)
3. Bernstein, P.A., Melnik, S., Churchill, J.E.: Incremental schema matching. In: *VLDB*, pp. 1167–1170 (2006)
4. Dou, D., McDermott, D.V., Qi, P.: Ontology translation on the semantic web. *J. Data Semantics* 2, 35–57 (2005)
5. Halevy, A., Ives, Z., Suciu, D., Tatarinov, I.: Schema Mediation in Peer Data Management Systems. In: *ICDE* (2003)
6. Kalfoglou, Y., Schorlemmer, M.: Ontology Mapping: The State of the Art. *Knowl. Eng. Rev.* 18(1), 1–31 (2003)
7. Kantere, V., Tsoumakos, D., Sellis, T., Roussopoulos, N.: GrouPeer: Dynamic Clustering of P2P Databases. Technical Report TR-2006-4, National Technical University of Athens (2006) (to appear in *Information Systems Journal*),
<http://www.dbnet.ece.ntua.gr/pubs/uploads/TR-2006-4>
8. Mota, L., Botelho, L.: OWL Ontology Translation for the Semantic Web. In: *Proceedings of the Semantic Computing Workshop of the 14th International World Wide Web Conference* (2005)
9. Fridman Noy, N.: Semantic integration: A survey of ontology-based approaches. *SIGMOD Record* 33(4), 65–70 (2004)
10. Fridman Noy, N., Musen, M.A.: Prompt: Algorithm and tool for automated ontology merging and alignment. In: *AAAI/IAAI*, pp. 450–455 (2000)
11. Ooi, B., Shu, Y., Tan, K.L., Zhou, A.Y.: PeerDB: A P2P-based System for Distributed Data Sharing. In: *ICDE* (2003)
12. Rahm, E., Bernstein, P.: A Survey of Approaches to Automatic Schema Matching. In: *VLDB Journal* (2001)
13. Shvaiko, P., Euzenat, J.: A survey of schema-based matching approaches. *J. Data Semantics* IV, 146–171 (2005)
14. Tatarinov, I., Halevy, A.: Efficient Query Reformulation in Peer-Data Management Systems. In: *SIGMOD* (2004)

Sequential Patterns for Maintaining Ontologies over Time

Lisa Di-Jorio, Sandra Bringay, Cline Fiot, Anne Laurent,
and Maguelonne Teisseire

LIRMM – Université de Montpellier 2 – CNRS
161 rue Ada, 34392 Montpellier, France

Abstract. Ontologies are known as a quality and functional model, allowing meta data representation and reasoning. However, their maintenance plays a crucial role as ontologies may be misleading if they are not up to date. Currently, this work is done manually, and raises the problem of expert subjectivity. Therefore, some works have developed maintenance tools but none has allowed a precise identification of the relations that could link concepts. In this paper, we propose a new fully generic approach combining sequential patterns extraction and equivalence classes. Our method allows to identify terms from textual documents and to define labeled association rules from sequential patterns according to relevance and neighborhood measures. Moreover, this process proposes the placement of the found elements refined by the use of equivalence classes. Results of various experiments on real data highlight the relevance of our proposal.

1 Introduction

Ontologies offer a generic model for knowledge representation. These special structures are widely used for capturing knowledge of a particular domain of interest, and for easing data manipulation and exchange. As this knowledge is constantly evolving, ontologies must be updated, by adding, deleting or replacing knowledge. Often, new knowledge is reported on textual documents. Ontology updating implies selection of interesting terms from various documents in a first time, and then a placement of these terms, either as new concepts, or as a new relation labels. For a human expert, regarding the quantity of data to mine, this is a difficult, time consuming and tedious work. Moreover, it differs from one expert to another one depending on their point of view.

One way to overcome expert subjectivity is the use of automatic tools. Mostly based on statistical or syntactic analysis, these tools focus on finding and adding new concepts. However, as far as we know, none of them allows detection of one important specific knowledge: the links between concepts, and the terms labeling them. These elements are the specificity of ontologies as they model semantic knowledge.

Expanding an ontology can be done through a feedback loop combining web mining and formal semantic [1]: data mining extracts new terms to add, and the

placement is done using some semantic logic rules. With these new elements, the process is repeated as many times as possible. Most of the time, the user is involved into the loop, either to manually place a new discovered element, or to label a relation, or to validate an adding... However, applied data mining techniques often result in a large quantity of elements, making such tools inefficient. Furthermore, no tool automatically fulfills the entire process: that is, look for new concepts/relations and place them into the ontology at the right level of abstraction. We propose in this paper a fully automatic process to expand a given ontology, based on data mining techniques and on equivalence classes. Our contribution is twofold. On the one hand we propose a method to select new terms through sequential patterns mining and to categorize them as concepts or as relation labels. On the other hand we define an equivalence class based approach to allow the most precise term placement, and to process a large quantity of discovered data. Uncovered elements are grouped according to concepts and are added at the right place through labeled relations.

The rest of this paper is organized as follows. First, we give an overview of the problem statement, presenting current work (Section 2). Then, we detail our contribution, explaining each step of the ontology expanding process (Section 3). Finally, experiments described in Section 4 highlight the relevance of our proposal.

2 Related Work

2.1 Ontologies and Maintenance

It is a challenging issue to define precisely what is an ontology, since this term is used in many areas, from philosophy or linguistic to artificial intelligence. According to [2], an ontology can be viewed as "*an explicit specification of a conceptualization*". They allow both data exchange and human / machine readability. Often, ontologies describe objects of the real world as concepts, and formalize relations linking them either as hierarchical relations, or as semantical relations. Most of the time, these relations are labeled by a word or an expression. These relations stress the difference between ontologies and other structures giving a semantic description of concepts interaction.

Example 1. *Let us consider the environmental area. **Storm** and **rainstorm** are specific to the atmospheric disturbance. Figure 2 illustrates part of this ontology.*

All the approaches presented in this section follow two generic steps. First, documents are preprocessed, i.e. words are replaced by their lemma, that is the generic form of a word. For example, the word "*is*" will be replaced by "*to be*", allowing to consider words regardless of their grammatical form. At this stage, lemmatized words are called *terms*. Among them are new potential candidates for the maintenance, such as new concepts or new relation labels. Then, enriching ontologies consist in selecting these terms as a first step. Lately, we have

noticed two major tendencies for term selection: statistical based methods, and syntactic based methods. We describe these methods in the following section.

2.2 Statistical Based Methods

Statistical methods consist in counting the number of occurrences of a given term in the corpus. The more frequent a term is (according to a measure), the more it will be considered as a candidate. In a statistical context, many measures have been proposed, mostly based on term distribution in the corpus. The easiest way is to count the number of apparitions of each term among the entire corpus [3,4,5]. More complex measures have also been defined. For instance, [6,7] successively test mutual information, Tf.Idf, T-test or statistic distribution laws, resulting in a good terms selection. However, these measures never take the domain into account, nor terms appearing alone. To overcome the domain representation problem, [8] proposes a new definition of mutual information. Whereas experiments show that representative terms are extracted, some relevant terms remain uncovered. Moreover, as these methods only select terms without any external information, it is impossible to distinguish concept terms from relation labels. So, all the extracted terms are considered as potential new concepts.

Then, discovered elements shall be placed into the ontology. Because of the quantity of extracted terms, this step cannot be manually done. To avoid a manual placement of a huge quantity of terms, [4] proposes to use term co-occurrence implying one or more existing concepts of the ontology. This involves knowing where a new concept should be added. However, they only bring the closest new concepts, and never add them at a precise level with a precise relation (ie, no relation label found). We thus argue that this method will not generate semantic knowledge.

Other approaches use data mining techniques, such as classification (grouping items in an already known class) or clustering (grouping items to, but in a class found during the process). [7] and [9] bring new terms close to existing ones in the ontology thanks to a classification method. Similarly, [3] and [5] group extracted terms using a clustering method. Each cluster shows a possible relation between grouped terms. However, it is not possible to define what kind of relation it is, or to label it.

Therefore, statistical methods allow new term selection and correlation detection with existing terms, but do not place new terms directly into the ontology. Moreover, statistical methods ignore the linguistic structure of the analyzed sentences, which can give information about relations linking concerned concepts. Using syntactic methods can overcome the problem.

2.3 Syntactic Based Methods

Methods based on syntax consist of a grammatical sentence analysis, most of the time preceded by a part of speech tagging (POS) process. Syntactic methods suppose that grammatical dependencies reflect semantic dependencies [10,11]. Thus, two syntagms are considered as concepts, and the grammatical relation linking them

as a semantic relation. These considerations allow adding extracted terms as new concepts at the right place, linking them to the right existing concept.

However, these approaches suffer from the same problems as statistical ones: a huge quantity of related terms are extracted, as there exists more than one grammatical dependency into a sentence. Therefore, data mining techniques are also applied in some approaches. [1,10,12] extract association rules from the syntactic dependencies. Association rules have been proposed by [13] and allow strong correlation detection like "*when the term A is employed, the term B is employed too*". These correlations highlight frequent grammatical dependencies and thus are a good way to prune many insignificant dependencies. Some syntactic based work defines regular expressions in order to find terms corresponding to one and only one kind of relation. For instance [14] looks for hyponyms form large text corpora.

Even though syntactic based approaches automatically put new terms into the existing ontology, they do not label new relations. Once again, the extraction of a common semantic structure from a large text corpus is missed. Relation labeling has to be done by the user. Moreover, data mining techniques are always used as a second step of the generic enriching process whereas these techniques can be directly used to extract new terms. Indeed, efficient techniques have been developed, allowing among other options to restrict term selections by semantic or time constraints [15,16].

However no work uses sequential patterns in order to detect or add new elements, eventhough these structures have been proved to be efficient [17] for large text databases mining. Sequential patterns lead to a finest text analysis as they store word apparition order and frequently co-occurring words. Moreover, sequential patterns keep a track of the document structure without requiring any external knowledge. We present in this paper a fully automatic approach based on sequential patterns extraction. Indeed, we propose to use them to discover new concepts and relations labels which link them to other concepts.

3 SPONTEX: Sequential Patterns for Ontology Expansion

3.1 Overview

In this section, we introduce **SPONTEX**, a new algorithm for expanding an ontology, by means of sequential patterns. Our general process, described by Figure 1, starts from these patterns. However, taken as an ordered list of words, sequential patterns need to be transformed and refined to raise a new kind of knowledge: concepts and semantic relations. We call them **labeled association rules**, associative relations between concepts expressed by a term (label).

Labeled association rules are generated through two steps: (1) words that may share semantic with an existing concepts are first selected and organized around this concept. These candidates constitute a **neighbor set**. In order to refine the search space, we define a new measure, called **closeness**. (2) Among these words are concept terms and relation labels. The generation of labeled association rules disambiguates the role of terms. At the end of the second step, the process is

close to be finished: we know new concepts terms, the existing concept they can be attached to, and the label of the relation linking them. However, we have not organized our new terms around concepts so far. This is the aim of the third step. We propose to use **equivalence classes**. All the details and formalization are provided at section 3.6. Finally, we add elements that have been discovered to the existing ontology during step 4. In order to respect our formal ontology definition (following section), we check that we do not add a same word as a term concept and a label relation.

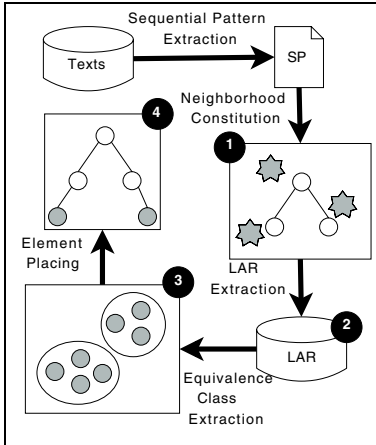


Fig. 1. General Process

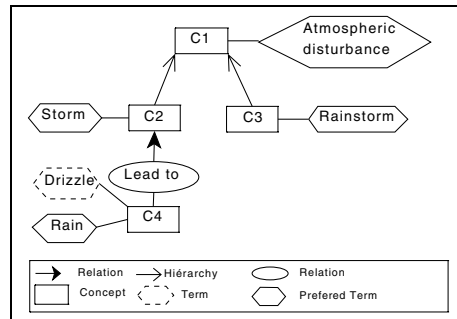


Fig. 2. Ontology Example

3.2 Sequential Patterns

Sequential patterns were originally introduced by [18]. They model an ordered list of itemsets usually associated to a given time period.

Let $\mathcal{O}$ be an object set and $\mathcal{I}$ be a set of **items** stored in a database **DB**. Each record E is a triplet $(id\text{-}object, id\text{-}date, itemset)$ as illustrated by table 1. An **itemset** is a non empty set of items from $\mathcal{I}$ represented by $(i_1, i_2, \dots, i_n)$. A **sequence** S is defined as an ordered and non empty list of itemsets $\langle s_1 s_2 \dots s_n \rangle$. A n -sequence is a sequence of length n (containing exactly n items).

Thus, **DB** associates a list of items to the object $id\text{-}obj$ at the date $id\text{-}date$ and can be represented in a *object-sequence* manner, as shown in table 2. In this table, the sequence $S = \langle (a)(b\ c) \rangle$ associated to object 1 means that the item a has been recorded, then b and c together. S is a 3-sequence.

Given a sequence $S' = \langle s'_1 s'_2 \dots s'_n \rangle$ and a sequence $S = \langle s_1 s_2 \dots s_m \rangle$, S' is included into S if and only if there exist integers $a_1 < a_2 < \dots < a_n$ such that $s'_1 \subseteq s_{a_1}, s'_2 \subseteq s_{a_2}, \dots, s'_n \subseteq s_{a_n}$. S' is then called a *subsequence* of S and S is a *supersequence* of S' .

Table 1. A transaction database example

<i>id-object</i>	<i>id-date</i>	<i>itemset</i>
1	1	a
1	3	b c
2	2	d e
2	3	d

Table 2. A sequence database example

<i>id-object</i>	<i>sequence</i>
1	<(a)(b c)>
2	<(d e)(d)>

For example, $S' = \langle (a)(b) \rangle$ is a subsequence of S because $(a) \subseteq (a)$ and $(b) \subseteq (b\ c)$. On the other hand, $\langle (b)(c) \rangle$ is not a subsequence of $\langle (b\ c) \rangle$.

An object o supports a sequence S if and only if S is included into the data sequence of this object. The *sequence support* (also called *frequency*) $Freq()$ is defined as the number of objects of the database **DB** supporting S . Given a threshold $minSupp$, a sequence S is frequent if $Freq(S) \geq minSupp$.

Extracting sequential patterns from a database like **DB** means finding all the maximal sequences (not included in others) which support $Freq()$ is at least equal to $minSup$. Each maximal sequence is a **sequential pattern**.

During this last decade, efficient algorithms for sequential patterns extraction have been proposed [18,19,20,21].

As sequential patterns were initially introduced to deal with market data, we need to transpose the context in order to apply extraction from a text documents database. Here, a date is represented by one or more sentences, and an item corresponds to a lemmatized word. Considering that a sentence is a unit of time, extracting the sequence $\langle (rain)(cause\ inundation) \rangle$ means that among all the documents, the word "rain" frequently occurs in a sentence, followed by the co-occurrence of the words "cause" and "inundation".

3.3 Formal Definitions of an Ontology and a Neighbor Set

We need to formally state what ontologies are, in order to properly use them during the adding process. In agreement with a common point of view, an ontology is constituted by concepts organized into a hierarchy. A concept is an object with associated terms describing their semantic. Computers infer knowledge starting from these concepts, and human understand their sense when reading the words associated to these ones. Moreover, our aim is to represent possible interactions between these concepts. This is done thanks to the definition of semantic relations. We use the following definition, initially proposed by [22]:

Definition 1. Let $\mathcal{C}$ be a concept set, $\mathcal{T}$ a term set, $\mathcal{R}_c$ a relation (between concepts) set, $\mathcal{R}_t$ a relation (between terms) set and $\mathcal{L}$ a relation's label set (semantic name of a relation). An ontology **O** is defined by:

$$\mathbf{O} = \{\mathcal{C}, \mathcal{T}, \mathcal{R}_c, \mathcal{R}_t, \mathcal{L}, <_c, f_{tc}, f_{rc}\}$$

with :

- $<_c : \mathcal{C} \times \mathcal{C}$ is a partial order relation on $\mathcal{C}$ defining concept hierarchy,
 $<_c(c_1, c_2)$ means that c_1 is more generic than c_2
- $f_{tc} : \mathcal{C} \rightarrow \mathcal{T}$ is the association function between a preferred term and a concept
- $f_{rc} : \mathcal{R}_c \rightarrow \mathcal{C} \times \mathcal{C}$ is an associative function between concepts

To avoid any confusion, we consider that an ontology concept is designed by one of its associated *term*. This term is said to be the **preferred term** of the concept. Talking about semantic relation amounts to use its **relation label**.

Example 2. Figure 2 shows a part of an ontology about atmospheric disturbances. Rectangles represent concepts, diamonds represent terms and ellipsis show relations.

The concept set $\mathcal{C}$ group $\{C1, C2, C3, C4\}$, the term set is $\mathcal{T} = \{\text{Atmospheric disturbance, Storm, Rainstorm, Rain, Drizzle}\}$, and the relation set $\mathcal{R}_c$ is formed by only one relation, which label is *Lead to*. "Atmospheric disturbance" is the preferred term of C_1 concept: when we talk about C_1 , we talk about all the atmospheric disturbances phenomena. $f_{rc}(\text{Lead to}) = (C_2, C_4)$ is a relation meaning that storm leads to inundation.

A concepts hierarchy $<_c$ is indicated by simple arrows. For example, $C1$ is more specific than $C2$.

By keeping track of frequent words co-occurring and their order, sequential patterns are an efficient tool for data extraction. However, taken as an ordered list of words, they do not allow to directly infer semantic knowledge. We thus raise the following questions: *how can we use them for an ontology updating process? How can we distinguish a word associated to a concept from a word associated to a relation?*

Among extracted sequential patterns are some already known terms, as they are already referenced by the ontology. We propose to refine the search space by only considering neighbors of these concepts. A neighbor of a given concept c_o is a term that can be accessed by using one link (hierarchical or semantic) from c_o :

Definition 2. Let c_o be a concept, the neighbor set $\mathcal{V}_{c_o}$ of c_o is defined as the concepts c and relations r set:

$$\forall c \in \mathcal{V}_{c_o}, \exists r \subseteq \mathcal{R} \mid f_{rc}(r) = (c_o, c) \vee f_{rc}(r) = (c, c_o) \vee <_c(c_o, c) \vee <_c(c, c_o)$$

This notion allows the association of new terms extracted by sequential patterns with existing concepts. In the rest of this paper, a term candidate appearing in a sequential patterns is called an **item**.

Example 3. The neighbor set associated to the "Storm" concept is $\mathcal{V}_{storm} = \{\text{"Rain", "Atmospheric disturbance"}\}$, because $f_{rc}(\text{Lead to}) = (\text{"Storm", "Rain"})$, and $<_c(\text{"Atmospheric disturbance", "Storm"})$.

The term "Rain" represents one of the ontology's concept of the picture 2 and "Lead to" is a label of a relation of the ontology, whereas "Cause" or "Inundation" are items of the sequential pattern $<(\text{Rain})(\text{Cause Inundation})>$.

3.4 Constructing the Neighbor Set

Each sequential pattern containing a concept term may participate to the neighbor set construction. Obviously, we cannot consider that every item of such a sequence is a neighbor, as the set cardinality will quickly explode. Therefore we propose to use a measure based on sequence support in order to add an item as a neighbor of a known term. This measure, called **closeness**, indicates the neighborhood degree between a term and is defined as follow:

Definition 3. Let S be a sequential pattern, i and c_o two different items from this sequence such as $c_o \in \mathcal{T}$. The **Closeness measure** of the item i as a term or a relation label of the c_o neighbor set is defined by :

$$Closeness(c_o, i) = \max \left(\begin{array}{l} \max(\frac{Freq([(i \ c_o)])}{Freq([(c_o)])}, \frac{Freq([(i \ c_o)])}{Freq([(i)])}), \\ \max(\frac{Freq([(i)(c_o)])}{Freq([(i)])}, \frac{Freq([(i)(c_o)])}{Freq([(c_o)])}), \\ \max(\frac{Freq([(c_o)(i)])}{Freq([(c_o)])}, \frac{Freq([(c_o)(i)])}{Freq([(i)])}) \end{array} \right)$$

Here, word order does not infer on the neighbor set. Three configurations are possible : (1) the word frequently appears in the same sentence than the term, (2) the word frequently appears before the term and (3) the word frequently appears after the term. By keeping the best apparition proportion of a configuration order relative to only one item, we consider the influence of each item on another. Moreover, as we are only interested in the best configuration, we keep the maximum proportion rate among the three possible configurations. Example 4 illustrates this idea:

Example 4. Table 3 shows extracted sequences from a document set.

Table 3. Extracted sequences

Sequence	<i>Freq</i>	Sequence	<i>Freq</i>
[(rain inundation cause)]	0.4	[(inundation)(cause rain)]	0.2
[(rain inundation)(cause)]	0.3	[(rain inundation)]	0.5
[(rain)(inundation cause)]	0.3	[(rain)(inundation)]	0.5
[(rain)(inundation)(cause)]	0.2	[(inundation)(rain)]	0.6
[(rain cause)(inundation)]	0.5	[(rain cause)]	0.5
[(rain)(cause)(inundation)]	0.3	[(rain)(cause)]	0.6
[(inundation)(rain)(cause)]	0.5	[(cause)(rain)]	0.5
[(cause)(rain)(inundation)]	0.3	[(rain)]	1
[(inundation)(cause)(rain)]	0.3	[(inundation)]	0.7
[(inundation cause)(rain)]	0.3	[(cause)]	0.7

The item "rain" already belongs to the ontology shown on the Figure 2 as a concept term. Let us compute "rain" and "inundation" closeness rate.

$$\begin{aligned}
 \text{Closeness}(\text{"rain"}, \text{"inundation"}) &= \\
 \max \left(\max \left(\frac{\text{Freq}([(\text{inundation rain})])}{\text{Freq}([(\text{rain})])}, \frac{\text{Freq}([(\text{inundation rain})])}{\text{Freq}([(\text{inundation})])} \right), \right. & (1) \\
 \max \left(\frac{\text{Freq}([(\text{inundation})(\text{rain})])}{\text{Freq}([(\text{rain})])}, \frac{\text{Freq}([(\text{inundation})(\text{rain})])}{\text{Freq}([(\text{inundation})(\text{rain})])} \right) & \\
 \left. \max \left(\frac{\text{Freq}([(\text{rain})(\text{inundation})])}{\text{Freq}([(\text{rain})])}, \frac{\text{Freq}([(\text{rain})(\text{inundation})])}{\text{Freq}([(\text{rain})(\text{inundation})])} \right) \right) & \\
 = \max(\max(\frac{0.5}{1}, \frac{0.5}{0.7}), \max(\frac{0.6}{1}, \frac{0.6}{0.7}), \max(\frac{0.5}{1}, \frac{0.5}{0.7})) & \\
 = \max(0.71, 0.86, 0.71) = 0.86 &
 \end{aligned}$$

The building of the neighbor set is done through Algorithm 1, called *Neighbor-Generation*. Given a set of sequential patterns, an already known concept set and a minimal closeness threshold given by the user, the algorithm returns the set $\mathcal{V}$ of all the neighbor sets V_{C_i} of the ontology concepts. More formally:

$$\forall V_{C_i} \in \mathcal{V}, V_{C_i} = \{(\text{item } j_1, \text{closeness}(C_i, j_1)), \dots, (\text{item } j_n, \text{closeness}(C_i, j_n))\}$$

This allows to store the closeness rate between a concept C_i and an item j . Example 5 shows such a set $\mathcal{V}$.

Example 5. Bold sequences from table 3 are sequential patterns. Algorithm 1 will successively test the following closeness threshold :

- $\text{Closeness}(\text{Rain}, \text{Inundation}) = 0.86$
- $\text{Closeness}(\text{Rain}, \text{Cause}) = 0.71$

Algorithm 1. NeighborGeneration

Data: Sequential patterns set $\mathcal{S}$,
The pattern prefixed tree **PSP**,
The ontology **O**
 minProx the minimal closeness
threshold fixed by the user

Result: $\mathcal{V}$, the set of all closeness relations

```

1  $\mathcal{V} \leftarrow \emptyset$ 
2 foreach  $s \in \mathcal{S}$  do
3   foreach  $c_o \in \mathcal{C}$  such as  $c_o \in s$  do
4     foreach  $i \in s$  such as  $i \neq c_o$  do
5       if  $\text{Prox}(c_o, i) \geq \text{minProx}$  then
6          $\mathcal{V}_{c_o} \leftarrow i$ 
7       end
8     end
9    $\mathcal{V} \leftarrow \mathcal{V}_{c_o}$ 
10 end
11 end
12 return  $\mathcal{V}$ 

```

With a minimal closeness threshold of 0.5, algorithm 1 will return the set $\mathcal{V} = \{\mathcal{V}_{C_4}\}$, with $\mathcal{V}_{C_4} = \{(Inundation, 0.86), (Cause, 0.71)\}$.

Note that building this neighbor set is only selecting new items to add to the ontology. However, we ignore if these items are concept terms or relation labels. For example, we do not know if the words "Inundations" or "Cause" selected in example 5 are new concept terms, or new relation labels. This will be determined through the extraction of labeled association rules, as it is explained in the following section.

3.5 Extracting Labeled Relations

We notice that when a document addresses two concepts linked by a relation, *the relation label is frequently employed in the same sentence as one of the two concepts*. Therefore, we need to determine which item is frequently used in the same sentence as a known concept term. Sequential patterns provide this information, so we propose to exploit it with the following relationship measure:

Definition 4. Let c_o be a term such that $\mathcal{V}_{c_o} \in \mathcal{V}$, i and j two items from $\mathcal{V}_{c_o}$ such as $i \neq j$. Then, the **relationship level** of the item i as a relation label between c_o and j and is defined by:

$$RL_i(c_o, j) = \max \left(\begin{array}{l} \frac{Freq([(i \ j \ c_o)])}{Freq([(j \ c_o)])}, \frac{Freq([(c_o)(i \ j)])}{Freq([(c_o)(j)])}, \\ \frac{Freq([(c_o \ i)(j)])}{Freq([(c_o)(j)])}, \frac{Freq([(j)(i \ c_o)])}{Freq([(j)(c_o)])}, \\ \frac{Freq([(j \ i)(c_o)])}{Freq([(j)(c_o)])} \end{array} \right)$$

The relationship level represents the proportion of documents which employed terms c_o and i in the same sentence, or terms c_o and j in the same sentence. This proportion could be considered as a kind of confidence, as it represents the maximal probability that i co-occurs with c_o knowing j or that i co-occurs with j knowing c_o .

The relationship level is not bounded to confidence rating. As a relation between two concepts is frequently co-occurring with one of these concepts, RL permits to distinguish a word role as a concept or as a relation. If $RL_i(c_o, j) > RL_j(c_o, i)$, this means that $(i-c_o)$ and $(i-j)$ co-occurs more frequently than $(j-c_o)$ and $(j-i)$. In that case, i is more likely to be a relationship according to our observations. This is why an item role is determined by the greatest confidence, i.e., the greatest RL.

Example 6. From the pattern set shown in Figure 3, two relationship levels can be computed: $RL_{cause}(rain, inundation)$ and $RL_{inundation}(rain, cause)$. $RL_{cause}(rain, inundation)$ represents the relationship level of "cause" as being a relation linking the concepts "rain" and "inundation"; and $RL_{inundation}(rain, cause)$ represents the relationship level of "inundation" as being a relation linking "rain" and "cause" concepts.

Here, we only detail the $RL_{cause}(rain, inundation)$ calculation:

$$\begin{aligned}
 & RL_{cause}(rain, inundation) \\
 = & \max \left(\frac{\frac{Freq([(cause\ inundation\ rain)])}{Freq([(inundation\ rain)])}}{\frac{Freq([(rain)(cause\ inundation)])}{Freq([(rain)(inundation)])}}, \frac{\frac{Freq([(rain\ cause)(inundation)])}{Freq([(rain)(inundation)])}}{\frac{Freq([(inundation\ cause)(rain)])}{Freq([(inundation)(rain)])}}, \right) \\
 & \frac{Freq([(rain)(inundation)]}{Freq([(inundation)(cause\ rain)])}}{\frac{Freq([(inundation)(rain)]}{Freq([(inundation)(rain)])}} \\
 = & \max(\max(\frac{0.4}{0.5}, \frac{0.4}{0.5}), \frac{0.5}{0.5}, \frac{0.3}{0.5}, \frac{0.3}{0.5}) \\
 = & \max(0.8, 1, 0.6, 0.5, 0.33) = 1
 \end{aligned}$$

In the same fashion, we find that $RL_{inundation}(rain, cause) = 0.8$.

As $RL_{inundation}(rain, cause) < RL_{cause}(rain, inundation)$, we can consider that "inundation" is a new concept, and "cause" is a relation linking it to the existing concept "rain".

Summarizing, at this stage, we have started from sequential patterns to select a set of potential new knowledge that can be linked to existing concepts. By building the neighbor set, we kept only interesting words. Thanks to the relationship level, we have decided if these new elements will be considered as new concept terms, or as new relation labels. We can now formalize a new kind of association rules integrating semantic knowledge, called **labeled association rules**:

Definition 5. A **labeled association rule** or LAR, denoted by $i \xrightarrow{r} j$, defines the implication of an item j by an item i , according to the relation r .

The existence of such a rule between an item and a concept from the ontology indicates the existence into the ontology of a relation linking this concept to this item.

Definition 6. The left part of a labeled association rule is the **acting** concept of the relation, and the right part is the **receiving** concept of the labeled relation.

A labeled association rule characterizes a relationship level as well as the relation direction. So, for each association of three items i , j and k with k corresponding to a concept c_o of the ontology, we can determinate by the relationship level calculation if one of the others items i or j define a relation between c_o and the third item.

The relation direction has been calculated during the relationship level determination. We define the **implication rate** of a labeled association rule. It represents the document proportion from the textual base for which having items c_o and j imply having item i .

Definition 7. Let i , j , c_o be a triplet such that i is the label of a labeled association rule between j and c_o . The **implication rate** of a labeled association rule $c_o \xrightarrow{i} j$ is given by:

$$IR(c_o \xrightarrow{i} j) = \max \left(\frac{Freq([(c_o)(i\ j)])}{Freq([(c_o)(j)])}, \frac{Freq([(c_o\ i)(j)])}{Freq([(c_o)(j)])} \right)$$

A high implication rate confirms that the relation i is a link between c_o and j . Therefore, from a triplet formed by a candidate concept j , an item i and a concept c_o , we compute the implication rates of the rules $(j \xRightarrow{i} c_o)$ and $(c_o \xRightarrow{i} j)$. The rule having the best implication rate is kept while the other one is left out.

In order to optimize the iteration number on sequential patterns and subsequences and consequently to reduce the execution time, we conceived Algorithm 2, named *LARGeneration*. This one computes through a single iteration the relationship level and the direction of the labeled association rules together.

Algorithm *LARGeneration* determines, from ontology concept neighborhood, the items which label relations or which are new terms.

Algorithm 2. LARGeneration

Data: The neighborhood set $\mathcal{V}$,

The pattern prefixed tree **PSP**

Result: The labeled association rules set RAL

```

1   $RAL \leftarrow \emptyset$ 
2  foreach  $\mathcal{V}_{c_o} \in \mathcal{V}$  do
3    foreach  $j \in \mathcal{V}_{c_o}$  do
4      foreach  $k \in \mathcal{V}_{c_o}$  such as  $k > j$  do
5         $ral = \text{Max}(RL_j(c_o, k), RL_k(c_o, j))$ 
6         $\text{ImplicationRate}(ral)$ 
7         $RAL \leftarrow ral$ 
8      end
9    end
10 return  $RAL$ 
11 end

```

This algorithm takes as input a prefixed tree called **PSP**. Proposed by [19], the prefixed tree is an efficient way to store sequential patterns. Each node is associated with an item. A PSP tree contains two kinds of edges: dashed edge representing the notion "and after" and plain edge representing the notion "in the same time". Leaves of a PSP are sequential patterns that can be read from the root to the leaves. This structure is output from our sequential pattern mining algorithm, and we choose to exploit it directly during the LAR mining process.

Given a set of all the neighbors and the PSP tree obtained from the sequential patterns extraction process, Algorithm 2 generates all possible labeled association rules. We iteratively compute the implication level $RL_j(c_o, i)$ and $RL_i(c_o, j)$ by combining each item couples i, j of the neighbor set $\mathcal{V}_{c_o}$ and a concept c_o (lines 1-5). Once the item role is determined, we apply the implication rate calculation in order to fix the new LAR sense (line 6) and finally add it to the LAR set (line 7).

3.6 Grouping LAR

At this stage, we have obtained a large quantity of new concepts and labels of relations which link them to existing concepts. It is now possible to add these new elements to the ontology. However, this means creating a concept for each selected term, and thus associating only one term by concept. We would overload the existing ontology with too many new concepts, missing benefit from concept philosophy (a concept allows for grouping equivalent words) and biasing user navigation during the validation step. Therefore, new terms need to be effectively grouped around common concepts.

Terms linked to a same concept through the same relation label share a common semantic. It is thus possible to exploit this property in order to group them by the use of equivalence classes. That way, homograph terms will be distinguished.

In the Set Theory, an equivalence class is defined on a set and through the definition of an equivalence relation. In our context, we consider the labeled association rules set E_{RAL} , on which we define two binary relations.

Definition 8. Let $\mathcal{R}$ (resp. $\mathcal{S}$) be a binary relation on the labeled association rules set $\mathcal{E}$ such as:

$$\mathcal{R} = \{(a, b) \mid a, b \in \mathcal{E} \wedge a.rec = b.rec \wedge a.label = b.label\}$$

$$\mathcal{S} = \{(a, b) \mid a, b \in \mathcal{E} \wedge a.act = b.act \wedge a.label = b.label\}$$

where $RAL.act$ is the acting concept of the rule RAL , $RAL.label$ is the rule label and $RAL.rec$ is the rule recipient.

The relation $\mathcal{R}$ (resp. $\mathcal{S}$) allows to select rules having the same label and the same recipient concept (resp. acting concept).

Proposition 1. Any relation $\mathcal{R}$ (resp. $\mathcal{S}$) as defined in definition 8 is an equivalence relation, as $\mathcal{R}$ (resp. $\mathcal{S}$) is reflexive, symmetric and transitive.

Proof. The proof is omitted as it is easy to see that all properties hold.

These equivalence relations allow building equivalence classes from labeled association rules. These classes are then added as a concept into the existing ontology.

Example 7. Given an extracted LAR set $\mathcal{RL} = \{Rain \xrightarrow{cause} Inundation, Rain \xrightarrow{cause} Landslide, Rain \xrightarrow{cause} Submersion\}$, then we can build an equivalence class based on the label "cause" and on the concept which preferred term is "Rain":

$$[\xrightarrow{cause}]_{RAIN} = \mathcal{RL}$$

This allows us to group terms *Inundation*, *Landslide*, and *Submersion* under the same concept.

3.7 Expanding the Ontology

Adding new elements is the final step of our method, before validation by a human expert. We create new concepts around which are associated equivalent terms. We add then our new concepts linking them to existing ones by the mean of labeled relations.

Some classes give a different role to a same item, either as a term concept, or as a relation label. As in works using syntactic based methods, we consider that an item can only have one role into the same ontology. So, before adding a new equivalence class **ce**, the algorithm checks if the label relation linking **ce** to the ontology is not already considered as a concept term, and rejects it otherwise.

Example 8. Figure 3 shows the evolution of the ontology presented in figure 2. We notice that a new concept which preferred term is "Inundation" have been added, linked to the existing concept "Rain" using the relation label "Cause".

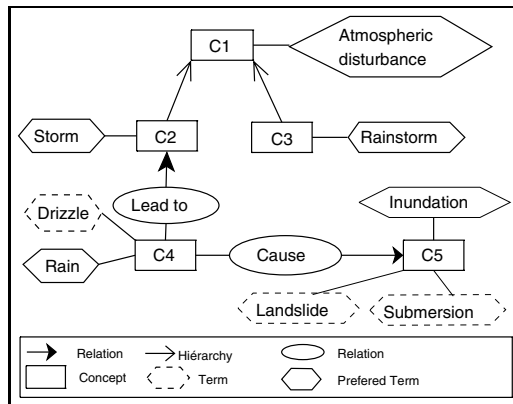


Fig. 3. Expanded Ontology

4 Experiments

4.1 Data

Our method has been tested on the EMWIS ontology¹, Euro-Mediterranean System of Water Domain Information. EMWIS is an European project concentrating his efforts on developing an ontology about water domain. The major goal is to improve communication between all the water area protagonists. This ontology is formed by 1006 concepts organized around three hierarchy levels and 29 non-labeled relations.

Concepts have been grouped around 25 themes in order to ease navigation. We decided to use them during our experiments. We built for each concept term

¹ http://www.semide.net/portal_thesaurus

a query which have been ran on a search engine. We downloaded the first 20 retrieved documents, obtaining thus a thematic corpus.

Experiments presented here have been realized on the "Water Needs" thematic, composed by 136 concepts. This theme was chosen for its ontology representativity: it was the one grouping the greatest number of concepts. Then, the textual corpus built from the Web have 2,720 documents.

4.2 Preprocess and Sequential Patterns Extraction

The relevance of the extracted patterns depends on document preprocessing, which is done according to three steps: (1) content extraction, (2) lemmatization, and (3) item selection.

Content extraction erases noise in documents (advertising, images, hypertext links...) and only keeps the main text of the page. In our experiments, lemmatization have been realized with TreeTagger [23], which is able to process french and english texts. After this step, all words are in their generic form: they are now considered as *terms*.

In order to keep only relevant words, we used tf.idf measure [24], allowing us to evaluate how important a word is to a document in a collection or corpus. At the end of this process, we obtain the most important terms according to tf.idf, and we keep terms already present into the ontology.

Sequential patterns have been extracted using a JAVA implementation of the algorithm VPSP [21], which combines prefixed tree projection of PSP [19] and memory projection of SPADE [20]. VPSP is a generate-prune algorithm which uses $(k-1)$ -sequences to generate k -sequences, and after a frequency computation prunes sequences under minimal support fixed by the user. We adapted VPSP to keep in memory sequences of length 1, 2 and 3, necessary for the RL measure computation. We optimized this step in order to minimize required memory space, gaining the time which would have been used to compute again these information. For our dataset about *Water Needs*, we efficiently generated and stored about 14,000,000 of 3-sequences.

4.3 Results

Results obtained with various closeness clue highlight our proposal relevance. Indeed, we noticed that our method allows detection of a high number of new concepts, and proposes a correct placement of them into the existing ontology.

We studied closeness clue value impact on the enrichment process, as it is a determining factor concerning the item selection. We noticed that the lower the closeness clue value, the larger the neighborhood set cardinality. Consequently the greater neighbors we obtain, the greater is the combination possibilities to generate labeled association rules. This is why we decided to use the relation level to prune rules, whereas closeness clue is very low.

We noticed that closeness clue has a strong influence on the number of LAR generated: when it is low we obtain more rules with a 100% confidence. Actually,

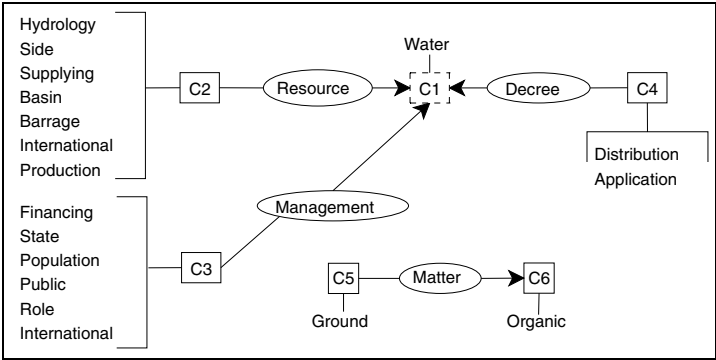


Fig. 4. Expanding Ontology Results

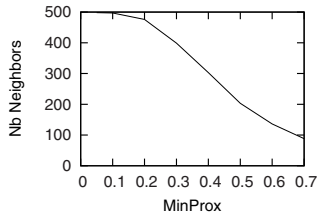


Fig. 5. Neighbor/minP

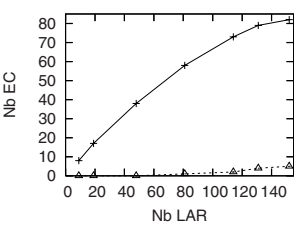


Fig. 6. Eq Class/LAR

if occurrences of the triplet (concept relation item) are lower, then these element are more often found together, which explains these observations.

We tested results obtained without using equivalence classes: this led to a new concept creation for each added terms, and mostly using a common relation label. Equivalence classes empirically prove to significantly improve results. First, we noticed that there is no loss of information: each selected term is placed into the existing ontology. Equivalence classes group terms sharing a common lexical field. As an example, we can see on the figure that terms "Basin" and "Hydrology" are grouped under the same concept and linked as "Resource" to the "Water" concept. Secondly, adding elements driven by equivalence classes eases the navigation of the expanded ontology.

Furthermore, this method detects homographic terms. Homographics are words sharing the same spelling, but having different meanings. For example "shift" can be used as "a change" or as "a period at work". In our experiments, we found the term "Source" was associated to the financial area, whereas it was already associated into the ontology to the "Water" concept, meaning "a river origin". In such a case, equivalence classes lead to the placement of two distincts terms "Source", each one associated to a different concept.

Figure 6 shows the number of equivalence classes found according to the number of LAR. Curve with cross points represents classes based on receiving concept, whereas curve with triangular points represents acting based ones. No-

tice that our real dataset sample led to a high number of received based classes. The number of equivalence classes grows as the number of LAR grows, but the number of new added concepts is significantly reduced: as an example, we can see that 140 LAR led to the creation of half less new concepts (i.e. 80 concepts).

These results, i.e. high number of labeled association, let us choose a low and therefore less selective minimal closeness clue. To obtain great quality results, we preferred to put a restriction on the implication level of our labeled association rules, keeping the more interesting ones.

Figure 4 presents some obtained results, realized with a minimal closeness clue fixed to 40%, and keeping labeled association rules having an implication level of at least 50%. Indeed, the chosen thematic - Water Needs - is large and then includes distant concepts. We fixed a large closeness value in order to catch less evident relations. Once labeled association rules have been generated, we noticed the high number of rules having an implication level superior to 80%, confirming that a threshold of 50% was sufficient for the extraction of most of the necessary rules to obtain relevant process result.

Figure 4 displays existing concepts, presented with dashed lines, and added element (concepts, relations and labels) by plain lines. A 40% minimal closeness value allowed to retrieve 303 of concepts/items pairs (or neighbor couples) potentially candidates.

From these pairs, 498 labeled association rules have been generated. We retained those having an implication level of at least 50% to build equivalence classes. The obtained results are coherent, as most of them have been correctly placed into the ontology. Elements added are rather general terms, which is a normal phenomenon considering that the built corpus cover a large part of the ontology. We can observe the high number of concepts added around the water concept, which seems normal as we have an ontology constructed for the water domain.

Finally, EMWIS ontology has 29 non-labeled relations. We observe that our method allowed to name one of them: the one linking "*Hydrological basin*" and "*Watercourse*".

Even if it seems to be limited, this result goes ahead compared to existing approaches. Moreover, this weak number of existing labelization can be explained by two reasons: firstly, the specific character of these relations, as they indeed concern only 0.02% of the total ontology and secondly, these relations link sub-concepts at a very specific level of the hierarchy. Last, we noticed that these relations mainly concern other themes, such as politics or agriculture.

5 Conclusion

Ontologies are powerful structures, allowing knowledge representation and sharing. However, their relevance directly depends on their maintenance. Ontology updating mainly consists in adding concepts and/or relations and is mostly manually done.

Some works propose tools for automatically updating ontologies. However, none of them has obtained complete results. This mainly comes from their lack of automatism, i.e. the user is involved in the enrichment loop, or only one type of knowledge (mostly concepts) is considered; none allows automatic relation label detection.

In this paper, we propose the use of sequential patterns, allowing term research and element extraction (concept or relation). Our proposal is fully automatic, as it places the user at the end of the process in order to validate obtained results. Our main contribution is the automatic discovery of relation label and new concepts, and a finest terms analysis by the mean of equivalence classes. We have described in details all the process steps. From sequential patterns, we determined items which can be used to expand the ontology through the building of neighbor sets. From these sets we construct a new kind of knowledge named Labeled Association Rules, allowing by their implication level to distinguish between concept and label. Finally, an approach based on equivalent classes is used to fine the results and to drive to a coherent and readable adding to the ontology.

We noticed during our experiments the real quality of added elements: not only chosen elements are relevant but element grouping and placement is totally coherent too. This proves that sequential patterns allow to highlight semantically correlated terms, and that an equivalence based method correctly groups elements sharing a common lexical field. This results in the detection of homograph terms, and improves the ontology readability. Moreover, this opens some interesting perspectives. First of all, refined method for sequential patterns extraction have been proposed [13,15], such as the use of a minimal or maximum window size in order to restrict itemset selection to close or distant one. This work can be adapted in our context in order to select more close patterns. Besides, we could use an ontology driven extraction method to integrate the concept hierarchy during the mining process.

References

1. Stumme, G., Hotho, A., Berendt, B.: Semantic web mining: State of the art and future directions. *Web Semantics: Science, Services and Agents on the World Wide Web* 4(2), 124–143 (2006)
2. Gruber, T.R.: A translation approach to portable ontology specifications. *Knowledge Acquisition* 5(2), 199–220 (1993)
3. Agirre, E., Ansa, O., Hovy, E., Martinez, D.: Enriching very large ontologies using the WWW. In: *ECAI 2000 workshop on Ontology Learning* (2000)
4. Faatz, A., Steinmetz, R.: Ontology enrichment with texts from the WWW. In: *The Semantic Web Mining Conference (WS 2002)* (2002)
5. Parekh, V., Gwo, J.P., Finin, T.: Mining Domain Specific Texts and Glossaries to Evaluate and Enrich Domain Ontologies. In: *International Conference of Information and Knowledge Engineering* (2004)
6. Xu, F., Kurz, D., Piskorski, J., Schmeier, S.: A domain adaptive approach to automatic acquisition of domain relevant terms and their relations with bootstrapping. In: *The 3rd international conference on language resources and evaluation* (2002)

7. Neshatian, K., Hejazi, M.R.: Text categorization and classification in terms of multi-attribute concepts for enriching existing ontologies. In: 2nd Workshop on Information Technology and its Disciplines, pp. 43–48 (2004)
8. Velardi, P., Missikoff, M., Fabriani, P.: Using text processing techniques to automatically enrich a domain ontology. In: Proceedings of ACM- FOIS (2001)
9. Han, E.H., Karypis, G.: Centroid-based document classification: Analysis and experimental results. In: The 4th European Conference of Principles of Data Mining and Knowledge Discovery, pp. 424–431 (2000)
10. Bendaoud, R.: Construction et enrichissement d'une ontologie à partir d'un corpus de textes. In: RJCRI 2006, pp. 353–358 (March 2006)
11. Roux, C., Proux, D., Rechermann, F., Julliard, L.: An ontology enrichment method for a pragmatic information extraction system gathering data on genetic interactions (2000)
12. Maedche, A., Staab, S.: Mining ontologies from text. In: Dieng, R., Corby, O. (eds.) EKAW 2000. LNCS (LNAI), vol. 1937, pp. 189–202. Springer, Heidelberg (2000)
13. Srikant, R., Agrawal, R.: Mining generalized association rules. *Future Generation Computer Systems* 13(2–3), 161–180 (1997)
14. Hearst, M.A.: Automatic acquisition of hyponyms from large text corpora. Technical Report S2K-92-09 (1992)
15. Fiot, C., Laurent, A., Teisseire, M.: Extended time constraints for sequence mining. In: 14th International Symposium on Temporal Representation and Reasoning (2007)
16. Masseglia, F., Poncelet, P., Teisseire, M.: Pre-processing time constraints for efficiently mining generalized sequential patterns. In: 11th International Symposium on Temporal Representation and Reasoning, pp. 87–95 (2004)
17. Jaillet, S., Laurent, A., Teisseire, M.: Sequential patterns for text categorization. *Intelligent Data Analysis* 10(3), 199–214 (2006)
18. Agrawal, R., Srikant, R.: Mining Sequential Patterns. In: The 11th IEEE International Conference on Data Engineering, pp. 3–14 (1995)
19. Masseglia, F., Cathala, F., Poncelet, P.: The PSP approach for mining sequential patterns. In: The Second European Conference on Principles of Data Mining and Knowledge Discovery, pp. 176–184 (1998)
20. Zaki, M.J.: SPADE: An efficient algorithm for mining frequent sequences. *Machine Learning* 42(1/2), 31–60 (2001)
21. Di-Jorio, L., Jouve, D., Kraemer, D., Serra, A., Raissi, C., Laurent, A., Teisseire, M., Poncelet, P.: VPSP: extraction de motifs séquentiels dans weka. In: Démonstrations dans les 22èmes journées Bases de Données Avancées (BDA 2006) (2006)
22. Di-Jorio, L., Fiot, C., Abrouk, L., Hérin, D., Teisseire, M.: Enrichissement d'ontologie: Quand les motifs séquentiels labellisent des relations. In: 23 ème journées Bases de Données Avancées (2007)
23. Schmid, H.: Probabilistic part-of-speech tagging using decision trees. In: International Conference on New Methods in Language Processing, Manchester, UK, unknown (1994)
24. Robertson, S.E., Jones, K.S.: Relevance weighting of search terms, pp. 143–160 (1988)

Evaluating Automatically a Text Miner for Ontologies: A Catch-22 Situation?

Peter Spyns

Vrije Universiteit Brussel - STAR Lab, Pleinlaan 2 Gebouw G-10,
B-1050 Brussel, Belgium
Tel.: +32-2-629.1237; Fax: +32-2-629.3819
`Peter.Spyns@vub.ac.be`

Abstract. Evaluation of ontologies is increasingly becoming important as the number of available ontologies is steadily growing. Ontology evaluation is a labour intensive and laborious job. Hence, the importance to come up with automated methods. Before automated methods achieve reliability and widespread adoption, these methods themselves have to be assessed first by human experts. We summarise experiences acquired when trying to assess an automated ontology evaluation method. Previously we have implemented and evaluated a light-weight automatic ontology evaluation method that can be easily applied by knowledge engineers to rapidly determine whether or not the most important notions and relationships are represented in a set of ontology triplets. Domain experts have contributed to the assessment effort. Various assessment experiments have been carried out. In this paper, we focus particularly on the practical lessons learnt, in particular the limitations that result from real life constraints, rather than on the precise method to automatically evaluate results of an ontology miner. A typology of potential evaluation biases is applied to demonstrate the substantial impact conditions in which an evaluation happens can have on the reliability of the outcomes of an evaluation exercise. As a result, the notion of “meta-evaluation of ontologies” is introduced and its importance illustrated. The main conclusion is that still more domain experts have to be involved, which is exactly what we try to avoid by applying an automated evaluation procedure. A catch-22 situation?

1 Introduction and Background

The development of the Semantic Web (of which ontologies constitute a basic building block) has become a very important research topic for the information based society. However, the process of conceptualising an application domain and its formalisation require substantial human resources and efforts. Therefore, techniques applied in human language technology (HLT) and information extraction (IE) are used to create or grow ontologies with a quality as high as possible in a period of time as limited as possible. Work is still in progress - recent

overviews of the state of the art (in particular for machine learning techniques) can be found in [5,6,26].

Even in the ideal case that (semi-) automated ontology learning methods have become mature, there still remains the problem of assessing and evaluating the results. Various proposals for evaluation methods¹ have recently been put forward [2,6,14,34]. All these approaches basically share the same problem, i.e. how to evaluate the outcomes of automated ontology learning methods in a way that goes beyond the context of a specific evaluation setting (task, domain, ...). Rare are the experts willing to devote their precious time to validate output generated by a machine or establish in agreement with colleague stakeholders and experts a gold standard. In addition, current evaluation methods require specialised skills and infrastructure almost solely available in an academic environment.

In an answer to these issues, we have tried to define a light-weight assessment procedure that is easy to understand and apply by "standard knowledge workers" (basically a domain expert, a computer scientist, an engineer, ...) outside academia [28]. The evaluation method should be generally applicable (any kind of text miner, any kind of text collection) and able to provide a rough but good enough and reliable indication whether or not results of a text miner on a particular corpus are worthwhile. Ontologies can be evaluated from many angles [7,12]. Our method wants to measure to which extent an ontology includes the important domain notions. Hence, in this paper quality of an ontology refers to the degree with which the lexical material delivered by the ontology miner covers the important notions conveyed in a text corpus. Furthermore, this is only one dimension of judging the quality of an ontology. Other dimensions are equally important and should also be taken into account - see the related work in section 6. Typical of our approach will be that only the "raw" corpus (lemmatised² but otherwise unmodified) constitutes the reference point, and not an annotated corpus or another reference ontology. However, the automatic evaluation procedure itself still needs validation, and therefore we do need human experts and/or a gold standard built by human experts.

As this is an ambitious endeavour, we have to realise it in several stages. The first step has been to define and try out some lexicometric scores for triplets generated automatically by a text miner [28,29,33]. A next step is to validate the evaluation procedure using these scores [31,32]. Trying out the method in various situations and synthesising the outcomes is a subsequent logical step. Finally, the experiences from the validation experiments have to be summarised as provide valuable insights to determine the set-up of new experiments.

The remainder of this paper is organised as follows. The next two sections present the material (section 2) and methods (section 3). An overview of the various experiments and their setting is presented in section 3.1. Subsequently, we

¹ The EON2006 workshop has been devoted to ontology evaluation - see <http://km.aifb.uni-karlsruhe.de/ws/eon2006>

² Lemmatise means to reduce words to their base form. E.g., working, works, worked → work. Incidentally note that in this paper, the terms 'word', 'term', and 'lemma' are used interchangeably.

explain how the machine gold standard is established (section 3.2) and validated (section 3.3). In section 4 (Results), we discuss how the automated evaluation procedure rates the results of an ontology miner on the one hand (section 4.1) as well as how the domain experts rate the automated procedure (section 4.2) on the other. In addition, not only the results of the ontology miner (section 5.1) and the evaluation experiments (section 5.2) are discussed but also their organisation (section 5.3). Related work is outlined in section 6. Indications for future research are given in section 7, and some final remarks (section 8) conclude this paper.

2 Material

The *memory-based shallow parser for English*, being developed at CNTS Antwerp and ILK Tilburg [4]³, has been used. It is an unsupervised parser that has been trained on a large general purpose language model. No additional training sessions (= supervised) on specific corpora are needed. Hence, the distinction between learning and test corpus has become irrelevant for our purposes. Semantic relations that match predefined syntactic patterns have been extracted from the shallow parser output. Additional statistics and clustering techniques using normalised frequencies and probabilities of occurrence are calculated to separate noise (i.e. false combinations generated) from genuine results. The unsupervised memory-based shallow parser with the additional statistical modules constitute the ontology miner. More details can be found in [22,23].

The privacy and VAT corpora (two separate documents) consist of 72,1K resp. 49,5K words. They constitute two *directives* (English version), namely the 95/46/EC of 18/12/2000 (privacy) and the 77/388/EC of 27/01/2001 (VAT), which EU member states have to adopt and transform into local legislation. The VAT directive has served as input for the ontology modelling and terminology construction activities in the EU FP5 IST FF Poirot project⁴ (IST-2001-38248). These two documents are the sole official legal reference texts for the domain. The texts have been lemmatised. The size of both texts however is rather small, when compared to other machine learning experiments. As a consequence, the quality of the ontology miner might be compromised. A possible workaround is to include unofficial documents that provide comments or points of view on the official directives. However, this might distort the outcomes as well as these don't represent an official EU position.

We were lucky to be able to use a list of *900 VAT terms selected manually* by domain experts on basis of the EU VAT Directive. According to the VAT experts the notions represented by these terms should be included in a VAT ontology.

The *Wall Street Journal (WSJ) corpus* (a collection - 1290K words - of English newspaper articles) serves as a corpus representing the general language that is to be contrasted with the specific technical vocabulary of the two Directives. The WSJ is not really a neutral corpus (the articles are about economic topics).

³ See <http://ilk.kub.nl> for a demo version.

⁴ <http://www.ffpoirot.org/>

It is easily available, also included in the WordSmith tool - see below - and a standard in corpus linguistics.

An off-the-shelf available *lexicographic program* (Oxford WordSmith Tools v4⁵) has been used to create the frequency lists and to easily filter out non words⁶. Further manipulation of exported WordSmith files and the calculation of the statistics are done by means of small scripts implemented in *Tawk* v.5 [35], a commercial version of (G)awk, in combination with some manipulations of the data in *MS Excel*.

3 Methods

3.1 Overview

As an ontology is supposed to represent the most relevant concepts and relationships of a domain of discourse or application domain, all the terms lexicalising these concepts and relationships should be retrieved from a corpus of texts about the application domain concerned when building an ontology for the domain. The key question is thus how to determine in an automated way to which extent the important terms of a text corpus have been retrieved. In addition, an algorithm is needed to distinguish relevant combinations (i.e. two concepts in a valid relationship - a triplet) from irrelevant ones. These key issues evidently hold for any ontology miner as well as for any method producing a machine gold standard⁸(section 3.2).

The machine gold standard has to be validated by humans (section 3.3) in order to assign some authority to the automated method. This step is in essence similar to a manual evaluation of an individual ontology, but the purpose is different. In addition, we are well aware that current term extractors are more sophisticated than the methods we use. For our experiments, we have intentionally sacrificed scientific state of the art for the simplicity of an off the shelf product.

We also discuss in detail how the validation by human experts has been organised. The work of Friedman and Hripcsak [8] has been our main source of inspiration for a meta-evaluation.

Note that the human-based evaluation step only serves to validate the automated procedure. Figure 1 displays the process flow of the evaluation experiments. Part A of the figure shows the CNTS ontology miner that produces lexical triplets that are validated by human experts (part C), as happens usually. Parts B and D represent "the automated expert" (a term extractor combined with

⁵ URL:<http://www.lexically.net/wordsmith/>

⁶ Another interesting tool is the on-line available term identification tool described in [19]. Other heuristics, next to the classical term frequency and document frequency statistics, are taken into account, such as domain relevance, domain consensus, lexical cohesion, and stylistic relevance⁷.

⁸ In the ideal case an ontology miner's result completely coincides with the machine gold standard.

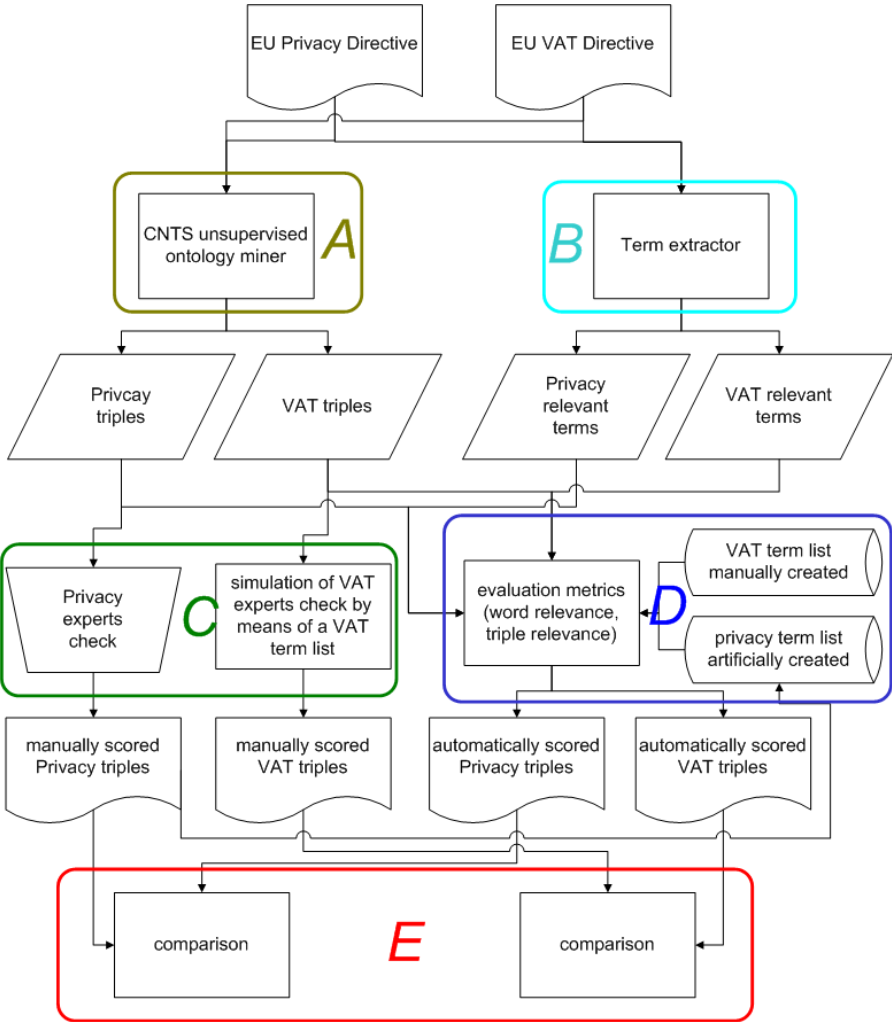


Fig. 1. Overall process flow of the experiments

term and triplet scoring heuristics) as a cheap and fast alternative for the human experts. Part E of the figure stands for the comparison of triplets validated by the human experts (human gold standard) and the ones automatically validated (machine gold standard). The greater the overlap between these two sets, the more closely the automated expert resembles the human experts. In the ideal case, the automated expert can consistently (no inter and intra rater differences) create a gold standard for any text and give a rough but fast impression of the quality of the material mined.

3.2 Establishing the Machine Golden Standard

Finding the relevant words. Basically, we try to answer the following fundamental questions by calculating an associated lexicometric score. Guarino [11] has proposed similar metrics but without a concrete implementation.

- is the vocabulary of the triplets retrieved representing the domain ? *coverage*
- is the vocabulary of the triplets retrieved not too general but reflecting the specialised terms of the domain ? *accuracy*
- has all the relevant domain vocabulary been captured by the triplets retrieved ? *recall*
- is the vocabulary of the triplets retrieved relevant for the domain ? *precision*

We have combined various insights from quantitative linguistics, in particular foundational insights by Zipf and Luhn, a statistical formula to compare two proportions, with the traditional IE evaluation metrics (recall and precision). The central notion linking everything together is "frequency class" (FC), i.e. the set of (different) lemmatised words that appear n times in a document d . E.g., for the Privacy Directive, there are 416 words that appear only once (hence FC 1 contains 416 elements), and there is one word that appears 1163 times (FC 1163 is a singleton). According to Zipf's law [39], the latter one ('the') is void of meaning, while the former ones (e.g., 'assurance') are very meaningful, but may be of only marginal interest to the domain. Subsequently Luhn [16] introduced the notion of "resolving power of significant words" by defining intuitively a frequency class upper and lower bound. In his view, the most significant words are found in the middle of the area of the frequency classes between these boundaries.

We propose to approximate the resolving power of significant words by simply calculating whether a FC is relevant or not. Only if a FC is composed by 60% or more of relevant words, the FC is considered to be relevant. A word is said to be relevant or not based on the outcome of a statistical formula that compares two relative proportions. Technically speaking, we compute the z-values of the relative difference between the frequency of a word in a technical text (the Privacy resp. VAT Directives) vs. a more general text (WSJ), which enables us to determine the words that are statistically typical of the technical text. These are the relevant words. Calculations have been done with a 99% confidence level. The gold standard for words is now defined in a very easy, fast and cheap way.

The assumption is that the ontology miner should be able to retain the words that belong to the relevant frequency classes, and hence simulate "the resolving power of words". The notion of relevant words is distributed over all members of a FC if 60% and more of its population is statistically relevant (see above). Subsequently, we have defined the following lexicometrics:

- The *coverage* of a text by the vocabulary of triplets automatically mined is measured by counting for each frequency class (FC) the number of words, constituting the triplets, that are identical with words from the text for that FC. This number is compared to the overall word count for the same FC. The mean value of these proportions constitutes the overall coverage percentage.

- The *accuracy* of triplets automatically mined to lexically represent the important notions of a text is measured by averaging the coverage percentage for the relevant frequency classes. An FC is considered to be relevant if it contains more than 60% of typical vocabulary, i.e. words considered as characteristic of a text on basis of statistical calculations (= machine gold standard). Characteristic words of a domain specific corpus are determined by comparison with a general language corpus (by calculating the relative difference of relative frequencies).
- The *recall* is defined as the vocabulary common to the triplets mined and the machine gold standard compared to the machine gold standard.
- The *precision* is defined as the vocabulary common to the triplets mined and the machine gold standard compared to the vocabulary of the triplets mined.

Finding the relevant triplets. After having determined how well (or bad) the overall triplet vocabulary (= all the words making up the triplets generated by the ontology miner) covers the terms representing important notions of the domain (as established by the machine gold standard for words), entire triplets are examined.

Again, the machine gold standard is used as reference. A triplet is considered relevant if it is composed by at least two terms statistically relevant (i.e. belong to the machine gold standard). We did not use a stopword list, as this list might change with the nature of the corpus, and as a preposition can be potentially relevant since they are included in the triplets automatically generated. The lexicometrics should cope with these issues.

A triplet score indicates how many characters of the three triplet parts (expressed as an averaged percentage) are matched by words of the machine gold standard. E.g., the triplet *< rule, establish, by_national_competent_body >* receives a score of 89 as only 'competent' is not included in the machine gold standard with a 95% confidence level ($89 = ((4/4)*100 + (11/11)*100 + (17/25)*100)/3$)⁹.

3.3 Validating the Machine Golden Standard

The ontology miner itself has not been modified during the experiments. In addition, the developers of the memory-based shallow parsers have not been involved in the experiments, and the developer (computational linguist) of the additional statistical measures only became knowledgeable of the test corpora and results when performing the batch runs of the ontology miner. She has not been involved in the evaluations. Nor had the experts performing the evaluation experiment anything to do with the ontology miner. The computer scientist responsible for the automated evaluation procedure had no knowledge of the internals of the ontology miner and was not involved in the actual assessment by the domain experts. He

⁹ A slight imprecision occurs due to the underscores that are not always accounted for.

merely distributed and collected the files (input to computational linguist, mining results to domain experts, assessments from experts) and implemented and ran the automated evaluation procedure. This strict separation of roles guarantees that the various persons involved do not influence each other. Also the set up of the experiments is not biased in one way or the other. Here we fully respect the criteria of Friedman and Hripcsak to minimise the bias [8, p.335].

Assessing the relevant terms. We have determined a baseline against which the results of our method can be compared. In earlier work, we showed that our method performs better than this random baseline (see [29,32]).

Unfortunately, the VAT experts were not available to evaluate the VAT terms and triplets automatically generated. The vocabulary of the 900 VAT terms manually selected constitutes a substitute for humans directly assessing the triplet vocabulary automatically generated. It has not been communicated how the VAT experts have reached agreement on the terms, which constitutes a negative aspect [8, p.336].

Even if not ideal from a scientific point of view, this corresponds to real life situations where on the one hand lists of terms generally accepted by a community are put forward as standard reference, and on the other hand, experts check machine generated outcomes. The former situation can be problematic for consistency and completeness, while the latter corresponds to what is called "leading the witness"¹⁰ [8, p.336]. From a methodological point of view, one can argue that the list of terms collected by experts does not necessarily adequately reflect the important terms in the text(s) submitted to the ontology miner. On the other hand, in many cases such term lists are compiled by several representative experts on behalf of standardisation committees and are (publicly) available. Thus, even if not ideal, it is as close as one can get to some objective and qualitative reference if experts are otherwise not available.

Note that a similar reference term list for the privacy domain was not available. As the privacy experts, at the time of the experiments, still had to construct a term list, such a list has been constructed artificially for the sake of the experiments. The terms contained in the triplets produced by the ontology miner that have been positively assessed by the experts make up the privacy machine gold standard. The privacy experts did not assess the machine gold standard for the privacy directive. Instead, the human gold standard was constituted by the vocabulary of the privacy triplets judged relevant by the privacy experts. Inevitably, such an approach runs the danger of missing terms. Not only can the ontology miner fail to erroneously produce a triplet for a relevant term, also human experts can (falsely) reject or unfortunately miss to approve a triplet containing such a term.

Assessing the relevant triplets. The basic questions to assess the quality of the automatic triplet scoring procedure are:

¹⁰ Without a golden reference, evaluators show a tendency to agree with the system output - unless there is a glaring error.

- Have all the relevant triplets been positively scored ? *recall*, also called *sensitivity*
- Are the triplets positively scored indeed relevant for the domain ? *precision*
- Are the triplets that have been negatively scored not relevant for the domain ? *specificity*

These metrics using the machine gold standard are applied to estimate the precision score of the miner. Note that we do not determine whether the miner has retrieved all the relevant triplets (recall score for the miner) as there will be no gold standard available (this is the point of setting up an automatic evaluation procedure instead of having experts produce a reference). We use the lexicometric scores to indirectly answer this question.

Two experts in privacy protection matters have been asked to independently validate the list of privacy triplets as produced by the ontology miner. One has been a privacy data commissioner and still is a lawyer while the other is a knowledge engineer specialised in privacy and trust. Ontology engineering involves experts of various background and affiliations to come to a commonly agreed upon conceptualisation. Hence, we consider them as appropriate for the experiments. Unfortunately, we didn't receive any information on the VAT experts involved. Friedman qualifies this as a source of potential bias [8, p.336]. It would have been better if more than two human (privacy) experts would have been involved [8, p.335], but unfortunately many experts are quite reluctant to perform this kind of validation as it is quite tedious and boring. That is also why we have not been able to perform a similar human assessment on the VAT triplets. As an approximation, we have re-applied the automated triple scoring procedure with the VAT term list, instead of the machine gold standard, to the VAT triplets.

The experts only knew they had to assess a set of triplets and were unaware of its origin and related purposes. For them, the goal was to assist in the semi-automated construction of a privacy ontology. The experts have assessed all the privacy triplets output by the ontology miner. They were unaware of the scores of the automatic validation procedure as well as each other's scores (so there was no mutual influence). The experts have marked the list of triplets with '+' or '-' indicating whether or not the triplet is valid, i.e. useful in the context of the creation of a privacy ontology. Their assessments have been merged subsequently. Only those triplets positively scored by both experts have been retained as the human triplet reference. More or less one year after the first rounds of experiments, the privacy experts agreed to perform a second round of scoring - again to all the triplets generated by the ontology miner. This round of scoring was meant to calculate the intra rater agreement. The long interval between the experiments served to avoid a learning effect with the experts. Otherwise, they might remember their previous assessment (or desired outcome).

In addition, a suggestion put forward by the privacy experts has been tested. When discussing the results of the first round of experiments, they had suggested to cut away manually irrelevant parts of the Privacy Directive before inputting it to the ontology miner. Even though the amount of text to be processed decreases (which might compromise the statistical calculations), the hypothesis was that

important terms would be detected more easily. As - except for a rough manual cutting away of irrelevant deemed sections - nothing else in the set up of the experiment has been changed, a comparison with the results obtained earlier became possible (regression test)¹¹.

4 Results

The CNTS ontology miner has been applied to the VAT and privacy texts. After some format transformation, the miner outputs 315 "subject-verb-object" triplets, such as *<person, pay, tax>*, and 500 "noun phrase-preposition-noun phrase" triplets such as *<accordance, with, article>* resulting in a total of 815 VAT triplets. Concerning the privacy corpus, 1115 *privacy* triplets have been generated by the ontology miner: 276 "subject-verb-object" triplets *<person, have, right>*, 554 "noun phrase-preposition-noun phrase" triplets (*<protection, of, individual>*) and 285 "subject-verb-prepositional object" triplets *<situation, result from, finding>*.

4.1 The Ontology Miner Rated by the Automated Evaluation Procedure

For the VAT corpus, only a list of terms was available. For the Privacy corpus, the "human" gold standard is approximated (consisting of the vocabulary of the triplets positively assessed by the human experts).

Table 1. Lexicometric scores

metrics	VAT	Privacy
coverage	49,26%	79,88%
accuracy	55,97%	95,04%
recall	36,06%	89,91%
precision	55,44%	27,43%

Table 2. VAT machine gold standard vs. human gold standard: $\kappa = 0,3757$

Word reference	"Expert" +	"Expert" -	
Statistics +	299	153	452
Statistics -	379	1375	1754
	788	1528	2206

Word relevance. Table 1 shows the lexicometric scores. There is a clear difference between the two sets of scores (VAT vs. privacy). We explain them by the origin of the human gold standard. For the VAT test, a list of expressions, not necessarily including the same words as used in the VAT directive, has been used. Hence, it is not a surprise that less terms match. For the privacy test, it basically concerns the same words, which might account for the good recall score. Precision is quite low. Probably because only the vocabulary of the positively scored triplets is considered as reference, which might be a too drastic limitation (both the miner and the experts might discard or miss out valid words). Hence, we only calculated the agreement (expressed by the κ -value) between the machine and human gold standard - see Table 2. A rather modest agreement was found.

¹¹ Due to space restrictions, this aspect is not presented here.

Triplet relevance. In a previous experiment [29], we have investigated 22 different scenarios to distinguish relevant triplets from superfluous ones.

In the situation of ontology engineering we estimate that a high specificity is more interesting than a high sensitivity (less false positives at the detriment of less true positives): a relevant triplet might be missed in order to have less rubbish triplets. The rationale is that it is probably more efficient to reduce the extent of the material ontology engineers have to check and reject compared to their effort needed to detect missing material. Fully automated ontology learning is still not achievable to completely dismiss human experts, so important misses will most probably not remain unnoticed. In the experiments described in [31], we have kept the 95% word relevance confidence level and set the threshold for the triplet scores at 70% (V1 and P1), 70% (V2 and P2) and 90% (V3 and P3). 3091 VAT triplets (V) and 1116 privacy triplets (P) have been generated by the ontology miner. They have been automatically validated in the way described above. Figure 2 ¹² displays the outcomes.

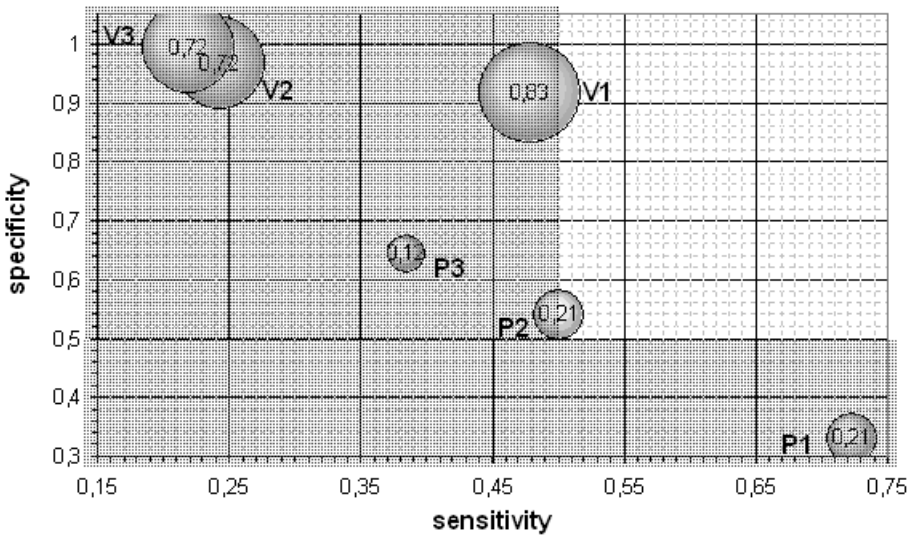


Fig. 2. Sensitivity, specificity and precision scores for the VAT and privacy corpora

As one can see on Figure 2, the 70% threshold produces the best results for the VAT corpus (V1) even if the sensitivity is slightly below 0,5 (shaded zone), and the worse for the privacy corpus (P1: low precision and specificity scores), while the 90% threshold results in almost moderate scores for the privacy corpus (P3) but in an unsatisfactory one in the VAT case (V3: low sensitivity). The 70% threshold gives the most acceptable results for the privacy corpus (P2) but a low sensitivity score in the VAT case (V2).

¹² The size of the bubble represents the precision score.

Table 3. Inter rater agreement (first round) on privacy triplets: $\kappa = -0,0733$

triplets mined	expert 1 -	expert 1 +	
expert 2 -	463	292	755
expert 2 +	248	112	360
	711	404	1115

Table 4. Inter rater agreement (second round) on privacy triplets: $\kappa = 0,1169$

triplets mined	expert 1 -	expert 1 +	
expert 2 -	793	117	910
expert 2 +	216	65	281
	1009	182	1191

Table 5. Intra rater expert 1 agreement on privacy triplets: $\kappa = 0,292$

triplets mined	round 1 -	round 1 +	
round 2 -	672	279	951
round 2 +	39	125	164
	711	404	1115

Table 6. Intra rater expert 2 agreement on privacy triplets: $\kappa = 0,4714$

triplets mined	round 1 -	round 1 +	
round 2 -	679	165	844
round 2 +	76	195	271
	755	360	1115

4.2 The Automated Evaluation Procedure Rated by Domain Experts

The 1116 privacy triplets have been rated by two human experts. The inter rater agreement expressed by the κ value is $-0,0733$ (first round) and $0,1169$ (second round). This almost equals contradiction - see Tables 3 and 4, which means that they agree in a way even less than expected by chance. One of the experts clearly behaved in a rather inconsistent way (intra rater agreement of $\kappa = 0,2936$ vs. $0,4714$) over the two test rounds - see Tables 5 and 6. The very low inter rater agreement becomes less surprising. These findings support the statement by Friedman and Hripcsak that two experts are not enough [8, p.335] to establish a gold standard. Only the privacy triplets commonly agreed upon by both experts (in a positive (112) and negative (463) sense) have been retained as the human triplet reference. For the VAT corpus, the triplet reference or gold standard has been constructed artificially (see section 3.3).

This probably explains why a modest agreement between the automated scoring procedure and the artificially simulated experts is found (κ value = $0,407$). Contrarily, the privacy experts (during the first round of experiments) apparently behaved almost completely in contradiction with the automated procedure.

Table 7. Automated scoring procedure vs. VAT simulated "experts" (threshold 70%) with $\kappa = 0,407$

triplets mined	"Expert" +	"Expert" -	
automaton +	684	136	2271
automaton -	748	1523	820
	1432	1659	3091

Table 8. Automated scoring procedure vs. Privacy experts (threshold 70%, round 1) with $\kappa = 0,026$

triplets mined	Expert +	Expert -	
automaton +	56	213	269
automaton -	56	250	306
	112	463	575

5 Discussion

The important point of applying these metrics, how imperfect they currently might be, is that the scores can be used to monitor changes (preferably improvements) in the behaviour of the text miner (regression tests). Currently this has not been explored yet, although the required data are available. As soon as the scores for a particular (and commonly agreed upon) textual source have been scientifically validated, the source and the scores together can be re-used as an evaluation standard in bench-marking tests involving other ontology miners, or even to some extent any RDF-based ontology producing tool. A logical next step would be that ontologies, automatically created by an ontology miner, are documented with performance scores on their textual source material as well as with scores for that particular miner on an evaluation reference (commonly agreed corpus and outcomes) - as e.g. customarily happens in the speech recognition industry.

5.1 Rating the Ontology Miner

Unfortunately, we cannot add a lot to our findings in the previous section for the VAT corpus as the VAT experts did not participate in assessing the triplets generated by the miner. The privacy experts did provide some comments. As a result, the following improvements could be implemented.

- The background ("neutral") corpus (here the WSJ) is key in establishing the machine gold standard. However, legal documents have many terms which are relevant to the legal domain in general, but not relevant to the particular legal domain under consideration. In future experiments using legal documents it is recommended to use a background corpus of terms taken from a set of European legal documents. For example, the term "Member State" is highly relevant to European Legislation in general, but has no specific relevance to the privacy domain. This is an example of a term that was judged highly relevant by the miner, but totally irrelevant by the experts.
- An issue not addressed is that of abstraction. Human experts extracting terms from a corpus are able to amalgamate synonyms and instances of higher level concepts where the use of lower level terms is of no use to the application domain. For instance, the privacy directive gives a list of data types which it is prohibited to collect without the data subject's consent. To a human expert, these classes of data are clearly what is known as "sensitive data". The inclusion of a synonym dictionary would go some way towards term abstraction although it can only take account of equivalence and not subclass relationships between terms. Currently, the tests and calculations depend too much on string matching.

The automated evaluation procedure assessed the ontology miner as rather "modestly" producing material reliably suitable for ontology engineering. Not only the miner misses more or less half of the interesting material but additionally

the quality of the material generated is not consistent (see Figure 2). It currently seems infeasible to define an appropriate threshold setting suited for both cases. Even if these results might be not so unexpected for an unsupervised miner (for supervised miners still perform better), ontology engineers in the field most probably are less impressed or helped by such material.

5.2 Rating the Automated Evaluation Procedure

The experiments do not allow to draw valid conclusions concerning the "goodness of fit" of the automated scoring procedures. The main reason is the insufficient (or even completely missing) availability of experts in general to establish a valid human gold standard. Although two privacy experts have participated, they did not rate in a consistent way casting doubt on the validity of the privacy human gold standard. A more detailed analysis should reveal whether or not this is due to one or the other expert. The mere fact that the machine gold standard does not represent state of the art techniques and methods is irrelevant in this matter.

5.3 "Rating" the Evaluation Set-Up

Following the criteria of Friedman and Hripcsak [8], we have clearly described the method applied to evaluate the results. Inter and intra rater agreement scores have been calculated, showing that one of experts did not score in a consistent way. Also the lexicometric and triplet overlap scores have been described in detail as they are used to establish a gold standard. This also allows to easily discover the limits of our experiments, which also complies with criteria set by Friedman and Hripcsak [8, pp.336-337]. In particular, the fact that the experiments are not completely symmetric. Also, it would have been interesting to have experts build an ontology completely by hand and use this as a human gold standard instead of validating machine generated output.

By involving two different domains in the evaluation experiment, we tested to which extent outcomes can be generalised over several application domains. Currently, due to the practical circumstances of the evaluation, one should not generalise the findings, either in a positive or negative way. Basically, conclusions can only be indecisive as for the VAT corpus, the expert involvement was to a large extent lacking, while for the Privacy corpus, not enough experts have been involved. One can wonder how many conclusions concerning evaluations of ontological material or ontologies reported in the literature will survive an analysis of the evaluation set-up as scrutinous as the presented here. E.g., one rarely finds inter and intra rater agreement numbers. Often developers of ontology learning applications also perform the evaluation - sometimes even as a sole evaluator.

6 Related Work

Previous reports on our work contain additional details on the unsupervised miner [22], its application to a bio-medical corpus and a qualitative evalua-

tion [23]. The method and previous quantitative experiments have been presented in [28,29,30,31,32]. Various researchers are working on different ways to evaluate an ontology from various perspectives. Good overviews of the recent state of the art that also contain a comparison of the characteristics of the various methods are [2,6,7,9,14,21].

A somewhat related topic is that of ontology selection and ranking: ontologies are evaluated as part of a selection process to choose the most appropriate ontology for a purpose or task (e.g. [1,24]). Some researchers have evaluated methods and metrics to select the most appropriate terms (e.g. [10,19]) from texts for building an ontology. However, these latter do not evaluate entire triplets. Others are active in ontology based information extraction (OBIE) and present metrics to evaluate OBIE performance - e.g., [18]. Next to that, one could consider additionally work that measures the similarity between two ontologies [17].

Only a few other approaches address the quantitative and automated evaluation of an ontology by referring to its source corpus. *Brewster* and colleagues have presented a probabilistic measure to evaluate the best fit between a corpus and a set of ontologies as a maximised conditional probability of finding the corpus given an ontology [3]. Unfortunately, till now no concrete results or test cases have been presented.

Velardi and colleagues have proposed to use the combination of "domain relevance" and "domain consensus" metrics to prune non domain terms from a set of candidate terms [36]. They use a set of texts typical of the domain next to other ones. *Domain relevance* is in fact the proportion of the relative frequency of a term in the domain text compared to the maximum relative frequency of that term over several non domain texts. *Domain consensus* is defined as the entropy of the distribution of a term in all the texts of the corpus. In our approach, we have computed the difference between two proportions, more specifically the z-values of the relative difference between the frequency of a word in a technical text vs. a general text, which enables us to filter out words that are only seemingly typical of the technical text. In [19], the authors also present a method to semantically interpret novel complex terms with the help of WordNet and to organise them in a hierarchy. An evaluation of these latter aspects is also provided. Remark that both of the proposed methods clearly (and correctly) differentiate a term or word from a concept.

Another statistical approach is elaborated by *Gillam and Tariq* [10] as part of a method to extract technical complex terms. They as well try to compare a specific text with a general text and characterise words by their weirdness (z-score for the ratio of the two relative frequencies of a word).

There is the - no longer continued - work of *Sabou* [25] who has examined how to learn ontologies for web services from their descriptions. Although the practical aspects of her work on the ontology learning aspects are quite tailored towards the application domain, the evaluation method resembles ours. She has "established a one-to-one correspondence between phrases in the corpus and derived concepts", so that our lexicometric scores are comparable to her ontology ratios. In more or less the same vein, *Gulla et colleagues* [13] use a

keyphrase extraction techniques to semi-automatically build an ontology. They involve domain experts to evaluate the ontology in a more or less task independent way. Queries are run against a separate manually built ontology and the semi-automatically constructed one.

Concerning the meta-evaluation of ontologies, *Gangemi* [9] provides in his impressive overview (and ontology) of ontology evaluation metrics and measures some elements and insights that come close to some of our findings. We have rather focused on a meta-analysis of how content material for an ontology can be assessed by means of a gold standard approach and which pitfalls are to be avoided to obtain methodologically sound outcomes.

7 Future Work

The same experiments can be repeated using another text miner - e.g., [15] - based on other algorithms or heuristics. Also other scoring measures (e.g. the weirdness measure [10]) to determine whether or not terms are relevant can be tried in the future. In the same line of thinking, we could look for other user friendly term extraction tools. We hope to test our method on other domains, pending the availability of sufficient appropriate domain experts. In addition, a regression test can be performed with the set up as described in this paper. All these experiments will also provide new insights to extend the framework for the meta-evaluation of ontologies.

8 Conclusion

The current experiments give an indecisive answer to the question whether the automatic evaluation procedure is up to providing a reliable indication on the quality of triplets produced by a ontology miner. The main reason is the imperfect manner in which the experiments had to be set up, constrained as they were by practical limitations in working conditions. The main lesson to be drawn is that ontology evaluation, especially when it concerns aspects that go beyond mere consistency checking, counting of ontology nodes or other "mechanical" or structural checks [38], is a fragile exercise. In order to produce scientifically valid results, an important number of conditions has to met with - as illustrate our experiences. Evaluating how ontology evaluation should happen is a rather novel research topic, which is not a surprise as the topic of ontology evaluation itself still offers many further avenues of research to be explored. The growing number of recent publications in this area illustrates that the topic is becoming a valuable research area. And the criterion whether or not an ontology adequately covers a domain cannot be addressed only in an impressionistic way by having people rate ontologies- cf. [20]. If well designed (e.g., [37]), computer assisted evaluation of ontologies is possible. And maybe introducing gaming aspects [27] could alleviate the psychological burden?

The lightweight automated evaluation procedure reported on in this paper aims at reducing the need to call upon experts, who are, in general, reluctant

to participate in evaluation procedures. However, the experiments and results show clearly that an active involvement of several appropriate experts of various backgrounds is still crucially needed at this stage. How to break this catch-22 (or deadlock) situation?

Acknowledgments

Parts of this research have been supported by the the OntoBasis project (GBOU 2001 #10069) of the IWT Vlaanderen (Institution for the Promotion of Innovation by Science and Technology in Flanders) and by the EU FP6 IP PRIME (IST 2002-507591) project. We are particularly indebted to dr. Marie-Laure Reinberger (at the time at the Universiteit Antwerpen - CNTS), who has produced the VAT and privacy triplets as well as a lemmatised version of the WSJ, to dr. Giles Hogben (at the time at the EU Joint Research Centre IPSC, Italy) and to drs. John Borking (Borking Consultancy, The Netherlands). Both acted as the privacy domain experts. In addition, we gratefully acknowledge Prof. dr. Patrick Wille (VAT@ NV, Belgium and partner of the EU FP5 IST FF Poiriot consortium) for putting at our disposal the hand-crafted list of 900 VAT terms.

References

1. Alani, H., Brewster, C., Shadbolt, N.: Ranking ontologies with aktiverank. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L.M. (eds.) ISWC 2006. LNCS, vol. 4273, pp. 1–15. Springer, Heidelberg (2006)
2. Brank, J., Grobelnik, M., Mladeníć, D.: Ontology evaluation. SEKT Deliverable #D1.6.1, Jozef Stefan Institute, Prague (2005)
3. Brewster, C., Alani, H., Dasmahapatra, S., Wilks, Y.: Data driven ontology evaluation. In: Shadbolt, N., O'Hara, K. (eds.) Advanced Knowledge Technologies: selected papers, pp. 164–168. AKT (2004)
4. Buchholz, S., Veenstra, J., Daelemans, W.: Cascaded grammatical relation assignment. In: Proceedings of EMNLP/VLC 1999, PrintPartners Ipskamp (1999)
5. Buitelaar, P., Cimiano, P., Loos, B. (eds.): Proceedings of the 2nd Workshop on Ontology Learning and Population: Bridging the Gap between Text and Knowledge. Association for Computational Linguistics (2006)
6. Buitelaar, P., Cimiano, P., Magnini, B. (eds.): Ontology Learning from Text: Methods, Applications and Evaluation. IOS Press, Amsterdam (2005)
7. Burton-Jones, A., Storey, V., Sugumaran, V.: A semiotic metrics suite for assessing the quality of ontologies. *Data and Knowledge Engineering* 55(1), 84–102 (2005)
8. Friedman, C., Hripcsak, G.: Evaluating natural language processors in the clinical domain. *Methods of Information in Medicine* 37(1-2), 334–344 (1998)
9. Gangemi, A., Catenacci, C., Ciaramita, M., Gil, R., Lehmann, J.: Ontology evaluation and validation: an integrated formal model for the quality diagnostic task. Technical report (2005), <http://www.loa-cnr.it/Publications.html>
10. Gillam, L., Tariq, M.: Ontology via terminology? In: Ibekwe-San Juan, F., Lainé Cruzel, S. (eds.) Proceedings of the Workshop on Terminology, Ontology and Knowledge Representation (2004), <http://www.univ-lyon3.fr/partagedessavoirs/termino2004/programb.htm>

11. Guarino, N., Persidis, A.: Evaluation framework for content standards. *OntoWeb Deliverable #D3.5*, Padova (2003)
12. Guarino, N., Welty, C.: Evaluating ontological decisions with OntoClean. *Communications of the ACM* 45(2), 61–65 (2002)
13. Gulla, J., Borch, H., Ingvaldsen, J.: Ontology learning for search applications. In: Meersman, R., Tari, Z., et al. (eds.) *OTM 2007, Part I. LNCS*, vol. 4803, pp. 1050–1062. Springer, Heidelberg (2007)
14. Hartmann, J., Spyns, P., Maynard, D., Cuel, R., de Figueroa, S., Sure, Y.: Methods for ontology evaluation. *KnowledgeWeb Deliverable #D1.2.3*, 3 (2005)
15. Judge, J., Sogrin, M., Troussov, A.: Galaxy: IBM ontological network miner. In: *CSSW. LNI*, vol. 113, pp. 157–160 (2007)
16. Luhn, H.P.: The automatic creation of literature abstracts. *IBM Journal of Research and Development* 2(2), 159–195 (1958)
17. Maedche, A., Staab, S.: Measuring similarity between ontologies. In: Gómez-Pérez, A., Benjamins, V.R. (eds.) *EKAU 2002. LNCS (LNAI)*, vol. 2473, pp. 251–263. Springer, Heidelberg (2002)
18. Maynard, D., Peters, W., Li, Y.: Evaluating evaluation metrics for ontology-based applications: Infinite reflection. In: Calzolari, N., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., Tapias, D. (eds.) *Proceedings of the Sixth International Language Resources and Evaluation (LREC 2008)*, Paris, European Language Resources Association (2008)
19. Navigli, R., Velardi, P.: Learning domain ontologies from document warehouses and dedicated web sites. *Computational Linguistics* 30(2), 151–179 (2004)
20. Noy, N., Guha, R., Musen, M.: User ratings of ontologies: who will rate the raters? In: *AAAI 2005 Spring Symposium on Knowledge Collection from Volunteer Contributors* (2005)
21. Obrst, L., Ashpole, B., Ceusters, W., Mani, I., Ray, S., Smith, B.: Semantic Web: Revolutionizing Knowledge Discovery in the Life Sciences. In: *The Evaluation of Ontologies: toward Improved Semantic Interoperability*, pp. 139–158. Springer, Heidelberg (2007)
22. Reinberger, M.-L., Spyns, P.: Unsupervised text mining for the learning of DOGMA-inspired ontologies. In: Buitelaar, Ph., Cimiano, P., Magnini, B. (eds.) *Ontology Learning from Text: Methods, Applications and Evaluation*, pp. 29–43. IOS Press, Amsterdam (2005)
23. Reinberger, M.-L., Spyns, P., Pretorius, A.J., Daelemans, W.: Automatic initiation of an ontology. In: Meersman, R., Tari, Z. (eds.) *OTM 2004. LNCS*, vol. 3290, pp. 600–617. Springer, Heidelberg (2004)
24. Sabou, M., Lopez, V., Motta, E., Uren, V.: Ontology selection: Ontology evaluation on the real semantic web. In: *Proceedings of the 4th International EON Workshop, Evaluation of Ontologies for the Web (2006)*, <http://eprints.aktors.org/487/>
25. Sabou, M., Wroe, C., Goble, C., Mishne, G.: Learning domain ontologies for web service descriptions: an experiment in bioinformatics. In: *Proceedings of the 14th International World Wide Web Conference* (2005)
26. Shamsfard, M., Barforoush, A.: The state of the art in ontology learning: a framework for comparison. *Knowledge Engineering Review* 18(4), 293–316 (2003)
27. Siorpaes, K., Hepp, M.: Games with a purpose for the semantic web. *IEEE Intelligent Systems* 23(3), 50–60 (2008)
28. Spyns, P., Reinberger, M.-L.: Lexically evaluating ontology triples automatically generated from text. In: Gómez-Pérez, A., Euzenat, J. (eds.) *ESWC 2005. LNCS*, vol. 3532, pp. 563–577. Springer, Heidelberg (2005)

29. Spyns, P.: Evalexon: assessing triples mined from texts. Technical Report 09, STAR Lab, Brussel (2005)
30. Spyns, P.: Object role modelling for ontology engineering in the DOGMA framework. In: Meersman, R., Tari, Z., Herrero, P., et al. (eds.) OTM-WS 2005. LNCS, vol. 3762, pp. 710–719. Springer, Heidelberg (2005)
31. Spyns, P.: Validating evalexon: validating a tool for evaluating automatically lexical triples mined from texts. Technical Report x6, STAR Lab, Brussel (2005)
32. Spyns, P., Hogben, G.: Validating an automated evaluation procedure for ontology triples in the privacy domain. In: Moens, M.-F., Spyns, P. (eds.) Proceedings of the 18th Annual Conference on Legal Knowledge and Information Systems (JURIX 2005), pp. 127–136. IOS Press, Amsterdam (2005)
33. Spyns, P., Pretorius, A.J., Reinberger, M.-L.: Evaluating DOGMA-lexons generated automatically from a text. In: Cimiano, P., Ciravegna, F., Motta, E., Uren, V. (eds.) EKAW 2004. LNCS (LNAI), vol. 3257, pp. 38–44. Springer, Heidelberg (2004)
34. Stvilia, B.: A model for ontology quality evaluation. *First Monday* 12(12) (2007)
35. Thompson Automation Software, Jefferson OR, US. Tawk Compiler, v.5 edition
36. Velardi, P., Missikoff, M., Basili, R.: Identification of relevant terms to support the construction of domain. In: Maybury, M., Bernsen, N., Krauwer, S. (eds.) Proc. of the ACL-EACL Workshop on Human Language Technologies (2001)
37. Vossen, P., Agirre, E., Calzolari, N., et al.: Kyoto: a system for mining, structuring and distributing knowledge across languages and cultures. In: Calzolari, N., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., Tapias, D. (eds.) Proceedings of the Sixth International Language Resources and Evaluation (LREC 2008), Paris, European Language Resources Association (2008)
38. Vrandečić, D., Sure, Y.: How to design better ontology metrics. In: May, W., Kifer, M. (eds.) ESWC 2007. LNCS, vol. 4519, pp. 311–325. Springer, Heidelberg (2007)
39. Zipf, G.K.: *Human Behaviour and the Principle of Least-Effort*. Addison-Wesley, Cambridge (1949)

Conceptual and Lexical Prototypicality Gradients Dedicated to Ontology Personalisation

Xavier Aimé^{1,3}, Frédéric Furst², Pascale Kuntz¹, and Francky Trichet¹

¹ LINA - Laboratoire d'Informatique de Nantes Atlantique (UMR-CNRS 6241)

University of Nantes - Team "Knowledge and Decision"

2 rue de la Houssinière BP 92208 - 44322 Nantes Cedex 03, France

pascale.kuntz@univ-nantes.fr, francky.trichet@univ-nantes.fr

² MIS - Laboratoire Modélisation, Information et Système

University of Amiens

UPJV, 33 rue Saint Leu - 80039 Amiens Cedex 01, France

frederic.furst@u-picardie.fr

³ Société TENNAXIA

37 rue de Châteaudun - 75009 Paris, France

xaime@tennaxia.com

Abstract. Since a long time, Domain Ontologies have been limited to scientific and technical domains. This situation has advantaged the sustainable development of "unbiased and universal knowledge". With the current emergence of Cognitive Sciences and the application of Ontology Engineering to Social and Human Sciences, the need to deal with subjective knowledge becomes more and more crucial. The aim of our work is to develop the notion of Personalised Vernacular Domain Ontology (PVDO). The principle underlying a PVDO consists in considering that an ontology O is not only specific to a delimited domain D , but is also peculiar to an endogroup E which shares a common pragmatics of D . This pragmatics, which complements the formal semantics of D , is defined during a process of ontology personalisation. This process is dependent on a context of use which includes several parameters, and in particular: culture, educational background and emotional state. Thus, ontologies co-evolve with their communities of use, and human interpretation of context in the use. Inspired by works in Cognitive Psychology, our contribution to ontology personalisation is based on the formal definition of two measures which aims at capturing subjective knowledge (*i.e.* the pragmatics of an ontology for knowledge (re)-using): (1) the *conceptual prototypicality gradient* evaluates the representativeness of a concept (resp. relation) within a local decomposition of a hierarchy and (2) the *lexical prototypicality gradient* evaluates the representativeness of a term within a set of terms used to denote a concept (resp. relation). In this way, these gradients aims at reflecting the degree of truth users of ontologies perceive on the *is-a* hierarchies and to what extent the terms associated to the concepts and relations are representative, respectively.

Keywords: Contextual Ontology, Typicality, Categorisation, Conceptual prototypicality, Lexical prototypicality, Information Retrieval, Subjective knowledge, Personalisation, Semantic Web, Pragmatic Web.

1 Introduction

Personalisation in the World Wide Web is a process of filtering the access to Web content according to the individual needs and requirements of each particular user [6]. Since the beginning of the nineties, *Personalisation* has become one of the major endeavours of research on the Web, in particular in Adaptive Hypermedia and Web Mining. The current advent of the Semantic Web, which leads to the introduction of machine-processable semantics accompanied with the development of automatic reasoning techniques, clearly improves the potentiality of the personalisation process. As Semantic Web is a content-aware navigation and fruition of resources, it is deeply connected to the idea of personalisation in its very nature [3]. At the heart of any personalisation system based on machine-processable semantics, we find ontologies which are mainly used to capture knowledge about the resources that can be queried (*i.e.* knowledge used for the semantic-based annotation process and the interpretation of the user's requests), and therefore are used to support the personalisation process. In our work, we claim that ontologies are of course crucial for personalisation, but that they must also be the *subject* of this process because ontologies only integrate *objective knowledge* whereas personalisation also requires *subjective knowledge*. In other words, we argue in favor of *Ontology Personalisation* as a means for Web - and Semantic Web - Personalisation.

This paper deals with *Subjective knowledge*, that is knowledge which is included in the semantic and episodic memory (*i.e.* declarative memory) of Human Being [10]. *Objective knowledge*, which can be expressed through documents of different natures (text, graphic, sound, etc.), corresponds to what must be captured within a Domain Ontology, as it is specified by the consensual definition of T. Gruber [9]: "*an ontology is a formal and explicit specification of a shared conceptualisation*". The advent of the Semantic Web and the standardisation of a Web Ontology Language (OWL) have led to the definition and the sharing of a lot of ontologies dedicated to scientific or technical fields. However, with the current emergence of Cognitive Sciences and the development of Knowledge Management applications in Social and Human Sciences, *subjective knowledge* becomes an unavoidable subject and a real challenge, which must be integrated and developed in Semantic Web, and more generally in Ontology Engineering [8]. Indeed, it can be relevant to wonder whether a Domain Ontology (DO) represents - in an objective and universal way - the entire knowledge of the field which is considered or if it just corresponds to individual and subjective interpretations of the reality or a sum of unshakable beliefs.

From a linguistic point of view, *pragmatics* studies how people comprehend and produce a communicative act in a concrete speech situation which is usually a conversation. In other words, pragmatics focuses on the parts of the language whose significance can only be understood according to a precise context of interpretation. *Syntax* is the study that relates signs to one another. *Semantics* is the study that relates signs to things in the world and patterns of signs to corresponding patterns that occur among the things the signs refer to. *Pragmatics* is the study that relates signs to the agents who use them to refer to things in the

world and to communicate their intentions about those things to other agents who may have similar or different intentions concerning the same or different things. Thus, in the context of Ontology Engineering, pragmatics aims at enriching the intrinsic formal semantics of an ontology by using elements describing the context of use. In our work¹, we particularly focus on the following dimensions of such a context: culture, educational background and emotional state of the end-user. Inspired by works in Cognitive Psychology, we advocate the definition of a specific process dedicated to *Ontology Personalisation*. Practically, this process consists in decorating the specialisation/generalisation links (“*is-a*” links) of the concepts and relations hierarchies of an ontology with 2 gradients. The goal of the first gradient, called **conceptual prototypicality gradient**, is to capture the user-sensitive degree of truth of the *categorisation process*, that is the one which is perceived by the end-user. The objective of the second gradient, called **lexical prototypicality gradient**, is to capture the user-sensitive degree of truth of the *lexicalisation process*, *i.e.* the definition of a set of terms used to denote a concept or a relation. These gradients aims at representing the multiple points of view that can be associated to the same ontology. They enrich the initial formal semantics of an ontology by adding a pragmatics defined according to a context of use which is dependent on parameters like culture, educational background or emotional context.

In order to intuitively justify the interest and the need of these gradients, let consider a simple example of information retrieval by using the current web search engine. If you enter the label *Dog*, the search engine returns documents whose contents have been indexed by this label. But “*Doggie*”, “*Poodle*” or “*Labrador*”² are not considered, whereas they are also interesting for searching documents related to “a dog”. Ontology-based information retrieval approaches can deal with this problem. But, this is not sufficient since the same results will be produced for all the users which, of course, do not really share the same searching background. Our approach, based on ontology personalisation, is more relevant since it aims at adapting the results to the profile of the end-user, by the means of the typicality which is subjectivity applied to the *is-a* semantic links between a concept and a sub-concept (resp. a relation and a sub-relation). By adopting this approach, the search engine, after having given the results about “*Dogs*”, can provide a set of mixed results related to *Labradors*, and then *Poodles*, etc. In other words, it can adapt and mix the results to the profile of the end-user - for instance, the fact that for the European community, Labradors are more representative, more typical, of the Dog concept than Poodles. This principle can also be applied to the synonyms of the term used to denote a

¹ This ongoing research project is funded by the French company Tennaxia (<http://www.tennaxia.com>). This “IT Services and Software Engineering” company provides industry-leading software and implementation services dedicated to Legal Intelligence in the following areas: Hygiene, Safety and Environment.

² which are synonyms of the term “*Dog*” or synonyms of a term used to denote a sub-concept of the concept intentionally defined as follows: “*a familiar mammal with four legs and able to bark*”.

concept (*e.g.* the label *Toutou* in French community is more common to denote the concept of *Dog* than the label *Tobby* or *Cabot*).

Our approach clearly contributes to the current trend related to the Pragmatic Web [16] which claims that it is not necessary to reach for context-independent ontological knowledge, as most ontologies used in practice assume a certain context and perspective of some community. The vision of the Pragmatic Web is to augment human collaboration effectively by appropriate technologies, such as systems for ontology negotiations, for ontology-based business interactions and for pragmatic ontology-building efforts in communities of interest and practice. In this view, the Pragmatic Web complements the Semantic Web by improving the quality and legitimacy of collaborative, goal-oriented discourses in communities. As recalled in [17], “*the best hope for the Semantic Web is to encourage the emergence of communities of interest and practice that develop their own consensus knowledge on the basis of which they will standardize their representations*”. Note that [5] proposes an extension of OWL (called C-OWL) for representing contextual ontologies. However, their interpretation of the notion of context is clearly different since they argue that not all knowledge should be integrated by an ontology, in the sense that knowledge can be mutually inconsistent. In that case, the ontology is contextualised in order to keep a local and consistent semantics. Our goal is clearly different since we consider that the semantics of the ontology O is fixed; we only use the context for defining a pragmatics of O and, therefore, for reflecting subjective knowledge.

The rest of this paper is structured as follows. Section 2 introduces the formal definition of the conceptual and lexical prototypicality gradients which are founded on the cognitive process of categorisation. Section 3 presents the distributional analysis of the gradients and the tool TOOPRAG which implements our approach; it also introduces some experimental results defined in the context of an application dedicated to the analysis of texts describing the Common Agricultural Policy (CAP) of the European Union.

2 Prototypicality Gradients

Defining an ontology O of a domain D at a precise time T consists in establishing a consensual synthesis of individual knowledge belonging to a specific endogroup; an endogroup is a set of individuals which share the same distinctive signs and, therefore, identify a community. For the same domain, several ontologies can be defined by different endogroups. We call *Vernacular Domain Ontologies* (VDO) this kind of resources; the qualifier *vernacular*, which comes from the latin word *vernaculus*, means native. For instance, vernacular architecture, which is based on methods of construction which use locally available resources to address local needs, tends to evolve over time to reflect the environmental, cultural and historical context in which it exists. This property is also described by E. Rosch as *ecological* [7,15], in the sense that although an ontology belongs to an endogroup,

it also depends on the context in which it evolves. Thus, given a domain D , an endogroup G and a time T , a VDO depends on three factors, characterising a precise context: (1) the culture of G , (2) the educational background of G and (3) the emotional state of G . In this way, a VDO can be associated with a pragmatic dimension. Indeed, a same VDO can be viewed (and used) from multiple points of view, where each point of view, although not reconsidering the formal semantics of D , allows to adapt (1) the truth degrees of the *isa* links defined between concepts (resp. relations) and (2) the expressivity degrees of the terms used to denote the concepts (resp. relations). We call *Personalised Vernacular Domain Ontologies* (PVDO) this kind of resources.

Our work is based on the fundamental idea that all the sub-concepts (resp. sub-relations) of a decomposition are not *equidistant* members, and that some sub-concepts (resp. sub-relations) are more representative of the super-concept (resp. super-relation) than others. This phenomenon is also applicable to the set of terms used to denote a concept (resp. a relation). This assumption is validated by works in Cognitive Psychology [11,10]. In order to calculate these differences of conceptual and lexical representativeness, we propose two measures:

1. the **conceptual prototypicality gradient**, which corresponds to a weighting of the *is-a* links (of the hierarchy of concepts and relations), and which qualifies the variations of representativeness of the sub-concepts and sub-relations;
2. the **lexical prototypicality gradient**, which corresponds to a weighting of the different terms used to denote a concept/relation, and which qualifies the variations of expressivity of the terms.

Formally, a VDO (given a field D and an endogroup G) is defined by the following t-uple:

$$O_{(D,G)} = \{\mathcal{C}, \mathcal{P}, \Omega_{(D,G)}, \leq^C, \sigma_P, L\} \text{ where}$$

- $\mathcal{C}, \mathcal{P}$ represent respectively the disjointed sets of concepts and properties³;
- $\Omega_{(D,G)}$ is a set of documents (*e.g.* text, graphic or sound documents) related to a domain D and shared by the members of the endogroup G ;
- $\leq^C$: $\mathcal{C} \times \mathcal{C}$ is a partial order on $\mathcal{C}$ defining the hierarchy of concepts ($\leq^C (c_1, c_2)$ means that the concept c_1 subsumes the concept c_2);
- σ_P : $\mathcal{P} \rightarrow \mathcal{C} \times \mathcal{C}$ which defines the domain and range of a property;
- $L = \{L_C \cup L_P, f_{term_C}, f_{term_P}\}$ is the lexicon related to dialect of G where:
 - L_C represents the set of terms associated to $\mathcal{C}$;
 - L_P represents the set of terms associated to $\mathcal{P}$;
 - the function $f_{term_C} : \mathcal{C} \rightarrow (L_C)^n$ which returns the tuple of terms used to denote a concept.
 - the function $f_{term_P} : \mathcal{P} \rightarrow (L_P)^n$ which returns the tuple of terms used to denote a property.

³ Properties include both attributes of the concepts and domain relations.

2.1 Intentional Component Based on Properties

The intentional component of our gradient aims at considering the intentional dimension of a conceptualisation.

For the concepts, our approach consists in comparing the properties of the concepts. The role of the features of a category (within the categorisation process) has been developed in [18]. In the context of our work, we advocate the following principle: the more a concept adds properties to those inherited from its super-concept, the more it is on the way of the specialisation and less prototypic it is, *i.e.* representative of its category. We consider this value as being the ratio between (1) the number of properties of the sub-concept and (2) the number of properties of the super-concept. Formally, the function $intentional : \mathcal{C} \times \mathcal{C} \rightarrow [0, 1]$ is defined as follows:

$$intentional(c_f, c_p) = \left(\frac{properties(c_p)}{properties(c_f)} \right)^n \quad (1)$$

where:

- $properties(c_p)$ is the number of properties of the super-concept c_p ;
- $properties(c_f)$ is the number of properties of the sub-concept c_f ;
- and n the number of the sub-concepts of c_p .

We raise the ratio to the power n in order to take the structure of the ontology into account, and thus to increase the strength of the concepts which own the higher values for the *intentional* function. The most prototypical concept of a decomposition is thus reinforced proportionally to the number of sub-concepts of this decomposition.

For the domain relations, we use a similar approach by considering the algebraic properties of the relations (symmetry, reflexivity, transitivity, irreflexivity, antisymmetry). Thus, the function $intentional : \mathcal{P} \times \mathcal{P} \rightarrow [0, 1]$ is defined as follows:

$$intentional(r_f, r_p) = \left(\frac{properties_{algebraic}(r_p)}{properties_{algebraic}(r_f)} \right)^n \quad (2)$$

where:

- $properties_{algebraic}(r_p)$ is the number of the algebraic properties of the super-relation r_p ;
- $properties_{algebraic}(r_f)$ is the number of the algebraic properties of the sub-relation r_f ;
- and n the number of the sub-relations of c_p .

However, the objectivity of an endogroup is an universal subjectivity. This component, in spite of being completely objective, is at least consensual for the endogroup which is considered. We study another way for this component with the amount of properties shared with other sub-categories [1,2].

2.2 Extensional Component Based on Frequencies

The extensional component of our gradient aims at taking the extensional view of a conceptualisation into account, through the terms used to denote the concepts/relations. This approach is based on the appearance frequency of a concept/relation related to a domain D , in an universe of the endogroup G . In this way, the more an element is frequent in the universe, the more it is considered as *representative/typical* of its category. This notion of typicality is introduced in the work of E. Rosch [7,15]. In our context, the universe of an endogroup is composed of the set of documents identified by $\Omega_{(D,G)}$. Our approach is inspired by the idea of Information Content introduced by Resnik [14]. Indeed, this is not because an idea is often expressed that it is really true and objective. Psychologically, it is recognised that the more an event is presented (in a frequent way), the more it is *judged* probable without being really true for an individual or an endogroup; this is one of the ideas defended by A. Tversky in its work on the evaluation of uncertainty [20].

Formally, the function $extensional_{G,D}(c_f, c_p) : \mathcal{C} \times \mathcal{C} \rightarrow [0, 1]$ (for a relation $extensional_{G,D}(r_f, r_p) : \mathcal{P} \times \mathcal{P} \rightarrow [0, 1]$) is defined as follows⁴:

$$extensional_{G,D}(c_f, c_p) = \frac{Information(c_f)}{Information(c_p)} \quad (3)$$

where:

$$Information(c) = \sum_{term \in world(c)} \left(\frac{count(term)}{N} * \frac{count(document, term)}{count(document)} \right) \quad (4)$$

with:

- $Information(c)$ defines the information content of the concept c (resp. the relation r);
- $count(term)$ returns how many times the *term* occurs in the documents of $\Omega_{(D,G)}$;
- $count(document, term)$ returns the number of documents of $\Omega_{(D,G)}$ where the *term* appears;
- $count(document)$ returns the number of documents of $\Omega_{(D,G)}$;
- $world(c)$ returns all the terms concerning the concept c via the function f_{term_c} (resp. the relation r via the function f_{term_p}) and all its sub-concepts (resp. sub-relations) from generation 1 to generation n ;
- N is the number of terms of $\Omega_{(D,G)}$.

⁴ This function, which is the same for the concepts and the relations, is only applicable if it exists:

- a direct *is-a* link between the super-concept c_p (resp. super-relation r_p) and the sub-concept c_f (resp. sub-relation r_f), with an order relation $c_f \leq c_p$ (resp. $r_f \leq r_p$),
- or an indirect link composed of a serie of *is-a* links between the c_p and c_f (resp. r_p and r_f).

Intuitively, the function *Information(c)* allows us to calculate “the ratio of use” of a concept/relation in an universe, by using first the terms directly associated to the concept/relation and then, by using the terms associated to all its sub-concepts (resp. sub-relation), from generation 1 to generation n . We balance each frequency by the ratio between the number of documents where the term is present and the global number of documents. An idea which is frequently presented in few documents is less relevant than an idea which is perhaps less defended in each document but which is presented in a lot of documents of the endogroup’s universe.

Note that it is possible to wonder on the adequacy of using the Inverse Document Frequency (IDF) [19] for the calculation of the function *Information*. Our approach is defined in a context of cognitive psychology, where the phenomenon of categorisation is combined with the recognition of forms. As IDF is a measure dedicated to semantic similarity, we considered that it is not really appropriated to our work which requires a measure dedicated to psychological proximity (which is clearly different to semantic similarity). However, we are currently making an experimentation in order to compare our gradients with semantic similarity measures.

2.3 Prototypicality Gradients

In the context of the process of categorisation (and classification), our gradients aims at taking both the *intentional* dimension (through the structure of the hierarchies) and the *extensional* dimension (through the appearance frequency of the terms used to denote the concepts/relations in the corpus) of an ontology into account. We use the expression *conceptual prototypicality gradients* because (1) what is calculated is not a distance but a real *gradient* which is used to balance the *is-a* links of the conceptual hierarchies, (2) we are able to define various degrees of *typicality* (the concept/relation which has the higher gradient is considered as the *prototypical category*) and (3) the gradients are calculated from the concepts/relations defined in intention and extension. To illustrate the use of our gradients, let consider a simple and intuitive example.

Let assume that we have defined the concept *Dog* with two properties - *Can bark* and *Can gnaw bones* - and two sub-concepts: *War Dog* (a War Dog *is a* Dog) and *Shepherd Dog* (a Shepherd Dog *is a* Dog). These two sub-concepts inherit the properties of the concept *Dog*; they can also add their own properties which are necessarily different from the ones of his brother. A *War Dog* can bark and can gnaw bones, but it *can also protect a human being*, it *can protect materials* and *can be considered as a weapon*: three additional properties compared to its super-concept *Dog*. A *Shepherd Dog* owns the same properties of any dog, but it can also *cluster sheep* (only one additional property compared to its super-concept). Thus, from a structural point of view, the sub-concept *War Dog* clearly specialises the concept *Dog* (since it adds 3 new properties). The sub-concept *Shepherd Dog*, because it only adds one property, is considered as being more representative of the concept *Dog*. This appreciation, based on the structure of the ontology, corresponds to the *intentional* component of our gradient, in

the sense that the intentional definitions of the concepts represent a stable and consensual agreement of the endogroup.

Then, why, in our Western community, we consider that Labrador is more representative of the concept *Dog* than Rotweiler? The answer can be related to the appearance frequency of the phenomenon in practice, to the familiarity with the phenomenon, etc. This *extensional* component, only based on our percepts and their interpretations, is calculated by the appearance frequency of a concept in the universe of the endogroup - knowing that each new document added to the universe can modify this judgment.

Now, the question is to know how our evaluation can change according to our emotions. Multiple works have been done in psychology on this subject [4,13]. The conclusion of these works can be summarized as follows: when we are in a negative mental state (*e.g.* fear or nervous breakdown), we tend to centre us on what appears to be the more important from an emotional point of view. In the context of our approach, it consists in reducing the universe to what is very familiar; for instance, our personal dog (or the one of a neighbor) - which at the beginning is inevitably the most characteristic of the category - becomes **the** and quasi unique dog. Respectively, in a positive mental state (*e.g.* love or joy), we are more open in our judgment and we accept more easily the elements which are not yet be considered as so characteristic. Thus, judgment and emotional states are strongly linked, and this relationship is clearly integrated (as parameters) in the definition of our gradients.

Conceptual Prototypicality Gradient (CPG). We define $cpg_{G,D} : \mathcal{C} \times \mathcal{C} \rightarrow [0,1]$ (resp. $\mathcal{P} \times \mathcal{P} \rightarrow [0,1]$) the function which, for all couple of concepts $c_f, c_p \in \mathcal{C}$ such as it exists an *is-a* link between the super-concept c_p and the sub-concept c_f ⁵, returns a real (null or positive value) which represents the conceptual prototypicality gradient of this link, in the context of a PVDO dedicated to a domain D and an endogroup G . For two concepts c_p and c_f (resp. two relations r_p and r_f), this function is defined as follows:

$$cpg_{G,D}(c_p, c_f) = [\alpha * \text{intentional}(c_f, c_p) + \beta * \text{extensional}_{G,D}(c_f, c_p)]^\gamma \quad (5)$$

with (1) $\alpha + \beta = 1$ and $\frac{\beta}{\alpha} = \mathcal{X}$, where $\alpha \geq 0$ a weighting of the intentional component, $\beta \geq 0$ a weighting of the extensional component, $\mathcal{X}$ the percentage of concepts of $\mathcal{C}$ which are directly (or indirectly) evoked in $\Omega_{(D,G)}$ and (2) $\gamma \geq 0$ a weighting of the mental state of the endogroup G .

The values of α and β can vary according to the domain which is considered, the will of the experts during the ontology development process, the context of use, etc. Stating the constraint $\frac{\beta}{\alpha} = \mathcal{X}$ allows us to fix the importance of knowledge expressed in the universe of the endogroup (*i.e.* extensional component) in comparison with innate knowledge (*i.e.* intentional component). These values, which are parameters of our algorithm, enable us, for instance, to reinforce the influence of the structural dimension of the ontology compared with the lexical

⁵ Respectively, for all couple of relations $r_f, r_p \in \mathcal{P}$ such as it exists an *is-a* link between the super-relation r_p and the sub-relation r_f .

dimension (in this case, the value of α is high and the value of β is low). The γ parameter aims at taking the mental state of the endogroup into consideration. According to [12], a negative mental state leads to the reduction of the value of representation, and conversely for a positive mental state. Thus, we characterize: (1) a *negative* mental state by a value $\gamma \in]1, +\infty[$, (2) a *positive* mental state by a value $\gamma \in]0, 1[$, and (3) a *neutral* mental state by the value 1.

When the value of γ is low, the value of the gradients associated to the concepts which are initially not considered as being so representative increases considerably, because a positive state facilitates the open mind, the valorisation, etc. Conversely, when the value of γ is high (*i.e.* a strongly negative mental state), the effect is to *select* only the concepts which own a high value of typicality, eliminating *de facto* the other concepts.

Lexical Prototypicality Gradient (LPG). The goal of this gradient is to evaluate the fact that the terms used to denote a concept or a relation have not the same representativeness within the endogroup. Indeed, the question is the following: “*why do we more frequently name the concept/relation x with the term y rather than z ?*”. To define these lexical variations, we propose to adapt the gradient previously defined, with the difference that we do not use the intentional component related to the properties. We base the formula on the Information Content of a concept or a relation, by using the ratio between the frequency of use of the term and the sum of the appearance frequencies of all the terms related to the concept/relation in $\Omega_{(D,G)}$.

Formally, we define $lpg_{G,D} : L_C \times \mathcal{C} \rightarrow [0, 1]$ the function, which for all concept/relation $c \in \mathcal{C}$ and the term $t \in L_C$ such as $t \in f_{termC}(c)$, returns a positive or null value representing the lexical prototypicality gradient of this term, and this for a domain D and an endogroup G . For the concepts, this function is defined as follows:

$$lpg_{G,D}(t, c) = \frac{1}{1 - \log\left(\frac{count(t)}{\sum count(f_{termC}(c))}\right)} \quad (6)$$

The form $\frac{1}{1 - \log(x)}$ has been adopted in order to obtain a non-linear behavior which is more close to human judgment. For the relations, this function $lpg_{G,D} : L_R \times \mathcal{R} \rightarrow [0, 1]$ is defined as follows:

$$lpg_{G,D}(t, r) = \frac{1}{1 - \log\left(\frac{count(t)}{\sum count(f_{termR}(r))}\right)} \quad (7)$$

3 Experimental Results

3.1 Distributional Analysis of the Gradients

In order to evaluate the distributional analysis of the CPG values on different types of hierarchies of concepts, we have developed a specific prototype whose parameters (given an ontology O) are: N the number of concepts of O , H the

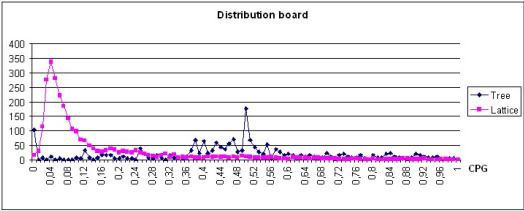


Fig. 1. Influence of the number of edges (with a constant number of concepts)

depth of O , and W the max width of O . From these parameters, the prototype automatically generates a random hierarchy of concepts. The results presented in figure 1 have been calculated in the following context:

- a hierarchy O_1 based on a tree described by $(N=800, H=9, W= 100)$;
- a hierarchy O_2 based on a lattice with a density of 0.5 described by $(N=800, H=9, W= 100)$;
- $\alpha = 0.5, \beta = 0.5$ and $\gamma = 1$.

These results clearly attest the fact that multiple inheritance leads to a dilution of the typicality notion.

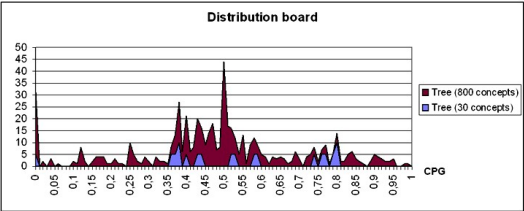


Fig. 2. Influence of the number of concepts in a tree

The results presented in figure 2 have been calculated in the following context:

- a hierarchy O_1 based on a tree described by $(N=800, H=9, W= 100)$;
- a hierarchy O_2 based on a tree described by $(N=50, H=2, W= 30)$;
- $\alpha = 0.5, \beta = 0.5$ and $\gamma = 1$.

These results indicate a relative stability of the distribution of CPG values, proportionally to the volume of the hierarchies, for a same density of graphs.

3.2 TOOPRAG: A Tool Dedicated to the Pragmatics of Ontology

TOOPRAG (*A Tool dedicated to the Pragmatics of Ontology*) is a tool dedicated to the automatic calculation of our gradients. This tool, implemented in Java 1.5,

```

...
<owl:Class rdf:ID="agricultural_labour_force">
  <rdfs:label xml:lang="EN" xml:lpg=0.7>farm worker</rdfs:label>
  <rdfs:label xml:lang="EN" xml:lpg=0.3>agricultural labour force</rdfs:label>
  <rdfs:subClassOf rdf:resource="#working_population_engaged_in_agriculture" xml:cpg=0.0074/>
</owl:Class>

<owl:Class rdf:ID="farmer">
  <rdfs:label xml:lang="EN" xml:lpg=0.375>grower</rdfs:label>
  <rdfs:label xml:lang="EN" xml:lpg=0.0>peasant</rdfs:label>
  <rdfs:label xml:lang="EN" xml:lpg=0.0>raiser</rdfs:label>
  <rdfs:label xml:lang="EN" xml:lpg=0.625>farmer</rdfs:label>
  <rdfs:subClassOf rdf:resource="#working_population_engaged_in_agriculture" xml:cpg=0.9841/>
</owl:Class>

<owl:Class rdf:ID="forest_ranger">
  <rdfs:label xml:lang="EN" xml:lpg=0.0>forest ranger</rdfs:label>
  <rdfs:subClassOf rdf:resource="#working_population_engaged_in_agriculture" xml:cpg=0.0/>
</owl:Class>

<owl:Class rdf:ID="agricultural_adviser">
  <rdfs:label xml:lang="EN" xml:lpg=0.0>agricultural adviser</rdfs:label>
  <rdfs:subClassOf rdf:resource="#working_population_engaged_in_agriculture" xml:cpg=0.0/>
</owl:Class>
...

```

Fig. 3. Extract of an OWL file produced by TOOPRAG

is based on Lucene⁶ and Jena⁷. It takes as inputs (1) an ontology represented in OWL 1.0, where each concept and relation is associated with a set of terms defined via the primitive *rdfs:label* (for instance, *<rdfs:label xml:lang="EN">farmer </rdfs:label>*) and (2) a corpus composed of text files. Thanks to the Lucene API, the corpus is first indexed. Then, the ontology is loaded in memory (via the Jena API) and the CPG values of all the *is-a* links of the concepts/relations hierarchies are computed. The LPG values of all the terms used to denote the concepts and the relations are also computed.

These results are stored in a new OWL file which extends the current specification of OWL 1.0. Indeed, as shown by figure 3, a LPG value is represented by a new attribute *xml:lpg* which is directly associated to the primitive *rdfs:label*. For instance, the LPG values of the terms “grower” and “peasant”, used to denote the concept “agricultural labour force” (*<owl:Class rdf:ID="agricultural_labour_force">*), are respectively 0.375 and 0. In a similar way, a CPG is represented by a new attribute *xml:cpg* which is directly associated to the primitive *rdfs:subClassOf*. For instance, the CPG values of the *is-a* links defined between the super-concept “working-population-engaged-in-agriculture” and its sub-concepts “agricultural-labour-force”, “farmer”, “forest-ranger” and “agricultural-adviser” are respectively 0.0074, 0.9841, 0 and 0.

3.3 Application in Agriculture

TOOPRAG has been used in a project dedicated to the analysis of texts describing the Common Agricultural Policy (CAP) of the European Union. In

⁶ Lucene is a high-performance, full-featured text search engine library written entirely in Java. Lucene is an open source project available at <http://lucene.apache.org/>.

⁷ Jena is a Java framework for building Semantic Web applications. It provides a programmatic environment for RDF, RDFS, OWL and SPARQL and includes a rule-based inference engine. Jena is an open source project available at <http://jena.sourceforge.net/>.

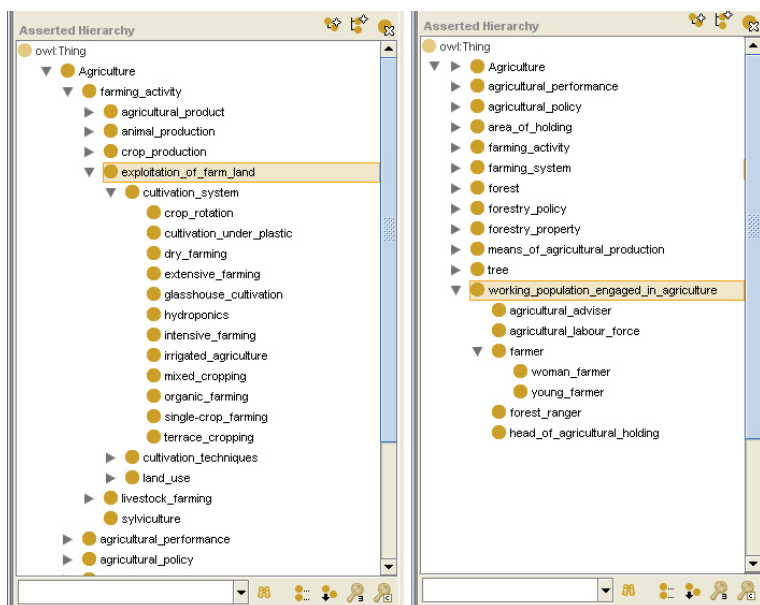


Fig. 4. Extract of the hierarchy of concepts of an ontology dedicated to Agriculture

this project, we have defined a specific ontology from the multilingual thesaurus Eurovoc (<http://europa.eu/eurovoc/>). This thesaurus, which exists in 21 official languages of the European Union, covers multiple fields (*e.g.* politics, education and communications, science, environment, agriculture, forestry and fisheries, energy, etc.). It provides a means of indexing the documents in the documentation systems of the European institutions and of their users (*e.g.* the European Parliament, some national government departments and European organisations). From the Eurovoc field dedicated Agriculture, we have defined a first hierarchy of concepts by using the hyponymy/hyperonymy relationships (identified by the “Broader Term” links in Eurovoc) and the synonymy relationships (identified by the “Used For” links in Eurovoc). Then, this hierarchy has been modified and validated by an expert in Agriculture and Forestry.

In its current version, this ontology includes a hierarchy of concepts based on a tree described by 283 concepts (depth=4 and max width=11). The lexicon of this ontology is composed of 597 terms. In average, each concept is associated with 2,1 terms (min=1 and max=11). Figure 4 shows an extract of this hierarchy loaded in Protégé (<http://protege.stanford.edu>).

The corpus used for this experimentation is composed of 55 texts published in the Official Journal of the European Union (<http://eur-lex.europa.eu>) since 2005, and in particular 43 regulations, 1 directive, 8 decrees, 3 community opinions. It includes 1.360.000 words. From a statistical point of view, 61 concepts of the ontology are directly evocated in the corpus through the terms and 37 indirectly (via inheritance). Thus, the reverse ratio is 34,63%.

```

...
<owl:Class rdf:ID="organic_farming">
  <rdfs:label xml:lang="EN" xml:lpq=1.0>organic farming</rdfs:label>
  <rdfs:subClassOf rdf:resource="#cultivation_system" xml:cpg=0.7/>
</owl:Class>

<owl:Class rdf:ID="intensive_farming">
  <rdfs:label xml:lang="EN" xml:lpq=0.0>intensive farming</rdfs:label>
  <rdfs:subClassOf rdf:resource="#cultivation_system" xml:cpg=0.0/>
</owl:Class>

<owl:Class rdf:ID="single-crop_farming">
  <rdfs:label xml:lang="EN" xml:lpq=0.0>single-crop farming</rdfs:label>
  <rdfs:subClassOf rdf:resource="#cultivation_system" xml:cpg=0.0/>
</owl:Class>

<owl:Class rdf:ID="extensive_farming">
  <rdfs:label xml:lang="EN" xml:lpq=1.0>extensive farming</rdfs:label>
  <rdfs:subClassOf rdf:resource="#cultivation_system" xml:cpg=0.0333/>
</owl:Class>

<owl:Class rdf:ID="dry_farming">
  <rdfs:label xml:lang="EN" xml:lpq=0.0>dry farming</rdfs:label>
  <rdfs:subClassOf rdf:resource="#cultivation_system" xml:cpg=0.0/>
</owl:Class>

<owl:Class rdf:ID="crop_rotation">
  <rdfs:label xml:lang="EN" xml:lpq=1.0>crop rotation</rdfs:label>
  <rdfs:subClassOf rdf:resource="#cultivation_system" xml:cpg=0.2667/>
</owl:Class>
...

```

Fig. 5. Extract of an OWL file produced by TOOPRAG

Although the ontology considered in this project does not yet include domain relations, the results provided by TOOPRAG (in the context of this specific corpus) are interesting because they help the expert to analyse the Common Agricultural Policy (CAP) of the European Union through regulatory texts. For instance, as shown by the figure 5, the CPG values clearly underline that since 2005, the cultivation system which is particularly encouraged by the PAC is the organic farming. In a similar way, the CPG values presented in Figure 3 state that the blue-collar workers of the agricultural secteur (*e.g.* farm workers, farmers or peasants) are more supported by the PAC than the white-collar workers (*e.g.* agricultural advisers or head of agricultural holdings).

4 Conclusions and Future Work

The purpose of our work, which is focused on the notion of “Personalised Vernacular Domain Ontology”, is to deal with *subjectivity knowledge* via (1) its specificity to an endogroupe and a domain, (2) its ecological aspect and (3) the prominence of its emotional context. This objective leads us to study the pragmatic dimension of an ontology. Inspired by works in Cognitive Psychology, we have defined two measures - identifying two complementary gradients - which are respectively dedicated to (1) the conceptual prototypicality which evaluates the representativeness of a concept/relation within a decomposition and (2) the lexical prototypicality which evaluates the representativeness of a term within a set of terms used to denote a concept. It is important to underline that these

gradients do not modify the formal semantics of the ontology which is considered; the subsumption links remain valid. These gradients only reflect the pragmatics of an ontology for knowledge (re)-using. Our gradients can be of effective help in different activities, such as:

- *Ontology Evaluation*. The prototypicality gradients are relevant indicators for judging *a fortiori* the quality of a categorisation, and consequently of a domain ontology (represented for instance in OWL). Indeed, to know which are the less typical concepts of a hierarchy (according to a context of use described by an endogroup and its universe) is a good way to wonder if these concepts are at the right place? Do we have to keep them for a given mental state? Conversely, when a concept is considered as being the most typical of a category, is it really in conformity with the judgement of the experts? And is this judgement (which is based on an *a priori* decision) the good one? In this context, what we claim is that our gradients are efficient and relevant measures in the sense that they tend to reflect the real appropriation of an ontology by an endogroup. The experts are free to confirm and to objectivize (or not) these results, in the context of a “reverse ontology engineering” process [8]. Of course, when the ontology has been developed from texts, the extensional component is very strong, and it can be interesting to equilibrate the points of view by adaptating the parameters of our gradients.
- *Information Retrieval*. The prototypicality gradients can be used to classify the results of a query, and more particularly an *extended* query, according to a relevance criteria which consists in considering the most representative element of a given concept (resp. a given term) as being the most relevant result of a query expressed by a (set of) term(s) denoting this concept (resp. corresponding to this term). This approach permits a classification of the extended results from a qualitative point of view. Moreover, our approach also allows us to proportion the number of results according to the value of the gradients (*i.e.* a quantitative point of view). Thus, information retrieval becomes customizable, because it is possible to adapt the results to the pragmatics of the ontology, *i.e.* privileging the intentional dimension (and not the extensional one) or conversely, working with different mental states, etc. In this way, *Ontology Personalisation* is used as a means for Web - and Semantic Web - Personalisation.
- *Ontological Analysis of text Corpora*. As introduced in section 3.3, our gradients can be used to evaluate the ontological contents of text corpora: which are the main concepts involved in a text corpus? By using the same ontology applied on different corpora related to the same domain, it is possible to compare, at the conceptual level, the Information Content of these corpora. We currently evaluate this approach in the context of an experimentation which aims at making a comparative analysis of the healthcare preoccupations of different populations, in particular French, English and American population. For this purpose, we currently define an multilingual

ontology from the MeSH⁸. In order to really deal with the preoccupations of people, we have selected the three most popular and complete medical websites where french, english and american people can find and exchange all the information they are looking about their health care needs: Doctissimo in France (*www.doctissimo.fr*), Healthcare Republic in United Kingdom (*www.healthcarerepublic.com*) and HealthCare.com in USA (*www.healthcare.com*). These websites will be considered as three distinctive corpora from which our gradients will be calculated, by using the same multilingual ontology.

Our work is currently in progress towards the improvement of the gradients according to many works related to Cognitive and Social Psychology. We also study how to enrich the intentional component by taking the axiomatic part of an ontology into account. From an application point of view, we currently evaluate our approach in the context of a project dedicated to Legal Intelligence within regulatory documents related to the areas “Hygiene, Safety and Environment”.

References

1. Au Yeung, C.M., Leung, H.F.: Formalizing typicality of objects and context-sensitivity in ontologies. In: AAMAS 2006: Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems, pp. 946–948. ACM, New York (2006)
2. Au Yeung, C.M., Leung, H.F.: Ontology with likeliness and typicality of objects in concepts. In: Embley, D.W., Olivé, A., Ram, S. (eds.) ER 2006. LNCS, vol. 4215, pp. 1611–3349. Springer, Heidelberg (2006)
3. Baldoni, M., Baroglio, C., Henze, N.: Personalization for the semantic web. In: Reasoning Web, pp. 173–212 (2005)
4. Bluck, S., Li, K.: Predicting memory completeness and accuracy: Emotion and exposure in repeated autobiographical recall. *Applied Cognitive Psychology* (15), 145–158 (2001)
5. Bouquet, P., Giunchiglia, F., van Harmelen, F., Serafini, L., Stuckenschmidt, H.: Contextualizing ontologies. *Journal of Web Semantics* 1(4), 325–343 (2004)
6. Brusilovsky, P., Kobsa, A.: The Adaptive Web: Methods and Strategies of Web Personalization. Springer, Heidelberg (2007) ISBN 978-3-540-72078-2
7. Gabora, L.M., Rosch, E., Aerts, D.: Toward an ecological theory of concepts. *Ecological Psychology* 20(1-2), 84–116 (2008)
8. Gomez-Perez, A., Fernandez-Lopez, M., Corcho, O.: Ontological Engineering. In: Advanced Information and Knowledge Processing. Springer, Heidelberg (2003)
9. Gruber, T.: Toward principles for the design of ontologies used for knowledge sharing. In: Guarino, N., Poli, R. (eds.) Formal Ontology in Conceptual Analysis and Knowledge Representation, Deventer, The Netherlands. Kluwer Academic Publishers, Dordrecht (1993)

⁸ MeSH is the U.S. National Library of Medicine’s controlled vocabulary (<http://www.nlm.nih.gov/mesh/meshhome.html>). MeSH terminology provides a consistent way to retrieve information that may use different terminology for the same concepts.

10. Harnad, S.: Categorical perception. *Encyclopedia of Cognitive Science* LXVII(4) (2003)
11. McEvoy, M.E., Nelson, D.L.: Category norms and instance norms for 106 categories of various sizes. *American Journal of Psychology* 95, 462–472 (1982)
12. Mikulinger, M., Kedem, P., Paz, D.: Anxiety and categorization-1, the structure and boundaries of mental categories. *Personality and individual differences* 11(11), 805–814 (1990)
13. Park, J., Nanaji, M.: Mood and heuristics: The influence of happy and sad states on sensitivity and bias in stereotyping. *Journal of Personality and Social Psychology* (78), 1005–1023 (2000)
14. Resnik, P.: Using information content to evaluate semantic similarity in a taxonomy. In: 14th International Joint Conference on Artificial Intelligence (IJCAI 1995), vol. 1, pp. 448–453 (1995)
15. Rosch, E.: Cognitive reference points. *Cognitive Psychology* (7), 532–547 (1975)
16. Schoop, M., de Moor, A., Dietz, J.L.G.: The pragmatic web: a manifesto. *Commun. ACM* 49(5), 75–76 (2006)
17. Singh, M.P.: The pragmatic web. *IEEE Internet Computing* 6(3), 4–5 (2002)
18. Smith, E.E., Shoben, E.J., Rips, L.J.: Structure and process in semantic memory: a featural model for semantic decisions. *Psychological Review* (81), 214–241 (1974)
19. Sparck-Jones, K.: A statistical interpretation of term specificity and its application to retriever. *Journal of documentation* 28(1), 11–21 (1972)
20. Tversky, A., Kahneman, D.: Judgment under uncertainty: Heuristics and biases. *Science* (185), 1124–1131 (1974)

Explanation in the *DL-Lite* Family of Description Logics

Alexander Borgida¹, Diego Calvanese², and Mariano Rodriguez-Muro²

¹ Dept. of Computer Science
Rutgers University, USA

borgida@cs.rutgers.edu

² Faculty of Computer Science

Free University of Bozen-Bolzano, Italy

{calvanese,rodriguez}@inf.unibz.it

Abstract. In ontology-based data access (OBDA), access to (multiple) incomplete data sources is mediated by a conceptual layer constituted by an ontology. In such a setting, to correctly compute answers to queries, it is necessary to perform complex reasoning over the constraints expressed by the ontology. We consider the case of ontologies expressed in *DL-Lite*, a family of DLs that, in the context of OBDA, provide an optimal tradeoff between expressive power and computational complexity of reasoning; notably conjunctive query answering is LOGSPACE in the size of the data. However, query answering with reasoning comes at a price: the justification of the presence of tuples in answers is no longer trivial, and requires *explanation*. In this paper, we characterize reasoning in *DL-Lite*, through deduction rules for building proofs, and we provide several novel contributions: (i) For standard ontology level reasoning, explanation is relatively simple, and our contribution comes mainly from a novel focus on brevity of proofs. (ii) Motivated by the use of *DL-Lite* for OBDA, we analyze and provide explanation for reasoning in finite models. (iii) We provide a facility for the explanation of an answer to a conjunctive query over a *DL-Lite* ontology. This algorithm is able to exploit the relational query engine to extract from the data the information necessary for finding the explanation more efficiently, and thus scales to large data sets. The presented approach has been implemented in a prototype for constructing explanations.¹

1 Introduction

Semantic data models such as the Extended ER model (EER) and UML are well known to provide a view of an application domain that is closer to the users' conceptualization of it than standard databases. As a result, there have long been proposals for querying databases through interfaces that offer such conceptual schemas to users (e.g., [2]).

On the other hand, Description Logics (DLs) [3] are a family of knowledge representation schemes developed over the past three decades that deal with concepts (unary relationships) and roles (binary relationships), which can be built up from atomic symbols using special concept and role *constructors*. These logics have precise formal semantics, and support sound and complete reasoning about judgments such as whether one concept subsumes/is more general than another, or whether a concept is unsatisfiable.

¹ Preliminary results on the research reported in this paper appeared in the Working Notes of the 2008 Workshop on Description Logics [1].

One application of DLs is providing the formal foundation of web ontology languages such as OWL-DL². More relevantly to this paper, it is known that EER and UML conceptual models can be translated into sets of DL axioms (called “TBoxes”), as in [4,5]. As a result, it is possible to use DL reasoners to detect inconsistencies in EER and UML models (e.g., classes that cannot possibly have any instances).

However, it is also well known that more expressive DLs (those with more constructors) tend to have a higher complexity of reasoning. The DL *DL-Lite* [6] was introduced to capture as much as possible of EER and UML conceptual models while still having effective reasoning. Moreover, in various contexts, such as data integration and ontology-based data access [7], data sources can be queried through a conceptual schema (or an ontology) that provides a formalization of the domain of interest. The key difficulty in such a setting is that the data stored in the sources is in general incomplete w.r.t. the constraints imposed by the schema, and hence have to be considered under the “open world assumption”. As a consequence, sophisticated reasoning may be required to obtain answers. For example, if the schema specifies that *Undergrads* are *Students*, and *Students* must be enrolled in at least two courses, then, in answering the query $q(x) \leftarrow \text{Undergrads}(x) \wedge \text{enrolledIn}(x, y)$, a system should be able to infer that all instances of *Undergrads* should be returned, without checking for each of it if there is an *explicitly* mentioned course it is enrolled in. Nevertheless, when the schema is expressed in *DL-Lite*, conjunctive queries can be answered with low data complexity (LOGSPACE, as in ordinary databases), while fully taking into account the constraints imposed by the schema [6].

Reasoning comes however at a price: end-users of information systems that do more than simple fact retrieval require some sort of facility for having answers *explained* to them. For example, in the area of deductive databases there has been work on explaining answers returned by Datalog-query processors [8,9]. Finding such explanations is non-trivial since the performance systems that do query answering are optimized, and do not use straightforward inference rules, such as back-chaining.

In the field of DLs, starting from [10], there have been papers studying the explanation of deductions such as concept subsumption [11,12] and knowledge base inconsistency [13,14,15,16]. More generally, the work on the Inference Web [17] has produced a substrate on which general explanation facilities for reasoners can be built.

Here we consider the problem of explaining reasoning and query answering for *DL-Lite_A*, the most expressive variant of the *DL-Lite* family considered in [6]. As for any DL, there are standard judgments such as concept/role subsumption, concept/role satisfiability and consistency of an ontology, requiring more or less standard explanations. Because of its use for database conceptual modeling, and the fact that databases are almost always considered to represent *finite structures* in which the conceptual models are interpreted, one important novel feature of the above reasoning tasks is the possibility of requiring finite models (which for *DL-Lite* enable additional inferences because the logic lacks the finite-model property). Example 3 in Section 3.4 illustrates first in English and then using inference rules the kind of reasoning that is needed in finite models. A second distinguishing feature of *DL-Lite* is the emphasis on *conjunctive query answering*, which requires new kinds of explanations. Considering the query in

² <http://www.w3.org/2007/OWL/>

Algorithm: breadth-first search for finding disjoint ancestors.

Input: concept B ; set T of disjointness and acyclic concept inclusion assertions.

Output: all pairs of concepts X, Y that are $\sqsubseteq$ -ancestors of B and are declared disjoint in T .

/ The search is breadth-first in the sense that, if (X_1, Y_1) and (X_2, Y_2) are two output pairs, and $\max(\text{dist}(B, X_1), \text{dist}(B, Y_1)) < \max(\text{dist}(B, X_2), \text{dist}(B, Y_2))$, then (X_1, Y_1) is output first. */*

/ Data structures: queues q_1 and q_2 hold pairs (V, k) , where V is a node and k is an integer representing the distance from B to V . */*

```
{
  new( $q_1$ );
   $q_1.\text{enter}(B, 0)$ ;  /* start outer BFS from node B */
  while (not  $q_1.\text{empty}()$ ) {
    ( $X, n$ )  $\leftarrow q_1.\text{leave}()$ ;
    for all  $D$  in parents( $X, T$ ) {  $q_1.\text{enter}(D, n+1)$ ; }
    new( $q_2$ );
     $q_2.\text{enter}(B, 0)$ ;  /* start new BFS from node B */
    while (not  $q_2.\text{empty}()$ ) {
      ( $Y, m$ )  $\leftarrow q_2.\text{leave}()$ ;
      if ( $m > n$ ) then exit loop /* to look at smallest  $m+n$  pairs only */
      for all  $D$  in parents( $Y, T$ ) {  $q_2.\text{enter}(D, m+1)$ ; }
      if disjoint( $X, Y, T$ ) then
        print  $X + ", " + Y + "$  are a source of unsatisfiability at proof length " +  $(n+m)$ ;
    }
  }
}
```

Fig. 1. Breadth-first search algorithm for finding disjoint ancestors

```
Student(BOB)
  { by Subconcept rule from
    PhD  $\sqsubseteq$  Student { Axiom 1 }
    PhD(BOB) { DB fact } }
supervisedBy(BOB, @1) (*)
  { by Subconcept rule from
    PhD  $\sqsubseteq$  dom(supervisedBy) { Axiom 2 }
    PhD(BOB) { DB fact } }
teaches(@1, @2)
  { by Subconcept rule from
    Professor  $\sqsubseteq$  dom(teaches) { Axiom 4 }
    Professor(@1)
      { by Subconcept rule from
        rng(supervisedBy)  $\sqsubseteq$  Professor { Axiom 3 }
        supervisedBy(BOB, @1) { see (*) } } }
```

Fig. 2. Proof tree generated from Step 4 for $Q_5(\text{BOB})$

Example 4, the kind of explanation needed in this case for the answer BOB is shown in Figure 2.

The rest of the paper has the following structure: Section 2 provides formal background on *DL-Lite* and general desiderata for explanations; Section 3 considers the

relatively straightforward reasoning tasks associated with *DL-Lite*, characterizing them in terms of inference rules, but also looks at the more unusual notion of finite-model reasoning; Section 4 considers in detail the task of explaining why some value is returned as one of the answers to a conjunctive query posed to a *DL-Lite* ontology.

2 Background

In this section we provide the formal background for the techniques and results in the rest of the paper. Specifically, we first introduce the Description Logic we deal with, and then provide some basic notions about explanations.

2.1 The *DL-Lite* Family of Description Logics

Description Logics (DLs) [3] are logics that represent the domain of interest in terms of *concepts*, denoting sets of objects, and *roles*, denoting binary relations between (instances of) concepts. Complex concept and role expressions are constructed starting from a set of atomic concepts and roles by applying suitable constructs, that depend on the DL at hand. In this paper, we deal with the *DL-Lite* family [6,18], which comprises tractable DLs particularly suited for accessing through an ontology large amounts of data managed through relational database technology. Specifically, we consider *DL-Lite_A* [19], one of the most expressive members of the family still enjoying LOGSPACE data complexity of query answering³.

Concepts and roles in *DL-Lite_A* are formed according to the following syntax:

$$\begin{array}{ll} B \longrightarrow A \mid \exists Q & Q \longrightarrow P \mid P^- \\ C \longrightarrow B \mid \neg B & R \longrightarrow Q \mid \neg Q \end{array}$$

where A , B , and C respectively denote an *atomic concept*, a *basic concept*, and a *general concept* (or simply, concept), whereas P , Q , and R respectively denote an *atomic role*, a *basic role*, and a *general role* (or simply, role).

Intuitively, a basic role of the form P^- denotes the *inverse* of the relation denoted by role P . A basic concept of the form $\exists P$ (resp., $\exists P^-$) denotes the projection of the relation denoted by P on its first (resp., second) component. An arbitrary concept $\neg B$ (resp., an arbitrary role $\neg Q$) denotes the complement of B (resp., Q).

A DL *knowledge base* (KB) $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ represents the domain of interest and consists of two parts, a TBox $\mathcal{T}$, representing intensional knowledge, and an ABox $\mathcal{A}$, representing extensional knowledge. In *DL-Lite_A*, a TBox is formed by a set of assertions of the following forms:

$$\begin{array}{ll} B \sqsubseteq C & \text{concept inclusion assertion} \\ Q \sqsubseteq R & \text{role inclusion assertion} \\ (\text{funct } Q) & \text{functionality assertion} \end{array}$$

³ We ignore here the distinction, present in *DL-Lite_A*, between abstract objects and data values.

The concept inclusion assertion $B \sqsubseteq C$ expresses that all instances of the (basic) concept B are also instances of the (general) concept C . Analogously for a role inclusion assertion. A functionality assertion expresses the (global) functionality of a basic role⁴.

Example 1. The following TBox

$$PhD \sqsubseteq Student \quad (1) \qquad \exists takes^- \sqsubseteq Course \quad (3)$$

$$\exists takes \sqsubseteq Student \quad (2) \qquad audits \sqsubseteq (\neg takes) \quad (4)$$

asserts that *PhDs* are a subclass of *Students*, who are the only ones who can *take* things, while things taken must be *Courses*; the last axiom is used to express that *takes* and *audits* are disjoint. ■

An *ABox* is formed by a set of *membership assertions* on atomic concepts and on atomic roles of the form

$$A(d) \qquad P(d_1, d_2)$$

stating respectively that the object (denoted by the constant) d is an instance of A , and that the pair (d_1, d_2) of objects is an instance of the role P .

Formally, the semantics of a DL is given in terms of interpretations, where an *interpretation* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ consists of an interpretation domain $\Delta^{\mathcal{I}}$ and an *interpretation function* $\cdot^{\mathcal{I}}$ that assigns to each concept C a subset $C^{\mathcal{I}}$ of $\Delta^{\mathcal{I}}$, and to each role R a binary relation over $\Delta^{\mathcal{I}}$. In particular, for the constructs of $DL-Lite_A$ we have:

$$\begin{aligned} A^{\mathcal{I}} &\subseteq \Delta^{\mathcal{I}} & P^{\mathcal{I}} &\subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \\ (\exists Q)^{\mathcal{I}} &= \{o \mid \exists o'. (o, o') \in Q^{\mathcal{I}}\} & (P^-)^{\mathcal{I}} &= \{(o_2, o_1) \mid (o_1, o_2) \in P^{\mathcal{I}}\} \\ (\neg B)^{\mathcal{I}} &= \Delta^{\mathcal{I}} \setminus B^{\mathcal{I}} & (\neg Q)^{\mathcal{I}} &= \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \setminus Q^{\mathcal{I}} \end{aligned}$$

An interpretation $\mathcal{I}$ *satisfies* an inclusion assertion $B \sqsubseteq C$ (resp., $Q \sqsubseteq R$) if $B^{\mathcal{I}} \subseteq C^{\mathcal{I}}$ (resp., $Q^{\mathcal{I}} \subseteq R^{\mathcal{I}}$). Furthermore, $\mathcal{I}$ satisfies an assertion (funct Q) if the binary relation $Q^{\mathcal{I}}$ is a function, i.e., $(o, o_1) \in Q^{\mathcal{I}}$ and $(o, o_2) \in P^{\mathcal{I}}$ implies $o_1 = o_2$. To specify the semantics of membership assertions, we extend the interpretation function to constants, by assigning to each constant a a *distinct* object $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$. Note that this implies that, as usual in DLs, we enforce the *unique name assumption* on constants [3]. An interpretation $\mathcal{I}$ satisfies a membership assertion $A(d)$ (resp., $P(d_1, d_2)$) if $d^{\mathcal{I}} \in A^{\mathcal{I}}$ (resp., $(d_1^{\mathcal{I}}, d_2^{\mathcal{I}}) \in P^{\mathcal{I}}$). A *model* of a KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ is an interpretation that satisfies all assertions in $\mathcal{T}$ and $\mathcal{A}$. A KB is *satisfiable* if it has at least one model. A KB $\mathcal{K}$ *logically implies* (an assertion) α , written $\mathcal{K} \models \alpha$, if all models of $\mathcal{K}$ satisfy α . Specifically, a concept C_1 (resp., role R_1) is *subsumed by* a concept C_2 (resp., role R_2) w.r.t. $\mathcal{K}$ if $\mathcal{K} \models C_1 \sqsubseteq C_2$ (resp., $\mathcal{K} \models R_1 \sqsubseteq R_2$). A concept C (resp., role R) is *satisfiable* w.r.t. $\mathcal{K}$ if there is a model $\mathcal{I}$ of $\mathcal{K}$ such that $C^{\mathcal{I}} \neq \emptyset$ (resp., $R^{\mathcal{I}} \neq \emptyset$). Satisfiability and subsumption are the fundamental reasoning tasks over a TBox. Unsatisfiable concepts are typically the result of modeling errors, and should be removed from a KB. Subsumption is the basis of classification, making the structure of the modeled knowledge explicit.

⁴ In order to guarantee the computational properties that allow for dealing efficiently with large amounts of data, $DL-Lite_A$ requires that, roughly, functional roles cannot be specialized in TBoxes [18,19].

In *DL-Lite_A*, due to the interaction of inclusion and functionality assertions, there may be inferences that do not hold in arbitrary models, but that do hold when only models with a finite domain are considered. In other words, reasoning w.r.t. arbitrary models differs from *finite model reasoning*. All the above reasoning tasks can be defined for the latter case as well.

We are also interested in query answering over *DL-Lite_A* KBs, and specifically in answering conjunctive queries. A *conjunctive query* (CQ) over a *DL-Lite_A* KB $\mathcal{K}$ has the form

$$Q(\mathbf{x}) \leftarrow \text{conj}(\mathbf{x}, \mathbf{y})$$

where $\text{conj}(\mathbf{x}, \mathbf{y})$ is a conjunction of atoms of the form $A(z)$ or $P(z_1, z_2)$, with A and P respectively atomic concepts and roles of $\mathcal{K}$, and z, z_1, z_2 either constants in $\mathcal{K}$ or variables in $\mathbf{x}$ or $\mathbf{y}$. The variables $\mathbf{x}$ are the so-called *distinguished variables* (which will be bound with constants in the KB), while $\mathbf{y}$ are the *non-distinguished variables* (which are existentially quantified). For example, the following simple query $q(w) \leftarrow \text{PhD}(w) \wedge \text{takes}(w, z)$ asks for PhDs who are taking something.

Given an interpretation $\mathcal{I}$, the conjunctive query $Q(\mathbf{x}) \leftarrow \text{conj}(\mathbf{x}, \mathbf{y})$ is interpreted as the set $Q(\mathbf{x})^{\mathcal{I}}$ of tuples $\mathbf{o}$ of elements of $\Delta^{\mathcal{I}}$ such that, when we assign $\mathbf{o}$ to $\mathbf{x}$, the first-order formula $\exists \mathbf{y}. \text{conj}(\mathbf{x}, \mathbf{y})$ evaluates to true in $\mathcal{I}$.

The reasoning service we are interested in is (*conjunctive*) *query answering*: given a knowledge base $\mathcal{K}$ and a conjunctive query $Q(\mathbf{x})$ over $\mathcal{K}$, compute the *certain answers* to $Q(\mathbf{x})$ over $\mathcal{K}$, i.e., the tuples $\mathbf{d}$ of constants in $\mathcal{K}$ such that $\mathbf{d}^{\mathcal{I}} \in Q(\mathbf{x})^{\mathcal{I}}$ for every model $\mathcal{I}$ of $\mathcal{K}$. We observe that query answering (properly) generalizes a well known reasoning service in DLs, namely *instance checking*, i.e., logical implication of an ABox assertion. In particular, instance checking can be expressed as the problem of answering (boolean) conjunctive queries constituted by just one ground atom.

2.2 Explanations

It is widely accepted that an explanation corresponds to a formal *proof*. A formal proof is constructed from premises using rules of inference. Although [20] suggests a specific XML-based syntax for inference rule schemas to be used in constructing proofs, we will use the more concise notation used in Programming Languages, and first applied to DLs in [21], which is illustrated in the following inference rule, expressing in one way the transitivity of the $\sqsubseteq$ relationship:

$$\text{Isa-trans} \quad \frac{\begin{array}{l} \mathcal{T} \vdash B_1 \sqsubseteq B_2 \\ \mathcal{T} \vdash B_2 \sqsubseteq B_3 \end{array}}{\mathcal{T} \vdash B_1 \sqsubseteq B_3} \quad B_1, B_2, B_3 \text{ concepts}$$

Here, the name of the rule schema is **Isa-trans**; the antecedent requires that from the TBox $\mathcal{T}$ one can deduce $B_1 \sqsubseteq B_2$ and also $B_2 \sqsubseteq B_3$; the consequent allows one to also deduce from the same TBox that $B_1 \sqsubseteq B_3$; the side-condition of the rule requires B_1, B_2 , and B_3 to be concept expressions.

The rules of inference used and the proof itself have certain intuitively desirable properties as far as the understandability of the resulting explanation⁵. These include:

⁵ Ideally, empirical user studies would support these claims; as it is, we rely on our and the readers' intuitions.

- *Simplicity*: while some rules of inference (e.g., concluding p from q and “if q then p ”) are self-evident, others may be so complex that in explaining an inference step one needs to also explain the validity of the inference rule, in addition to explaining the antecedents. Such rules should be avoided, if possible.
- *Brevity*: all things being equal, shorter proofs are preferred, since they take less time to present and understand; note that one way in which to make proofs shorter is to find portions, called lemmas, that are re-used.

Note that the above principles may conflict (e.g., a simpler proof may be longer), and therefore in general there is likely to be no single “ideal” explanation system – more likely one with “knobs” that can be adjusted.

Although not frequently articulated, an explanation involves not only a proof but also a *proof presentation strategy*. For example, there is a decided preference for tree-shaped proofs, produced by rules of inference with a single conclusion, and zero or more antecedents. So-called “natural deduction proofs” produce such proofs, and many explanation facilities for description logics and the semantic web follow these principles (e.g., [10,17,22]). One of the advantages of such proofs is that they support interactive and gradual unfolding of only relevant parts under user control. These ideals are exemplified by Horn logic, where the explanation of goal g provides as a first step the rule $p \wedge q \rightarrow g$ from which g was deduced, and then allows the user to choose follow-up questions concerning the derivation of p and/or q .

While it is possible to present proofs using a very mechanical approach, which produces the same format for all rules of inference, this is not a necessity, and flexibility can lead to improvements. For example, most inference systems from sets of axioms have a reiteration rule of the form

$$\text{Given } \frac{}{\mathcal{T}, \psi \vdash \psi} \psi \text{ any axiom}$$

which allows any axiom from the theory to be used in a proof. It is better to replace this by a scheme where all axioms in $\mathcal{T}$ are numbered, and whenever some other inference rule uses ψ as an antecedent, ψ is listed, with its number as justification. More interestingly, certain sub-proofs may be judged to be too trivial/obvious, and can therefore be eliminated from the proof when presented as an explanation. A simple example of this involves conjunction exploitation, where we simply allow a proof to use axiom $p \wedge q$, when what is required is p . In a similar vein, in [10] some kinds of inheritance were explained by indicating the ancestor from which the inherited constraint was obtained, without explicitly listing all the intermediate concepts through which inheritance passed.

Of course, there is also the possibility of generating graphical explanation proof trees, or natural language text [23].

3 Explanations of Standard Inferences

3.1 Modified Syntax of $DL\text{-}Lite_{\mathcal{A}}$

A number of notions in $DL\text{-}Lite_{\mathcal{A}}$ (and their notation), such as existential constraints, inverses of roles, and complements of concepts or roles are, in our opinion, mathematically

too sophisticated for users familiar only with notations like UML diagrams⁶. For this reason, we propose to alter the surface syntax shown to users.

As far as $\neg$ is concerned, we observe that the restricted occurrence of negated concepts (resp., roles) in *DL-Lite*_A, in axioms of the form $B_1 \sqsubseteq \neg B_2$ (resp., $Q_1 \sqsubseteq \neg Q_2$), means that these are really only used to describe that two (un-negated) concepts (resp., roles) are *disjoint*. Hence, we replace each assertion of the form $B_1 \sqsubseteq \neg B_2$ (resp., $Q_1 \sqsubseteq \neg Q_2$) by the assertion (disjoint $B_1 B_2$) (resp., (disjoint $Q_1 Q_2$)), having the same semantics. This eliminates $\neg$ from our axioms, and also from our explanations.

Next, we propose to eliminate the notations $\exists P$ and $\exists P^-$, and replace them by the more familiar notions of “the current domain” and “the current range of role P ”, respectively, written as $\text{dom}(P)$ and $\text{rng}(P)$ ⁷. As a result of the above simplifications, concept inclusion assertions will now only relate atomic concepts and/or current domains/ranges of roles. And in addition to subsumption, we have axioms for disjointness of concepts.

The above transformation also has the desirable effect of eliminating role inverses from concept inclusions. The remaining use of role inverses is in functionality assertions of the form $(\text{funct } P^-)$, and in role inclusion assertions. In general, ontology designers are encouraged to declare names for inverse roles (as in UML and OWL 1.1) by using an assertion of the form $(\text{inverseRoles } P \text{ idForInvOf } P)$ (e.g., $(\text{inverseRoles makes madeBy})$) and then using $\text{idForInvOf } P$ instead of P^- .

Unfortunately, this will not allow us to completely ignore the role inverse notation, as illustrated by the following example.

Example 2. Suppose the TBox $\mathcal{T}$ contains the role inclusion assertions:

$$P_1 \sqsubseteq P_2^- \quad P_2 \sqsubseteq P_3 \quad P_1 \sqsubseteq P_4^- \quad P_4 \sqsubseteq P_5 \quad P_3 \sqsubseteq \neg P_5$$

Observe that P_1 is unsatisfiable in $\mathcal{T}$. The reason is that the first two inclusions imply (by **Isa-trans** and **IsaInv** below) that $P_1 \sqsubseteq P_3^-$; similarly, the next two inclusions imply that $P_1 \sqsubseteq P_5^-$. By the last inclusion, P_3 and P_5 are disjoint, and hence so are P_3^- and P_5^- . Hence, P_1 , being subsumed by two disjoint roles, is unsatisfiable. Note that in this proof we referred to P_3^- and P_5^- in explaining the unsatisfiability of P_1 , even though neither of these role inverses appears in the TBox. ■

Therefore, we will in general not be able to avoid the need of confronting the user with role inverses, when she has not specified an alternate name for the inverse of a role⁸.

In the following subsections we present the rules of inference required for sound and complete reasoning about a variety of judgments. Because these will be relatively simple, we will not spend any time on issues of proof presentation.

3.2 TBox Reasoning

Subsumption reasoning in *DL-Lite*_A is a particularly simple form of structural subsumption, in part because there are no nested concepts. Therefore, one does not need

⁶ Ideally, this would be supported by experimental results.

⁷ The use of the word “*current*” is meant to emphasize the distinction from OWL “domain”, which describe the *potential* set of objects to which a property may apply.

⁸ We might consider prompting the user for an explicit name for the inverses that are needed before any particular explanation is begun.

any of the complications suggested in [10], such as atomic concepts, normalized concepts, etc.; the standard **ISA**-inference rules for reiteration (givens), reflexivity, and transitivity suffice. The rule for givens is shown in Section 2.2, as is the transitivity rule for concepts; we assume that the latter rule applies also to roles. The reflexivity rule is:

$$\textbf{Isa-refl} \frac{}{\mathcal{T} \vdash X \sqsubseteq X} \quad X \text{ a concept or a role}$$

We also need inference rules to relate the domains and ranges of a role and its inverse. In such rules (and ones further one), Q denotes a (basic) role, i.e., either an atomic role P or the inverse P^- of an atomic role. Moreover, we assume the syntactic simplification $(P^-)^- = P$.

$$\textbf{Dom-rng-inv} \frac{}{\mathcal{T} \vdash \text{dom}(Q) \sqsubseteq \text{rng}(Q^-)} \quad Q \text{ a role}$$

$$\textbf{Rng-dom-inv} \frac{}{\mathcal{T} \vdash \text{rng}(Q) \sqsubseteq \text{dom}(Q^-)} \quad Q \text{ a role}$$

Finally, the following inference rules take into account role subsumption when considering domains, ranges, and inverses:

$$\begin{array}{c} \textbf{Isa-dom} \frac{\mathcal{T} \vdash Q_1 \sqsubseteq Q_2}{\mathcal{T} \vdash \text{dom}(Q_1) \sqsubseteq \text{dom}(Q_2)} \quad \begin{array}{c} Q_1, Q_2 \\ \text{roles} \end{array} \quad \textbf{Isa-rng} \frac{\mathcal{T} \vdash Q_1 \sqsubseteq Q_2}{\mathcal{T} \vdash \text{rng}(Q_1) \sqsubseteq \text{rng}(Q_2)} \quad \begin{array}{c} Q_1, Q_2 \\ \text{roles} \end{array} \\ \textbf{IsaInv} \frac{\mathcal{T} \vdash Q_1 \sqsubseteq Q_2}{\mathcal{T} \vdash Q_1^- \sqsubseteq Q_2^-} \quad \begin{array}{c} Q_1, Q_2 \\ \text{roles} \end{array} \end{array}$$

Let us now introduce the $\perp$ symbol, denoting the empty set in all interpretations⁹. By the definition of $\sqsubseteq$, we therefore have the following additional inference rule:

$$\textbf{Nothing} \frac{}{\mathcal{T} \vdash \perp \sqsubseteq X} \quad X \text{ a concept or a role}$$

By definition, an *unsatisfiable/inconsistent* concept or role must be subsumed by $\perp$. In any DL, a concept or role can be shown to be unsatisfiable *indirectly*, by finding, via **Isa-trans**, a superconcept that itself is “directly” unsatisfiable. In *DL-Lite_A*, a concept will be said to be “*directly*” unsatisfiable due to subsumption by disjoint concepts, or due to being the current domain or range of an unsatisfiable role (which fall out of rules **Isa-dom** and **Isa-rng**, when $Q_2 = \perp$), or subsumption by another unsatisfiable concept. Similarly, a role can only be unsatisfiable due to subsumption by another unsatisfiable role, subsumption by disjoint roles, or due to its current domain or range being unsatisfiable. Hence, we need the following inference rules:

$$\begin{array}{c} \textbf{Inc-disj} \frac{\mathcal{T} \vdash X \sqsubseteq X_1 \quad \mathcal{T} \vdash X \sqsubseteq X_2}{\mathcal{T}, (\text{disjoint } X_1 \ X_2) \vdash X \sqsubseteq \perp} \quad X, X_1, X_2 \text{ concepts or roles} \\ \textbf{Inc-role-d} \frac{\mathcal{T} \vdash \text{dom}(P) \sqsubseteq \perp}{\mathcal{T} \vdash P \sqsubseteq \perp} \quad \begin{array}{c} P \text{ atomic} \\ \text{role} \end{array} \quad \textbf{Inc-role-r} \frac{\mathcal{T} \vdash \text{rng}(P) \sqsubseteq \perp}{\mathcal{T} \vdash P \sqsubseteq \perp} \quad \begin{array}{c} P \text{ atomic} \\ \text{role} \end{array} \end{array}$$

In the absence of unsatisfiability, all reasoning about $\sqsubseteq$ reduces to simple classification of atomic concepts and expressions denoting the current domains/ranges of roles,

⁹ This will serve as both the empty concept and the empty role, and we assume the convention that $\text{dom}(\perp) = \perp$, $\text{rng}(\perp) = \perp$, and $\perp^- = \perp$.

i.e., computing the so-called Hasse diagram $\mathcal{G}_{\mathcal{T}}$ induced by the $\sqsubseteq$ assertions in the TBox $\mathcal{T}$. Once this is done, explaining $B_1 \sqsubseteq B_2$ for satisfiable concepts involves only finding the shortest path between them.

The proof of unsatisfiability of a concept or role, though polynomially computable, can in fact be quite cumbersome because it can involve a chain of alternate demonstrations of role unsatisfiability and concept unsatisfiability, connected by the unsatisfiability of role domains and ranges. Moreover, in order to find *shortest proofs*, one needs to consider both indirect and direct ways of showing unsatisfiability. For lack of space, we consider here only the core of the algorithm searching for proofs of direct concept unsatisfiability w.r.t. a TBox $\mathcal{T}$ without unsatisfiable roles.

If a concept B is unsatisfiable w.r.t. $\mathcal{T}$, there must be $\sqsubseteq$ -paths from B to two concepts, say X, Y , asserted to be disjoint. For shortest proofs, we require that the sum of the lengths of these paths be minimal. To find this, an algorithm can start from B and explore in *breadth-first order* paths to $\sqsubseteq$ -ancestors X and Y until two disjoint such concepts are found (see Figure 1, where $\text{parents}(X, \mathcal{T})$ returns the set of concepts Y such that $X \sqsubseteq Y$ is in $\mathcal{T}$, while $\text{disjoint}(X, Y, \mathcal{T})$ is a predicate that returns true if $(\text{disjoint } X \ Y)$ is in $\mathcal{T}$). Supposing that the lengths of the paths to these concepts are n and m respectively, with $n \geq m$, then the length of this explanation is $n + m$.¹⁰ Unfortunately, this may not be the shortest explanation: if there exist disjoint concepts X' and Y' that are, respectively, j and k steps away from B , these yield an explanation of length $j + k$, which could be less than $n + m$ even if $j > n$, as long as $0 < k < n + m - j$. However, once we have detected the first pair X, Y at distance $n + m$, to detect the shortest explanation, we only have to search up to the limit when $j = n + m$. Adapting the algorithm in Figure 1 for this task is straightforward, as is keeping track of the paths leading from B to X and Y . Interestingly, the algorithm will also find indirect proofs of unsatisfiability, where the paths from B share an initial fragment π ; these have a shorter presentation, omitting one of the π .

3.3 ABox Reasoning

In *DL-Lite*_A, one infers new facts about existing individuals by applying inclusion axioms on concepts and roles, and recognizing that $P(a, b)$ entails $\text{dom}(P)(a)$ and $\text{rng}(P)(b)$. This is formalized by the following inference rules:

$$\begin{array}{l}
 \textbf{Subconcept} \quad \frac{\mathcal{T} \vdash B_1 \sqsubseteq B_2 \quad \langle \mathcal{T}, \mathcal{A} \rangle \vdash B_1(a)}{\langle \mathcal{T}, \mathcal{A} \rangle \vdash B_2(a)} \quad \begin{array}{l} B_1, B_2 \text{ concepts;} \\ a \text{ an individual} \end{array} \\
 \textbf{Subrole} \quad \frac{\mathcal{T} \vdash P_1 \sqsubseteq P_2 \quad \langle \mathcal{T}, \mathcal{A} \rangle \vdash P_1(a, b)}{\langle \mathcal{T}, \mathcal{A} \rangle \vdash P_2(a, b)} \quad \begin{array}{l} P_1, P_2 \text{ atomic roles;} \\ a, b \text{ individuals} \end{array} \\
 \textbf{Subrole-inv} \quad \frac{\mathcal{T} \vdash P_1 \sqsubseteq P_2^- \quad \langle \mathcal{T}, \mathcal{A} \rangle \vdash P_1(a, b)}{\langle \mathcal{T}, \mathcal{A} \rangle \vdash P_2(b, a)} \quad \begin{array}{l} P_1, P_2 \text{ atomic roles;} \\ a, b \text{ individuals} \end{array} \\
 \textbf{Dom-intro} \quad \frac{\langle \mathcal{T}, \mathcal{A} \rangle \vdash P(a, b)}{\langle \mathcal{T}, \mathcal{A} \rangle \vdash \text{dom}(P)(a)} \quad \begin{array}{l} P \text{ an atomic role;} \\ a, b \text{ individuals} \end{array}
 \end{array}$$

¹⁰ For simplicity, in the algorithm we assume that the set of concept inclusion assertions contains no cycle. Such cycles would make all involved concepts equivalent. Either they are detected a priori, or we would need to add to the algorithm a loop-checking condition.

$$\text{Rng-intro} \frac{\langle \mathcal{T}, \mathcal{A} \rangle \vdash P(a, b)}{\langle \mathcal{T}, \mathcal{A} \rangle \vdash \text{rng}(P)(b)} \begin{array}{l} P \text{ an atomic role;} \\ a, b \text{ individuals} \end{array}$$

To detect unsatisfiability in a KB, one simply looks for objects belonging to concepts which can be deduced to be subsumed by $\perp$, or for objects violating functionality constraints. The one nontrivial aspect is when we prefer shorter explanations. In the case of unsatisfiable ABoxes, one wants *the shortest derivation of a conflict* from the original ABox – one with fewest rule applications. To find this, one can use a strategy similar to the one described above for finding the shortest proof of unsatisfiability, assuming that the graph also has instance as well as subclass edges.

We note that while the above looks for evidence of knowledge base unsatisfiability, this is not the same problem as diagnosing errors in the knowledge base. Pinpointing [13], and related orthogonal techniques are much more likely to be useful for this task.

3.4 Reasoning in Finite Models

As mentioned, $DL\text{-}Lite_{\mathcal{A}}$ does not enjoy the finite model property, and hence inferences that hold specifically in finite models require to be explained.

Example 3. Consider the following TBox

$$\begin{array}{ll} (\text{func } tutors) & Student \sqsubseteq \text{rng}(tutors) \\ \text{dom}(tutors) \sqsubseteq TA & TA \sqsubseteq Student \end{array}$$

Since, *tutors* is a function, there can be at most as many values in its range as in its domain. Since the current range of *tutors* contains all *Students*, there can be at most as many *Students* as values in the domain of *tutors*. And since $\text{dom}(tutors)$ is contained in the set of *TAs*, there can be at most as many *Students* as *TAs*. If, in addition, now one has that $TA \sqsubseteq Student$, this implies that there are at most as many *TAs* as *Students*, and therefore the number of *TAs* and *Students* is the same. In an infinite model, this leads to no new conclusions, even if one recalls that *TA* is a subset of *Student*. However, in a finite model, these two facts imply that the extensions of *TA* and *Student* must be identical, which means that a new subsumption has been inferred: $Student \sqsubseteq TA$. ■

Clearly, the above pattern can be generalized by replacing *tutors* with the composition of an arbitrary set of roles $Q_1 \circ Q_2 \circ \dots \circ Q_k$, obtaining rule **Same-cardinality**:

$$\frac{\begin{array}{l} \mathcal{T} \vdash (\text{func } Q_1 \circ \dots \circ Q_k) \quad \mathcal{T} \vdash B_2 \sqsubseteq \text{rng}(Q_1 \circ \dots \circ Q_k) \\ \mathcal{T} \vdash \text{dom}(Q_1 \circ \dots \circ Q_k) \sqsubseteq B_1 \quad \mathcal{T} \vdash B_1 \sqsubseteq B_2 \end{array}}{\mathcal{T} \vdash B_2 \sqsubseteq B_1} \begin{array}{l} B_1, B_2 \text{ concepts;} \\ Q_1, \dots, Q_k \text{ basic roles} \end{array}$$

The remaining question is how one can deduce properties of a composition of roles, given only $DL\text{-}Lite_{\mathcal{A}}$ axioms. First, the following rule captures that if all roles are functions, then their composition will be a function:

$$\text{Func-comp} \frac{}{\mathcal{T}, (\text{func } Q_1), \dots, (\text{func } Q_k) \vdash (\text{func } Q_1 \circ \dots \circ Q_k)} \begin{array}{l} Q_1, \dots, Q_k \\ \text{basic roles} \end{array}$$

And since the current domain of a composition is contained in the current domain of the first role, we also have:

$$\text{Dom-comp} \frac{\mathcal{T} \vdash \text{dom}(Q_1) \sqsubseteq B_1}{\mathcal{T} \vdash \text{dom}(Q_1 \circ \dots \circ Q_k) \sqsubseteq B_1}$$

However, $B_2 \sqsubseteq \text{rng}(Q_1 \circ \dots \circ Q_k)$ does not follow from $B_2 \sqsubseteq \text{rng}(Q_k)$ alone, because the *current* range of the composition may be smaller, if not all values in $\text{dom}(Q_k)$ are reached by $Q_1 \circ \dots \circ Q_{k-1}$. So one also needs the entire current domain of Q_k to be contained in the current range of $Q_1 \circ \dots \circ Q_{k-1}$, leading to the rule:

$$\text{Rng-comp} \frac{\begin{array}{l} \mathcal{T} \vdash \text{dom}(Q_k) \sqsubseteq \text{rng}(Q_1 \circ \dots \circ Q_{k-1}) \\ \mathcal{T} \vdash B_2 \sqsubseteq \text{rng}(Q_k) \end{array}}{\mathcal{T} \vdash B_2 \sqsubseteq \text{rng}(Q_1 \circ \dots \circ Q_k)} \quad k \geq 2$$

It follows from results in [24] that these are *all* the possible additional subsumption inferences needed for the finite model case.

As far as explanations are concerned, this is a prime example where the user will need separate explanations for the rules of inference themselves.

4 Explaining Answers to Conjunctive Queries over a *DL-Lite* ABox

Consider first the simpler issue of answering conjunctive queries over a regular database. To explain why $Q(b)$ is true in a database requires showing why the database, treated as an interpretation, makes the body of the query evaluate to true. For conjunctive queries, this means exhibiting the values used for the existentially quantified variables in the body of the query. For example, if MIMI is an answer to query

$$Q_0(x) \leftarrow \text{Student}(x), \text{supervisedBy}(x, y), \text{teaches}(y, z)$$

one would need to locate some “witness” values ANNA and ENG101 for variables y and z , and then explain that $\text{Student}(\text{MIMI})$, $\text{supervisedBy}(\text{MIMI}, \text{ANNA})$ and $\text{teaches}(\text{ANNA}, \text{ENG101})$ are atoms present in the database. In general, it is possible that a value/tuple¹¹ appears in the answer for multiple reasons. In the above example, there may be alternate bindings of y and z , which together with $x = \text{MIMI}$, satisfy the query. In these case the user needs to be given the option of seeing an enumeration of the different explanations. The principle of minimality would not seem to enter into the choice of explanations here because all explanations are identical in form.

In the case of *DL-Lite_A*, the difficulty is that the ABox is not a closed database, but instead must be “completed” according to the axioms. For example, if we have $\text{Student}(\text{MIMI})$ and $\text{Student} \sqsubseteq \text{Person}$, then we must also add $\text{Person}(\text{MIMI})$; and if we have $\text{Professor}(\text{GINA})$ and $\text{Professor} \sqsubseteq \text{dom}(\text{teaches})$, then one can conclude

¹¹ In this example, and the rest of this paper, we deal only with queries that return a single value. This is only a presentation strategy – the theory and implementation we present applies equally to queries that have multiple variables in the head.

that there is some (hypothetical) individual, say $@c$, representing what GINA teaches, and that $teaches(GINA, @c)$ holds. Such hypothetical individuals may also get additional properties of their own. Unfortunately, the result can be an infinite database since the axioms may contain cyclic dependencies; e.g., $Person \sqsubseteq \text{rng}(\text{hasParents})$, $\text{rng}(\text{hasParents}) \sqsubseteq Person$.

From theoretical results [6], we know that from the TBox $\mathcal{T}$ and the ABox $\mathcal{A}$ one can derive a (generally infinite) canonical model $\text{can}(\mathcal{T}, \mathcal{A})$ ¹², by introducing “hypothetical individuals”, whose existence is posited by axioms, as illustrated in the example above. A crucial property of $\text{can}(\mathcal{T}, \mathcal{A})$ is that each conjunctive query $Q(x)$ can be answered by evaluating it, as in regular databases, over only a finite “small” portion of $\text{can}(\mathcal{T}, \mathcal{A})$, whose size depends on $Q(x)$. We exploit this fact to explain why $Q(b)$ holds, by essentially constructing the *finite part* of $\text{can}(\mathcal{T}, \mathcal{A})$ that is needed to justify the truth of the query body for $Q(b)$.

In order to generate the relevant part of $\text{can}(\mathcal{T}, \mathcal{A})$, we resort to the algorithm for query answering in $DL\text{-}Lite_{\mathcal{A}}$. Query answering in $DL\text{-}Lite_{\mathcal{A}}$ [6] is performed by first rewriting the original query $Q(x)$ into a set $\mathcal{S} = \{Q_0(x), \dots, Q_n(x)\}$ of alternate queries, then evaluating these over the original ABox (treated as a closed database), and finally returning the union of the results. Each query in $\mathcal{S}$ expresses *necessary* conditions on values x to satisfy the original query $Q(x)$, and the entire set $\mathcal{S}$ has the property that the answer to the original query $Q(x)$ is the union of the answers to the queries in $\mathcal{S}$ when executed over the ABox.

Example 4. Consider the following TBox $\mathcal{T}$

$$PhD \sqsubseteq Student \quad (5) \quad \text{rng}(\text{supervisedBy}) \sqsubseteq Professor \quad (7)$$

$$PhD \sqsubseteq \text{dom}(\text{supervisedBy}) \quad (6) \quad Professor \sqsubseteq \text{dom}(\text{teaches}) \quad (8)$$

and the query $Q_0(x) \leftarrow Student(x), supervisedBy(x, y), teaches(y, z)$. The $DL\text{-}Lite_{\mathcal{A}}$ rewriting algorithm would rewrite Q_0 into the following set of queries:

$$\begin{aligned} Q_0(x) &\leftarrow Student(x), supervisedBy(x, y), teaches(y, z) \\ Q_1(x) &\leftarrow PhD(x), supervisedBy(x, y), teaches(y, z) \\ Q_2(x) &\leftarrow PhD(x), supervisedBy(x, y), Professor(y) \\ Q_3(x) &\leftarrow PhD(x), supervisedBy(x, y), supervisedBy(w, y) \\ Q_4(x) &\leftarrow PhD(x), supervisedBy(x, y) \\ Q_5(x) &\leftarrow PhD(x), PhD(x) \\ Q_6(x) &\leftarrow PhD(x) \end{aligned}$$

We recall briefly, using the above query as an example, the basic steps of the rewriting algorithm that are necessary to understand its use in our explanation setting; for the full details we refer to [6]. Essentially, the algorithm makes use of replacement and unification steps. A replacement can be applied to an atom when the corresponding predicate appears on the right hand side of an inclusion axiom; e.g., query $Q_1(x)$ is obtained from $Q_0(x)$ by replacing $Student(x)$ with $PhD(x)$, due to axiom (5). Similarly, $Q_2(x)$ is obtained from $Q_1(x)$ by making use of axiom (8), which can be seen

¹² The canonical model corresponds to what in databases is called *chase* of a database w.r.t. a set of constraints [6].

in FOL as $\forall v. \exists w. \text{Professor}(v) \leftarrow \text{teaches}(v, w)$ ¹³. On the other hand, a unification step collapses two atoms with the same predicate when the corresponding arguments can be unified; e.g., query $Q_4(x)$ is obtained from $Q_3(x)$ by unifying the two atoms $\text{supervisedBy}(x, y)$ and $\text{supervisedBy}(w, y)$. Unifications are essential in order to enable replacements where a variable is required to occur only once, as per the previous technical footnote.

Note that the algorithm need not produce a linearly ordered set of rewritings: if the TBox had an additional axiom $MSc \sqsubseteq \text{Student}$, there would also be four rewritings $Q_{1b}, \dots, Q_{4b}$ paralleling $Q_1, \dots, Q_4$, with MSc replacing PhD . ■

The key observation is that the replacement rewriting steps correspond to the “inverses” of the additions made to the canonical model of the knowledge base. Hence, we can use them to guide the explanation of why a certain individual is in the answer to the original query. Suppose that an individual b is part of the answer to $Q(x)$, and we want to explain why. In our example, suppose the ABox contains $PhD(\text{BOB})$, and no other facts about BOB, and we want to explain why BOB is in the answer to $Q_0(x)$. To do so, we proceed as follows.

Step 1. Since $Q(b)$ was true, we select from $\mathcal{S}$ the rewritings that produce b as an answer when directly evaluated over the ABox (viewed as a closed database)¹⁴. Let $Q_k(x)$ be one such rewriting. We (re)compute the derivation of $Q_k(x)$ from $Q(x)$, building a data structure that tracks how (changed) atoms in one query are derived from the predecessor query. This is easy for substitutions, if we replace each atom in the query by a stack/list whose top is the most recently rewritten form of the original atom. For unifications, we put a pointer on the stack from one of the unified atoms to the stack of the other atom.

In our example, BOB would be returned by both $Q_5(x)$ and $Q_6(x)$, so the derivation of $Q_5(x)$ and $Q_6(x)$ from $Q(x)$ is reconstructed. The derivation of Q_5 would yield roughly the following data structure:

```
[ [ PhD(x), axiom1, Student(x) ],
  [ PhD(x), axiom2, supervisedBy(x,y) ],
  [ pointer(atom(2)), unify(w,x), supervisedBy(w,y),
    axiom3, Professor(y), axiom4, teaches(y,z) ] ]
```

Step 2. Since $Q_k(b)$ is true, there is an assignment θ of ABox individuals to the variables in the head and body of $Q_k(x)$ such that $\theta(x) = b$, and for each atom β in $Q_k(x)$, $\theta(\beta)$ is an ABox fact.

In our example, the ABox contains $PhD(\text{BOB})$, and for $Q_5(x)$ (or $Q_6(x)$), we would start with $\theta(x) = \text{BOB}$.

Step 3. Traversing backwards the sequence of rewritings from $Q_k(x)$ to $Q(x)$, we extend the substitution θ to the variables in the intervening queries, by keeping track of unifications, or assigning to such variables newly introduced *Skolem constants* (corresponding to objects introduced by the inclusion assertions). More precisely, when going

¹³ Technical note: a replacement where a variable is removed from the resulting query (w , in the example) is allowed only when such a variable does not occur anywhere else in the query.

¹⁴ By completeness of the query answering algorithm [6], at least one such rewriting $Q_k(x)$ will always exist.

$$\begin{aligned}
& supervisedBy(BOB, @1) \quad (*) \\
& \quad \{ \text{by meaning of domain, from} \\
& \quad \quad dom(supervisedBy)(BOB) \\
& \quad \quad \{ \text{by } \mathbf{Subconcept} \text{ rule from} \\
& \quad \quad \quad PhD \sqsubseteq dom(supervisedBy) \quad \{ \text{Axiom 2} \} \\
& \quad \quad \quad PhD(BOB) \quad \{ \text{DB fact} \} \} \}
\end{aligned}$$

Fig. 3. Domain rule expansion

backwards from Q_i to Q_{i-1} , from which Q_i was generated, if no variable has been eliminated when rewriting Q_{i-1} to Q_i , then θ need not be extended. When a variable z has been eliminated because it has been unified with a variable y , then we set $\theta(z) = \theta(y)$. While when a variable z has been eliminated in Q_i by replacing an atom $R(y, z)$ (resp., $R(z, y)$) with $A(y)$, due to inclusion axiom $A \sqsubseteq dom(R)$ (resp., $A \sqsubseteq rng(R)$), then we set $\theta(z) = @c$, where $@c$ is a fresh constant representing a new hypothetical individual. In this way, when one reaches the original query $Q(x)$, θ will have assigned to each variable appearing in it either an ABox individual or a Skolem constant. In analogy to the case of standard databases, one then initially shows to the user $\theta(Q(x))$, i.e., the set of atoms of the original query to which θ has been applied.

In our example, we would get $\theta(x) = BOB$ for $Q_5(x)$, and the following new assignments: $\theta(y) = @1$ for $Q_4(x)$, $\theta(w) = \theta(x) = BOB$ for $Q_3(x)$, and $\theta(z) = @2$ for $Q_1(x)$. The resulting initial explanation shown to the user would be the sequence of ground atoms matching the query conjuncts:

$$Q_0(BOB) \leftarrow Student(BOB), supervisedBy(BOB, @1), teaches(@1, @2)$$

Step 4. The data structure constructed in Step 1, together with the substitutions gathered in Step 1 unifications, as well as Steps 2 and 3, yields a complete proof tree of the atoms in the original query (see Figure 2). Note that the leftmost part of this proof tree (its first level) corresponds to the initial explanation we suggested in Step 3. In an interactive session, users can control the iterative expansion of this proof tree to lower depths in whatever order they desire. It is interesting to note that pointers in the data structure become Lemmas – previously justified atoms, thereby providing a much shorter global proof. So unification in the rewriting algorithm has an interesting benefit for explanation.

We observe that an explanation always exists, and that the above algorithm will *always* provide one. Note also that one can manipulate here the proof tree, to make proof steps more understandable to humans. For example, in this case one can expand the rules dealing with domain and range. This could lead to the modified proof fragment depicted in Figure 3.

Several issues need to be addressed at this point. First is the selection of the rewriting(s) $Q_k(x)$ from $\mathcal{S}$ in Step 1. For this, we need an efficient way of finding only the indexes j of those rewritings that actually make $Q(b)$ be true. This can be done by associating to each of the queries $Q_i(x)$ in $\mathcal{S}$ a distinct tag, and returning such tags as part of the answer. Originally, if $\mathcal{S} = \{Q_1(x), \dots, Q_n(x)\}$, then the rewriting would normally have been written in SQL as:

```
SELECT x FROM ... WHERE Q1(x) UNION
SELECT x FROM ... WHERE Q2(x) UNION ...
```

We would instead use the query

```
SELECT x, 1 AS tag FROM ... WHERE Q1(x) UNION
SELECT x, 2 AS tag FROM ... WHERE Q2(x) UNION ...
```

Secondly, in order to execute Step 3, for each of the tag values j actually occurring in the answer, we need to get the ABox tuples that would contribute to making the body of query $Q_j(b)$ be true. In our example, if the ABox was $\{PhD(BOB), Student(MIMI), supervisedBy(BOB, ALICE)\}$, then tags 4, 5 and 6 would all be returned for answer $x = BOB$ of $Q(x)$, and we would obtain the set of atoms $\{PhD(BOB), supervisedBy(BOB, ALICE)\} \cup \{PhD(BOB)\} \cup \{PhD(BOB)\}$.

Note that in case more than one rewriting returns the answer b , and the rewritings are not reducible one to the other by unification, then we have alternate explanations for why $Q(b)$ holds. In this case, it seems that explanations involving fewer rewritings are preferable since they involve less abstract reasoning. In our example, the explanation based on Q_4 , shown in Figure 4, seems clearly preferable to the previous explanation for $Q_5(BOB)$ because it is shorter (involving fewer rewritings of the original query). All things being equal, we also believe an explanation would be preferred if it introduces fewer hypothetical (Skolem) individuals.

Moreover, Q_5 is preferable to Q_6 because both atoms would be supported by grounding to the same atomic value, thereby saving a unification step in the explanation, which would have resulted in a pointer/Lemma call in the explanation.

Therefore, to find the better rewritings first, we need a search strategy that tests whether a rewriting is satisfied in the database of atoms selected above, before applying any other replacement or unification. And one which uses replacements that introduce Skolem constants as late as possible.

We have implemented the above-described algorithm for generating explanations in a prototype Prolog program, which exploits Prolog's unification technique (for unifying conjunctive query atoms, testing applicability of axiom replacements, and for finding database substitutions), and its backtracking control structure (through an invertible

```
Student(BOB)
{ by Subconcept rule from
  PhD  $\sqsubseteq$  Student { Axiom 1 }
  PhD(BOB) { DB fact } }
supervisedBy(BOB, ALICE)
teaches(ALICE, @1)
{ by Subconcept rule from
  Professor  $\sqsubseteq$  dom(teaches) { Axiom 4 }
  Professor(@1)
  { by Subconcept rule from
    rng(supervisedBy)  $\sqsubseteq$  Professor { Axiom 3 }
    supervisedBy(BOB, ALICE) { DB fact } } }
```

Fig. 4. Alternate (shorter) explanation for $Q(BOB)$

append predicate) to search through all possible sequences of rewritings of length less than $n = \max\{\text{tag value returned by the query}\}$.

5 Conclusions

The paper tackles the problem of explaining *DL-Lite* reasoning, viewed in part as (i) requiring inference rules for building proofs, and (ii) finding short proofs. (The use of inference rules makes possible, among others, a connection to generic explanation software available on the Semantic Web [17].) In general, an alternate, more accessible syntax is introduced for *DL-Lite*, and an algorithm for finding shortest proofs of inconsistency is presented. Of greater novelty and complexity is the *explanation* of reasoning in finite models, which is particularly relevant for conceptual models of databases.

Since *DL-Lite* is intended to support efficient conjunctive query evaluation over a DL KB, we address for the first time explanation for this. In particular, we provide a theoretically sound technique and a prototype implementation for finding explanations of the fact that some value b is returned as an answer to query $Q(x)$ over knowledge base $(\mathcal{T}, \mathcal{A})$. These explanations are minimal length in the sense that fewest transformations steps are applied from the original query. The performance system used for query-answering is used to retrieve a minimal set of tuples needed to explain the truth of $Q(b)$, thus making the explanation component scalable even for very large ABoxes.

Future work includes a component for finding *all* different explanations for some conclusion (which is useful for the case when the conclusion is no longer desired), and especially explaining why a value e was *NOT* returned by a conjunctive query.

Acknowledgments. This research has been partially supported by the EU FP6 FET project TONES (Thinking ONtologiES) under contract number FP6-7603, by the Italian PRIN 2006 project NGS (New Generation Search), funded by MIUR, and by the U.S. DHS under ONR grant N00014-7-1-0150.

References

1. Borgida, A., Calvanese, D., Rodriguez-Muro, M.: Explanation in DL-Lite. In: Proc. of the 2008 Description Logic Workshop (DL 2008). CEUR Electronic Workshop Proceedings (2008), <http://ceur-ws.org/>
2. Elmasri, R., Wiederhold, G.: GORDAS: A formal high-level query language for the entity-relationship model. In: Proc. of the 2nd Int.Conf.on the Entity-Relationship Approach (ER 1981), pp. 49–72 (1981)
3. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P.F. (eds.): The Description Logic Handbook: Theory, Implementation and Applications. Cambridge University Press, Cambridge (2003)
4. Calvanese, D., Lenzerini, M., Nardi, D.: Unifying class-based representation formalisms. J. of Artificial Intelligence Research 11, 199–240 (1999)
5. Berardi, D., Calvanese, D., De Giacomo, G.: Reasoning on UML class diagrams. Artificial Intelligence 168(1–2), 70–118 (2005)
6. Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Rosati, R.: Tractable reasoning and efficient query answering in description logics: The DL-Lite family. J. of Automated Reasoning 39(3), 385–429 (2007)

7. Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Rosati, R.: Tailoring OWL for data intensive ontologies. In: Proc. of the Workshop on OWL: Experiences and Directions (OWLED 2005). CEUR Electronic Workshop Proceedings, vol. 188 (2005), <http://ceur-ws.org/Vol-188/>
8. Shmueli, O., Tsur, S.: Logical diagnosis of LDL programs. In: Proc. of the 7th Int. Conf. on Logic Programming (ICLP 1990), pp. 112–129 (1990)
9. Arora, T., Ramakrishnan, R., Roth, W.G., Seshadri, P., Srivastava, D.: Explaining program execution in deductive systems. In: Ceri, S., Tsur, S., Tanaka, K. (eds.) DOOD 1993. LNCS, vol. 760, pp. 101–119. Springer, Heidelberg (1993)
10. McGuinness, D.L., Borgida, A.: Explaining subsumption in description logics. In: Proc. of the 14th Int. Joint Conf. on Artificial Intelligence (IJCAI 1995), pp. 816–821 (1995)
11. Deng, X., Haarslev, V., Shiri, N.: A framework for explaining reasoning in description logics. In: Working Notes of the AAAI Fall Symposium on Explanation-aware Computing, pp. 189–204 (2005)
12. Schlobach, S.: Explaining subsumption by optimal interpolation. In: Alferes, J.J., Leite, J.A. (eds.) JELIA 2004. LNCS (LNAI), vol. 3229, pp. 413–425. Springer, Heidelberg (2004)
13. Schlobach, S.: Debugging and semantic clarification by pinpointing. In: Gómez-Pérez, A., Euzenat, J. (eds.) ESWC 2005. LNCS, vol. 3532, pp. 226–240. Springer, Heidelberg (2005)
14. Schlobach, S., Cornet, R.: Explanation of terminological reasoning: A preliminary report. In: Proc. of the 2003 Description Logic Workshop (DL 2003) (2003)
15. Godfrey, P., Minker, J., Novik, L.: An architecture for a cooperative database system. In: Risch, T., Litwin, W. (eds.) ADB 1994. LNCS, vol. 819, pp. 3–24. Springer, Heidelberg (1994)
16. Godfrey, P.: Minimization in cooperative response to failing database queries. Int. J. of Cooperative Information Systems 6, 95–149 (1997)
17. McGuinness, D.L., Pinheiro da Silva, P.: Explaining answers from the Semantic Web: the Inference Web approach. J. of Web Semantics 1(4), 397–413 (2004)
18. Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Rosati, R.: Data complexity of query answering in description logics. In: Proc. of the 10th Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR 2006), pp. 260–270 (2006)
19. Poggi, A., Lembo, D., Calvanese, D., De Giacomo, G., Lenzerini, M., Rosati, R.: Linking data to ontologies. J. on Data Semantics X, 133–173 (2008)
20. Pinheiro da Silva, P., McGuinness, D.L., Fikes, R.: A proof markup language for semantic web services. Information Systems 31(4–5), 381–395 (2006)
21. Borgida, A.: From type systems to knowledge representation: Natural semantics specifications for description logics. J. of Intelligent and Cooperative Information Systems 1(1), 93–126 (1992)
22. Borgida, A., Franconi, E., Horrocks, I., McGuinness, D.L., Patel-Schneider, P.F.: Explaining *ALC* subsumption. In: Proc. of the 1999 Description Logic Workshop (DL 1999). CEUR Electronic Workshop Proceedings, vol. 22 (1999), <http://ceur-ws.org/Vol-22/>
23. Fiedler, A.: Natural Language Proof Explanation. In: Hutter, D., Stephan, W. (eds.) Mechanizing Mathematical Reasoning. LNCS, vol. 2605, pp. 342–363. Springer, Heidelberg (2005)
24. Rosati, R.: Finite model reasoning in DL-Lite. In: Bechhofer, S., Hauswirth, M., Hoffmann, J., Koubarakis, M. (eds.) ESWC 2008. LNCS, vol. 5021. Springer, Heidelberg (2008)

Using Ontologies for an Intelligent Patient Modelling, Adaptation and Management System

Matt-Mouley Bouamrane^{1,2}, Alan Rector¹, and Martin Hurrell²

¹ School of Computer Science

Manchester University, UK

`{mBouamrane,Rector}@cs.man.ac.uk`

² CIS Informatics, Glasgow, UK

`martin.hurrell@informatics.co.uk`

Abstract. Health Information Management Systems (HIMS) face considerable technical and organisational barriers before successful deployment in hospitals. In addition, many existing systems have significant limitations, including: lack of flexibility and adaptability to complex requirements and processes and a general lack of “intelligence”. They offer basic patient management functionalities but do not go far beyond core functionalities. Due to their rigid architectures, these systems are hard to maintain and update. Recent advances in knowledge representation, including ontologies, can offer powerful and appealing solution to these problems. In this paper, we describe our current work on using ontologies for adapted information collection and patient representation. We describe the iterative transformation of a basic risk assessment software into a “knowledge-aware” system. We argue that using ontologies is both conceptually appealing and a pragmatic solution to implementing a shift from simple management systems to intelligent systems in healthcare. In turn, we believe such systems will efficiently support clinicians in their daily activities and will result in improved delivery of tailored patient care.

1 Introduction

The use of Information Management Systems in Healthcare (HIMS) offer many advantages including: reducing information and tasks duplication, reducing paper trails, reduction of administrative tasks, provision of centralised information leading to improved retrieval of patient medical information and records and reduction in waiting time. HIMS may also reduce the incidence of clinical adverse events, many of whom arise from insufficient information about a patient’s medical history. Computer-based screening systems have had measurable benefits in reducing omission and errors arising as the result of clinicians dealing with information of high complexity. As an example, physician computer order entry have proved useful in error prevention and preventive intervention through the use of structured entry, rule-based reminders and triggering of alerts relating to

allergies and adverse drug interaction [1, 2, 3]. Likewise, in his survey of patient-computer interview systems, Bachman highlights that face-to-face information collection with a clinician is often less complete than computer-based history taking [4].

Despite of their potential advantages, HIMS still face considerable challenges before successful deployment in hospitals. Potential issues include: the perceived lack of immediate return on investment from trust managers, resistance to process changes from staff and clinicians, the initial effort required to deploy a HIMS in a hospital which often leads to disruption of service due to the necessity for staff training and technical breakdowns, software incompatibility and the lack of enterprise-wide solutions, leading to the complexity and overheads involved in integrating various individual departmental software solutions. In addition to these technical and organisational barriers, many existing commercial systems also suffer from a significant number of limitations. We identify among them: lack of flexibility and adaptability to complex requirements and processes and a general lack of “intelligence”. Many existing commercial HIMS are “mechanical” systems, based on a combination of database systems and distributed technologies. They offer basic patient management functionalities but do not go far beyond these core functionalities. Due to their rigid architectures, these systems are hard to maintain and update.

In this paper, we propose to overcome some of the challenges commonly faced by patient management systems by transforming “mechanical” systems into “knowledge-aware” systems. Our solution involves adding a layer of ontologies on top of the functionalities commonly required from the management system. The benefit of the approach is two fold. First, the resulting system is more convenient to update as modifying the ontology layer can be done without the need for additional and costly software engineering work. The clean separation between system functionalities and the knowledge base used by the system means that the latter can be modified if the face of evolving knowledge or changing requirements. Secondly, the ontology layer enables the system to perform operations, such as decision support, which were cumbersome to implement when using database and distributed system technologies on their own. We illustrate our approach by describing the iterative steps performed to transform an existing preoperative assessment system into a “knowledge-aware” system. The paper presents two of a total of four expected iterations: (i) the implementation of an adaptive questionnaire and (ii) patient modelling system using ontologies. Future iterative steps will consist in developing a (iii) clinical rule ontology and (iv) an information relevance ontology. The paper is organised as follows: the next section describes a basic patient risk assessment software and its main limitations. We then describe the first two iterative transformation steps towards a “knowledge-aware” system. The first one consists of developing a context-sensitive patient questionnaire based on an ontology. The second one consists in generating a patient model ontology, which can be used as a service provider to a number of client interfaces. We conclude with a discussion of the benefits of the current approach and directions for future research.

2 A Basic Risk Assessment Software

2.1 System Description

Figure 1 presents a basic risk assessment software. This is an existing system for preoperative assessment of patients prior to elective (i.e. non-emergency) surgery currently used in the preoperative clinic of Utrecht Hospital, Netherlands. The user interface consists of a web-based form which connects to the system server through a standard browser (Fig. 2). The aim of the software is to gather patient medical history so an informed patient risk assessment can be performed by anaesthetists prior to surgery so potential complications can be anticipated and pre-emptive actions taken where necessary. The patient medical history essentially consists of: general health condition, history of past surgery, cardiovascular and respiratory history, medication and miscellaneous health conditions, including allergies. The patient answers a number of questions from a static questionnaire which contains around 50 questions. A limited number of questions are implemented using conditional branching through the use of IF-THEN constructs.

In an hospital setting, data input is typically performed by a nurse, who will read questions as they come up on the screen. In certain cases, patient who are judged to have the necessary abilities (physical, technical and cognitive) can fill in the questionnaire in a dedicated computer room, under the supervision of preoperative nurses. Patients can therefore request support and seek clarification whenever necessary. Other options currently under consideration include: providing the software to general practitioners in primary care as a decision support tool for potential referral to specialist care and using the software as a phone-based screening tool prior to potential admission to hospital.

In addition to collecting information about a patient's medical history, a preoperative nurse or junior doctor will usually carry a physical examination and add in the system additional information about the patient including: height, weight, nutrition, verbal response etc. A number of tests (e.g. blood test) can also be requested and the results are entered in the system. All the information collected concerning the patient is stored into a database (Fig. 1). A rule engine then uses a combination of best-practice and local rules on the patient data to derive a number of scores. These scores range from simple calculations (e.g. Body Mass Index¹) to more sophisticated algorithms to derive predictors of overall perioperative² outcomes such as the ASA (American Society of Anaesthesiologists) physical status classification³, cardiac scores (e.g. Goldman and Detsky cardiac risk index [5]), etc. Risk scores and predictors are then combined to produce an overall risk assessment score (step 4. in Fig. 1). The risk assessment essentially consists of: perioperative and postoperative cardiac, respiratory

¹ BMI = Weight / (Height)² in metric units.

² Period surrounding a patient's surgical procedure, typically including ward admission, anaesthesia, surgery and recovery.

³ Ranging from ASA I (healthy patient) to ASA V (moribund).

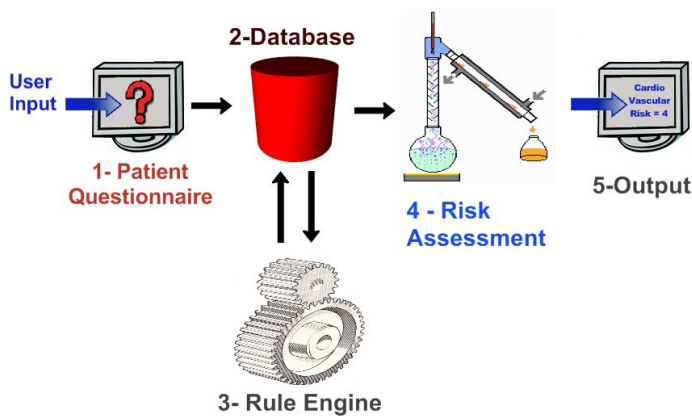


Fig. 1. A basic patient risk assessment software

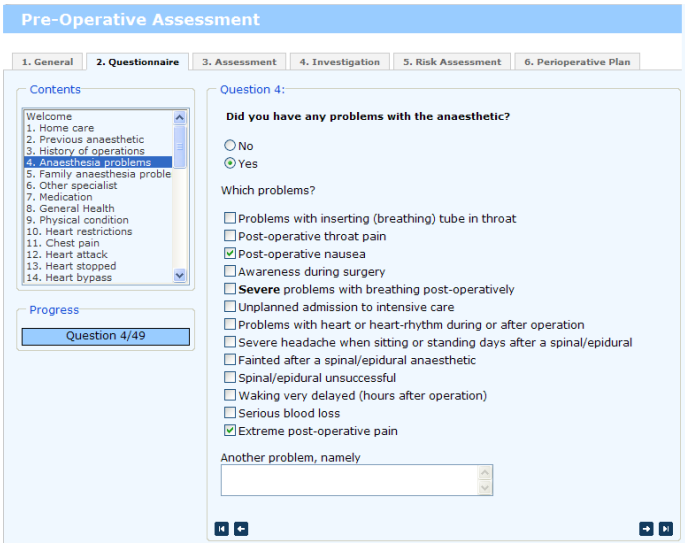


Fig. 2. Web-based preoperative questionnaire

and infection risks. Finally a preoperative form is produced in HTML and PDF formats, which can be printed out for archival purposes.

2.2 System Limitations

The system presented has been in use for over three years. It has resulted in significant improvements in work processes, including reduction of paper trail, standardised workflow and risk score calculation, centralised information leading to improved efficiency in retrieving and accessing patient records, reduced incidence

of unnecessary tests, etc. However, the system still has many shortcomings. We here discuss the system's main limitations:

Patient Information Gathering. The challenge is to provide brief general questionnaires that suite the majority of patients while, at the same time, capturing sufficient details about the minority of patients with special problems for which more information is critical. The system needs to make information for the majority quick and efficient without sacrificing completeness. This is obviously very difficult to achieve using static questionnaires. Conditional branching can, to some extent, be used to alleviate this problem. However, this method quickly becomes hard to manage in more complex cases. Another issue is that systems designed on branching are hard to maintain since dependencies are usually hard coded in the implementation. The sequence of potentially related questions can not easily be altered and additional questions can not be introduced without considerable engineering work.

System Maintenance. As highlighted by [6], a major challenge faced by HIMS are continuously evolving work processes and practices due to emerging guidelines, advances in healthcare and organisational changes. In a system such as the one previously described, patient data stored in the database have no longer any intrinsic meaning. The data can only be correctly used and interpreted via surrounding software components used to input data and extract data from the database. This means that even small structural changes to the system will often require significant software engineering work. Updating the system on clinical sites will generally cause delays and disruptions to the service.

Clinical Rule Management. There are in existence more clinical rules and guidelines than anyone could possibly manage. Also, many hospitals use their own local rules in addition to other rules and guidelines. The existing system currently uses the same rules and calculates the same scores regardless of patient profile (although it will obviously reach different outcomes given different circumstances). However, there is a strong clinical case for choosing to run different risk scores depending on the specifics of a patient or a particular procedure. An example consists in using different rules to calculate the cardiac risks of a cardiac patient (i.e. undergoing cardiac surgery) and a non-cardiac patient.

Display of Critical Information. A thorough documentation of a patient's medical history is widely recognised as providing good indicators of potential intra-operative and postoperative complications. However, for a risk assessment to be effective, the clinician must not be overloaded with information. As an example, a BMI value may be sufficient for the clinician to form a judgement about the safety of a procedure without the need for him to know the specific height and weight details of a patient, while the details of previous surgery may only be useful if their are relevant to a planned procedure, etc. The challenge here is to prominently display critical information while reducing or perhaps even hide less relevant information.

3 An Ontology-Driven Adaptive Questionnaire

A solution to the challenge of making the information collection process quick for the majority and efficient without sacrificing completeness, is to develop an adaptive questionnaire. By “*adaptive*” we mean a dynamic modification of the behaviour of the application (i.e. structure of the questionnaire) in response to user interaction (context-sensitive self-adaptation) [7]. Previous method used to implement context sensitive adaptation in medical questionnaire include, conditional branching, using tree models and finite state machines [8,9,10]. Limitations of these proposed methods include complexity, scalability and lack of flexibility for system maintenance. Our proposed solution to context-sensitive adaptation is to use an ontology as the basis for adaptation of information collection. The proposed method permits to iteratively capture finer-grained information with each successive step, should this information be relevant according to a questionnaire ontology. The proposed method intends to replicate the investigating behaviour exhibited by clinicians when presented with items of information which may be cause for concern or require further attention. While the system has the potential to reduce the number of questions and thus save time and costs for healthy patients, the emphasis is rather on collecting *more* information *whenever* relevant so a proper informed patient risk assessment can be performed. We argue that this method is robust, scalable and highly configurable and although the method is presented in a medical context, the principles are generic. The system

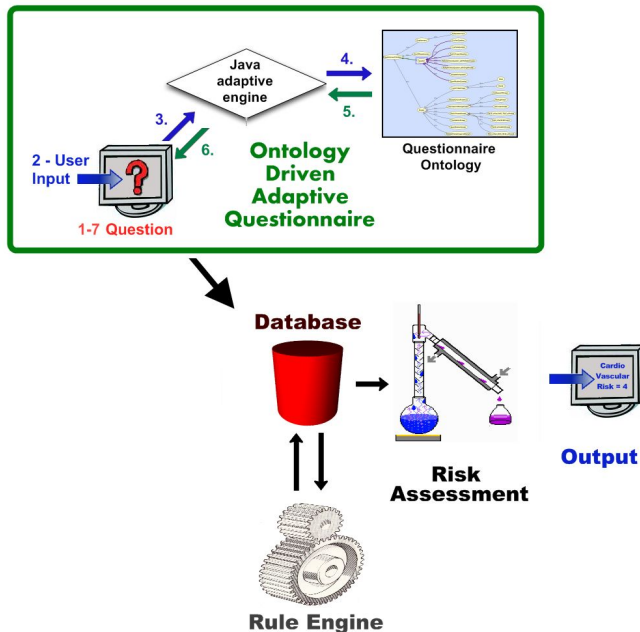


Fig. 3. First iterative step towards a knowledge-aware system: an ontology-driven adaptive questionnaire

implementation is illustrated in Fig. 3 while technical details have been described in elsewhere [11,12] Note that the main difference with Fig. 1 is that an ontology is now responsible for managing user interaction. This is thus the first iterative step in transforming the previous “mechanical” risk assessment system into a “knowledge-aware system”.

4 Patient Medical History Modelling

There is currently extensive work on developing information models, electronic patient health records and terminologies and ontologies in the medical domain [13, 14, 15, 16, 17]. Specific applications of ontologies include modeling medical errors [18], clinical examinations in oral medicine [19], etc. In our system, the information collected by the adaptive questionnaire could be directly input in a database, as is the case in the original system described in Fig. 1. However, as previously mentioned, this means that the medical data in isolation have lost all intrinsic meaning. We refer to this information representation as the “*Data level*” representation, with the associated lack of flexibility in the structure of the risk assessment system previously highlighted in section 2.2. In the new system implementation, the information collection based on an ontology creates the opportunity to simultaneously generate a patient profile automatically generated from the medical ontology and thus to preserve the semantics of the information collected. This information representation is what we describe as the “*Semantic level*” and constitutes the second iterative step in transforming the patient system into a knowledge-aware system.

The main benefit of this approach is that a single information repository, a semantic patient medical profile, can now provide a number of services to various clients: to input data in the database, as input to a rule engine, a clinical document or a patient record, as illustrated in Fig. 4. This system design provides greater flexibility to the current implementation as new software components can be added and older one withdrawn without affecting the whole structure of the system. If the type of information collected about the patient remains static, changes to the clients of the patient profile ontology are restricted within the interface layer of Fig. 4. If the type of information collected about the patient is updated, changes will occur both in the OWLPatientModeler and the interface layer. However, the latter changes are typically incremental (e.g. a new item of information about the patient is now required) and therefore updates to the system ought to be manageable.

4.1 Patient Medical History Ontology

We here want to stress to the reader that the patient semantic medical profile generated by our system is *not* in any case a *patient medical record*. It is instead a formal representation of the information collected during the preoperative questionnaire. The main difference here is essentially one of scale. While attempting to model *any* potential item of medical information for any patient is extremely

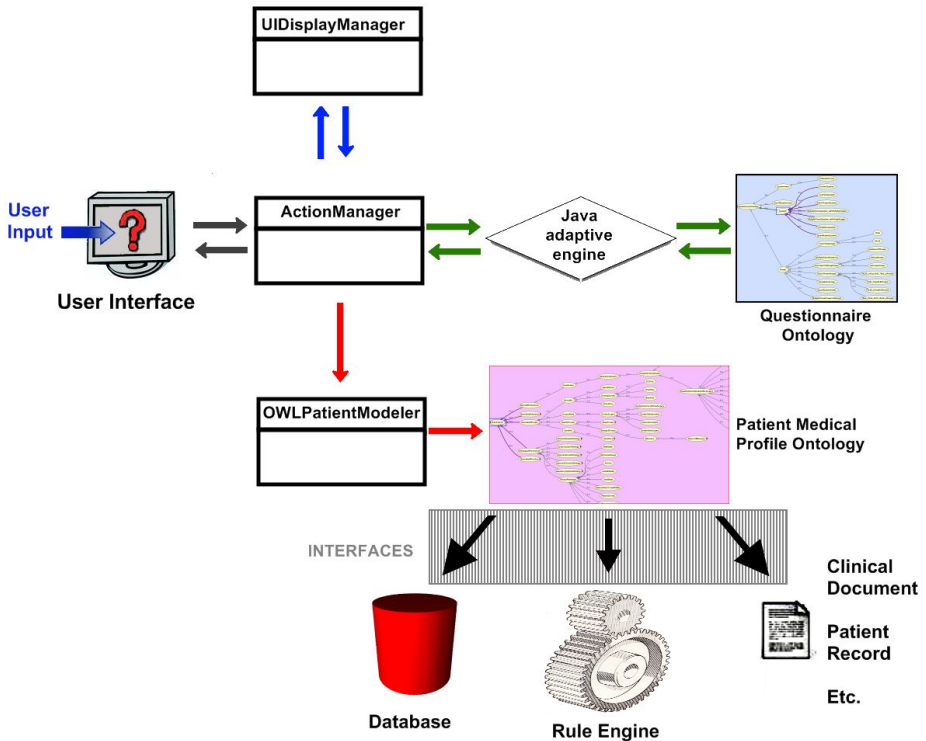


Fig. 4. Second iterative step towards a knowledge-aware system: a patient profile ontology now provides a number of services to various clients

challenging, doing so in a very constrained domain is somehow more manageable, as we will shortly demonstrate using a practical example. The preoperative questionnaire currently in use in the system is composed of between 30 to 90 questions (the variation is due to the adaptive behaviour of the questionnaire). Thus, the scope of the information which needs to be modelled is well defined and thus constrained and manageable. Figure 5 illustrates a patient profile generated by the system. We here describe in more details the type of medical information modelled by the patient medical profile ontology using the example of a specific patient.

Medical Condition. Many items of information of relevance to clinicians consist of Boolean-type information regarding a patient's medical history. More precisely: the absence or presence of specific conditions. Example include: “has the patient got diabetes?”, “is the patient epileptic?”, etc. For this type of information, modeling is done through the use of the *hasPresence* and *hasAbsence* functional properties. The syntax of information is $\{hasAbsence \text{ some } MedicalCondition\}$ as illustrated by items labelled 1 in Fig. 5. Remember that this only models information obtained through the preoperative questionnaire. The purpose of this information is to flag down to the

Label

Class Description: PatientProfile

6

hasTemporalUnit some Year

and hasAge value 29

hasAbsence some (FamilyMember

that isAffectedBy some AnaestheticComplication)

7

hasAbsence some (InvestigationAct

that consistsOf some (ChestXRay

that hasTemporalRange some (Month

that hasValue some int(<= 61)))

7

hasAbsence some (InvestigationAct

that consistsOf some (ECGTest

that hasTemporalRange some (Month

that hasValue some int(<= 61)))

2

hasAbsence some (PastHealthEvent

that consistsOf some AnaestheticComplication)

2

hasAbsence some (PastHealthEvent

that consistsOf some BloodClot)

2

hasAbsence some (PastHealthEvent

that consistsOf some HeartAttack)

2

hasAbsence some (PastHealthEvent

that consistsOf some Stroke)

3

hasAbsence some (PastSurgery

that hasLocation some Heart)

1

hasAbsence some Diabetes

1

hasAbsence some Epilepsy

hasFutureExpectation some HomeCare

hasHealthFeature some (DrinkingHabit

that hasTemporalStatus some TRUE_atPresent

and hasQuantityRange some (QuantityRangeFeature

that hasValue some int(<= 3)

and hasQuantityUnit some AlcoholUnit

and hasTemporalUnit some Day))

8

hasHealthFeature some (SmokingHabit

that hasTemporalStatus some FALSE_atPresent TRUEInThePast)

hasHealthStatus some FitAndAble

5

hasMedication some (AntiInflammatoryDrug

that hasTemporalStatus some TRUE_atPresent)

5

hasMedication some (AsthmaDrug

that hasTemporalStatus some FALSE_atPresent TRUEInThePast)

4

hasPresence some (Asthma

that hasSeverityFeature some Mild)

7

hasPresence some (InvestigationAct

that consistsOf some (BloodTest

that hasTemporalRange some (MonthRange

that hasValue some int(<= 61)))

3

hasPresence some (PastSurgery

that hasSpecificLocation some Appendix)

hasPresence some (PastSurgery

Fig. 5. OWL Patient Profile as viewed through the Protégé-OWL User Interface

clinicians the potential existence of certain medical conditions. In case of a positive response, this is likely to be followed up by further investigation into the condition (e.g. “*what type of diabetes?*”, “*does the patient take medication?*”). The exact nature of further investigations will usually depends on local hospital policies. Hence, the advantage of a flexible information collection system as described in section 3. In keeping with the open world assumption of OWL⁴, a medical condition is only assumed to be absent if a question was specifically asked to the patient (e.g. the answer to the question “*do you have diabetes?*” was explicitly stated as “*No*”).

Specific Medical Event. In some cases, critical information for the clinician is whether a patient has had an occurrence in the past of a specific medical event, such as a heart attack, stroke, etc. This information is modelled in the ontology through the use of the special classes *PastHealthEvent* and *PastSurgery*. For specific event (items 2), information syntax is in the form: *{hasPresence\Absence some (PastHealthEvent that consistsOf some SpecificEvent)}*. For specific surgery (items 3), information syntax is in the form: *{hasPresence\Absence some (PastSurgery that hasLocation some SpecificAna-tomicLocation)}*. In the example of Fig. 5, the ontology tells us that the patient has never had any of: anaesthetic complications, blood clot, heart attack, stroke, or heart surgery but she did have appendix surgery.

Qualitative Information. Qualitative information can be expressed as shown by example 4: the patient has asthma with mild symptoms. What actually corresponds to mild symptoms can be asserted in the questionnaire ontology, thus giving the flexibility to see those criteria being adapted to specific sites.

Temporal Information. In many cases, it is important to know whether some information about the patient is currently true or if it was true in the past even if it is no longer true. Information which fall under this remit include smoking, drug taking, medication, etc. In order to express these nuances, we use the following classes: *TRUE-atPresent* (e.g. “*smoker*”), *TRUE-atPresent-And-TRUE-InThePast*, *TRUE-atPresent-And-FALSE-InThePast*, *FALSE-atPresent-And-TRUE-InThePast* (e.g. *patient is an “ex-smoker”*), and finally: *FALSE-at-Present-And-FALSE-InThePast* (e.g. “*never smoked*”). This is illustrated by the items labelled 5: the patient is an ex-smoker, she uses to take medication for asthma but no longer does and is currently taking some anti-inflammatory medication. More specific information, such as the exact date of an event can also be specified if necessary.

Cardinal Information. It is possible to express cardinal information (e.g. numbers and ranges) in an OWL ontology using cardinal restrictions. In this case it is important to specify what is the nature on the units used so the information in the patient ontology remains self-explanatory. Therefore, one needs to define unit classes. These are essentially of two types: temporal unit and quantity units. Thus, item 6 says that this patient’s age is 29 and the unit to interpret this value is *Year*. Another advantages in defining unit classes is that

⁴ Information which is not asserted in the ontology can not be inferred to be false.

this information is stored in a single location and can easily be updated to suit new guidelines or local trust rules.

Range Information. Items 7 tell us that the patient did not have a chest X-ray or ECG (electrocardiogram) within the last 6 months but she did have a blood test.

Combining more Complex Information. This last example shows how to combine the previous syntaxes to express more complex information: items 8, tells us that the patient currently drinks alcohol and that her alcohol intake is less than 3 units of alcohol per day. Once again, what exactly constitutes an alcohol unit can be asserted in the ontology, providing a convenient method for updating the system.

4.2 System Description

In order to achieve system flexibility and maintenance requirements highlighted in section 2, the system illustrated in Fig. 4 is implemented along independent self-contained software components. The interaction loop is as follows: the user answers the current medical question currently being asked, the ActionManager dispatches this information to the adaptive engine which consults the questionnaire ontology to extract the next question which is returned to the ActionManager. The ActionManager hands out the next question (an OWL class) to the UIDisplayManager whose purpose is to interpret the information in the question class and to render it appropriately on the user interface (e.g. depending whether it is a multiple choice questions, whether the user is allowed one or several answers, etc.) Simultaneously, the action manager hands out the current question and corresponding answer classes to the OWLPatientModeler, whose purpose is to produce the patient semantic medical profile described in the previous section. The system is implemented using Java technology, the UIDisplayManager, the OWLPatientModeler and the adaptive engine are implemented using the OWL 1.1 API [20].

5 Discussion and Future Work

We have presented 2 of 4 iterative steps towards transforming a “mechanical” patient risk assessment system into a “knowledge-aware system”. Using an ontology for context-sensitive adaptation means that the information collection process can be tailored to patients’ individual circumstances and thus enables finer-grained information collection. In the second step, we have argued that using a patient ontology to model the information collected offer several advantages. The first is that the semantics of the information collected are preserved and self-contained in the ontology and thus remains interpretable regardless of surrounding technology and software implementation. Then, the patient semantic medical profile can be used to provide services to a number of software clients. This design has significant implication for system flexibility, maintenance and update. For compatibility with other HMS, we are currently into the process of

mapping the concepts in the patient ontology to the IOTA terminology (International Organization for Terminologies in Anaesthesia). The IOTA terminology is itself mapped to the SNOMED-CT terminology⁵. In order to complete the transformation of the system into a fully-fledged knowledge-aware system, 2 iterative steps remain: to develop an ontology of clinical rules in order to infer which are the most relevant given specific patients' circumstances or a particular procedure. Although this may a-priori sound a daunting task, it is quite manageable in practice. Any one hospital will typically use a very limited number of clinical rules and risk scores for preoperative risk assessment (e.g. between 10 to 30 cardiac, respiratory, nutrition risk scores, etc.) This means that the task of developing the clinical rule ontology is manageable while the ontology can be iteratively updated to meet new requirements. The final step will be to develop an information relevance ontology, to ensure that the most critical clinical information is prominently displayed given patients' specific circumstances and surgical procedures, while minimising less relevant information.

References

1. Bates, D.W., Leape, L.L., Cullen, D.J., Laird, N., Petersen, L.A., Teich, J.M., Burdick, E., Hickey, M., Kleeefield, S., Shea, B., Vliet, M.V., Seger, D.L.: Effect of computerized physician order entry and a team intervention on prevention of serious medication errors. *Journal of the American Medical Association* 280, 1311–1316 (1998)
2. Kuhn, K., Giuse, D.: From hospital information systems to health information systems problems, challenges, perspectives. *Methods of Information in Medicine* 40, 275–287 (2001)
3. Kuperman, G.J., Gibson, R.F.: Computer physician order entry: Benefits, costs, and issues. *Annals of Internal Medicine* 139, 31–39 (2003)
4. Bachman, J.W.: The patient-computer interview: a neglected tool that can aid the clinician. *Mayo Clinic Proceedings* 78, 67–78 (2003)
5. Palda, V.A., Detsky, A.S.: Perioperative assessment and management of risk from coronary artery disease. *Annals of internal medicine* 127(4), 313–328 (1997)
6. Lenz, R., Kuhn, K.A.: Towards a continuous evolution and adaptation of information systems in healthcare. *International Journal of Medical Informatics* 73(1), 75–89 (2004)
7. Jameson, A.: Adaptive interfaces and agents. In: *Human-computer interaction handbook: fundamentals, evolving technologies and emerging applications*, pp. 305–330. Lawrence Erlbaum Associates, Inc., Mahwah (2003)
8. Houziaux, M.O., Lefebvre, P.J.: Historical and methodological aspects of computer-assisted medical history-taking. *Informatics for Health and Social Care* 11(2), 129–143 (1986)
9. van Ginneken, A.M., de Wilde, M., Blok, C.: Generic computer-based questionnaires: an extension to opensde. In: Fieschi, M., Coiera, E., Li, Y.C. (eds.) *Proceedings of 11th World Congress on Medical Informatics, MEDINFO*, pp. 688–692. IOS Press, Amsterdam (2004)

⁵ http://www.nlm.nih.gov/research/umls/Snomed/snomed_main.html

10. Vahabzadeh, M., Epstein, D., Mezghanni, M., Lin, J.L., Preston, K.: An electronic diary software for ecological momentary assessment (ema) in clinical trials. In: Proceedings of 17th IEEE Symposium on Computer-Based Medical Systems, CBMS 2004, Bethesda, US, pp. 167–172. IEEE Computer Society, Los Alamitos (2004)
11. Bouamrane, M.M., Rector, A., Hurrell, M.: Ontology-Driven Adaptive Medical Information Collection System. In: An, A., Matwin, S., Raś, Z.W., Ślezak, D. (eds.) Foundations of Intelligent Systems. LNCS (LNAI), vol. 4994, pp. 574–584. Springer, Heidelberg (2008)
12. Bouamrane, M.M., Rector, A., Hurrell, M.: Gathering precise patient medical history with an ontology-driven adaptive questionnaire. In: Proceedings of Computer-Based Medical Systems, CBMS 2008, Jyväskylä, Finland, pp. 539–541. IEEE Computer Society, Los Alamitos (2008)
13. Iakovidis, I.: Towards personal health record: current situation, obstacles and trends in implementation of electronic healthcare record in europe. *International Journal of Medical Informatics* 52, 105–115 (1998)
14. Dieng-Kuntz, R., Miniera, D., Ruzicka, M., Corby, F.C.O., Alamarguya, L.: Building and using a medical ontology for knowledge management and cooperative work in a health care network. *Computers in Biology and Medicine* 36(7-8), 871–892 (2006)
15. Rector, A., Rogers, J.: Ontological and practical issues in using a Description Logic to represent medical concept systems: experience from GALEN. In: Barahona, P., Bry, F., Franconi, E., Henze, N., Sattler, U. (eds.) Reasoning Web 2006. LNCS, vol. 4126, pp. 197–231. Springer, Heidelberg (2006)
16. Yu, A.: Methods in biomedical ontology. *Journal of Biomedical Informatics* 39(3), 252–266 (2006)
17. Hu, B., Dasmahapatra, S., Dupplawa, D., Lewisa, P., Shadbolta, N.: Reflections on a medical ontology. *International Journal of Human-Computer Studies* 65(7), 569–582 (2007)
18. Stetson, P.D., McKnight, L.K., Bakken, S., Curran, C., Kubose, T.T., Cimino, J.J.: Development of an ontology to model medical errors, information needs, and the clinical communication space. *Journal of American Medical Informatics Association* 9(6 suppl. 1), s86–s91 (2002)
19. Gustafsson, M., Falkman, G.: Experiences in modeling clinical examinations in oral medicine using owl. In: Proceedings of the OWLED 2007 Workshop on OWL: Experiences and Directions, Innsbruck, Austria (2007)
20. Horridge, M., Bechhofer, S., Noppens, O.: Igniting the owl 1.1 touch paper: The owl api. In: Proceedings of the third International Workshop of OWL Experiences and Directions, OWLED 2007, Innsbruck, Austria (2007)

Context-Addressable Messaging Service with Ontology-Driven Addresses*

Jaroslav Domaszewicz, Michal Koziuk, and Radoslaw Olgierd Schoeneich

Institute of Telecommunications, Warsaw University of Technology
ul. Nowowiejska 15/19, 00-665 Warsaw, Poland
{domaszew,mkoziuk,rschoeneich}@tele.pw.edu.pl

Abstract. The context-addressable messaging service allows applications to send messages to mobile users described by their context. The context of these users is described in terms of ontology assertions, and an ontology-driven expression is used as a context-based address. This expression can be interpreted as a definition of a new ontology class. The recipients of a context-addressed message are all the users (nodes) whose context makes them instances of the address class. This paper presents a model for an ontology-based context-addressable messaging system, and provides performance results of its implementation based on available 'off-the-shelf' software.

1 Introduction

In mobile context-aware systems, a piece of context data can often be associated with a particular node in the network. Such node-specific context data are used to describe the node's location, its environment, the role of its user, etc. Special use of this node-specific context data can be made to provide support for a Context-Addressable Messaging (CAM) service. This service allows sending messages whose recipients are specified using context data.

A simple use case for Context-Addressable Messaging is presented in Fig.1¹. Imagine that on the scene of a major emergency, an injured person requires medical assistance. Such a person could inform nearby medical staff about his situation by sending a message with a request for help, with an appropriate context-based description of desired recipients (e.g. "All unoccupied medical staff which are within 1km from me"). The description is what we call a Context-based Address. A Context-based Address is used to describe the context of nodes to which a message is to be delivered. The Context-Addressable Messaging service would deliver the request for help to all users of the system meeting the specified criteria, in particular the two unoccupied nurses within 1km from the sender. Medical staff in the area who are currently occupied would not receive the message.

* This work was supported by the 6FP MIDAS IST project, contract no. 027055.

¹ Cliparts taken from Open Clip Art Library at <http://openclipart.org/>

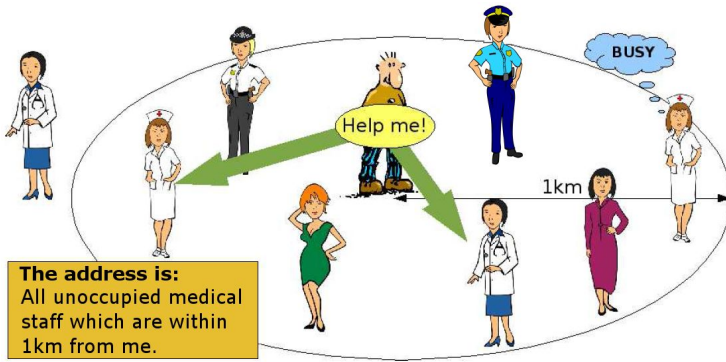


Fig. 1. Illustration of Context-Addressable Messaging

Medical staff further than 1 km from the injured person would not receive it either.

The CAM service, as envisioned in this paper, is: (a) unreliable (the service is best effort), (b) connectionless, (c) datagram-oriented (each message send operation causes a network packet to be sent), (d) group-oriented (a Context-based Address describes a group of message recipients), and (e) one-way (responses, if needed, can be handled by a regular, unicast communications service).

We have identified three major issues important for Context-Addressable Messaging. First, the system requires a context modeling mechanism. Second, a language for constructing Context-based Addresses is needed. Third, a dedicated routing protocol (not addressed in this paper) is required to efficiently route Context-Addressed Messages to their destinations.

The key contribution of this paper is the idea of using ontology-driven, runtime-formed class definitions as Context-based Addresses. A Context-based Address is constructed from terms taken from a context modeling ontology, using operators offered by the concept description language of a selected ontology language. Each address can be viewed as a definition of a new concept (class), which does not exist in the original ontology. The intended recipients of a Context-Addressed Message are all the nodes whose context makes them instances of the address class. Another contribution is the architecture for an ontology-driven Context-Addressable Messaging service, constructed as a middleware layer. The middleware is a domain-neutral framework customized for a specific domain by an exchangeable context modeling ontology. Finally, this paper presents an evaluation of a proof-of-concept implementation of the middleware, including performance results and conclusions for future research.

This paper is organized as follows. Section 2 presents existing solutions similar to the CAM service. Section 3 explains the proposed structure of Context-based Addresses. Section 4 presents the architecture of a middleware which provides the CAM service. Section 5 contains results of the performed experiments. Conclusions and directions for future work are presented in section 6.

2 Related Work

Addressing network nodes by their attributes has already been proposed in the literature. In particular, discarding node identifiers has been of particular interest to the wireless sensor networks research community, where focus is placed on creating a data-centric network. Solutions such as Directed Diffusion [1] or the EYES project [2] propose to use attributes of nodes (coupled with values of those attributes) in order to describe destination nodes.

A similar context-based messaging architecture for MANET networks, called FlavourCast [3], has been proposed. There, destination nodes can be described by an arbitrary attribute such as color.

Another area where nodes are not interested in directly addressing each other is Content-Based Addressing [4] [5]. In this solution communicating entities specify the content of the data they are producing and send out appropriate notifications. Nodes which would like to receive data, subscribe to it by creating appropriate filters. The system selects relevant messages, by matching the attributes and values from notifications with constraints contained in filters.

The main difference between all the mentioned work and our research is that (a) we propose to incorporate a domain model in the form of an ontology and (b) we use ontology-driven, runtime defined classes as addresses. All the above solutions rely on functions performed on attributes, their values, and arbitrary other parameters. There are however, a number of systems which aim at enhancing the publish/subscribe scheme with ontologies. S-ToPSS (Semantic Toronto Publish/Subscribe System) [6] proposes to introduce semantics into publish/subscribe matching algorithms, by using a taxonomy of concepts from a domain specific ontology (as one of three possible extensions).

A similar concept (which uses ontologies to match subscriptions and published data) is presented in [7] and [8], where the Siena subscription language is extended by ontologies and ontological operators. The proposed extension relies on the usage of ontological equivalence and subsumption to perform matching between filters and subscriptions.

Solutions described above are implemented according to the publish-subscribe scheme. The proposed Context-Addressable Messaging service differs from them in that the destination nodes do not need to subscribe to any events. Instead each node receives messages that match its current context (which can change dynamically).

3 Ontology-Driven Context-Based Addresses

A basic assumption behind Context-Addressable Messaging is that each node in the network has access to a common context (domain) model. The context model is an ontology which describes the current domain of the CAM service. Thus, there exists a collection of predefined classes, relations and individuals specific to the domain. Every piece of information entered into the system has to be structured in terms of the ontology. A simple example is presented in Fig.2.

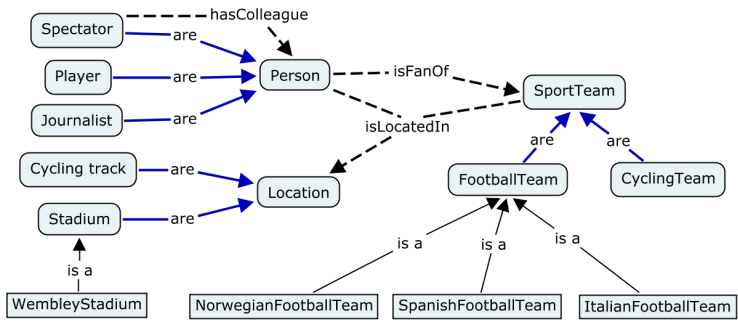


Fig. 2. A simple ontology

Constructor	DL Syntax	Example
intersectionOf	$C_1 \sqcap \dots \sqcap C_n$	Human $\sqcap$ Male
unionOf	$C_1 \sqcup \dots \sqcup C_n$	Doctor $\sqcup$ Lawyer
complementOf	$\neg C$	$\neg$ Male
oneOf	$\{x_1 \dots x_n\}$	{john,mary}
toClass	$\forall P.C$	$\forall$ hasChild.Doctor
hasClass	$\exists r.C$	$\exists$ hasChild.Lawyer
hasValue	$\exists r.\{x\}$	$\exists$ CitizenOf.{USA}
minCardinalityQ	$(\geq nr.C)$	$(\geq 2$ hasChild.Lawyer)
maxCardinalityQ	$(\leq nr.C)$	$(\leq 1$ hasChild.Male)
inverseOf	r^-	hasChild $^-$

Fig. 3. An example of a concept description language (taken from [9])

$$Spectator \sqcap (\exists isLocatedIn.\{ WembleyStadium\}) \sqcap (\exists isFanOf.\{ SpanishFootballTeam\})$$

Fig. 4. An example of a Context-based Address (in a Description Logic notation)

In our approach to CAM, Context-based Addresses are domain ontology-driven classes defined with a concept description language (a part of a selected ontology language). In other words, Context-based Addresses are definitions of new classes, formed from existing concepts (classes, relations, individuals) by means of concept constructors (operators). The structure of these definitions follows the formalism of the chosen ontology language. An example of concept constructors from a concept description language (taken from [9]) is presented in Fig.3.

Consider the example ontology of a sports domain, presented in Fig.2. A sample Context-based Address for this ontology is presented in Fig.4. This address denotes all spectators located at the Wembley Stadium who are fans of the

Address:	Payload:
Spectator and (isLocatedIn value WembleyStadium) and (isFanOf value SpanishFootballTeam)	(ANY DATA)

Fig. 5. An example of a Context-Addressed Message

Spanish football team. It is constructed by combining one class, two relations, and two individuals, all belonging to the sports domain ontology. Two concept-description language constructors (`intersectionOf` and `hasValue`) are used in the definition. Note that such an address class does not exist in the ontology; it is produced at runtime by an application or a user.

A binary representation of a Context-based Address is placed in the header of a Context-Addressed Message (see Fig.5). The message’s payload can be any data. Such message is sent into the network and is delivered to all nodes whose context matches the description contained in the attached address (i.e., whose context makes them instances of the address class). Note that the Manchester OWL Syntax [10] notation used in Fig.5 is equivalent to the Description Logic notation from Fig.4.

4 CAM Middleware Architecture

The CAM middleware architecture, for a single mobile node, is given in Fig.6. As can be seen, the domain model (i.e., an ontology) is not hardwired into the middleware but is imported into it. To use the middleware in a different environment (i.e., for a different domain), it is enough to exchange the domain model. Hence, the CAM middleware can be described as a domain-neutral framework customized by a domain model. All nodes in the network have to use the same domain model.

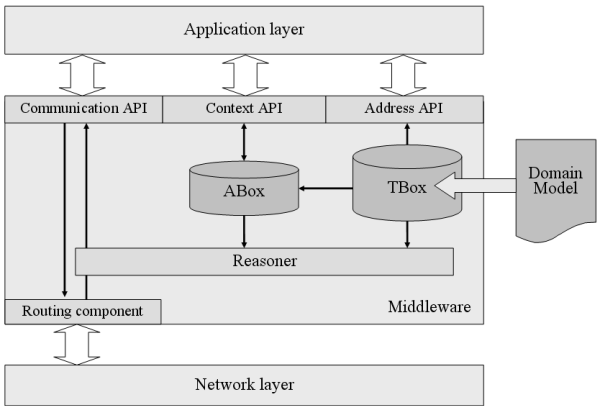


Fig. 6. CAM middleware architecture (one mobile node shown)

The CAM middleware consists of four main building blocks: the internal domain model representation (TBox), the node's context data (ABox), the reasoner, and the routing component. The TBox holds a runtime representation of all the information provided in the domain model ontology (such as, for example, the class hierarchy). The TBox, once produced from the imported domain model, remains unchanged at runtime, except for temporary insertion of Context-based Addresses (explained below). The context data (ABox) are statements entered into the middleware by applications and expressed in terms of the domain model (such as a statement that some instance belongs to a certain class). The ABox changes at runtime in response to changing context of the node. Also, while the TBox is the same at all nodes in the network (all the nodes share the domain model), the ABox is node-specific. A node's ABox contains context data collected and injected into the middleware by this node's applications (it is the applications' responsibility to acquire context from the environment and inject it into the middleware). In general, the context data stored in the ABox vary from node to node. The middleware reasoner performs address resolving, i.e., using the ABox and the TBox, it provides an answer to the question: "does this node's context match a certain Context-based Address". Finally the routing component routes Context-Addressed Messages. The goal of the routing component is to avoid flooding the network with Context-Addressed Messages. As a result, a Context-Addressed Message is delivered to only a subset of nodes, so that address resolving need not be performed by all nodes in the network².

Among the context data items stored in a node's ABox, we distinguish a special ABox individual which we refer to as **thisNode**. Facts about the context of the node are expressed as: (a) object properties linking **thisNode** with other individuals, (b) datatype properties assigning certain values to **thisNode** or (c) statements that **thisNode** belongs to certain classes. Of course, once a relation exists between **thisNode** and other individuals, the facts related to those individuals also become a part of the node's context. An important feature of this way of context representation is that each node holds a self-centric view of the context, centered around the **thisNode** individual.

The key part of the CAM middleware is the address resolving, i.e., checking if the node's context (i.e., all data in the node's ABox directly or indirectly related to the **thisNode** individual) matches the Context-based Address of a newly received Context-Addressed Message. The address resolving is done at each prospective message recipient. This is carried out as a three step process (illustrated in Fig.7). First, the definition of a class forming the Context-based Address is inserted to the TBox. Next, the reasoner is requested to determine (taking into account all relationships captured by the domain ontology) if the **thisNode** individual belongs to this new class. As the domain model has been changed, the reasoner has to perform a classification process, on a structure which includes the new class. Finally, after a response to the query has been produced, the address class is removed from the TBox. This is required to restore

² The routing component as envisioned by the authors, is described in [11].

Address: Spectator and (isLocatedIn value WembleyStadium) and (isFanOf value SpanishFootballTeam)
Step 1: - Add the class CL = Spectator and (isLocatedIn value WembleyStadium) and (isFanOf value SpanishFootballTeam) to the TBox
Step 2: - Check if thisNode individual belongs to the class CL
Step 3: - Remove the class CL from the TBox

Fig. 7. The steps of address resolving

a) Defining the context of a node. //entering context data ClientContextAPI thisNode = new ClientContextAPI(); thisNode.addBelongsToClass("Spectator"); thisNode.addUniqueProperty("isLocatedIn", "WembleyStadium"); thisNode.addProperty("isFanOf", "SpanishFootballTeam"); //retrieving context data String myTeam = thisNode.getProperty("isFanOf");
b) Creating Context-based Addresses. ClientAddressAPI a = new ClientAddressAPI(); int Spectator = a.getClass("Spectator"); int SpectatorAtWembley = a.addRestriction(Spectator, "isLocatedIn", "WembleyStadium"); int SpanishFanAtWembley = a.addRestriction(SpectatorAtWembley, "isFanOf", "SpanishFootballTeam");
c) Sending and receiving Context-Addressed Messages. //sending a message ClientCommAPI c = new ClientCommAPI(); c.send(a.getProperAddress(SpanishFanAtWembley), "The game of the Spanish team will start in 10 minutes !!!"); //receiving a message c.buffer.wait(); processMessage(c.buffer.data);

Fig. 8. Using the CAM middleware

the TBox of the ontology to its original state. Depending on the query response, the message is either passed to the application layer or discarded.

Fig.8 illustrates how the described middleware can be used by applications. Fig.8a shows an example of how the context of a node can be entered into the middleware using the Context API (retrieval and removal of context data is also possible). Fig.8b presents an example of how different Context-based Addresses can be constructed using the Address API. Fig.8c illustrates the Communication API, showing how a message can be sent to a constructed address, and how a message can be received by nodes whose context matches the message's address.

5 Experimental Results

The CAM middleware architecture has been implemented by using mainstream technologies and off-the-shelf software components presented in Fig.9. As the research on a dedicated routing protocol was still work in progress, we used flooding based on broadcasting for delivery of messages (i.e., all nodes in the network were prospective recipients).

Issue	Solution
Ontology language	OWL-DL
Tbox and ABox	JENA [12]
Reasoner	Pellet [13]
Routing	None (flooding)

Fig. 9. CAM middleware implementation summary

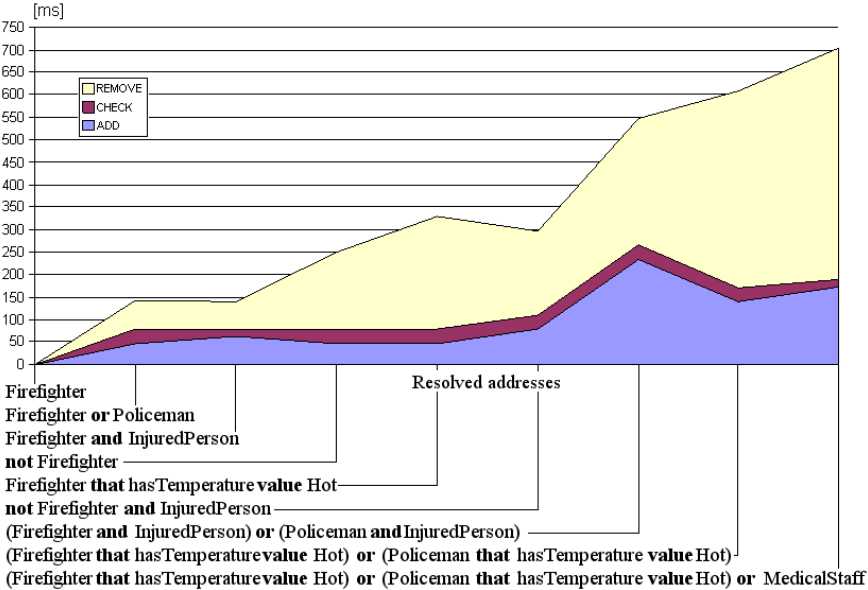


Fig. 10. Address resolution time at the receiver for a small emergency ontology

Performance testing of the CAM middleware was performed on a PC with a 2,66GHz Celeron processor and 1GB RAM. During the tests we used two ontologies. The first one was a very small ontology for emergency situations, developed internally for test purposes. The second ontology was the publicly available, much larger Pizza [14] ontology.

To evaluate the performance of the CAM middleware, we measured the times of different stages of address resolving (due to reasoning, it is by far the most time consuming operation in the middleware). The test procedure was the following: (a) the context of each node was set-up and did not change later on, (b) a Context-based Address was created on one node, (c) the message with this address was sent to the other nodes, and (d) one of the destination nodes measured the time required to resolve the address. As address resolving is a three phase process, we measured the time required to perform each phase.

Fig.10 presents the results obtained for the emergency ontology. The first resolved address is a class present in the TBox (**Firefighter**). In this case, both the TBox and the ABox do not change and resolving an address is instantaneous.

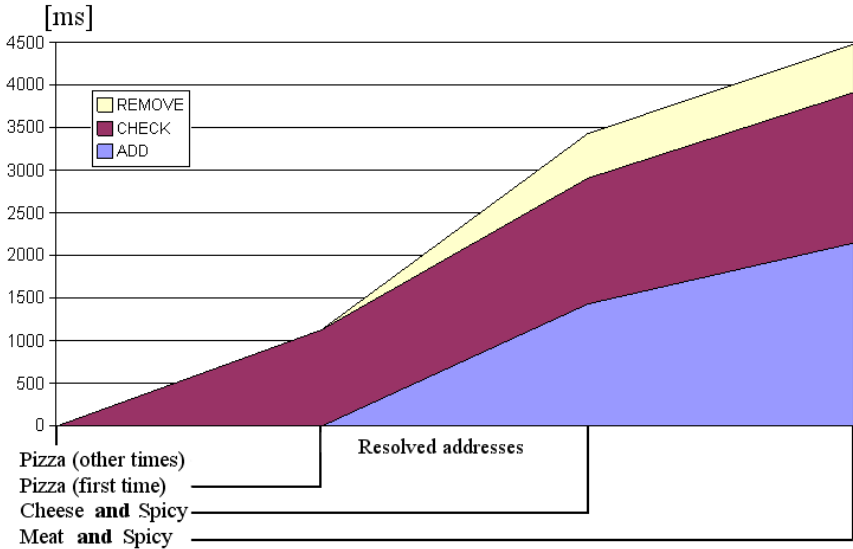


Fig. 11. Address resolution time at the receiver for the pizza ontology

More complicated addresses consist of an intersection or a union of two classes (e.g., *Firefighter* or *Policeman*), and we observe that the time required for resolving an address increases. The steps which were not required previously (adding a new class, and removing it later) are now performed and constitute the major part of the address resolving process. In general, the more complicated an address, the longer the address resolving process. Resolving the most complicated address (out of the tested ones) requires around 700ms. Looking at results for the significantly bigger *Pizza* ontology (Fig.11), we observe that the address resolving time has dramatically increased.

Based on the presented results we draw a number of conclusions. First, not surprisingly, the more complicated the address, the longer it takes to resolve it. Second, a significant amount of time required for address resolving is consumed when adding the new address class to the TBox, and later removing it. Finally, the time required to resolve an address grows with the size of the ontology; this has a severe impact on performance of the CAM middleware for large ontologies.

6 Conclusions and Future Work

The performance results of the CAM middleware show that, even for a PC, it is hard to obtain near real-time operation with mainstream, off-the-shelf software technologies. This is even more true for mobile devices. Clearly, the Pellet reasoner has not been optimized for real-time reasoning on a variable TBox

and ABox. We assume that this conclusion holds for other existing OWL-DL reasoners.

We intend to address the above performance problems by (a) choosing an ontology language less complex than OWL-DL (DL-Lite [15], along with its conjunctive query mechanism, seems to be a good candidate) and (b) implementing a dedicated reasoner that can handle the chosen ontology language on a mobile device. Another important part of the work on the CAM architecture is the Context-Based Routing [11] protocol; such a protocol is essential to avoid flooding the network with Context-Addressed Messages and having to do address resolving at every node.

References

1. Intanagonwiwat, C., Govindan, R., Estrin, D., Heidemann, J., Silva, F.: Directed diffusion for wireless sensor networking. *IEEE/ACM Trans. Netw.* 11(1), 2–16 (2003)
2. Handzinski, V., Koepke, A., Frank, Ch., Karl, H., Wolisz, A.: Semantic addressing for wireless sensor networks. Technical report, Telecommunication Networks Group, Technische Universität Berlin (May 2004)
3. Cutting, D., Corbett, D.J., Quigley, A.: Context-based Messaging for Ad Hoc Networks (May 8–13, 2005)
4. Carzaniga, A., Rosenblum, D., Wolf, A.: Content-based addressing and routing: A general model and its application (2000)
5. Carzaniga, A., Wolf, A.L.: Forwarding in a content-based network. In: *SIGCOMM 2003: Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, pp. 163–174. ACM, New York (2003)
6. Petrovic, M., Burcea, I., Jacobsen, H.A.: S-ToPSS: semantic Toronto publish/subscribe system. In: *VLDB 2003: Proceedings of the 29th international conference on Very large data bases, VLDB Endowment*, pp. 1101–1104 (2003)
7. Keeney, J., Lynch, D., Lewis, D., O’Sullivan, D.: On the Role of Ontological Semantics in Routing Contextual Knowledge in Highly Distributed Autonomic System. Technical report, Department of Computer Science, Trinity College Dublin (2006)
8. Keeney, J., Jones, D., Roblek, D., Lewis, D., O’Sullivan, D.: Knowledge-based semantic clustering. In: *SAC 2008: Proceedings of the 2008 ACM symposium on Applied computing*, pp. 460–467. ACM, New York (2008)
9. Baader, F., Horrocks, I., Sattler, U.: Description logics as ontology languages for the semantic web. In: *Festschrift in honor of Jörg Siekmann. LNCS (LNAI)*, pp. 228–248. Springer, Heidelberg (2003)
10. Horridge, M., Drummond, N., Goodwin, J., Rector, A., Stevens, R., Wang, H.H.: The Manchester OWL Syntax. In: *OWL: Experiences and Directions 2006*, Athens, Georgia, USA, November 10–11 (2006)
11. Schoeneich, R.O., Domaszewicz, J., Koziuk, M.: Concept-Based Routing in Ad-Hoc Networks. In: *The 10th International Conference on Distributed Computing and Networking - ICDCN 2009, January 3–6 (submitted, 2009)*
12. Carroll, J., Dickinson, I., Dollin, C., Reynolds, D., Seaborne, A., Wilkinson, K.: Jena: Implementing the semantic web recommendations (2003)

13. Sirin, E., Parsia, B., Grau, B.C., Kalyanpur, A., Katz, Y.: Pellet: A practical owl-dl reasoner. *Web Semant.* 5(2), 51–53 (2007)
14. Drummond, N., Horridge, M., Stevens, R., Wroe, C., Sampaio, S.: Pizza ontology v1.3 (October 18, 2005),
<http://www.co-ode.org/ontologies/pizza/2005/10/18/>
15. Calvanese, D., Giuseppe, D.G., Lembo, D., Lenzerini, M., Rosati, R.: DL-Lite: Tractable description logics for ontologies. In: *Proceedings of the Twentieth National Conference on Artificial Intelligence*, pp. 602–607 (2005)

Towards a System for Ontology-Based Information Extraction from PDF Documents

Ermelinda Oro¹ and Massimo Ruffolo²

¹ Department of Computer Science and System Science (DEIS)
linda.oro@deis.unical.it

² Institute of High Performance Computing and Networking of CNR (ICAR-CNR)
University of Calabria, 87036 Rende (CS), Italy
ruffolo@icar.cnr.it

Abstract. Ontologies enable to directly encode domain knowledge in software applications, so ontology-based systems can exploit the meaning of information for providing advanced and intelligent functionalities. One of the most interesting and promising application of ontologies is information extraction from unstructured documents. In this area the extraction of meaningful information from PDF documents has been recently recognized as an important and challenging problem. This paper proposes an ontology-based information extraction system for PDF documents founded on a well suited knowledge representation approach named *self-populating ontology* (*SPO*). The *SPO* approach combines object-oriented logic-based features with formal grammar capabilities and allows expressing knowledge in term of ontology schemas, instances, and extraction rules (called *descriptors*) aimed at extracting information having also tabular form. The novel aspect of the *SPO* approach is that it allows to represent ontologies enriched by rules that enable them to populate them-self with instances extracted from unstructured PDF documents. In the paper the tractability of the *SPO* approach is proven. Moreover, features and behavior of the prototypical implementation of the *SPO* system are illustrated by means of a running example.

Keywords: Ontology, Information Extraction, Attribute Grammars, Knowledge Representation, Datalog.

1 Introduction

Nowadays, there is a growing interest in ontologies that are expected to extend current information technologies capabilities. Ontologies enable to directly encode domain knowledge in software applications, so ontology-based systems can exploit the meaning of information for providing advanced and intelligent functionalities.

A very interesting and promising application of ontologies is *information extraction* (*IE*). The aim of information extraction is to recognize and extract relevant information, contained in unstructured documents, and to store them in structured knowledge bases. Thus extracted information can be queried and analyzed by means of already existing techniques coming from the database world. One of the most diffused unstructured document format is the Adobe portable document format (PDF). Information extraction from PDF has been recently recognized as a very important and challenging problem because PDF documents are completely unstructured and their internal encoding

is completely visual-oriented. So traditional wrapping/information extraction systems cannot be applied. In [1] Gottlob et al. point out that "there is a substantial interest from industry in wrapping documents in format such as PDF and PostScript. In such documents, wrapping must be mainly guided by a reasoning process over white spaces... it is very different from Web wrapping and will require new techniques and wrapping algorithms".

Currently there is available a large body of literature on (ontology-based) information extraction approaches and systems (see Section 2). Existing approaches, however, suffer from the following principal drawbacks: (i) information is extracted mainly by exploiting their syntactic structure and not their actual semantics; (ii) extraction rules are able to identify tabular information only when such a structure is explicitly declared (as happens in html documents); (iv) when existing systems adopt ontologies they mainly works in two steps: the first one is the actual information recognition and extraction, while the second one is the annotation of extracted information to already existing ontologies. Current systems generally do not directly exploit the knowledge represented in the ontology to perform information recognition and extraction. Most importantly, at the best of our knowledge no ontology based systems for extracting information from PDF documents exist.

This paper proposes a novel ontology-based information extraction system founded on a well suited and "ad hoc" knowledge representation approach named *self-populating ontology* (*SPO*). The *SPO* approach combines object-oriented logic-based capabilities with formal grammar features. In particular, ontology representation capabilities are obtained by extending Datalog [2,3] by means of object-oriented constructs, such as, *class*, *object* and *inheritance*. Such capabilities are combined with *Attribute Grammars* [4] that extends a context free grammars by *attributes*, *functions* and *predicates*. The peculiarity of the *SPO* approach is that it allows to represent ontologies enriched by *production rules* in which *non-terminal symbols* coincide with class declared in the ontology, *attributes* of a non-terminal coincide with the relative class attributes, *predicates* are constituted by queries on the ontology. This way, the *SPO* approach allows to represent ontology schemas and instances where classes and objects can be equipped by *descriptors*. Descriptors are production rules where the right-side (*descriptor-body*) constitutes an (*extraction*) *pattern* which recognition in a document means that the object in the left-side (*descriptor-head*) either: exists in the document (*object descriptor*) or can be extracted and stored as a class instance (*class descriptor*). Descriptors extend attribute grammars by means of 2-dimensional composition capabilities that enable to recognize and extract also information having tabular form. Roughly speaking, descriptors are rules that *describe* how information can be extracted from PDF unstructured documents, and stored as ontology objects. The main important feature of descriptors is that they can exploit each other in describing concepts, so each ontology object can be described by the composition of other objects. In the *SPO* approach domain knowledge, extraction rules, and extracted information live together in the same conceptual structure. The tractability of *SPO* approach is proven in the paper.

The *SPO* system is capable to exploit knowledge and descriptors represented in the ontology for extracting, from PDF documents, information that, in turn, populates the *SPO* itself. In order to allow extracting information having tabular form, the *SPO* system

also exploits a 2-dimensional PDF document pre-processing technique (see Section 3). Such technique allows to create an internal document representation in which each document is viewed as a set of strings contained in 2-dimensional document areas (named *portions*) placed over a Cartesian plane. This representation enables to best exploit descriptors features.

The primary contributions of this paper are: (i) the presentation of the novel ontology-based information extraction system that enables to directly populate an ontology with information (having also tabular form) extracted from PDF documents; (ii) the description of the tractable knowledge representation approach which the system is founded on. The presented system constitutes a semantic technology that can contribute to empower unstructured information management capabilities of existing applications for creating valuable solutions for enterprises.

The remainder of this paper is organized as follows. Section 2 briefly describes related work. Section 3 shows the PDF documents pre-processing method used by *SPO* approach. Section 4 defines the *SPO* approach. Section 5 describes how the prototypical implementation of the *SPO* approach works. Finally section 6 concludes the paper.

2 Related Work

A large body of work concerning approaches and systems aimed at extracting relevant information from semi-structured and unstructured documents is currently available in literature. Already existing approaches and systems, as described in [5], can be classified as *manual*, *semi-supervised* and *unsupervised* by considering the automation degree adopted in the definition of wrappers and extraction rules. They adopt techniques founded on query languages, grammar rules, natural language processing, HTML tree processing, fuzzy logic, ontologies. Most significant *manual* approaches are: TSIMMIS, Minerva, Web-OQL, W4F, XWRAP [5,6], JEDI [7], FLORID [8]. The *semi-supervised* group contains the following approaches: SRV, RAPIER, WHISK, WIEN, STALKER, SoftMealy, NoDoSe, DEByE [5,6] and LixTo [1,9]. In the *unsupervised* group the most significant approaches are: STAVIES [10], DeLa, RoadRunner, EX-ALG, DEPTA [5,6].

By considering the degree of structuring of the input documents, which the extraction tasks are performed on, IE approaches and systems can be classified in: structured, (wrapping from database and XML); semi-structured (extraction from HTML, fixed length record files like EDCDIC); and unstructured (extraction from flat text). The greater part of previously cited systems and approaches works on semi-structured documents (mainly Web document in HTML format). Unstructured document oriented approaches and systems can be nowadays split in two families: NLP-oriented and PDF-oriented. NLP-oriented systems, have its origins in the Message Understanding Conferences (MUCs) [11] and adopt NLP-based techniques in learning and applying extraction rules on flat text or weekly structured Web documents. Systems that belongs to this family are GATE [12], RAPIER, SRV, WHISK [5,6], KnowItAll [13], TextRunner [14], SnowBall [15], DIPRE [16], furthermore well famous works in this area are contained in [17]. PDF-oriented information extraction approaches appeared recently in literature after a seminal Gottlob's work. In particular Flesca et Al. [18] propose a fuzzy

logic approach that exploits spatial PDF feature in recognizing relevant information; Baumgartner et Al. [19] use document understanding techniques for identifying *atomic* elements of PDF documents on which apply spatial reasoning and ontology based wrapping that enable to identify significant document blocks; Gottlob in [20] describes the PDF document preprocessing techniques currently used by the LixTo system, but do not mention the technique used from extracting information. Before the Gottlob's papers only document understanding [21] and table recognition methods [22] was applied on PDF documents in attempting to identify and extract relevant information from them.

The ontology research area influenced information extraction. A lot of work concerning the use of ontology for extracting meaningful information from HTML Web documents has been proposed in literature. One of the first work in this area is [23]. Recently many relevant approaches appeared, some of the most relevant are described in [24,25,26,27]. A sub area of ontology named extraction ontologies propose methods to use ontologies in information extraction [28,29,30].

At the best of our knowledge no existing approach that deals with the problem of extracting meaningful information from PDF documents by using ontologies has already been proposed.

3 Unstructured PDF Documents Handling

The aim of the system presented in this work is to extract information from documents in Adobe's portable document format (PDF), even when information is in tabular form, and store them as ontology objects (population).

PDF is a well known unstructured document format thought for print and on screen visualization. It can be considered the standard format for document publication, sharing and exchange. PDF documents are encoded by means of the *document description language*. In this language a document consists of a collection of 2-dimensional objects (i.e. textual and graphical elements, images) contained in *content streams*. Each object has some metadata that contain presentation features and coordinates that express the position in which it must be shown on the page at visualization or printing time. Objects can appear in casual order in the content stream, so the appearance of contents of a PDF document can be understood only after page rendering or printing. PDF documents are widely used in enterprises and on the web. Thus the extraction of meaningful information from them is worthwhile, however the intrinsic print/visual oriented nature of PDF format poses many issues in defining "ad hoc" information extraction approaches. In fact, normally information extraction systems exploit the syntactic structure of electronic documents (e.g. HTML tags) for extracting information.

Since PDF documents are strongly unstructured, in order to extract information from them, a preprocessing technique that allows to obtain an internal document representation suitable for the *SPO* approach is required. This section describes such a representation and sketches the technique, named *portioning process*, used to create it.

The internal document representation adopted in the *SPO* approach is founded on the idea of *document portion*. A document portion is a three tuple of the form $\pi = \langle \sigma, \nu_1, \nu_2 \rangle$ where: (i) σ is a string of alphanumeric characters in which the special character \$ represents images; (ii) ν_1 and ν_2 are the Cartesian coordinates of two opposite vertices of the rectangular area of the document, that contains the string σ .

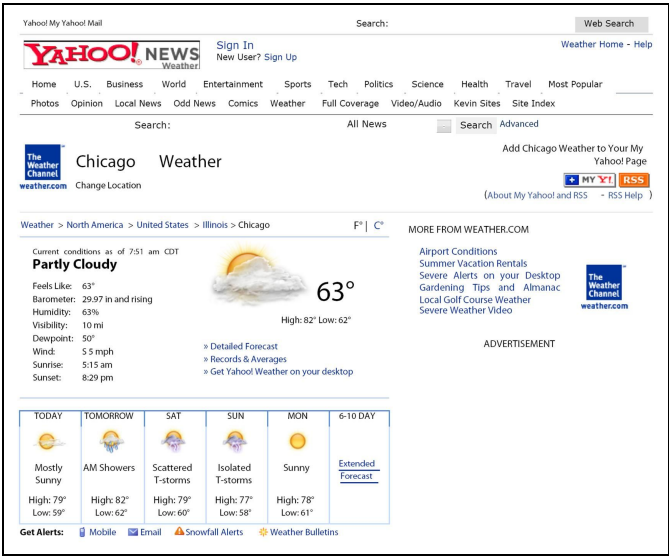


Fig. 1. The PDF version of the Yahoo Chicago weather page

The document portion idea enables to represent a PDF document as a set $\mathcal{I}$ of 2-dimensional strings arranged over a Cartesian plane. For instance, by considering the PDF document shown in Figure 1¹, Cartesian plane and portions are depicted in Figure 5. Coordinates assigned to each portions preserve relative positions among content elements belonging to the table. The portion in Figure 5 with coordinates $\nu_1 = (8, 19)$ and $\nu_2 = (12, 20)$ contains the string: $\sigma = "$ \$ Weather Bulletins".

This representation is exploited by the system as described in Section 5 and enables *SDO* languages to express 2-dimensional composition rules that permit to recognize and extract also tabular information.

Presented representation is obtained by applying to PDF documents an heuristic algorithm called *portioning process*. Because of the lack of space just a sketch of the algorithm is given. The algorithm execute the following step: (i) Content elements extraction. By accessing content streams, tokens and images are identified and acquired with their spatial coordinates (Figure 2). (ii) Space distribution analysis. This is a fundamental step that allows to identify how content elements are distributed in a page. The algorithm analyzes the horizontal and vertical space distribution among tokens and images of a page and defines two threshold values T_H and T_V . (iii) Cluster building. By considering T_H and T_V the algorithm groups elements which horizontal and/or vertical distance is below the related threshold value. Elements embraced in a cluster are arranged in a new portion (Figure 3) which coordinates depends from those of contained elements, and string is obtained by concatenating tokens and special chars representing images (preserving visualization order). (iv) Lines building. In this step the algorithm checks the horizontal coordinates of clusters and isolated elements in order to

¹ Obtained from <http://weather.yahoo.com> by converting in PDF the related HTML page.

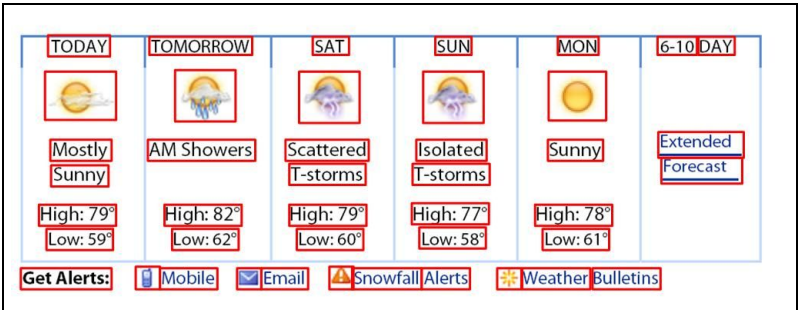


Fig. 2. Example of initial PDF elements

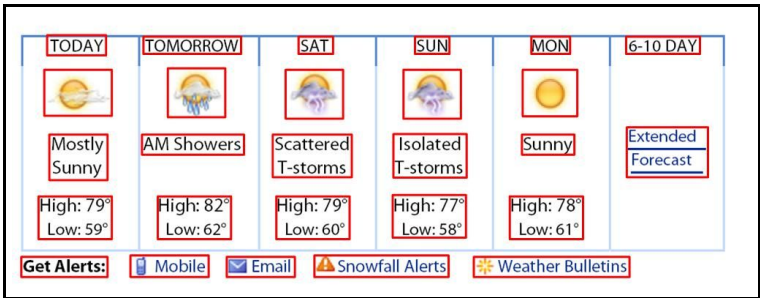


Fig. 3. Example of clustered elements

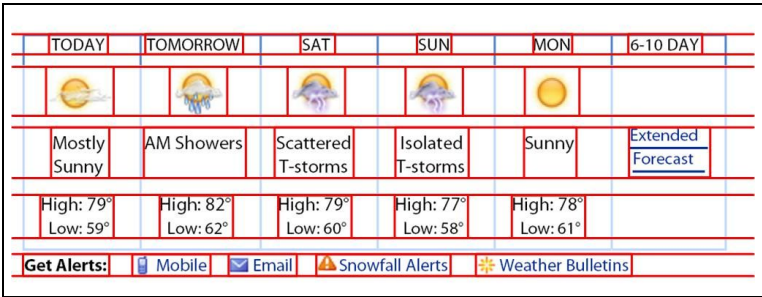


Fig. 4. Example of recognized lines

define those elements that can be considered belonging at the same line (Figure 4). (v) Lines tagging. Space distribution among line elements (i.e. clusters and isolated tokens and images) is evaluated. Depending from the space distribution, lines are heuristically tagged as text or table lines. (vi) Zones building. The algorithm analyzes the sequence of lines and identifies possible text and table zones as sequences respectively of text and table lines. When images that cover a significant number of lines are identified an image zone is defined. (vii) Table and paragraph building. In this step the algorithm applies a table recognition method on table zones by constructing a grid of table cells

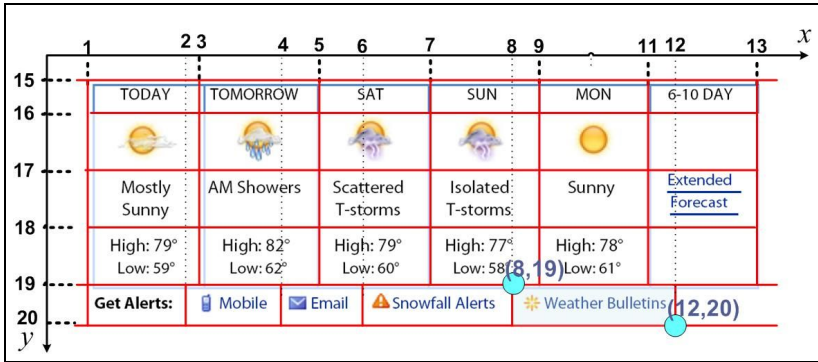


Fig. 5. Example of *document portions*

and assigning coordinates as shown in Figure 5. Text paragraphs are identified merging elements of text zones. This two activities are combined in order to avoid possible misclassification of lines.

It is worthwhile note that parameters used in the algorithm strongly depends from the document layout. For example, documents arranged in multiple columns or presenting complex composition of images and text requires different parameter settings. Furthermore step (vii) can be avoided when the user intends to extract information only from text, in this case in step (v) all lines are tagged as text.

4 The SPO Knowledge Representation Approach

In this section the object-oriented logic-based knowledge representation approach, that allows information extraction from PDF documents, is formally defined. Furthermore, consistency and semantics of the paradigm are described and some complexity results are drawn. In the following is assumed that the reader is familiar with formal grammars [31], attribute grammars [4] and Datalog [2,3].

A *SPO* is, formally, a couple $\mathcal{OG} = \langle \mathcal{O}^z, \mathcal{AG} \rangle$ where $\mathcal{O}^z = \langle D, A, C, R, \preceq, \sigma, \delta, \iota \rangle$ is an ontology; $\mathcal{AG} = \langle \mathcal{G}, Attr, Func, Pred \rangle$ is an attribute grammar in which $\mathcal{G} = \langle \Sigma, N, S, \Pi \rangle$ is the underlying context free grammar.

The ontology $\mathcal{O}^z$ is obtained extending Datalog by object-oriented constructs, such as, *class*, *object* and *inheritance*. More formally, let Z be a set of *constants* and $\tilde{Z}$ (the whole set of *values*) be that one obtained by the union of Z with all finite lists of elements in Z . An ontology on Z is an eight-tuple $\mathcal{O}^z = \langle D, A, C, R, \preceq, \sigma, \delta, \iota \rangle$ where:

- D, A, C, R are *disjoint* sets of entity names respectively called *data-types*, *attribute-names*, *classes* and *relations*. Set D contains only *integer* and *string* data-types. Set A contains the special attribute-name **id**. Elements in $C \cup D$ are called *flat-types*. For each flat-type t there is a *list-type* denoted by $[t]$. Their union forms all *types* denoted by T ;
- $\preceq$ is a partial order (called *isA*) on C ;
- $\sigma : C \cup R \rightarrow 2^{A \times T}$ is the *schema* function. For each $e \in C \cup R$, then set $\sigma(e)$ is the *schema* of e and any couple $\langle a, t \rangle \in \sigma(e)$ is the (schema) *attribute* of e with

name a and type t . The schema of a class c contains the attribute $\langle id, c \rangle$, whereas no relation schema contains attributes with name **id**;

- $\delta : D \rightarrow 2^Z$ (domain function) associates a value domain to each data-type;
- ι is the instance function associating to an element $e \in C \cup R$ set $\iota(e)$ from $2^{A \times \tilde{Z}}$ called the (direct) instances of e (also objects if e is a class and tuples otherwise). Let $\hat{e} \in \iota(e)$, then couple $\langle a, z \rangle \in \hat{e}$ is the (instance) attribute of $\hat{e}$ with name a and value z . If $\hat{e}$ is an instance of a class, then value $z \in Z$ of the attribute $\langle id, z \rangle \in \hat{e}$ is called the object identifier (*oid*) of $\hat{e}$.

The peculiarity of the *SPO* paradigm is that elements $\mathcal{O}^Z$ and $\mathcal{A}\mathcal{G}$ are strictly coupled as described in the following. Sets N and $Attr$ coincide, respectively, with C and A . Sets Σ and Z are disjoint. Moreover, if $a \in Attr(Y)$ with $Y \in N$, then t is the type of a if and only if $\langle a, t \rangle \in \sigma(Y)$. Set *Func* contains arithmetic expressions, string expressions, list expressions and all the functions that allow to manipulate attribute values in P-TIME. Furthermore, let p be any production in $\mathcal{O}\mathcal{G}$, then each attribute in the right-hand side of p appears at most once in all the expressions of *Func*(p). Set *Pred* contains: (i) comparison predicates on integers, strings or lists; (ii) decision queries on $\mathcal{O}_+^Z$ that extend $\mathcal{O}^Z$ by new instances (generated during the extraction process); (iii) a default predicate (associated to each production) checking whether computed attribute values are consistent with respect to $\mathcal{O}_+^Z$.

4.1 Representing Schemas and Instances

The *SPO* approach allows to express *SPO* schemas (classes and relations) and instances (objects and tuples) that both represent the semantic of information to extract and the structure of the ontology to populate.

The syntax for expressing schemas and instances is based on Datalog augmented with Object-Oriented features (existing systems [32,33] could be simply used in the *SPO* system). The syntax adopted for expressing schemas, instances, and descriptors in the *SPO* approach is shown in the following by considering a simple running example aimed at extract weather forecast information from the table having the structure depicted in figure 1.

A class is an aggregation of individuals (*objects*) that have the same set of properties (*attributes*). Attribute **id** is implicitly declared.

```
class koppenClimate(lTempF:integer,hTempF:integer,avgRainfallCm:integer).
class continentalClimate(summerHumidity:string,
    winterHumidity:string) isa {koppenClimate}.
    hotSummer:continentalClimate(lTempF:-36,hTempF:86,avgRainfallCm:90,
        summerHumidity:"dry",winterHumidity:"wet").
class place(name:string).
class state(areaKm2:integer, capital:city, neighborState:[state]) isa {place}.
    illinois:state(name:"Illinois",areaKm2:140998, capital:springfield,
        neighborState:[wisconsin,kentucky,iowa,missouri,indiana]).
class city(population:integer, inState:state) isa {place}.
    chicago:city(name:"Chicago",population:2833321,inState:illinois).
```

Class city shows the ability to specify user-defined classes as attribute types (e.g. inState:state). For class city is represented an object which oid is the constant chicago, while string "Chicago" is the value for the attribute

name:string. Class state has a list-type attribute neighborState:[state]. Classes koppenClimate and continentalClimate show how class hierarchies (taxonomies) can be built up by using isa key-word.

Relationships among objects are represented by means of *relations*, which like classes, are defined by a name and a list of attributes. Relations cityClimate, which *tuples* assert what is the climate of a given city, can be declared as follows:

```
relation cityClimate(c:city,climate:koppenClimate).
    cityClimate(c:chicago,climate:hotSummer).
...
```

Default Schemas and Instances for Documents Handling. The SPO language provides the following set of schemas and instances aimed at enabling document content handling.

```
class box(paragraph:[basicElement]).
class basicElement().
    class token (value:string) isa {basicElement}.
    class image (uri:string) isa {basicElement}.
relation hasLemma(tok:token, lemma:string, tag:posTag).
relation defBy(token:token, regex:regexPattern).
class PostTag(code:string).
    noun: PostTag ("noun").
    adj: PostTag ("adjective").
    verb: PostTag ("verb").
...
```

Class *box* aims at collecting objects representing document portions (see Section 3). Attribute *paragraph* keeps a list of *basicElements* objects that preserves the order which they appear in the related document portion. Class *basicElement* collects instances of the classes *token* and *image*. For example, the last portion in figure 5 is represented by a *box* instance that contains an image and a token.

The instances of the class *basicElement* may have a set of features that characterize them and that can be exploited in descriptors to guide the extraction process. In particular, instances of class *token* could have linguistic properties that are represented by means of the relation *hasLemma* that associates a token to its lemma and POS-tag. Linguistic properties of a token are obtained by exploiting existing free NLP tools.

Semantic information extraction exploits dictionaries that represent basic domain knowledge. Relation *defBy* binds a token with the related dictionary entry used to define it. Regular expressions are collected as instances of the class *regexPattern* as shown in the following.

```
class regexPattern (regex:string, flagMode:integer).

rx_d2: regexPattern("\d{2}",2). (1)
rx_circ: regexPattern("o",2). (2)
rx_lowHigh: regexPattern("low:|high:|min:|max:",2). (3)
rx_sun: regexPattern("sun(?:day)?",2). (4)
...
```

The attributes `regex:string` and `flagMode:integer` respectively contain a regular expression and its matching mode². In above examples, the value "2" of attribute `flagMode` represents case insensitive matching mode.

4.2 Representing Object and Class Descriptors

The main feature of the *SPO* language is the ability to equip classes and objects of an ontology with *descriptors*. Descriptors are rules that *describe* how information can be recognized and/or extracted from unstructured documents, and stored as ontology objects. Descriptors can exploit each other in describing concepts, so each ontology object can be described by the composition of other objects. Descriptors are extraction rules that constitutes an abstract way for expressing $\mathcal{AG}_{\leq}$ productions extended by 2-dimensional capabilities. Each right-side (*descriptor-body*) constitutes an (*extraction*) *pattern* which recognition in a document means that the object in the left-side (*descriptor-head*) either: exists in the document (*object descriptor*) or can be extracted and stored as a class instance (*class descriptor*). By considering the weather forecast example, in the following some descriptors required to recognize and extract the whole table are shown.

Descriptor for class `weatherForecastTable` describes a weather forecast table as a an horizontal sequence of `weatherForecast` objects (columns). The list-type attribute `weathers` is aimed at containing table columns. Its values are set by means of list concatenations `L:=L+X`; and `L:=L+L1` ; .

```
class weatherForecastTable(weathers:[weatherForecast]).
<weatherForecastTable(weathers:[L])> ->
  <X:weatherForecast()> {L:=L+X;}
  <weatherForecastTable(weathers:[L1])> {L:=L+L1;} .
<weatherForecastTable(weathers:[L])> ->
  <X:weatherForecast()> {L:=L+X;} .
```

(5)

A column containing the weather forecast for a day (class `weatherForecast`), can be recognized and extracted by means of the following descriptor.

```
class weatherForecast(day:weekDay, descr:token,
  hTemp:integer, lTemp:integer).
<weatherForecast(weekDay:WD, descr:D, hTemp:HT, lTemp:LT)> ->
  <X:weatherDayCell(day:T)> {WD:=T;}
  <B:box(paragraph:[I]), I:image()>
  <X:weatherDescriptionCell(descr:T)> {D:=T;}
  <X:weatherTemperaturesCell(high:H, low:L)> {HT:=T; LT:=T;}
  DIR "vertical".
```

(6)

This descriptor describes a column of the weather forecast table as a vertical sequence (operator `DIR`) of a `weatherDayCell`, a box that contains just an image, a

² Regular expressions and matching modes adopt the java syntax [34] that uses the following set of characters with the specified meaning: `'\w'` word character; `'\d'` digit; `'\s'` whitespace. Character `'\''` serves to introduce escaped constructs. Moreover, operator `'(?:'` is used instead classical `'('`.

weatherDescriptionCell, a weatherTemperaturesCell. In the following one of these descriptors (weathertemperatCell) is shown.

```
class weatherTemperaturesCell(high:integer, low:integer)
<weatherTemperaturesCell(high:H, low:L)> ->
  <B:box()> CONTAIN
  <X:temperaturesPair(high:H0, low:L0)>{H:=H0;L:=L0};
```

(7)

This descriptor exploits the CONTAIN construct that allows to check spatial containment among objects. Thus a weatherTemperaturesCell represents cells of the weather forecasting table that contain a temperaturesPair.

The following descriptors allow to recognize and extract object of the classes weatherTerm, temperaturesPair and temperature.

```
class weatherTerm(descr:string).
  sunny:weatherTerm("sunny").
  <sunny> -> <T:token(), hasLemma(tok:T, lemma:"sunny")>.
  shower:weatherTerm("showers").
  <shower>-> <T:token(), hasLemma(tok:T, lemma:"shower")>.
```

(8)
(9)

```
class temperaturesPair(high:integer, low:integer).
<temperaturesPair(high:H, low:L)> -> {integer TMP;}
  <temperature(value:T)> {TMP:=T;}
  <T:token(), defBy(T, rx_lowHigh)>
  <temperature(value:T)> {H:=#max(TMP, T); L:=#min(TMP, T);}
  SEPBY <blankSequence>.
```

(10)

```
class temperature(value:integer).
<temperature(value:V)> ->
  <T:token(value:S), defBy(T, rx_d2)>{V:=#str2int(S);}
  <T:token(value:"o")>.
```

(11)

The descriptor of the class weatherTerm exploits linguistic capabilities. For instance, it allows to recognize the object sunny when in the text a token with lemma "sunny" is present. A weather temperature object is a sequence of a number (with two digits) and the symbol 'o'. The function #str2int converts a string in the corresponding integer value. The objects recognized by means of the temperature descriptor, are exploited by the descriptor of the class temperaturesPair that describes a sequence of: a temperature, a token and a second temperature all separated by a blankSequence object. The construct SEPBY is "syntactic sugar" and expresses that each couple of objects must be *separated* by one or more blank characters. The higher and the lower temperatures are calculated in procedural mode.

4.3 Consistency and Semantics of SPO

Consistency of SPO schemas. Any schema contains only attributes with distinct names. Let c_1 and c_2 be two classes such that $c_2 \preceq c_1$. For each attribute $\langle a, t_1 \rangle \in \sigma(c_1)$ there

exists precisely one other attribute $\langle a, t_2 \rangle \in \sigma(c_2)$ with the same name. If t_1 is either a data-type or a list-type $[t]$, where t is a data-type, then $t_2 = t_1$. If t_1 is a class, then t_2 is a class and $t_2 \preceq t_1$ holds. If t_1 is a list-type $[t_1]$, where $\hat{t}_1$ is a class, then $t_2 = [\hat{t}_2]$ and $\hat{t}_2 \preceq \hat{t}_1$ holds.

Consistency of SPO instances. Given a type $t \in T$, then value z is *compatible* with t if one of the following conditions holds: (i) $t \in D$ and $z \in \delta(t)$; (ii) $t \in C$ and z is an *oid* of an instance of t ; (iii) $t = [\hat{t}]$ is a list-type, and z is a list of values all of which are *compatible* with $\hat{t}$. Set $\tilde{Z}^t$ denotes all values in $\tilde{Z}$ compatible with t . Let e be an element in $C \cup R$ that has schema $\sigma(e) = \{\langle a_1, t_1 \rangle, \dots, \langle a_h, t_h \rangle\}$. Then, each instance $\iota(e)$ of e has the following form $\{\langle a_1, z_1 \rangle, \dots, \langle a_h, z_h \rangle\}$ where each z_j is *compatible* with t_j ($1 \leq j \leq h$). There are no two instances sharing the same *oid*.

Semantics of SPO. The semantics of $\mathcal{O}^z$ is given in terms of Datalog. The datalog representation $\mathcal{C}(\mathcal{O}^z)$ of $\mathcal{O}^z$ is the set of all the clauses created from $\mathcal{O}^z$ as follows: (i) for each element $e \in C \cup R$ and instance $\hat{t} = \{\langle a_1, z_1 \rangle, \dots, \langle a_h, z_h \rangle\}$ in $\iota(e)$, create ground fact $e(z_1, \dots, z_h) \leftarrow$; (ii) for each couple of classes c_1, c_2 in C such that $c_2 \preceq c_1$, create clause $c_1(X_1, \dots, X_h) \leftarrow c_2(X_1, \dots, X_h, \dots, X_k)$ where $k \geq h$, $|\sigma(c_1)| = h$, and $|\sigma(c_2)| = k$.

Given a Datalog query $\mathcal{Q}$ where any of its predicate symbol $e \in C \cup R$ has arity $|\sigma(e)|$. So, the set of couples $\langle p(X_1, \dots, X_n), \mathcal{Q} \rangle$, where p is a generic predicate, is called *decision query* on $\mathcal{O}^z$. Whenever each variable X_i assume value z_i ($1 \leq i \leq n$), thus expression $\mathcal{C}(\mathcal{O}^z) \cup \mathcal{Q} \models p(z_1, \dots, z_n)$ may be evaluated³ for checking whether the ground atom $p(z_1, \dots, z_n)$ is **true** with respect to $\mathcal{O}^z$.

In order to formally define language $\mathcal{L}(\mathcal{OG})$, we extend $\mathcal{G}$ and $\mathcal{AG}$ by $\mathcal{G}_{\preceq} = \langle \Sigma, N, S, \Pi_{\preceq} \rangle$ and $\mathcal{AG}_{\preceq} = \langle \mathcal{G}_{\preceq}, \text{Attr}, \text{Func}_{\preceq}, \text{Pred} \rangle$, respectively, making use of $\mathcal{O}^z$ in such a way that: (i) $\Pi_{\preceq} = \Pi \cup \{c_1 \rightarrow c_2 \mid c_1, c_2 \in C, c_2 \preceq c_1\}$; (ii) $\text{Func}_{\preceq}(p) = \{c_1.a : c_2.a \mid \langle a, t \rangle \in \sigma(c_2) \text{ for some type } t\}$, for each production $p : c_1 \rightarrow c_2$ in $\Pi_{\preceq} \setminus \Pi$.

Definition 1. Grammar $\mathcal{AG}_{\preceq}$ is equivalent to $\mathcal{OG}$, that is $\mathcal{L}(\mathcal{OG}) = \mathcal{L}(\mathcal{AG}_{\preceq})$.

4.4 Complexity of SPO

Lemma 1. Let t be any parse tree⁴ of $\mathcal{AG}_{\preceq}$ and $x(t)$ the string that yields t such that $\|x(t)\| = n$. Then, each attribute value in t has length $\mathcal{O}(n)$.

Proof. Let $x_1, \dots, x_k$ be the attributes of $\mathcal{AG}_{\preceq}$. Without loss of generality, we assume to deal with attributes on strings. Given a node v in t , we denote by $\|v\|$ the length of the string obtained concatenating all the attribute values in v . Since any attribute can be exploited at most once in all the functions of a production, then $\|v\| \leq c + |x_1^1| + \dots + |x_k^1| + \dots + c + |x_1^h| + \dots + |x_k^h|$ holds, where any x_i^j is attribute x_i in the j^{th} child of v ($0 \leq i \leq k$), and c is the length of the longest (constant) string in $\text{Func}_{\preceq}$. So the rise in length of each node is at most $k * c$ and $\text{size}(t) \leq |N| * (2n - 1)$ where $|N|$ is the

³ Lists of elements are handled as constants.

⁴ Note that t could also be invalid for $\mathcal{AG}_{\preceq}$.

number of $\mathcal{AG}$ non-terminal symbols, then $\|\rho(t)\| \leq k * c * |N| * (2n - 1)$ where $\rho(t)$ is the root of t . Clearly any attribute value in t has length $\mathcal{O}(n)$. ■

Definition 2. Given $\mathcal{OG}$ in which $\mathcal{AG}_{\leq}$ is unambiguous, then problem OG-PARSE is defined as follows. Given an input string w , does w belong to $\mathcal{L}(\mathcal{AG}_{\leq})$?

Theorem 1. OG-PARSE is **P**-complete.

Proof. (Membership) Let w to parse for membership in $\mathcal{L}(\mathcal{AG}_{\leq})$. Since $\mathcal{G}_{\leq}$ is unambiguous, so if $w \in \mathcal{L}(\mathcal{G}_{\leq})$, there exists exactly one parse tree t for $\mathcal{G}_{\leq}$. Then, all the functions in $Func_{\leq}$ related to the nodes of t can be computed from the leaves of t to its root. The union of $\mathcal{O}^z$ with the new objects generated in t composes $\mathcal{O}_+^z$. Hence, for each non-leaf node v_0 of t , having children $v_1, \dots, v_h$, are evaluated the related predicates. In particular, to each query $\langle p(X_1, \dots, X_n), Q \rangle$ in v_0 corresponds the evaluation of expression $\mathcal{C}(\mathcal{O}_+^z) \cup Q \models p(z_1, \dots, z_n)$, where $z_1, \dots, z_n$ are attribute values of node v_i ($0 \leq i \leq h$). Consider now that t is generated in polynomial time, and all functions are polynomial time computable. Moreover, let $\mathcal{D} = \mathcal{C}(\mathcal{O}_+^z) \setminus \mathcal{C}(\mathcal{O}^z)$ the ground facts representing the new objects, then $\mathcal{D}$ is polynomial in the size of w as shown in the proof of by Lemma 1. Therefore, since deciding $Q \cup \mathcal{D} \models p(z_1, \dots, z_n)$ is **P**-complete [35] when Datalog query Q is fixed, whereas database $\mathcal{D}$ and atom $p(z_1, \dots, z_n)$ are an input (data complexity), then $\mathcal{C}(\mathcal{O}^z) \cup \mathcal{D} \cup Q \models p(z_1, \dots, z_n)$ is polynomial in $\|w\|$ because $\mathcal{C}(\mathcal{O}^z)$ and Q are fixed while $p(z_1, \dots, z_n)$ and $\mathcal{D}$ are an input.

(Hardness) It is enough to notice that since deciding $Q \cup \mathcal{D} \models p(z_1, \dots, z_n)$ is **P**-complete, so $\mathcal{C}(\mathcal{O}^z) \cup \mathcal{D} \cup Q \models p(z_1, \dots, z_n)$ is **P**-hard, then OG-PARSE cannot be easier. ■

5 SPO Approach at Work

In this section is synthetically described how the system prototype that implements the *SPO* approach works. Figure 6 shows the prototype architecture. A *SPO* is used for many *population processes* that are sequences of the following steps: pre-processing, 2-D matching, population. The input of a population process is constituted by the $\mathcal{AG}_{\leq}$ produced by the *compiler*, an *unstructured document* and a *user query*. The output is

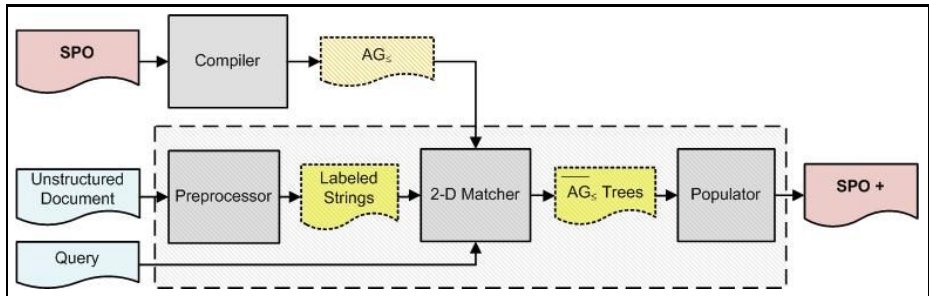


Fig. 6. Architecture of the SPO System Prototype

the *SPO*+ that is the original *SPO* augmented by concept instances extracted from the input document. Compiling, 2-dimensional matching and population are described in the following. Whereas preprocessing has already been described in Section 3.

5.1 *SPO* Compiling

The *compiler* translates a *SPO*, expressed by means of the *SPO* language, in terms of $\mathcal{AG}_{\leq}$ production rules. Productions are equipped by a label that express the spatial relationships existing among objects they combine (i.e. horizontal and vertical concatenation, containment).

For instance, an object descriptor number (8) is translated in term of $\mathcal{AG}_{\leq}$ production rules as follows:

$$\begin{aligned}
 \text{WEATHERTERM}_0 &\rightarrow \text{TOK\_SUNNY}_1, & id_{(0)} &:= \text{sunny}, & value_{(0)} &:= \text{"sunny"}, \\
 & & & \# \# \text{hasLemma}(id_{(1)}, \text{"sunny"}). \\
 \text{TOK\_SUNNY}_0 &\rightarrow \text{'sunny'}, & id_{(0)} &:= \# \text{newID}(), & value_{(0)} &:= \text{"sunny"}. \\
 \text{TOK\_SUNNY}_0 &\rightarrow \text{'sunnier'}, & id_{(0)} &:= \# \text{newID}(), & value_{(0)} &:= \text{"sunnier"}. \\
 \text{TOK\_SUNNY}_0 &\rightarrow \text{'sunniest'}, & id_{(0)} &:= \# \text{newID}(), & value_{(0)} &:= \text{"sunniest"}.
 \end{aligned} \tag{12}$$

In this production the attributes *id* and *value* of the non terminal *WEATHERTERM* are set respectively to *sunny* (constant in $\mathcal{O}^z$) and "sunny" (string) by using the related functions when the predicate $\# \# \text{hasLemma}(id_{(1)}, \text{"sunny"})$ answers that the string recognized in the document has as lemma the word *sunny*. This predicate is executed by the 2-D matcher because lemmas depend from the position of a word in a string.

The object descriptor number (11) is translated in term of production rules as follows:

$$\begin{aligned}
 \text{TEMPERATUREPAIR}_0 &\rightarrow \text{TEMPERATURE}_1 \text{ SEPARATOR}_2 \text{ TOK\_RX\_LOWHIGH}_3 \\
 &\text{SEPARATOR}_4 \text{ TEMPERATURE}_5, \\
 &id_{(2)} == \text{blankSequence}, \quad id_{(4)} == \text{blankSequence} \\
 &high_{(0)} := \# \text{max}(value_{(1)}, value_{(5)}), \\
 &low_{(0)} := \# \text{min}(value_{(1)}, value_{(5)}). \\
 \text{TOK\_RX\_LOWHIGH}_0 &\rightarrow \text{'low'}, & id_{(0)} &:= \# \text{newID}(), & value_{(0)} &:= \text{"low"}. \\
 \text{TOK\_RX\_LOWHIGH}_0 &\rightarrow \text{'high'}, & id_{(0)} &:= \# \text{newID}(), & value_{(0)} &:= \text{"high"}. \\
 \text{TOK\_RX\_LOWHIGH}_0 &\rightarrow \text{'min'}, & id_{(0)} &:= \# \text{newID}(), & value_{(0)} &:= \text{"min"}. \\
 \text{TOK\_RX\_LOWHIGH}_0 &\rightarrow \text{'max'}, & id_{(0)} &:= \# \text{newID}(), & value_{(0)} &:= \text{"max"}.
 \end{aligned} \tag{13}$$

This production is labeled as "horizontal" because it express an horizontal sequence of the non-terminals in left side. In particular, *TOK_RX_LOWHIGH* non terminal represents tokens defined by the regular expression (3) translated in standard *AG* productions. Predicates $\# \text{max}(value_{(1)}, value_{(5)})$ and $\# \text{min}(value_{(1)}, value_{(5)})$ are evaluated by the populator because they do not depends from positions.

5.2 2-D Matching

2-D matcher, shown in Figure 7, is composed by three main sub modules: *selector*, *parse tree builder* and *populator*.

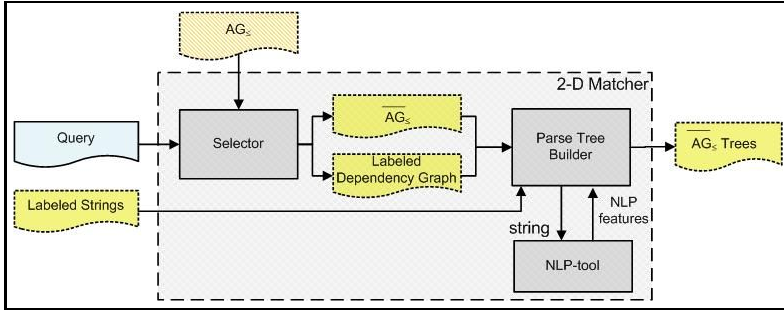


Fig. 7. Module 2-Dimensional Matcher of the SPO System Prototype

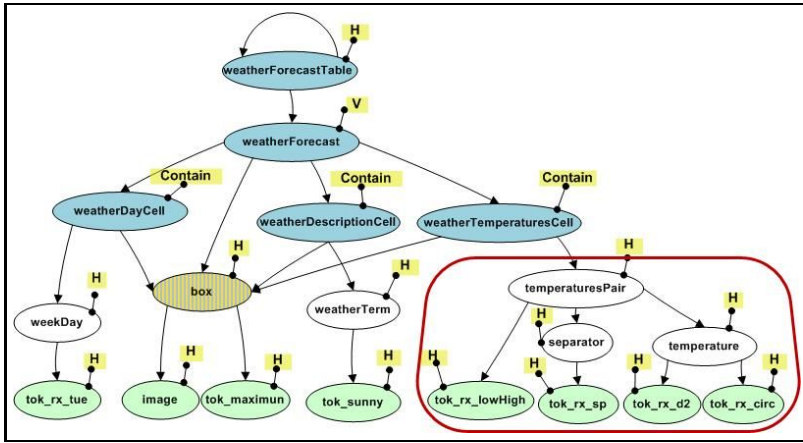


Fig. 8. Labeled Dependency Graph

Selector takes in input the user query and $\mathcal{AG}_\preceq$. It starts from the non-terminal symbol contained in the user query (start concept), explores $\mathcal{AG}_\preceq$ in order to identify the subset of productions $\overline{\mathcal{AG}_\preceq}$ that constitutes the grammar that has as axiom the start concept. Then the $\overline{\mathcal{AG}_\preceq}$ labeled dependency graph ($\mathcal{AG}_\preceq^{LDG}$) is built up. As shown in Figure 8 for the weather forecast example, each node of the $\mathcal{AG}_\preceq^{LDG}$ is equipped by a label that expresses which kind of spatial check must be executed.

Parse tree builder takes as input the document representation generated by the pre-processor, the $\overline{\mathcal{AG}_\preceq}$ and the $\mathcal{AG}_\preceq^{LDG}$. The output of the *parse tree builder* are all the *admissible* parse tree of $\overline{\mathcal{AG}_\preceq}$ obtained by applying a suitable variant of the bottom-up version of Earley's chart parsing algorithm [36] that is able to handle strings contained in the 2-dimensional document representation. Figure 9 shows a sketch of the output $\overline{\mathcal{AG}_\preceq}$ parse tree corresponding to the encompassed area of the $\mathcal{AG}_\preceq^{LDG}$ shown in Figure 8 when the user query is `isX : weatherForecastTable([L]) ?`.

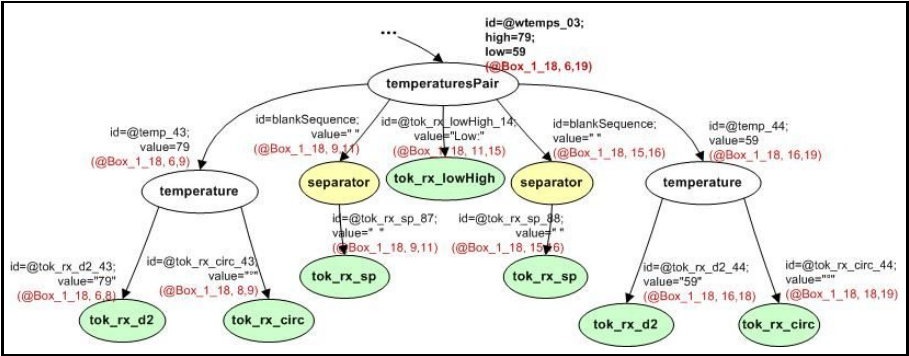


Fig. 9. A sketch of the parse tree resulting from the query $X:weatherForecastTable()$?

5.3 Population

The populator, takes all the admissible parse trees as input. For each of them it evaluates functions that assign values to object attributes (including OIDs) and predicates that are queries on the *SPO* that check if obtained objects are allowed for *SPO*. If the check goes well, obtained objects are added to the initial *SPO* in order to produce the *SPO+*. It is noteworthy that if new objects are added to *SPO* then a subset of the strings in input constitutes a language for $L(\mathcal{AG}_{\prec})$. By considering the user query $X : weatherForecastTable([L])?$ new objects added to the initial *SPO* are shown in the following.

```
@t_001:weatherForecastTable(weathers:[@wf_01,@wf_02,@wf_03,@wf_04,@wf_05]).
@wf_01:weatherForecast(day:@wday_03,descr:@tok_max09_01,hTemp:79,lTemp:59).
...
@wf_05:weatherForecast(day:@wday_07,descr:@tok_max23_01,hTemp:78,lTemp:61).
@wday_03:weekDay(day:"Thursday").
@tok_max_09_01:tok_maximun(value:"Mostly Sunny").
...
@wday_07:weekDay(day:"Monday").
@tok_max_23_01:tok_maximun(value:"Sunny").
```

6 Conclusion and Future Work

In this paper the prototypical implementation of a system for ontology based information extraction from PDF documents has been presented. The *self-populating ontology* approach (*SPO*) which the system is founded on has also been described and its tractability has been proved. The most interesting aspect of *SPOs* is that they are capable to exploit the knowledge in themselves represented for extracting, from PDF unstructured documents, information that, in turn, populates it-self. This way extracted information can be queried and analyzed by means of already existing techniques coming from the database world. *SPOs* empower unstructured information management capabilities of existing applications for creating valuable solutions for enterprises. As

future work we intend to: (i) define a visual approach for descriptors writing; (ii) define an ad-hoc more efficient chart parsing algorithm; (iii) robustly implement the system and extend it to other document format.

References

1. Baumgartner, R., Flesca, S., Gottlob, G.: Visual web information extraction with lixto. In: VLDB 2001: Proceedings of the 27th International Conference on Very Large Data Bases, pp. 119–128. Morgan Kaufmann Publishers Inc., San Francisco (2001)
2. Apt, K.R., Blair, H.A., Walker, A.: Towards a theory of declarative knowledge. In: Foundations of Deductive Databases and Logic Programming, pp. 89–148. Morgan Kaufmann, San Francisco (1988)
3. Abiteboul, S., Hull, R., Vianu, V.: Foundations of Databases. Addison-Wesley, Reading (1995)
4. Knuth, D.E.: Semantics of context-free languages. *Mathematical Systems Theory* 2(2), 127–145 (1968)
5. Kayed, M., Shaalan, K.F.: A survey of web information extraction systems. *IEEE Transactions on Knowledge and Data Engineering* 18(10), 1411–1428 (2006); Member-Chia-Hui Chang and Member-Moheb Ramzy Girgis
6. Laender, A., Ribeiro-Neto, B., Silva, A., Teixeira, J.: A brief survey of web data extraction tools. In: SIGMOD Record, vol. 31 (June 2002)
7. Huck, G., Fankhauser, P., Aberer, K., Neuhold, E.: Jedi: Extracting and synthesizing information from the web. *coopis* 0, 32 (1998)
8. Ludäscher, B., Himmeröder, R., Lausen, G., May, W., Schlepphorst, C.: Managing semistructured data with florid: a deductive object-oriented perspective. *Inf. Syst.* 23(9), 589–613 (1998)
9. Gottlob, G., Koch, C., Baumgartner, R., Herzog, M., Flesca, S.: The lixto data extraction project - back and forth between theory and practice. In: PODS, pp. 1–12 (2004)
10. Papadakis, N.K., Skoutas, D., Raftopoulos, K., Varvarigou, T.A.: Stavies: A system for information extraction from unknown web data sources through automatic web wrapper generation using clustering techniques. *IEEE Transactions on Knowledge and Data Engineering* 17(12), 1638–1652 (2005)
11. Marsh, E., Perzanowski, D.: Muc-7 evaluation of information extraction technology: Overview of results. In: Seventh Message Understanding Conference (MUC-7), pp. 1251–1256 (1998)
12. Ciravegna, F.: Adaptive information extraction from text by rule induction and generalisation. In: IJCAI, pp. 1251–1256 (2001)
13. Etzioni, O., Cafarella, M., Downey, D., Kok, S., Popescu, A.M., Shaked, T., Soderland, S., Weld, D.S., Yates, A.: Web-scale information extraction in knowitall (preliminary results). In: WWW 2004: Proceedings of the 13th international conference on World Wide Web, pp. 100–110. ACM, New York (2004)
14. Banko, M., Cafarella, M.J., Soderland, S., Broadhead, M., Etzioni, O.: Open information extraction from the web. In: Veloso, M.M. (ed.) IJCAI, pp. 2670–2676 (2007)
15. Agichtein, E., Gravano, L.: Snowball: extracting relations from large plain-text collections. In: DL 2000: Proceedings of the fifth ACM conference on Digital libraries, pp. 85–94. ACM, New York (2000)
16. Brin, S.: Extracting patterns and relations from the world wide web. In: WebDB, pp. 172–183 (1998)

17. Pazienza, M.: Information Extraction. Springer, Heidelberg (1997)
18. Flesca, S., Garruzzo, S., Masciari, E., Tagarelli, A.: Wrapping pdf documents exploiting uncertain knowledge. In: Dubois, E., Pohl, K. (eds.) CAiSE 2006. LNCS, vol. 4001, pp. 175–189. Springer, Heidelberg (2006)
19. Hassan, T., Baumgartner, R.: Intelligent text extraction from pdf documents. In: CIMCA/IAWTIC, pp. 2–6 (2005)
20. Carne, J., Ceresna, M., Frölich, O., Gottlob, G., Hassan, T., Herzog, M., Holzinger, W., Krüpl, B.: The lixto project: Exploring new frontiers of web data extraction. In: Bell, D.A., Hong, J. (eds.) BNCOD 2006. LNCS, vol. 4042, pp. 1–15. Springer, Heidelberg (2006)
21. Srihari, S.N., Lam, S.W., Cullen, P.B., Ho, T.K.: Document image analysis and recognition. In: Bourbakis, N. (ed.) Artificial Intelligence Methods and Applications, pp. 590–617. World Scientific Publishing, Singapore (1992)
22. Zanibbi, R., Blostein, D., Cordy, J.R.: A survey of table recognition. IJDAR 7(1), 1–16 (2004)
23. Embley, D.W., Campbell, D.M., Jiang, Y.S., Liddle, S.W., Ng, Y.K., Quass, D., Smith, R.D.: Conceptual-model-based data extraction from multiple-record web pages. Data Knowl. Eng. 31(3), 227–251 (1999)
24. Aitken, J.: Learning Information Extraction Rules: An Inductive Logic Programming approach. In: Proceedings of the 15th European Conference on Artificial Intelligence, pp. 355–359 (2002), <http://citeseer.ist.psu.edu/586553.html>
25. McDowell, L., Cafarella, M.J.: Ontology-driven information extraction with ontosyphon. In: International Semantic Web Conference, pp. 428–444 (2006)
26. Cimiano, P., Völker, J.: Text2onto. In: Montoyo, A., Muñoz, R., Métais, E. (eds.) NLDB 2005. LNCS, vol. 3513, pp. 227–238. Springer, Heidelberg (2005)
27. Maedche, E., Neumann, G., Staab, S.: Bootstrapping an ontology-based information extraction system. In: Studies in Fuzziness and Soft Computing, Intelligent Exploration of the Web. Springer, Heidelberg (2002)
28. Saggion, H., Funk, A., Maynard, D., Bontcheva, K.: Ontology-based information extraction for business intelligence. In: ISWC/ASWC, pp. 843–856 (2007)
29. Wood, M.M., Lydon, S.J., Tablan, V., Maynard, D., Cunningham, H.: Populating a database from parallel texts using ontology-based information extraction. In: Mezziane, F., Métais, E. (eds.) NLDB 2004. LNCS, vol. 3136, pp. 254–264. Springer, Heidelberg (2004)
30. Popov, B., Kiryakov, A., Kirilov, A., Manov, D., Ognyanoff, D., Goranov, M.: Kim - semantic annotation platform. In: International Semantic Web Conference, pp. 834–849 (2003)
31. Hopcroft, J.E., Motwani, R., Ullman, J.D.: Introduction to automata theory, languages, and computation. In: SIGACT News, 2nd edn., vol. 32(1), pp. 60–65 (2001)
32. Ricca, F., Leone, N.: Disjunctive logic programming with types and objects: The dlw+ system. J. Applied Logic 5(3), 545–573 (2007)
33. Kifer, M., Lausen, G., Wu, J.: Logical foundations of object-oriented and frame-based languages. J. ACM 42(4), 741–843 (1995)
34. java.util.regex. Pattern, <http://java.sun.com/j2se/1.5.0/docs>
35. Dantsin, E., Eiter, T., Gottlob, G., Voronkov, A.: Complexity and expressive power of logic programming. In: IEEE Conference on Computational Complexity, pp. 82–101 (1997)
36. Erbach, G.: Bottom-up earley deduction. CoRR cmp-lg/9502004 (1995)

Detecting Dirty Queries during Iterative Development of OWL Based Applications

Ramakrishna Soma¹ and Viktor K. Prasanna²

¹ Computer Science Department, University of Southern California, Los Angeles, CA 90089
rsoma@usc.edu

² Ming Hsieh Department of Electrical Engineering, University of Southern California,
Los Angeles, CA 90089
prasanna@usc.edu

Abstract. Incremental/iterative development is often considered to be the best approach to develop large scale information management applications. In an application using an ontology as a central component at design and/or runtime (*ontology based system*) that is built using this approach, the ontology itself might be constantly modified to satisfy new and changing requirements. Since many other artifacts, e.g., queries, inter-component message formats, code, in the application are dependent on the ontology definition, changes to it necessitate changes to other artifacts and thus might prove to be very expensive. To alleviate this, we address the specific problem of detecting the SPARQL queries that need to be modified due to changes to an OWL ontology (T-Box). Our approach is based on a novel evaluation function for SPARQL queries, which maps a query to the extensions of T-Box elements. This evaluation is used to match the query with the semantics of the changes made to the ontology to determine if the query is *dirty*- i.e., needs to be modified. We present an implementation of the technique, integrated with a popular ontology development environment and provide an evaluation of our technique on a real-life as well as benchmark applications.

1 Introduction

OWL and RDF, the ontology languages proposed as a part of the semantic web standards stack, provide a rich set of data modeling primitives, precise semantics and standard XML based representation mechanisms for knowledge representation. Although originally intended to address the problem of finding and interpreting content on the World Wide Web, these standards have been proposed as an attractive alternative to traditional technologies used for addressing the information and knowledge management problems in large enterprises [7]. As more robust and scalable tools appear in the market, the motivation for applying semantic web technologies in large-scale enterprise wide solution increases steadily. A common use of these technologies is to build *ontology-based* systems [4]. In such systems an ontology, which is a formal model of the problem domain, is a key entity in the design of other system elements like the knowledge bases (KBs), queries to the KBs, messages between the components in the system, and the code itself.

Vast experience in building large-scale information systems, including those based on traditional RDBMS, data warehouses etc., point to the use of an incremental/iterative

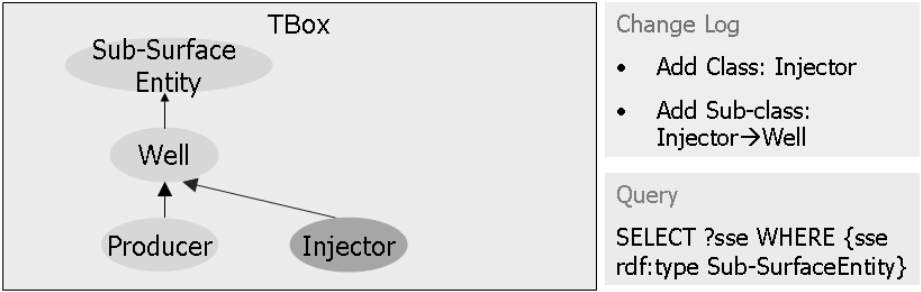


Fig. 1. A ontology change scenario

approach as being the most effective [2,6,13]. In such an approach, the requirements are assumed to be evolving as the system is developed (and used). The software itself is built in phases- with new requirements added at the beginning of each phase and working sub-systems delivered at the end of it. Frequent changes may thus be made to the ontology in order to accommodate the new requirements. Since other artifacts that form a part of the system are closely tied to the ontology, changes to the ontology necessitates changes to them. This may lead to a situation in which changes to an ontology could trigger an expensive chain of changes to other artifacts. Tools that ease the detection and performance of such changes can increase the productivity of the software engineers and hence reduce the cost of building software are thus very important for the success of such a development methodology.

We address a specific case of this broader problem for the class of applications that use OWL for representing the ontologies and SPARQL for the queries. In general, we assume that the ontologies are built ground up- perhaps reusing existing ontologies. Our technique uses the changes made to an OWL TBox to detect which queries need to be modified due to it- we call such queries *dirty* queries. To understand why this is non-trivial consider the following simple scenario shown in Fig. 1.

The original setup consists of a TBox with three classes (Sub-SurfaceEntity, Well, Producer) and a query to retrieve all Sub-SurfaceEntity elements. A new class Injector is then added to the TBox and specified as a sub-class to Well (this is recorded in the change log). A naive change detection algorithm [10] would have compared the entity names from the log (Well, Injector) to those in the query (Sub-SurfaceEntity), and determined that the query need not be modified. However, there could be a knowledge base consistent with the new ontology which contains statements asserting certain elements to be of type Injector. Since these elements/type assertions are not valid in any knowledge base consistent with the original ontology and will be returned as the results of the said query, we consider the query to be dirty. The naive algorithm does not detect this invalid example because the *semantics* are not considered in this approach (in this case the class hierarchy).

Thus our approach goes beyond the simple entity matching by considering the semantics of the ontology, the changes and the queries. A challenge we face in our approach arises because SPARQL is defined as a query language for RDF graphs and its

relationship with OWL ontologies is not very obvious. We address this challenge by defining a novel evaluation function that maps the SPARQL queries to the domain of OWL semantic elements (Sect. 4). Then the semantics of the changes are also defined on the same set of semantic elements (Sect. 5). This enables us to compare and match the the SPARQL queries with that of the changes made to the ontology and hence determine which queries have been effected (Sect. 6). We present an implementation of our technique, which seamlessly integrates with a popular, openly available OWL ontology development environment (Sect. 7). We show an evaluation of our technique for a real-life application for the oil industry and two publicly available OWL benchmark applications (Sect. 8). Finally we describe some related work (Sect. 9) and discussions and conclusions (Sect. 10).

2 Preliminaries

2.1 OWL

The OWL specification is organized into the following three sections [16]:

1. **Abstract Syntax:** In this section, the modeling features of the language are presented using an abstract (non-RDF) syntax.
2. **RDF Mapping:** This section of the specification defines how the constructs in the abstract syntax are mapped into RDF triples. Rules are provided that map valid OWL ontologies to a certain sub-set of the universe of RDF graphs. Thus RDF mappings define a subset of all RDF graphs called well-formed graphs (WF_{OWL}) to represent valid OWL ontologies.
3. **Semantics:** The semantics of the language is presented in a model-theoretic form. The OWL-DL vocabulary is defined over an *OWL Universe* given by the three tuple $\langle IOT, IOC, ALLPROP \rangle$, where:
 - (a) IOT is the set of all *owl:Things*, which defines the set of all individuals.
 - (b) IOC is the set of all *owl:Classes*, comprises all classes of the universe.
 - (c) $ALLPROP$ is the union of set of *owl:ObjectProperty* ($IOOP$), *owl:Datatype Property* ($IODP$), *owl: AnnotationProperty* ($IOAP$) and *OWL:Ontology Property* ($IOXP$).

In addition the following notations are defined in the specification, which we will use in the rest of this paper:

- A mapping function T is defined to map the elements from OWL universe to RDF format.
- An interpretation function $EXT_I: ALLPROP \rightarrow P(R_I \times R_I)$ is used to define the semantics of the properties.
- The notation $CEXT_I$ is used for a mapping from IOC to $P(R_I)$ defining the extension of a class C from IOC .

From now on we will use *ontology* to refer to the TBox, ABox combine as commonly used in OWL terminology.

2.2 SPARQL

SPARQL is the language recommended by the W3C consortium, to query RDF graphs [20]. The language is based on the idea of matching *graph patterns*. We use the following inductive definition of *graph-pattern* [17]

1. A tuple of form $(I \cup L \cup V) \times (I \cup L \cup V) \times (I \cup L \cup V)$ is a graph pattern (also a triple pattern). Where I is the set of IRIs, L set of literals and V is set of variables.
2. If P_1 and P_2 are graph patterns then $P_1 \text{ AND } P_2$, $P_1 \text{ OPT } P_2$, $P_1 \text{ UNION } P_2$ are graph patterns
3. If P is a graph pattern and R is a built-in condition then $P \text{ FILTER } R$ is a valid graph pattern.

The semantics of SPARQL queries are defined using a mapping function μ , which is a partial function $\mu: V \rightarrow \tau$, where V is the set of variables appearing in the query and τ is the triple space. For a set of such mappings Ω , the semantics of the AND ($\bowtie$), UNION and OPT ($\overleftarrow{\bowtie}$) operators are given as follows

$$\Omega_1 \bowtie \Omega_2 = \{\mu_1 \cup \mu_2 \mid \mu_1 \in \Omega_1, \mu_2 \in \Omega_2 \text{ are compatible mappings}^1\}$$

$$\Omega_1 \cup \Omega_2 = \{\mu \mid \mu \in \Omega_1 \text{ or } \mu \in \Omega_2\}$$

$$\Omega_1 \overleftarrow{\bowtie} \Omega_2 = (\Omega_1 \bowtie \Omega_2) \cup (\Omega_1 \setminus \Omega_2)$$

where

$$\Omega_1 \setminus \Omega_2 = \{\mu \in \Omega_1 \mid \forall \mu' \in \Omega_2, \mu \text{ and } \mu' \text{ are not compatible}\}$$

Since SPARQL is defined over RDF graphs, its semantics with respect to OWL is not very easy to understand. In an attempt to clarify this, a subset of SPARQL that can be applied to OWL-DL ontologies is presented in SPARQL-DL [22]. The kind of queries we use in our work are the same as those presented in their work but we also consider graph patterns that SPARQL-DL does not- more specifically the authors consider only conjunctive (AND) queries, where as we consider all SPARQL operators, viz, AND, UNION, OPTIONAL and FILTER. Note that the main goal of [22] is to define a (subset of the) language that can be implemented using current reasoners, whereas the goal of our work is to be able to detect queries that effected by ontology changes.

3 Overview

In Sect. 1 we provided the intuition that a dirty query with respect to two TBoxes is one which can match some triple from an ontology consistent with either one of the TBoxes but not both. We further formalize this notion here, using:

- O is the original ontology.
- C is the set of changes applied to O .
- O' is the new ontology obtained after C is applied to O .
- Q is a SPARQL query. We need to determine if it is dirty or not.

¹ μ_1 and μ_2 are compatible if for a variable x , $(\mu_1(x) = \mu_2(x)) \vee (\mu_1(x) = \phi) \vee (\mu_2(x) = \phi)$

- WF_O is the set of RDF graphs which represent ontologies that are consistent wrt. the statements in O .
- Similarly the set of well-formed OWL graphs wrt. O' is given by $WF_{O'}$.

The *extension of a query* is defined as follows:

Definition: The extension of a query Q containing a graph pattern GP , wrt. an OWL T-Box O (denoted $EXT_O(Q)$ or $EXT_O(GP)$), is defined as the set of all triples that match GP and are valid statements in some RDF graph from WF_O .

A more formal definition of a dirty query is given as:

Definition: A query is said to be dirty wrt. two OWL TBoxes O and O' , if it matches some triple in $WF_{O'} \setminus WF_O$ or $WF_O \setminus WF_{O'}$, i.e., $EXT_O(Q) \cap (WF_{O'} \setminus WF_O \cup WF_O \setminus WF_{O'}) \neq \emptyset$.

Thus to determine if a given query is dirty we find the extension of the given query. Apart from this, we also need to determine and compare it with the set of triples that are present in $WF_{O'} \setminus WF_O \cup WF_O \setminus WF_{O'}$. To do this we consider the changes C and determine the set of triples added, removed or modified in WF_O due to it- we call this as the *semantics of the change*. Thus our overall approach to detect dirty queries consists of the following four steps:

1. Capture ontology change: The changes made to the ontology are logged. Ideally, the change capture tool must be integrated with the ontology design tool, so that the changes are tracked in a manner that is invisible to the ontology engineer. Since many other works [14,15,8] have focused on this aspect of the problem we re-use much of their work and hence do not delve into it.
2. Determine the extension of the query.
3. Determine the semantics of change.
4. Matching: Determine if the ontology change can lead to an inconsistent result for the given queries, by matching the extension of the query with the changed semantics of the ontology.

Each of these steps is detailed in the following sections.

4 Extension of SPARQL Queries

Due to the complexity of consistency checking for DL ontologies [3,5], it is very hard to accurately determine the EXT of a query. In order to alleviate this we use a simplified function called *NEXT*, which determines the set of triples that satisfy a graph pattern by using a necessary (but not sufficient) condition for a triple to be a valid statement in an ontology K_O . From the SPARQL semantics point of view, *NEXT* can be thought of as a function that provides the range for each variable in a query, in the evaluation function Ω . The range itself is defined in terms of the semantic elements of an OWL TBox. In other words, $\Omega: Q \rightarrow T(NEXT(Q))$ - where T is the function to map OWL semantic elements to triples [16]. The semantics presented in the *RDF-Compatible Model-Theoretic Semantics* section of the OWL specification has been used as the basis for defining *NEXT*. We first show how *NEXT* is defined for simple triple patterns and then generalize it to complete queries.

4.1 Triple Patterns

Queries to OWL ontologies can be classified into three types: those that only query A-Box statements (A-Box queries), those that query only T-Box statements (T-Box queries) or those that contain a mix of both (mixed queries) [22]. In the interest of space and as all the queries in the applications/benchmarks we have considered are A-Box queries, we will only present the *NEXT* values for them. A similar method to the one below can be used to create the corresponding tables for the mixed and TBox queries.

Our evaluation of *NEXT* for triple patterns in A-Box queries is based on the following observations:

1. The facts/statements in an OWL A-Box can only be of three kinds: type assertions, or identity assertions (sameAs/ differentFrom) or property values.
2. A triple pattern contains either a constant (URI/literal) or a variable in each of the subject, object and predicate position. Correspondingly, triple patterns are evaluated differently based on whether a constant or a variable occurs in the subject, property, or object position of the query.

We illustrate how *NEXT* values for triple patterns are determined through an example. Consider a triple pattern of the form $?var1 \text{ } constProperty \text{ } ?var2$, where ($?var1$ and $?var2$ are variables and $constProperty$ is a URI). We know that for this triple to match any triple in the A-Box, $constProperty$ must be either $rdf:type$ or $owl:sameAs/ differentFrom$ or some datatype/object property defined in the T-Box.

Consider the case where $constProperty$ is $rdf:type$. The only valid values that can be bound to $?var2$ are the URIs that are defined as a class (or a restriction) in the T-Box. In other words it belongs to the set IOC . The valid values of the subject ($?var1$) is the set of all valid objects in O , i.e., IOT , because every element in IOT can have a type assertion. Therefore the $NEXT(tp)$ is given as $P(IOT \times \{rdf:type\} \times IOC)$ -the power-set of all the triples from $\{IOT \times rdf:type \times IOC\}$.

Note that this is a necessary but not sufficient condition because, although every triple in *NEXT* cannot be proved to be a valid statement with respect to O (not sufficient), but by the definition of these semantic elements, it is necessary for a triple to be in it. As a simple example to illustrate this, consider a TBox with two classes *Man* and *Woman* that are defined to be *disjoint* classes and a triple pattern $?x \text{ } rdf:type \text{ } ?var$. An implication of *Man* and *Woman* being specified as disjoint classes is that an individual cannot be an instance of both these classes i.e. $EXT(tp) \notin \{ \langle aInd \text{ } rdf:type \text{ } Man \rangle \wedge \langle aInd \text{ } rdf:type \text{ } Woman \rangle \}$. However as described above the *NEXT* for the triple pattern is $P(IOT \times \{rdf:type\} \times IOC)$ and does not preclude such a combination of triples from being considered in it.

Using similar analysis, we evaluate the *NEXT* values for other kinds of triple patterns as shown in Table 1.

4.2 Compound Graph Patterns

We now extend this notion to arbitrary graph patterns. Recall from Sect. 2 that a graph pattern Q is recursively defined as $Q = Q1 \text{ AND } Q2 \parallel Q1 \text{ UNION } Q2 \parallel Q1 \text{ OPT } Q2 \parallel$

Table 1. NEXT values for SPARQL queries to OWL A-Boxes

Type	Triple Pattern (TP)	Case	NEXT _O (TP)
1	?var1 ?var2 ?var3	-	$P(IOT \times Prop \times (IOT \cup LV_I))$
2	?var1 ?var2 Value	Value is a URI from the TBox (Class Name)	$P(IOT \times \{rdf:type\} \times \{Value\})$
		Value is an unknown URI	$P(IOT \times \{IOOP \cup owl:sameAs \cup owl:differentFrom\} \times \{Value\})$
		Value is a literal	$P(IOT \times \{IODP\} \times Value)$
3	?var1 Property?var3	Property is rdf:type	$P(IOT \times \{rdf:type\} \times (IOC))$
		Property is owl:sameAs(differentFrom)	$P(IOT \times \{owl:sameAs\} \times IOT)$
		Property is object property i.e. Property $\subset IOOP$	$P\left(\bigcup_{D \in DOM_P} CEXT(D) \times \{Property\} \times \bigcup_{R \in RAN_P} CEXT(R)\right)$
		Property is Data-type property i.e. Property $\subset IODP$	$P\left(\bigcup_{D \in DOM_P} CEXT(D) \times \{Property\} \times LV\right)$
4	?var1 Property Value	Property is rdf:type (and Value $\subset IOC$)	$P(CEXT(C)) \times \{rdf:type\} \times \{Value\}$ $C = T^{-1}(Value)$
		Property is sameAs(differentFrom) (Value is a URI $\subset IOT$)	$P(IOT \times \{owl:sameAs\} \times \{Value\})$
		Property is a object property or data type property (Correspondingly Value is a URI or a literal)	$P(\bigcup_{D \in DOM_P} CEXT(D) \times \{Property\} \times \{Value\})$
5	Value ?var1 ?var2	-	$P(\{Value\} \times ALLPROP \times \{IOT \cup LV \cup IOC\})$
6	Value ?var1 Value2	-	Same as case 2.
7	Value Property ?var2	-	Same as case 3.
8	Value Property Value2	-	TP

Q1 FILTER R. The NEXT value for a query Q is defined based on what the connecting operator is as follows:

1. Consider a simple example of the first case in which both Q1 and Q2 are triple patterns connected through AND: (*?x type A AND ?x type B*). For the variable x to satisfy the first (second) triple pattern, it has to have a value in CEXT (A) (CEXT (B)).

However, due to the AND, x has to be a compatible mapping. Thus the valid values of x are in $(\text{CEXT}(A) \cap \text{CEXT}(B))$. For a variable that only appears in either of the sub-patterns, the *NEXT* does not depend on the other sub-pattern.

2. For a UNION query, the mappings of the variables occurring in Q_1 and Q_2 are completely independent of each other and thus the evaluation can be independently performed.
3. If the two sub-queries are connected by OPT, which has the left join semantics, the variables on the left side (Q_1) is independent of variables in Q_2 . However the extension of the variables in Q_2 , is similar to the queries in an AND query.
4. When expressions are connected using the FILTER operator, the extension is determined as that of Q_1 (we examine two special cases later).

Once the *NEXT* values of the variables in each sub-query are computed, the *NEXT* values of the query can be computed as follows:

For a constant c , $\text{NEXT}(c, Q) = \{c\}$. The extension of the query Q is given as

$$\text{NEXT}(Q) = \bigcup_{tp \in Q} P(\{\text{NEXT}(\text{sub}_{tp}, Q) \times \text{NEXT}(\text{prop}_{tp}, Q) \times \text{NEXT}(\text{obj}_{tp}, Q)\})$$

where tp is each triple pattern in Q and sub_{tp} , prop_{tp} and obj_{tp} represent the constant/variable in the respective position in tp .

This procedure is summarized in algorithm 4.2. The algorithm takes as input a SPARQL query that is fully parenthesized, such that the inner most parenthesis contains the expression that is to be evaluated next. For each of the expressions surrounded by a parenthesis, we maintain the value of the *NEXT* value to which the variable is mapped. When this is modified during the evaluation of the expression in a different sub-query, it is updated to be the new value of the variable, based on the operator semantics described above.

Exceptions: Two exception cases which are treated separately are:

- An interesting use of the FILTER expression is used to express negation in queries [21]. E.g., to query for the complement of instances of a class C one can write a query of the form:

$$(?x \text{ type owl:Thing.OPT}(?a \text{ type } C.\text{Filter}(?x = ?a)).\text{Filter}(!\text{Bound}(a))$$

In this query $?x$ is bound to all objects that are *not* of type C i.e., the *NEXT* value for the variable $?x$ should be assigned as $\text{IOT} \setminus \text{CEXT}(C)$.

- Another interesting case is the use of *isLiteral* condition in a FILTER expression. Consider the triple pattern $?c \text{ type Student. ?c ?p ?val.FILTER}(\text{isLiteral}(\text{val}))$. Without the FILTER clause, we might conclude that the variable p is bound to all properties with domain Student. But since the filter condition specifies that val has to be a literal, p can be restricted to the set of data-type properties with domain C . Note that by not considering the FILTER we obtained the super-set of possible bindings. Therefore any change to one of these properties would have still been detected but some false positives may have been present.

Algorithm 1. Algorithm to compute the NEXT of a compound query**Require:** Fully Parenthesized Query in Normal form Q, Ontology O**Ensure:** NEXT of Q

```

1. while all patterns are not evaluated do
2.    $P \leftarrow$  innermost unevaluated expression in Q
3.   if P is a triple pattern(tp) then
4.      $NEXT(var, P) \leftarrow NEXT_S(var, tp)$ 
5.      $NEXT(var, tp) \leftarrow NEXT_S(var, tp)$ 
6.   else if P is of the form  $(P_1 \text{ AND } P_2)$  then
7.     for each variable v in  $P_1, P_2$  do
8.       if if v occurs in both  $P_1$  and  $P_2$  then
9.          $NEXT(v, P) = NEXT(v, P_1) \cap NEXT(v, P_2)$ 
10.        Update the NEXT of v in  $P_1$  and  $P_2$  as well all sub-patterns it may occur in to  $NEXT(v, P)$ 
11.       else if if v occurs in both  $P_1$  and  $P_2$  then
12.          $NEXT(v, P) = NEXT(v, P_i)$ 
13.       end if
14.     end for
15.   else if P is of the form  $(P_1 \text{ OPT } P_2)$  then
16.     for each variable v in  $P_1, P_2$  do
17.       if v occurs in  $P_1$  then
18.          $NEXT(v, P) = NEXT(v, P_1)$ 
19.       else if v occurs in  $P_2$  then
20.         if v also occurs in  $P_1$  then
21.            $NEXT(v, P_2) = NEXT(v, P_1) \cap NEXT(v, P_2)$ 
22.         else if v occurs only in  $P_2$  then
23.            $NEXT(v, P) = NEXT(v, P_2)$ 
24.         end if
25.       end if
26.     end for
27.   else if P is of the form  $(P_1 \text{ FILTER } R)$  then
28.     for each variable v in  $P_1, P_2$  do
29.        $NEXT(v, P) = NEXT(v, P_1)$ 
30.     end for
31.   else if P is of the form  $(P_1 \text{ UNION } P_2)$  then
32.      $NEXT(v, P) = NEXT(v, P_1)$ 
33.   end if
34. end while
35. return The union of NEXT of each triple pattern in the query

```

5 Semantics of Change

The second step of our change detection process is to map the changes made to the ontology to OWL semantic elements, which will enable the queries and the changes to be compared. We observe that the changes to a TBox can be classified as *lexical changes* and *semantic changes*. Lexical changes represent the changes made to the names(URIs) of OWL classes or properties. Such changes can be handled easily by a simple string match and replace in the query.

Semantic changes are more interesting because they effect one or more OWL semantic elements and need to be carefully considered. They can be further classified as:

- *Extensional changes*: Extensional changes are the changes that modify the extensional sets of a class or a property. E.g., adding an axiom that specifies a class as a super-class of another is an example of this because, the extension of the super-class is now changed to include the instances of the sub-class.
- *Assertional/rule changes*: Assertional changes do not modify the extensions of TBox elements but add additional inference rules or assertions. E.g., specifying a property to be transitive does not change the extension of the domain or range of the property but adds a rule to derive additional triples from asserted ones.
- *Cardinality changes*. The cardinality changes specify constraints on the cardinality of the relationship.

The complete list of semantic changes that can be made to an OWL (Lite) ontology is presented in [8]. We have used it as the basis of capturing and representing the ontology changes in our system. The *semantics of a change*, is the effect of the change to extension of the model is represented as a set of all OWL semantic elements that are effected by the change. By matching this to the extension (*NEXT* value) of the query, we can determine if the query is dirty or not. In table 2, we show some examples of the changes and their semantics.

Table 2. Changes to OWL ontologies and their semantics

Object	Operation	Argument(s)	Semantics of Change
Ontology	Add_Class	Class definition (C)	$IOC \neq IOC'$
Ontology	Remove_Class	Class ID (C)	$IOC \neq IOC', \text{ CEXT}(\text{SC}) \neq \text{CEXT}'(\text{SC})$ $\text{CEXT}(\text{Dom}(\text{P})) \neq \text{CEXT}'(\text{Dom}(\text{P})),$ $\text{CEXT} \neq \text{CEXT}'(\text{Ran}(\text{P})) \forall P \parallel C \in \text{Dom}(\text{P}) \text{ or } \text{Ran}(\text{P})$
Class (C)	Add_SuperClass	Class ID (SC)	$\text{CEXT}(\text{SC}) \neq \text{CEXT}'(\text{SC})$
Class(C)	Remove_SuperClass	Class ID (SC)	$\text{CEXT}(\text{SC}) \neq \text{CEXT}'(\text{SC})$
Property (P)	Set_Transitivity	Property ID	- (Assertional Change)
Property (P)	UnSet_Transitivity	Property ID	- (Assertional Change)

- Example 1: A class C is added to the TBox- IOC ² the set of classes defined in the TBox of the new ontology is different from the original one.
- Example 2: A more interesting case is when a class C is removed from the TBox. Not only is IOC changed as before, but also the extension of the super-classes of C because all the instances of C which were also instances of the super-class(es) in the original TBox are not valid in the modified TBox. The modification also effects the extensions of the classes (restrictions), *intersectionOf* in which the class C appears.

² The OWL spec [16] defines IOC as the set of all OWL classes, here we (ab)use the notation to denote the set of classes defined in the ontology (T-Box).

Finally, the domain and range resp. of the properties in which *C* appears are also modified. Note that a complete DL reasoner (like Pellet or Racer) can be used to fully derive the class subsumption hierarchy, which can then be used to derive the semantics of the change.

- Example 3: In the third example, an OWL axiom which defines a class *C* as subclass of *SC* is added. In this case, the extension of the class *SC* changes. Setting and un-setting transitivity of a property is an example of an assertional change to the ontology as described above.

In the interest of space the entire set of changes and the OWL semantic entities that it changes are not presented here but the interested reader can find it online³.

6 Matching

The matching algorithm is fairly straight forward and is presented in pseudo-code form in Algorithm 2.

Algorithm 2. Algorithm to detect dirty queries for a set of ontology changes

Require: Ontology Change log *L*, Query *Q*

Ensure: Dirty queries

1. Aggregate changes in *L*
 2. **for** each lexical change *l* in *L* **do**
 3. Modify the name of the ontology entity if it appears in it
 4. **end for**
 5. Let *N* ← NEXT of *Q*
 6. Let *C* ← set of semantically changed extensions of *O* due to *L*
 7. **for** each element *n* in *N* **do**
 8. **if** Check if *n* matches any element in *C* **then**
 9. Mark *Q* as dirty
 10. **end if**
 11. **end for**
 12. **return**
-

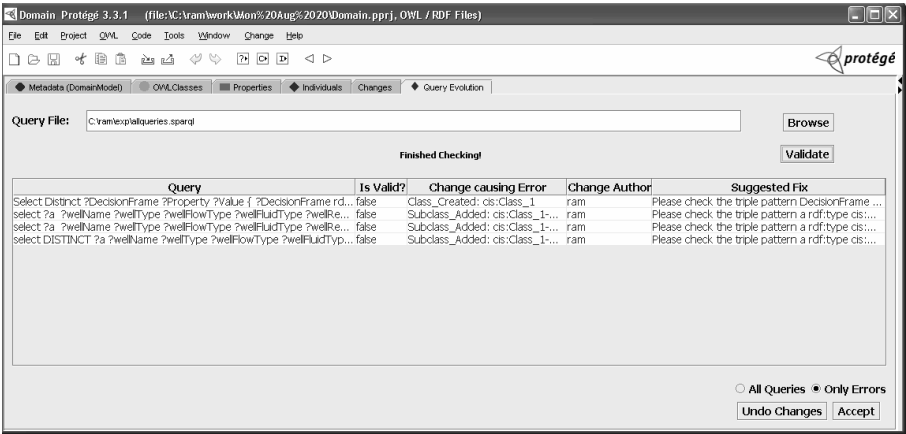
In the first step the log entries are aggregated to eliminate redundant edits. E.g., it is possible that the log contains two entries, one which deletes a class *C* and another which adds the same class *C*. Such changes are commonly observed when the changes are tracked through a user interface and the user often retraces some of the changes made. Clearly these need to be aggregated to conclude that the TBox has not been modified. Then the lexical changes are matched and the query is automatically modified to refer to the new names of the TBox elements. Finally the *NEXT* value of *Q* and the semantic implications of each change in *L* are matched. This is done by comparing the extension or element bound to the subject, object, property position of each triple pattern of *Q*, with extensions modified due to the changes made to the TBox. If any of these sets is effected, then the query is marked as dirty.

³ <http://pgroup.usc.edu/iam/papers/supplemental-material/SemanticsOfChange.pdf>

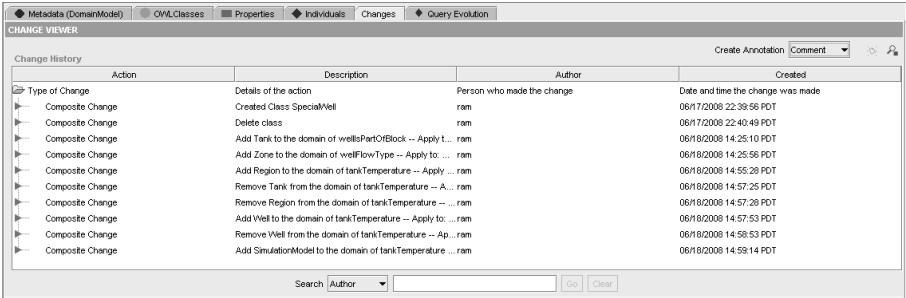
7 Implementation

Our technique has been implemented as a plug-in to the popular and openly available Protégé ontology management tool [19]. Since we have used Protégé for ontology development in our work, providing this service as a plug-in enables a seamless environment to the ontology engineer. Moreover, the Protégé toolkit is equipped with another plug-in that tracks the changes made to an ontology. We utilise this to capture the changes made to the ontology. After the user makes the changes to the ontology in the design tab of the tool, he proceeds to the dirty query detection panel and points to a file containing the SPARQL files for validation. The dirty queries are highlighted and the user can then decide if the changes have to be kept or discarded. The query validation plug-in is shown in Fig. 2(a) and the Protégé change tracking plug-in in Fig. 2(b).

Our implementation of the dirty query detection algorithm is in Java and uses a openly available grammar for SPARQL to create a parser for the queries⁴. Since all



(a) Query validation service implemented as a plug-in to the Protégé toolkit



(b) Change tracking service in Protégé

Fig. 2. Support for incremental development in Protégé environment

⁴ <http://antlr.org/grammar/1200929755392/index.html>

the queries in our applications were stored in a file, the problem of finding the queries was made easy. However, many of the queries were parameterized and we had to pre-process the queries to convert it to a form that was compliant with the specification. E.g., a query for a keyword search to find people with a user specified name is usually parameterized as follows “SELECT ?persons WHERE { ?persons rdf:type Person. ?persons hasName \$userParm\$}”. Here \$userParm\$ is replaced with a dummy string literal to make it into a parseable SPARQL query.

Once the queries are parsed, the *NEXT* values for the queries are evaluated as described in Sect. 4. We have used the Pellet reasoner⁵ for determining the class and property lattices needed for evaluating *NEXT*. The changes tracked by Protégé ontology plug-in are logged in a RDF file. We extract these changes using the Jena API⁶, perform the necessary aggregations and evaluate the semantics of the changes. Again the Pellet reasoner is used here for computing the class lattices etc. Finally the matching is done and all the details of the dirty queries- the triple pattern in the query that is dirty, the TBox change that caused it to be invalidated, the person who made the change and a suggested fix to the problem is displayed to the user.

8 Evaluation

We have evaluated our algorithm on three data-sets: the first two, LUBM [11] and UOBM [12] are two popular OWL knowledge base benchmarks which consist of OWL ontologies related to universities and about 15 queries. The third benchmark we have used (called CiSoft) is based on a real application that we have built for an oil company [23]. The schema for LUBM is relatively simple- it has about 40 classes. Although UOBM has a similar size it ensures that all the OWL constructs are exercised in the TBox. Both these benchmarks have simple queries- each query on an average has about 3 triple patterns in it and they are all conjunctive queries. The TBox of the Cisoft benchmark is larger than the other two (about 100 classes) and the queries we have chosen from the Cisoft application is a set of about 25 queries, and each query has on an average 6 triple patterns. These queries exercise all the SPARQL connectors (AND, OPT, UNION, FILTER).

To evaluate our algorithm, we compare it with two other algorithms. The simpler of these two is the *Entity name* algorithm which checks if the name of the entities modified in the TBox occur in the triple patterns of the query by string matching. If it occurs, then it declares the query to be dirty and if not it declares it clean. The second algorithm called the *Basic Triple Pattern* is a sub-set of our *Complete* algorithm. This algorithm does not consider the connectors between the SPARQL operators i.e., it only implements the rules presented in table 1.

We have used the two standard metrics from information retrieval- *precision* and *recall* for evaluation. Recall is given as the ratio of the no. of dirty queries retrieved by the algorithm to the total no. of dirty queries in the data-set. Precision is given by the ratio of the total no. of dirty queries detected by the algorithm to the total no. of results returned by the algorithm. The results show in Table 3 are the average of 50 runs for

⁵ <http://pellet.owldl.com/>

⁶ <http://jena.sourceforge.net/>

Table 3. Dirty query detection results for three algorithms

Benchmark	Algorithm	Recall	Precision
Cisoft	Complete	1	1
	Basic T.P	1	0.2
	Entity name	0.4	0.45
LUBM	Complete	1	1
	Basic T.P	1	0.6
	Entity name	0.4	0.85
UOBM	Complete	1	1
	Basic T.P	1	0.7
	Entity name	0.25	0.6

each data-set; in each run a small number (< 10) of random changes to the ontology was simulated and the algorithms were then used to detect the dirty queries with respect to those changes.

We see that the Basic TP algorithm has a recall of 1 i.e., always returns all the dirty queries in the data-set but it also returns a number of false positives (low precision). Since the results returned by BTP is always a super-set of the complete algorithm- the recall is always 1. To understand why a low precision is observed for BTP (especially for the Cisoft data-set), consider a query of the form *?a rdf:type Student. ?a ?prop ?value*. Since the algorithm considers each triple pattern in isolation, it infers that every valid triple in the ontology will match the second triple pattern (*?a ?prop ?value*). Therefore any ontology change will invalidate the query. However, this is incorrect because the first triple pattern ensures that only triples which refer to instances of *Student* will match the query and therefore only changes related to the OWL class *Student* will invalidate the query. The LUBM and UOBM queries do not have many triple patterns of this form, thus the precision of BTP for these data-sets is higher.

The entity name algorithm does not always pick out the dirty queries (recall < 1). The main shortcoming of this algorithm is that, it cannot detect the ontology changes that might affect values that might be bound to a variable.

9 Related Work

Much work has been done in the general area of ontology change management [8,9,24]. Most of these works deal with the semantic web applications in which ontologies are imported or built in a distributed setting. In such a setting, the main challenge is to ensure that the ontologies are kept consistent with each other. In our work, we address the problem of keeping the SPARQL queries consistent with OWL ontologies. Although, some aspects of the problem- e.g., the set of changes that can be made to an OWL ontology are the same, our key contributions are in defining the notion of dirty queries and the evaluation function which maps queries and (implications) of ontology changes onto the OWL semantic elements, which makes it possible to compare them to decide if the query is invalidated.

In [24], although the authors define evolution quite broadly as *timely adaptation of an ontology to the arisen changes and the consistent propagation of these changes to dependent artifacts*, they do not address the issue of keeping queries based on ontology definitions consistent with the new ontology. The authors do define a generic four stage change handling mechanism- (change) representation, semantics of change, propagation, and implementation, which is applicable to any artifact that depends on the ontology. Our own four step process is somewhat similar to this.

An important sub-problem in the ontology evolution problem is the change detection problem. Various approaches have proposed in literature to address this problem. Most of these address the problem setting of distributed ontology development and thus provide sophisticated mechanisms to compare and thus find the differences between two ontologies [15,18]. On the other hand we assume a more centralized setting in which we assume that the ontology engineers modify the same copy of the ontology definition file. We have used an existing plug-in developed for the Protégé toolkit [14], which tracks the changes made to the ontology.

An important artifact in a ontology based system is the knowledge base. In [25], the authors address the problem of efficiently maintaining a knowledge base when the ontology (logic program) changes. Similar to the work of view maintenance in the datalog community, the authors use the delta program to efficiently detect the data tuples that need to be added or deleted from the existing data store. This is an important piece of work addressing the needs of the class of applications that we target, and is complementary to our work.

In the area of software engineering, the idea of agile database [1] addresses the similar problem of developing software in an environment in which the database schema is constantly evolving. The authors present various techniques and best practices to facilitate efficient development of software in such a dynamic methodology. Unlike our work, the authors however, do not address the problem of detecting the queries that are affected by the changes to the schema.

10 Discussion and Conclusions

We have addressed a problem seen in the context of OWL based application development using an iterative methodology. In such a setting as the (OWL) TBox is frequently modified, it becomes necessary to check if the queries used in the application also need to be modified. The key element of our technique is a SPARQL evaluation function that is used to map the query to OWL semantic elements. This is then matched with the semantics of the changes to the TBox to detect dirty queries. Our evaluation shows that simpler approaches might not be enough to effectively detect such queries.

Although originally intended to detect *dirty* queries we have found that our evaluation function can be used as a quick way to check if a SPARQL query is *semantically incorrect* with respect to an ontology. Semantically incorrect queries are those that do not match any valid graph for the ontology- i.e., always return an empty result-set. For such queries, our evaluation function will not find a satisfactory binding for all the triple patterns.

An assumption we make in our work is that all the queries to the A-Box are available for checking when changes are made. In many application development scenarios this may not be feasible. Therefore whenever possible it is a good practice for an application development team working in such an agile methodology to structure the application so that the queries used in the application can be easily extracted for these kinds of analysis. If it is not possible to do so, one option for an ontology engineer is to use the OWL built-in mechanism to mark the changed entity as *deprecated*, and phase it out after a sufficiently long time.

Often times, the queries are dynamically generated based on some user input. In such cases it might be harder to check the validity of the queries. However, it might still be possible to detect dirty queries because, such queries are generally written as parameterized templates which are customized to the user input. If such templates are made available, it might still be possible to check if they are valid.

Acknowledgment

This research was funded by CiSoft (Center for Interactive Smart Oilfield Technologies), a Center of Research Excellence and Academic Training and a joint venture between the University of Southern California and Chevron. We are grateful to the management of CiSoft and Chevron for permission to present this work.

References

1. Ambler, S.: Agile Database Techniques: Effective Strategies for the Agile Software Developer. John Wiley and Sons, Chichester (2003)
2. Boehm, B.W.: A spiral model of software development and enhancement. IEEE Computer 21(5), 61–72 (1988)
3. Donini, F.M.: Complexity of reasoning. In: Description Logic Handbook, pp. 96–136 (2007)
4. Guarino, N.: Formal ontology and information systems. In: 1st International Conference on Formal Ontologies in Information Systems, FOIS 1998 (1998)
5. Horrocks, I., Sattler, U.: A tableau decision procedure for *SHOIQ*. J. of Automated Reasoning 39(3), 249–276 (2007)
6. Inmon, W.H.: Building the Data Warehouse. John Wiley and sons Inc., Chichester (2002)
7. Jorge, C., Martin, H., Miltiadis, L. (eds.): The Semantic Web. Real-world Applications from Industry. Springer, Heidelberg (2007)
8. Klein, M.: Change Management for Distributed Ontologies. Ph.D thesis, Vrije Universiteit Amsterdam (August 2004)
9. Leenheer, P.D., Mens, T.: Ontology evolution: State of the art and future directions. In: Hepp, M., Leenheer, P.D., Moor, A.D., Sure, Y. (eds.) Ontology Management. Springer, Heidelberg (2007)
10. Liang, Y.: Enabling active ontology change management within semantic web-based applications. Technical report, School of Electronics and Computer Science, University of Southampton (2006)
11. Lehigh university benchmark, <http://swat.cse.lehigh.edu/projects/lubm/>
12. Ma, L., Yang, Y., Qiu, Z., Xie, G., Pan, Y., Liu, S.: Towards a complete owl ontology benchmark. In: Sure, Y., Domingue, J. (eds.) ESWC 2006. LNCS, vol. 4011, pp. 125–139. Springer, Heidelberg (2006)

13. Martin, R.C.: *Agile Software Development: Principles, Patterns, and Practices*. Prentice Hall PTR, Upper Saddle River (2003)
14. Noy, N.F., Kunnatur, S., Klein, M., Musen, M.A.: Tracking changes during ontology evolution. In: *Third International Conference on the Semantic Web* (2004)
15. Noy, N.F., Musen, M.A.: Promptdiff: A fixed-point algorithm for comparing ontology versions. In: *Eighteenth National Conference on Artificial Intelligence (AAAI)*,
16. Patel-Schneider, P.F., Horrocks, I.: Owl web ontology language semantics and abstract syntax, w3c recommendation, <http://www.w3.org/tr/owl-semantics/>
17. Perez, J., Arenas, M., Gutierrez, C.: Semantics and complexity of sparql. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L.M. (eds.) *ISWC 2006*. LNCS, vol. 4273, pp. 30–43. Springer, Heidelberg (2006)
18. Plessers, P., Troyer, O.D., Casteleyn, S.: Understanding ontology evolution: A change detection approach. *Web Semantics: Science, Services and Agents on the World Wide Web* 5 (2007)
19. The protege ontology editor and knowledge acquisition system, <http://protege.stanford.edu/>
20. Prudhommeaux, E., Seaborne, A.: Sparql query language for rdf, w3c recommendation, <http://www.w3.org/tr/2008/rec-rdf-sparql-query-20080115/>
21. Schenk, S., Staab, S.: Networked graphs: a declarative mechanism for sparql rules, sparql views and rdf data integration on the web. In: *WWW 2008: Proceeding of the 17th international conference on World Wide Web*, pp. 585–594. ACM, New York (2008)
22. Sirin, E., Parsia, B.: Sparql-dl: Sparql query for owl-dl. In: *3rd OWL Experiences and Directions Workshop (OWLED)* (2007)
23. Soma, R., Bakshi, A., Prasanna, V., Sie, W.D., Bourgeois, B.: Semantic web technologies for smart oil field applications. In: *2nd SPE Intelligent Energy Conference and Exhibition* (February 2008)
24. Stojanovic, L.: *Methods and Tools for Ontology Evolution*. Ph.D thesis, University of Karlsruhe (2004)
25. Volz, R., Staab, S., Motik, B.: Incremental maintenance of materialized ontologies. In: Meersman, R., Tari, Z., Schmidt, D.C. (eds.) *CoopIS 2003, DOA 2003, and ODBASE 2003*. LNCS, vol. 2888, pp. 707–724. Springer, Heidelberg (2003)

Reusing the SIOC Ontology to Facilitate Semantic CWE Interoperability

Deirdre Lee, Vassilios Peristeras, and Nikos Loutas

Digital Enterprise Research Institute (DERI),
National University of Ireland Galway (NUIG),
Galway, Ireland
{firstname.lastname}@deri.org

Abstract. Due to the increasingly large volumes of data that today's businesses handle, effective data management is a key issue to be addressed. An important aspect of data management is data interoperability, or the ability of information systems to exchange data and knowledge. Collaborative Working Environments (CWEs) are fundamental tools for supporting data interoperability within the scope of a particular project or within a team of eProfessionals collaborating together. However CWE users may partake in multiple projects hosted on different platforms, or a particular project may be distributed over multiple CWE platforms. Therefore, it would be advantageous to be able to share and combine data from several independent CWEs. Currently, CWE platforms remain informational islands, with data expressed in a proprietary format. To address this data heterogeneity challenge and enable CWE interoperation, data must be structured in a semantically interpretable format. In this paper, we propose to reuse Semantically Interlinked Online Community (SIOC) to facilitate semantic CWE interoperability. Once proprietary CWE data is annotated with the SIOC ontology, it becomes interpretable by external CWEs. Based on this, the CWE Interoperability Architecture has been designed. Following this architecture, we developed the SIOC4CWE Toolkit, which allows the exporting, importing, and utilization of SIOC data.

Keywords: data interoperability, ontology, semantics, collaborative working environment (CWE).

1 Introduction

Data is one of the most valuable resources to businesses, institutes, and communities. In a time where copious amounts of data are accessible in databases, on internal networks, and online, data management and integration are emerging as key issues to be addressed. Park and Ram claim that data interoperability is the most critical issue facing businesses that need to access information from multiple information systems [1]. They contend that data management in a heterogeneous environment has been one of the most challenging problems for businesses because many a firm's valuable information lies in multiple legacy systems and is stored in multiple, conflicting formats. Collaborative Working Environments (CWEs) address the need for data

interoperability within the scope of a particular project or within a team of eProfessionals collaborating together. An eProfessional is a professional worker, whose work relies on being part of knowledge networks and being involved in distributed cooperation processes [2]. CWE platforms, such as SAP NetWeaver, BSCW, Business Collaborator (BC), and Microsoft SharePoint, provide a common workspace for a large number of independent users and applications, e.g. project and document management applications, calendars, forums, mailing lists, etc [3]. However, CWEs remain informational islands that cannot communicate with data originating in other systems or databases, or even with data originating in other projects or CWEs. In a business, eProfessionals typically do not work exclusively on one project: they participate in different projects on different workspaces. Additionally they maintain personal information spaces, where they keep track of overall planning and to-do items, as well as private information. The assumption that all collaborating eProfessionals will use and prefer to use the same CWE platform is not a realistic assumption. Therefore, in order to accommodate eProfessionals participating in different projects, with different requirements, and different personal preferences, it is necessary to facilitate the interoperation of data from several independent CWEs. Yet such data integration is currently a manual and laborious task. As a result of integrating data from multiple CWEs, third-party applications that provide higher-level functionality based on the correlation of the data may also be developed.

Research has shown that intelligent data integration and sharing requires explicit representations of information semantics. It is here that ontologies play a crucial role: shared formalized models of a particular domain, whose intended semantics is both shared between different parties and machine-interpretable [4]. Uschold and Gruninger [5] recognize that without addressing the reality of semantic heterogeneity, full seamless connectivity between systems will not be achieved. Gruber [6] defines an ontology to be a formal specification of a conceptualisation, meaning, ontologies provide a formal specification of all entities in a domain and the relationships between those entities. He affirmed that in order to support the sharing and reuse of formally represented knowledge among Artificial Intelligence systems, it is useful to define the common vocabulary in which shared knowledge is represented. Although a CWE may not be considered an Artificial Intelligence system, the premise of defining a common vocabulary is still relevant. For example, in the realm of relational databases, data integration is addressed using a mediated schema; a set of relations that is designed for a specific data integration application, and contains the common aspects of the domain under consideration [7]. Thus, if systems structure their data in terms of a common schema, the data is universally interpretable. We propose to use ontologies in the integration of legacy CWEs. More specifically, in this work we reuse the Semantically Interlinked Online Community (SIOC) ontology, as a common meta-language in CWE to address semantic interoperability issues, as shown in Fig. 1.



Fig. 1. Using SIOC for CWE Interoperability

The remainder of the paper is organized as follows: Section 2 illustrates a motivating scenario, which demonstrates the need of semantic CWE interoperability for eProfessionals. Section 3 introduces the area of data interoperability and presents the state of the art in using ontologies to enable semantic interoperability. Section 4 explains the motivation behind reusing the SIOC ontology to facilitate semantic CWE interoperability, as opposed to creating a new ontology. Section 5 describes SIOC in detail, outlining its core ontology concepts and relationships. In section 6, the CWE Interoperability Architecture is presented, including the SIOC4CWE Toolkit, which provides many tools and prototypes that implement the architecture. Section 7 contains an evaluation of the research presented in this paper. Finally, section 8 summarizes the main contributions of this paper and proposes some future research directions.

2 Motivating Scenario

Here we present a simple motivating scenario for our work. Anne is working on two EU research projects: Ecospace and SemanticSpace. Each project uses a different CWE platform to help partners collaborate efficiently: Ecospace uses BSCW, while SemanticSpace uses BC. Both projects are concerned with improving how eProfessionals collaborate, so there are common themes between them. In addition, like Anne, some participants are active in both projects. As the projects progress, items, such as documents, calendar events, discussions, etc., are created on both shared workspaces. It becomes increasingly difficult for Anne to keep track of and search relevant information on both platforms. For example, documents relating to a common conference may be uploaded to both platforms. In order to review these documents easily, Anne wishes to see all documents together in one CWE platform. Another example is if Anne wants to find all deliverables in both the Ecospace and SemanticSpace projects with the keyword “web 2.0” in the title, or to find out calendar events that are related to her, or what has been uploaded in both systems during the last two days. In all cases she must log on to both shared workspaces individually, perform a search, and then combine the search results manually. This may be acceptable with only two shared workspaces, however with multiple workspaces it quickly becomes a complex, time-intensive, and unmanageable task.

3 State of the Art

The European Commission define ‘interoperability’ as the ability of Information and Communication Technology (ICT) systems and of the business processes they support to exchange data and to enable the sharing of information and knowledge [8]. They propose that three aspects of interoperability need to be considered: organizational, technical, and semantic. Organizational interoperability is concerned with defining business goals, modeling business processes, and bringing about the collaboration of administrations that wish to exchange information but may have different internal structures and processes. Technical interoperability covers the

technical issues of linking computer systems and service, e.g. open interfaces, interconnection services, and protocols. Semantic interoperability is concerned with ensuring that the precise meaning of exchanged information is understandable by communicating systems.

At a technical level, CWEs may interoperate using a Service Oriented Architecture (SOA) approach. SOA promotes loose coupling between software components so that they can be reused. Applications in SOA are built based on services. This is done by dividing the business process in atomic services, which have their own logic. Web service technologies, as a SOA implementation, have been used as the gluing communication protocol of different groupware systems to facilitate CWE interoperation [9, 10]. As the issue of technical interoperability has been addressed by Service Oriented Architecture (SOA) related approaches, the research focus has now shifted to overcoming interoperability at the semantic level. We propose to address semantic interoperability issues in various CWE platforms through the use of ontologies. Ontologies, in the form of logical domain theories and their knowledge bases, offer the richest representations of machine-interpretable semantics for systems and databases in the loosely coupled world, thus ensuring greater semantic interoperability and integration [11].

There are three main approaches to using ontologies for data integration: single-ontology, multiple-ontology, and hybrid [4]. Single-ontology approaches use one global-ontology to provide a shared vocabulary for the specification of the semantics. The main problem with this approach is reaching a general consensus from all participants of what concepts should be included in the ontology and how these concepts relate to each other. In multiple-ontology approaches, each information source is described by its own ontology. However this leads to a higher-level interoperation problem, in that how ontologies relate to each other. To overcome this problem, an additional representation formalism defining ontology mappings must be provided. The construction of such mappings between ontologies is currently a core topic in the Semantic Web community [12-16]. The goal of ontology mapping is to generate correspondences between the concepts from different but related ontologies. In most cases, formal mapping rules with clear semantics need to be generated for integrating information systems. However, research in discovering and representing semantic mappings is still in very preliminary stages and many challenges remain, e.g. the ideal choice of a mapping language that carefully balances expressivity with scalability and what kind of interaction between system and domain experts should be supported [17]. To overcome the drawbacks of the single- or multiple-ontology approaches, hybrid approaches were developed. A hybrid approach entails aspects from both single-ontology and multiple-ontology approaches. To enable data integration a common shared ontology is used. However to allow for data structure discrepancies, data sources may extend the global ontology. Adopting an extended ontology provides a richer and more detailed specification of a domain. Moreover basic data interoperation is still possible using only the global ontology [4].

In this paper, it is proposed to use the SIOC ontology as the core ontology for data integration between CWEs; however, there is scope to extend the ontology with a CWE-specific module. This would enrich the data interoperation between CWEs.

4 Reusing the SIOC Ontology

In this section, we will explain the motivation behind reusing the SIOC ontology to facilitate semantic CWE interoperability. Each CWE platform models their CWE concepts in a proprietary format. Therefore, our first approach was to consider a multiple-ontology approach, with each platform associated with its own ontology. However, in order for platforms to interoperate, all ontologies would have to be mapped to each other, introducing a lot of ontology-engineering overhead, as discussed in the State of the Art section. Also, although CWE platforms

are unique, there is a lot of overlap between concepts, which justifies a common ontology. Therefore a single ontology, which would provide a shared vocabulary for the specification of the semantics, was then considered. In collaboration with CWE-platform engineers from BSCW, BC, and NetWeaver, a common ontology was designed, an extract of which can be seen in Fig 2. On closer examination of the CWE general ontology, many similarities in the data structure of online community sites and CWEs were found. For example, the relationship between a post and a forum on an online community site is comparable to the relationship between a document and a folder in a CWE. Although the SIOC ontology is designed for online community sites, we investigated if it would be possible to reuse the same ontology to achieve semantic interoperability among legacy CWE platforms. Ontology reuse is popular in semantic development, as it promotes knowledge sharing and interoperability. In an ontology engineering study carried out by Bontas-Simperl and Tempich, they found that in over 50% of the cases the final ontology was built with the help of other ontological sources [18]. Other advantages of reusing the SIOC ontology are:

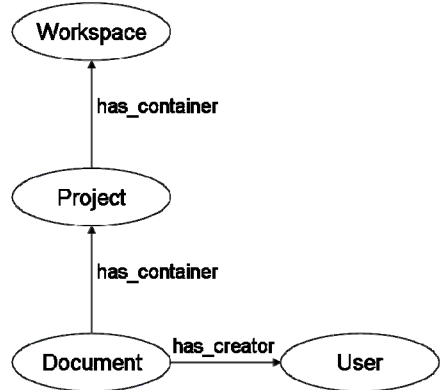


Fig. 2. Extract from the CWE General Ontology

- **Adoption:** It is difficult to persuade a community to adopt a new ontology. The SIOC ontology is widely accepted and a W3C Member Submission.
- **Bridging information on the Web and closed Information Systems:** SIOC gives the unique opportunity to link legacy data locked in CWE platforms with Web data
- **Lack of an already exiting and ready to be used ontology for CWE concepts**
- **Online community sites and CWEs share many integration issues:** For example, in the online community domain a user may partake in many discussions, forums, and blogs across several sites. In the CWE domain, an eProfessional may partake in many projects located on separate CWEs

There are, however, CWE-specific concepts that are not currently defined in the SIOC ontology e.g. workflow, project, task. We therefore propose to use a hybrid-ontology approach to modeling CWE platform data, which facilitates the extension of the core SIOC concepts. The SIOC ontology is used as the base ontology for modelling CWEs. However, there is also the possibility to extend the base ontology with a CWE-domain module for CWE- specific concepts. CWE platforms may opt-in or opt-out to the use of the CWE-domain module and, either way, will still be interoperable based on the core SIOC concepts. This offers a scalable and manageable approach to the semantic integration of CWEs.

5 Semantically-Interlinked Online Communities (SIOC)

As our approach incorporates SIOC, we briefly present this technology. The SIOC initiative aims at connecting online community sites, such as blogs, forums, wikis, and mailing lists. It consists of the SIOC ontology, an open-standard machine readable format for expressing the information contained both explicitly and implicitly in internet discussion methods; SIOC tools for leveraging SIOC data, e.g. metadata exporters, and storage and browsing systems; and the SIOC community, an expanding network of people using SIOC [19].

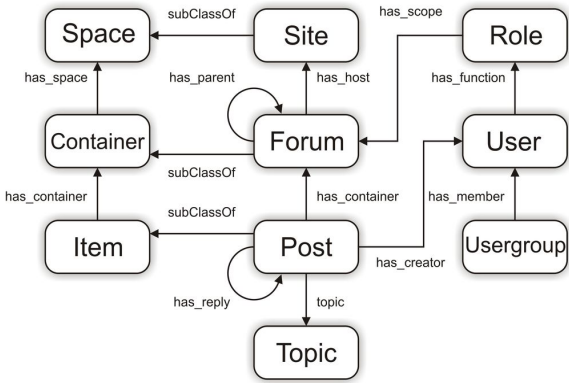


Fig. 3. Main Concepts and Properties in the SIOC Core Ontology

The SIOC ontology was recently published as a W3C Member Submission, which was submitted by 16 organizations [20]. The ontology is expressed in RDF and consists of the SIOC Core ontology (consisting of 11 concepts and 53 properties) and two ontology modules: SIOC Types and SIOC Services. The SIOC Core ontology defines the main concepts and properties required to describe information from online communities on the Semantic Web. The main terms in the SIOC Core ontology are shown in Fig. 3. The basic concepts in SIOC have been chosen to be as generic as possible, thereby enabling many different kinds of user-generated content to be described. The high-level concepts *sioc:Space*, *sioc:Container* and *sioc:Item* are at

the top of the SIOC concept hierarchy, and most of the other SIOC concepts are subclasses of these. A data space (*sioc:Space*) is a place where data resides, such as a website, personal desktop, shared file space, etc. It can be the location for a set of *Container(s)* of content *Item(s)*. Subclasses of *Container* can be used to further specify typed groupings of *Item(s)* in online communities. The concept *sioc:Item* is a high-level concept for content items and is used for describing user-created content. Properties defined in SIOC allow relations between objects, and attributes of these objects, to be described. SIOC modules are used to extend the available terms and to avoid making the SIOC Core Ontology too complex and unreadable. The SIOC Types module defines more specific subclasses of the SIOC Core concepts, which can be used to describe the structure and various types of content of social sites. The SIOC Services ontology module allows one to indicate that a web service is associated with (located on) a *sioc:Site* or a part of it. Export tools produce SIOC RDF data from online community sites. An important property of these SIOC exporters is information contained within a site available in RDF. By using RDF, SIOC gains a powerful extensibility mechanism, allowing SIOC-based descriptions to be mixed with claims made in any other RDF vocabulary, e.g. FOAF. As social sites begin to generate more SIOC data, this information can be reused, e.g. to provide better tools for finding related information across community sites or to transfer rich information about content items between online community sites.

6 CWE Interoperability Architecture

The CWE Interoperability Architecture provides a middleware that enables multiple independent CWE-platforms and third-party applications to share and correlate data, based on SIOC. Together with engineers and developers from many of the main CWE-platform providers, e.g. SAP NetWeaver, BSCW, and BC, the SIOC4CWE Toolkit has been developed. It provides many tools and prototypes that implement the CWE Interoperability Architecture and may be used in its evaluation.

Fig. 4 shows the CWE Interoperability Architecture. Firstly, proprietary CWE data is exported as SIOC RDF data. The SIOC data is then imported by other CWEs or by third-party applications. Finally the SIOC data is utilized accordingly. SIOC Exporters are responsible for translating CWE platform-specific data into SIOC RDF

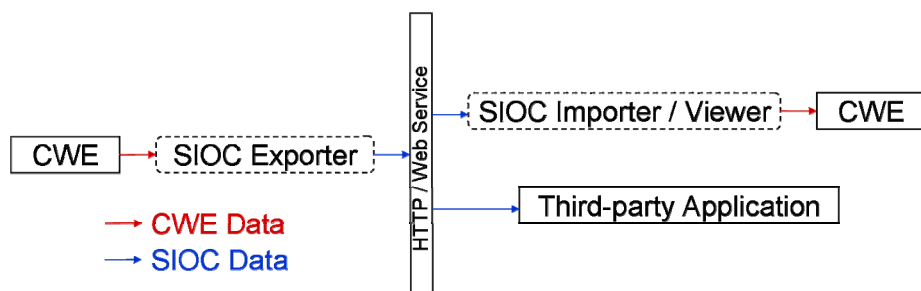


Fig. 4. CWE Interoperability Architecture

data. This SIOC data is made available to external systems through HTTP or web services (WSDL) so that it may be imported by other CWEs or third-party applications. If a CWE wishes to use the SIOC data, it must first convert the SIOC data into CWE proprietary data, so that it can be presented to the user in a homogenous fashion with the local data. A SIOC Importer/Viewer is responsible for this task. A third-party application may use SIOC data as it is, as it usually provides functionality based on the correlation of SIOC data from a variety of sources, e.g. a SIOC browsing tool, which allows a user to browse data from multiple CWE platforms.

6.1 CWE-SIOC Concept Mappings

In order to interoperate with outside systems, a CWE must expose its data in a semantically interpretable format. The SIOC ontology is used as a CWE meta-language to describe CWE data in a homogenous format. Before translating proprietary CWE data into SIOC RDF data, concepts that exist in a specific CWE domain must be mapped to concepts in the SIOC ontology (e.g. user, post, and document). The CWE-platform provider is responsible for defining the mappings between the existing CWE model and the SIOC ontology. Table 1 shows a sample of mappings from existing CWE-platform concepts, namely BSCW concepts, BC concepts, and SAP NetWeaver Portal concepts, to SIOC concepts. The SIOC Types module is used to define more specific subclasses of the SIOC Core concepts, which are necessary to fully capture all CWE concepts. This approach is simple and

Table 1. Concept Mapping from BSCW, BC, and NetWeaver to SIOC

BSCW	BC	NetWeaver	SIOC
BSCW Site	BC Host Server (identified by domain)	Collaboration Portal	sioc:Site
Workspace	Workspace/Project	Workspace/Project	sioc:Container
-	Members List	Team	sioc:UserGroup
Document	Document	Document	sioc:Item
Note	-	Post	sioc:Post
URL	-	URL	annotea:Bookmark
Folder	Folder	Folder	sioc:Container
-	Collection	Page	sioc:Container
Project	-	-	sioctype:project Directory
User	User	User	sioc:User, foaf:Person

scalable, as all that is required to integrate a new or legacy CWE is a set of concept mappings from CWE-specific concepts to existing concepts in the SIOC ontology.

6.2 Exporting CWE Data as SIOC Data

Based on the CWE-SIOC conceptual mappings, CWE providers have developed SIOC exporters to annotate the internal CWE data and export it as SIOC RDF data. For example, Fig. 5 shows a BC folder, as it would appear in the BC workspace.



Fig. 5. BC Folder from the BC CWE Platform

This folder may be converted to the following SIOC data fragment:

```
<rdf:Description
  rdf:about="http://namespace.groupbc.com/objects/ECOSPACE/25000">
  <dc:title>Galway images</dc:title>
  <sioc:id>25000</sioc:id>
  <sioc:has_owner
    rdf:resource="http://namespace.groupbc.com/objects/ECOSPACE/24866"/>
  <bc:public>1</bc:public>
  <sioc:has_creator
    rdf:resource="http://namespace.groupbc.com/objects/ECOSPACE/24866"/>
  <sioc:has_space
    rdf:resource="http://namespace.groupbc.com/objects/ECOSPACE/1077"/>
  <rdfs:label>Galway images</rdfs:label>
  <dc:description>Images from Galway</dc:description>
  <bc:class>Folder</bc:class>
  <dcterms:created>2008-03-05 11:47:50</dcterms:created>
  <rdf:type rdf:resource="http://rdfs.org/sioc/ns#Container"/>
  <rdfs:seeAlso
    rdf:resource="http://ecospace.withbc.com/bc/bc.cgi/0/2500?op=sioc"/>
  <bc:type_id>45</bc:type_id>
  <sioc:has_modifier
    rdf:resource="http://namespace.groupbc.com/objects/ECOSPACE/24866"/>
  <sioc:has_parent
    rdf:resource="http://namespace.groupbc.com/objects/ECOSPACE/1077"/>
  <dcterms:modified>2008-03-05 11:47:50</dcterms:modified>
</rdf:Description>
```

Once exported, the SIOC RDF data may be made available to the outside world via HTTP or web services. In our case, a web service has been developed, which exposes the contents of a CWE workspace as SIOC data. Also, documents and folders may be accessed, added, deleted, renamed, or replaced remotely from the CWE via web services.

6.3 Utilizing SIOC Data

The exported data, expressed in terms of the SIOC ontology concepts, may be:

- imported into another CWE platform so that it can be browsed locally, or
- it may be imported into a third-party application so that it can be merged and correlated with other SIOC data.

6.3.1 SIOC Importer/Viewer

Importing remote SIOC data into a CWE workspace allows external data to be integrated with local CWE data. A SIOC Importer/Viewer has been developed for the BC platform to investigate the practicality of using SIOC as an interchange data format to facilitate the integration of CWEs. Based on reverse mappings, the SIOC Importer/Viewer translates SIOC data into CWE platform-specific data. The BC user sees the remote data within the BC user interface as if the data was present on the local BC server. This allows folders, documents, and other data from a remote SIOC-enabled system to be accessible by other BC workspace users using the standard BC interface. This is somewhat analogous to a user mounting a read-only shared network drive using NFS where a remote folder may be viewed by an application running locally. The BC SIOC Importer/Viewer currently uses web services, as described in section 6.2, to access remote SIOC RDF data. Fig. 6 shows an example of external CWE folders (denoted by the  icon), which have been imported into a local CWE project on the BC platform.

☒ Type ↓	Name	Object Type	Description	Events	Creator	Modified
	Semantic Space To Do for 2008	Default Folder Type			Dave Walker (dwalker)	06/03/08 14:27
	Workpackages	Default Folder Type	This folder contains some more folders...		Dave Walker (dwalker)	23/01/08 15:40
	DERI Galway	Default Object Type	This is a photo from DERI Galway		Dave Walker (dwalker)	23/01/08 15:40
	SemanticSpace Requirement Analysis	Default Object Type			Dave Walker (dwalker)	23/01/08 15:40
	rfc2119.txt	Default Object Type			Dave Walker (dwalker)	23/01/08 15:40

Fig. 6. Example of external CWE folders, which have been imported into a local CWE project on the BC platform

6.3.2 Third-Party Application

As well as directly importing SIOC data into other CWEs, SIOC data may also be reused by independent applications and services. Exporting proprietary data as SIOC data facilitates the integration and correlation of data from multiple sources. Third-party applications may take advantage of this rich data collection and provide beneficial services based on the aggregation of the data. A SIOC4CWE Explorer for navigating and querying aggregated SIOC data from heterogeneous CWEs in a unified way has been developed [21]. The SIOC4CWE Explorer imports SIOC data from multiple SIOC-enabled CWE platforms and enables users to browse and query the disparate information in a homogenous manner using a single user interface. CWEs export their data as SIOC RDF data using SIOC Exporters. All SIOC content is then imported into a local RDF store, which is accessed by the SIOC4CWE Explorer. A faceted navigation interface is presented to the end-user. The faceted navigation interface allows to quickly narrow down number of results by choosing more and more precise values from various angles [22, 23]. Fig. 7 displays a screenshot of the SIOC4CWE Explorer query interface, with a list of items that have *deirdre* as the creator. As shown the Explorer present results from both the *Ecospace* and *SemanticSpace* projects. She may decide to browse within a particular project, or may choose to browse all or specific types of items (e.g. documents, discussions, etc.) aggregated from all projects. The item creator or when the item was created may also

be used as search criteria. After selecting particular parameter(s), the user is presented with the list of items that satisfy the criteria. Resultant items' metadata is summarized, and a link to the actual item in the original CWE is also provided, so that the user can access the full content from its source.

The SIOC Xplore Widget is another third-party application that is currently being developed. It builds upon the work of the SIOC4CWE Explorer, extending the Explorer's base functionality in widget form.



Fig. 7. Screenshot from the SIOC4CWE Explorer

7 Evaluation

This section evaluates the feasibility and effectiveness of reusing the Semantically Interlinked Online Community (SIOC) ontology to facilitate semantic CWE interoperability. This work has been carried out as part of a project, which specifies the need for interoperability amongst different platforms in its user/system requirements. These requirements were identified in the requirements elicitation phase of the project from real users participating in workshops and surveys. When investigating whether SIOC would be a viable option for modeling CWE data, CWE-platform engineers from BSCW, BC, and NetWeaver were consulted. Two use-cases are presented in this paper demonstrating the reuse of SIOC in the CWE domain.

7.1 Evaluating SIOC Importer/Viewer

The first use-case involves one CWE platform directly importing data from a remote CWE platform. This relates to the use-case in the motivation scenario (section 2),

where Anne wishes to see all documents relating to a common conference together in one CWE platform. The implementation of this use-case is described in section 6.3.1. The CWE developers found both mapping CWE-specific concepts to SIOC concepts and, based on these mappings, defining CWE data in terms of the SIOC ontology straight forward. A `GetFolderContent` service exposes the contents of a particular CWE folder in its SIOC format. It takes `folder ID` and `depth` as parameters and returns the SIOC-annotated data as a string. The `depth` specifies how many levels of subfolders will be returned; the greater the depth, the more time it will take to translate the data into SIOC RDF data. Only the structure of the folder and metadata about each item will be returned, not the actual items, e.g. if a folder contains a pdf document, only its title, description, and MIME-type will be returned. If the user wants to view the actual document, they must invoke the `GetItem` service, passing the `document ID` as a parameter. When importing a folder from a remote CWE, the user must enter their user credentials for the remote CWE. This security model is not optimal, as it exposes the user's credentials to a third-party, i.e. the local CWE. OAuth may be a solution for this, as it would enable the local CWE to access remote resources, without knowing the user-credentials to the remote CWE. [24]. Another security concern that is raised when a user imports a remote folder into a local CWE, is that all users of the local CWE have access to the remote folder, even if they are not a member of the remote project. One solution would be to delegate the responsibility of determining the confidentiality of the remote folder to the importing user. Another option is to place restrictions on each folder, as to whether or not they may be exported to remote CWEs.

Once imported, the local CWE must convert the SIOC data back into CWE-specific data, as is the role of the SIOC Importer/Viewer, so that it may be integrated with legacy CWE data. During development, it was noted that this may cause problems, as multiple CWE concepts may be mapped to the same SIOC concept. For example, from Table 1, we can see that the *bc:workspace/project* concept, *bc:folder* concept, and *bc:collection* concept all map to the *sioc:container* concept. However, when a *sioc:container* concept is reverted to a BC specific concept, it will always be translated as a *folder* concept. Although this is sufficient for general semantic data interoperability, in order to facilitate more fine-grained interoperability, a hybrid approach to ontology modeling will need to be adopted, as discussed in section 3. This means that the core SIOC concepts will be extended with a CWE module, which models CWE-specific concepts, such as *workspace*, *project*, and *folder*. For current implementations, the developers were satisfied with the generic solution of items being mapped to folders and items.

During development, it was decided that it was important that the replicated local data is synchronized with the original remote data. To achieve this, every time a remote folder is viewed by a user, it is refreshed. Also, if a change is made locally, this change is sent to the remote CWE, so that the original folder may reflect the updated version. This entails communication overhead; however we believe that this is acceptable as changes to remote folders are infrequent. The remote data is stored in the local CWE until it is specifically deleted by a user.

7.2 Evaluating Third-Party Application

The second use-case involves a third-party application importing SIOC data from one or more CWE platforms, so that higher-level functionality may be offered based on the correlation of the SIOC data. This relates to the use-case in the motivation scenario (section 2), where Anne wants to find all deliverables in both the Ecospace and SemanticSpace projects with the keyword “web 2.0” in the title, or to find out calendar events that are related to her, or what has been uploaded in both systems during the last two days. To address this cross-platform querying functionality, the SIOC4CWE Explorer has been developed. The implementation of this tool is described in section 6.3.2. The SIOC4CWE Explorer imports SIOC data from multiple CWEs, stores the data locally, and performs RDF queries over this data. This is obviously not a scalable solution, as the amount of data stored locally swiftly becomes unmanageable. One option is to specify a depth of 1 when invoking the `GetFolderContent` service. This will limit the size of the data returned, but also the effectiveness of the search. Alternatively, we propose to delegate the responsibility of searching to the CWE platform. A query would be submitted to a remote CWE and only the results satisfying the search criteria would be returned to the SIOC4CWE Explorer. This would curb the overhead on the SIOC4CWE Explorer substantially and improve the efficiency of searches, as internal-CWE searching is optimized. The SIOC4CWE Explorer may easily merge the results, as they are all defined in terms of the SIOC ontology, and display them to the user. Issues relating to this solution include the implementation of a ‘search’ service by all participating CWE platforms and a consensus on the structure of a query. In order for the query to be interpretable by all CWEs, it should be based on the SIOC ontology. These details are currently under development.

Security is another aspect that needs to be addressed. Similar to the first use-case, the SIOC4CWE Explorer involves a third-party accessing password-protected data. However, in this case the application accesses data from multiple CWE-platforms, each requiring authentication. Aside from security issues, having to enter a username and password multiple times may infuriate users. OpenID eliminates the need for multiple usernames across different websites by providing users with a single URI that they may use to log on to many different sites [25]. BSCW already implements the OpenID standard. While OpenID solves the authentication problem, once CWEs have a user’s log-in credentials, they technically have complete authorization to that user’s resources. Similar to the previous use-case, OAuth may be a suitable solution to granting the SIOC4CWE Explorer authorization to access the necessary CWE data without having to disclose user-credentials. The difficulty of this approach lies in the necessity of each CWE-platform to implement OAuth authentication in their APIs.

The next stage of evaluation is to hold user evaluations to determine the experience of use and provide further feedback. The evaluation will take place by end users, in the context of the parent project.

8 Conclusion and Future Work

SIOC is presented in this paper as a viable option for facilitating semantic data interoperability between disparate CWEs. Once proprietary CWE data is expressed in

terms of the SIOC ontology, it may be interpreted by other CWEs and correlated with SIOC RDF data from external sources. The concept mappings shown in Table 1 demonstrate how CWE-platform specific terms may be translated to SIOC concepts. The implementation of the key components of the CWE Interoperability Architecture demonstrates the great potential of using SIOC to describe CWE proprietary data and ultimately to facilitate the semantic interoperability of independent, distributed CWEs. There are still some open issues for future work. Following on from the successful usage of SIOC in our project so far, the existing work will be extended to facilitate seamless interoperability between all current co-operating CWE platform providers within the context of our project. We also hope that other CWE platform providers will adopt the SIOC approach to CWE interoperation. This would result in a greater CWE-SIOC data mass, which could be used for advanced integration functionality. Additionally, we want to integrate legacy CWE data with SIOC data produced by other sources, for example, online community sites. In this way CWE data could be integrated with data from other domains, opening additional data interoperation possibilities. Another open issue is the extension of the SIOC core ontology with richer descriptions of CWE concepts, as discussed in section 7. Using only the core SIOC concepts, the reverse mapping process from SIOC concepts to CWE concepts is far less expressive than the original mappings from CWE concepts to SIOC concepts. Therefore, it is proposed to extend the SIOC ontology with a module containing CWE-specific concepts. This may enable better interpretation by CWEs of SIOC RDF data.

References

1. Park, J., Ram, S.: Information Systems Interoperability: What Lies Beneath? *ACM Transactions on Information Systems (TOIS)* 22(4), 595–632 (2004)
2. Prinz, W., et al.: Towards an Integrated Collaboration Space for eProfessionals. In: *The Second International Conference on Collaborative Computing (CollaborateCom 2006)*, Atlanta, GA, USA (2006)
3. Bojars, U., et al.: Interlinking the Social Web with Semantics. *IEEE Intelligent Systems* 23(3) (2008)
4. Stuckenschmidt, H., Van Harmelen, F.: Information Sharing on the Semantic Web. In: *Advanced Information and Knowledge Processing*, p. 276. Springer, Heidelberg (2005)
5. Uschold, M., Gruninger, M.: Ontologies and Semantics for Seamless Connectivity. *SIGMOD Record* 33(4), 58–64 (2004)
6. Gruber, T.: A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition* 5(2), 199–220 (1993)
7. Halevy, A.Y.: Answering Qeries Using Views: A Survey. *The VLDB Journal* 10(4), 270–294 (2001)
8. European Interoperability Framework (EIF), European Interoperability Framework for pan-European eGovernment Services. 2004, The European Commission, Directorate General for Informatics (IDABC) (2004)
9. Martínez Carreras, M.A., et al.: Designing a Generic Collaborative Working Environment. In: *IEEE International Conference on Web Services, 2007. ICWS 2007*, Salt Lake City, UT, USA. IEEE Computer Society, Los Alamitos (2007)

10. Martínez Carreras, M.A., Gomez Skarmeta, A.F.: Towards Interoperability in Collaborative Environments. In: The Second International Conference on Collaborative Computing (CollaborateCom 2006), Atlanta, GA, USA (2006)
11. Obrst, L.: Ontologies for semantically interoperable systems. In: Proceedings of the twelfth international conference on Information and knowledge management. ACM, New Orleans (2003)
12. Van Harmelen, F.: Semantic Web Research anno 2006: main streams, popular fallacies, current status and future challenges. In: 10th International Workshop on Cooperative Information Agents. Springer, Heidelberg (2006)
13. Shvaiko, P., Euzenat, J.: A Survey of Schema-based Matching Approaches. *Journal on Data Semantics IV*, 146–171 (2005)
14. Noy, N.F.: Semantic integration: a survey of ontology-based approaches. *SIGMOD Rec.* 33(4), 65–70 (2004)
15. Kalfoglou, Y., Schorlemmer, M.: Ontology mapping: the state of the art. *Knowl. Eng. Rev.* 18(1), 1–31 (2003)
16. Wache, H., et al.: Ontology-based Integration of Information - A Survey of Existing Approaches. In: Workshop: Ontologies and Information Sharing IJCAI 2001 (2001)
17. Qin, H., Dou, D., LePendu, P.: Discovering Executable Semantic Mappings Between Ontologies. In: On the Move to Meaningful Internet Systems 2007: CoopIS, DOA, ODBASE, GADA, and IS. Springer, Heidelberg (2007)
18. Bontas Simperl, E.P., Tempich, C.: Ontology Engineering: A Reality Check. In: On the Move to Meaningful Internet Systems 2006: CoopIS, DOA, GADA, and ODBASE. Springer, Heidelberg (2006)
19. Breslin, J., et al.: SIOC: An Approach to Connect Web-based Communities. *International Journal on Web Based Communities* 2(2), 133–142 (2006)
20. Breslin, J., Bojars, U.: Semantically-Interlinked Online Communities (SIOC) Ontology Submission Request to W3C 2007 (2007)
21. Ning, K., et al.: A SIOC Enabled Explorer of Shared Workspaces. In: Workshop on Web 2.0/Computer Supported Co-operative Work in conjunction with ECSCW 2007, Limerick, Ireland (2007)
22. Kruk, S.R., et al.: MultiBeeBrowse - Accessible Browsing on Unstructured Metadata. In: Meersman, R., Tari, Z. (eds.) OTM 2007, Part I. LNCS, vol. 4803, pp. 1063–1080. Springer, Heidelberg (2007)
23. Oren, E., Delbru, R., Decker, S.: Extending Faceted Navigation for RDF Data. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L.M. (eds.) ISWC 2006. LNCS, vol. 4273, pp. 559–572. Springer, Heidelberg (2006)
24. OAuth - An open protocol to allow secure API authentication 2008 (2008)
25. OpenID - Single Digital Identity 2008 (2008)

An Ontology-Based Approach for Discovering Semantic Relations between Agent Communication Protocols

Maricela Bravo¹ and José Velázquez²

¹ Morelos State Polytechnic University, Cuauhnáhuac 566, Texcal,
Morelos, México, CP 62550
mbravo@upemor.edu.mx

² Electrical Research Institute, Reforma 113, Palmira,
Morelos, México, CP 62490
jconrado@iie.org.mx

Abstract. Traditionally autonomous agents communicate each other using a predefined set of communication primitives implicitly encoded inside the agent protocol. Nowadays, there are various research efforts for automating the deployment of agents in open environments such as Internet. Considering the existence of multiple heterogeneous agents, independently developed and deployed on the Web, the challenge is to achieve interoperability at the communication level, reducing the number of communication errors caused by differences in syntax and semantics of their particular languages implementations. Currently, to support communication interoperability, agent owners must redesign communication syntax and deploy manually their agents, which results in a tedious, time consuming and costly task. To solve this problem we propose an Ontology-based approach for discovering semantic relations between agent communication protocols, which considers the description of primitives and their pragmatics. We present a case study to show the applicability of our approach, and implemented a communication environment to evaluate the resulting set of relations in the Ontology. Results show that our approach reduces the level of heterogeneity among participating agents.

Keywords: Agent communication protocols, ontologies, translator.

1 Introduction

Communication in multi-agent systems (MAS) plays a key role in achieving coordination and cooperation for specific problems. Traditionally agents have been designed to communicate each other using a predefined set of primitives following a protocol. However, nowadays there is a growing interest in deploying and communicating autonomous agents in open environments such as Internet. The challenge of deploying multiple agents over Internet is to provide interoperability at the communication level. Currently, to support communication interoperability, agent owners must redesign communication syntax and deploy manually their agents, which results in a tedious, time consuming and costly task. Therefore, research efforts to

propose new approaches to automate the communication process will facilitate the automatic deployment of agents over Internet.

The language used by agents to exchange messages is defined as agent communication language (ACL). KQML [1] was the first standardized ACL from the ARPA knowledge project. KQML consists of a set of communication primitives aiming to support interaction between agents. Another ACL [2] standard comes from the Foundation for Intelligent Physical Agents (FIPA) initiative. FIPA ACL is based on speech act theory, and the messages generated are considered as communicative acts. The objective of using a standard ACL is to achieve effective communication without misunderstandings, but the software implementation of ACL is not explicitly defined, leaving developers to follow their own criteria. Furthermore, standard ACL specifications consider the incorporation of privately developed communicative acts.

Communication in MAS is executed through the exchange of messages following a protocol. A protocol is a sequence of exchanged messages conforming to a set of shared rules. For this work we are considering that a message has the following elements:

- *Sender* identifies the agent that is issuing the message,
- *Receiver* is the target agent, which will receive and analyze the incoming message,
- ***Primitive*** is a basic communication act, which has syntax, semantics and pragmatics.
- *Parameters* are the rest of input data, these parameters depend on the primitive.
- *Other data* is left for additional information.

In this paper we present a combined approach for discovering semantic relations between primitive instances. Our approach consists of a probabilistic-based classification technique and a pragmatic analysis of the primitive usage in the communication protocol. The rest of the paper is organized as follows. In section two we present related work with the subject of this research. In section three we describe the general process for discovering semantic relations between communication protocols. In section four we present a case study to show the applicability of our approach. In section five we present the communication environment to execute communications. In section six we evaluate the results and finally, in section seven we conclude.

2 Related Work

Many authors have presented different techniques and algorithms for discovering semantic relations between vocabularies in various research areas, such as data base schema integration, knowledge engineering, natural language processing and information systems integration. In data base research Rahm and Bernstein [3] presented a taxonomy of the existing approaches for schema matching and distinguished name-based, constraint-based, structure-based, text-oriented and a combination of matchers. Batini [4] presented various data base schema integration methods and established three integration phases: schema comparison, schema conforming and schema merging. Chimaera [5] is a semi-automatic merging and diagnosis tool developed by the Stanford University Knowledge Systems Laboratory.

It provides assistance in the task of merging knowledge bases produced by multiple authors in multiple scenarios. PROMPT [6] is an algorithm that provides a semi-automatic approach to ontology merging and alignment. The MOMIS project [7] consists of an architecture designed to integrate heterogeneous data sources. The mapping approach of MOMIS is based on human definitions of semantic relations and the use of supporting tools. GLUE [8] is a system which uses machine learning techniques to discover mappings. GLUE uses a multi-strategy learning approach: a Naive Bayes text-based classification method and a name learner.

Mostly of reported techniques use and analyze data, concepts or vocabularies provided by human developers. However, none of the reported techniques takes into consideration the analysis of pragmatics, which is the real usage of a concept in a protocol or conversation. In contrast to reported works, in this paper we present a combined approach, which considers the pragmatic information included in protocols.

3 Process for Discovering Relations

The general process for discovering relations between two sets of communication primitives is presented in Figure 1. The solution is based on the use of a probabilistic-based Bayes algorithm classification (a syntactical technique) and the use of Finite State Machines (FSM) (a pragmatical technique).

1. *Acquisition and processing of communication primitives descriptions.* Descriptions are acquired through a Web-based environment and are processed using taggers and text classifiers. The final result of this step consists of a set of keywords relative to the primitive description.
2. *Semantic classification.* The next step consists of classifying primitives according to the *Type* attribute of the *Primitives* class defined in the Ontology. The classification process takes as input the set of keywords generated in the previous

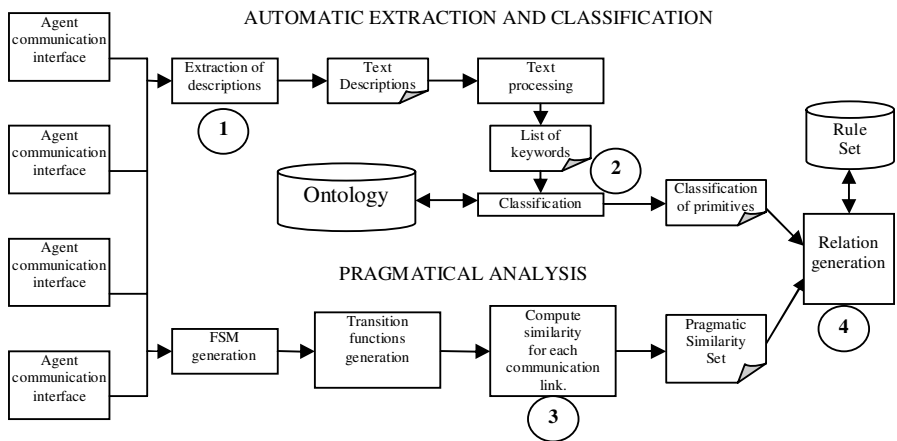


Fig. 1. Architecture for discovering semantic relations between communication protocols

phase and using a Bayes classifier algorithm to identify to which *Type* each communication primitive belongs to.

3. *Discovering pragmatic relations.* We use FSM-diagrams to compare the real usage of the primitive in the communication protocol. For a communication scenario, in a MAS environment, we adapted the classes presented by Müller [7]. He establishes that any communication protocol consists of three general states: *starter*, *reactor* or *completer* depending on the moment when the primitive occurs, but we added another the *modify* state to be more precise. In order to obtain real semantics of the primitive we evaluate the moment when the primitive is issued, and also map both sets of primitives in the same FSM, to compute similarity.
4. *Establish relationships.* Based on the previous steps we proceed to define semantic relations between communication primitives from different agents, according to the following rules:
 - Two communication primitives are *equal (EQ)* if their *Type* attribute is the same, and if they have the same usage in the FSM.
 - Two communication primitives are *similar_pragmatic (SP)* if their *Type* attribute is different, but they have the same usage in the FSM.
 - Two communication primitives are *similar_semantic (SS)* if their *Type* attribute is the same, but they do not have the same usage in the FSM.

4 An Example

1. Acquisition and processing of communication primitives descriptions

Considering a multi-agent system populated with three autonomous agents $MAS = \{ A, B, C \}$. The set of communication primitives are shown in Table 1. We used taggers and text preprocessors to extract keywords from the descriptions of the primitives.

2. Semantic classification

For classification process we used a probabilistic-based Bayes algorithm, and provided as input the descriptions of communication primitives and a set of keywords which represent the different states (classes) in which a primitive may be issued. The result of the algorithm is the *Type* attribute to which the primitive belongs. We consider this step as a semantic similarity approach. Table 2 presents the resulting classification of primitives of agents *A*, *B* and *C*.

3. Discovering pragmatic relations

The process for discovering semantic relations between communication protocols requires human intervention, because we need to draw the state diagrams in order to generate the FSM and obtain the set of transition functions to compute differences. In Figure 2 we present the resulting state transition diagrams of communication protocols of agents *A*, *B* and *C*.

For each arc in the FSM there is a transition function ft for a given initial state and an input primitive which produces a final state.

$$ft(initial-state, input-primitive) = final-state \quad (1)$$

Table 1. Communication primitives of agents A, B and C

Agent A	Communication primitives
Primitives	{(CFP , “Initiate a communication process by calling for proposals”), (Propose , “Issue a proposal or a counterproposal”), (Accept , “Accept the terms specified in a proposal without further modifications”), (Terminate , “Unilaterally terminate the current communication process”), (Reject , “Reject the current proposal with or without an attached explanation”), (Acknowledge , “Acknowledge the receipt of a message”), (Modify , “Modify the proposal that was sent last”), (Withdraw , “Withdraw the last proposal”)}
Agent B	Communication primitives
Primitives	{(Initial_offer , “Send initial offer”), (RFQ , “Send request for quote”), (Accept , “Accept offer”), (Reject , “Reject offer”), (Offer , “Send offer”), (Counter-offer , “Send counter offer”)}
Agent C	Communication primitives
Primitives	{(Call for proposal , “Initiate a call-for-proposal”), (Propose proposal , “Send a proposal or a counterproposal”), (Reject proposal , “Reject the received proposal with or without an attached explanation”), (Withdraw proposal , “Withdraw the previous proposal that was sent”), (Accept proposal , “Accept the terms and conditions specified in a proposal without further modifications”), (Change proposal , “Change the proposal that was sent”), (Inform proposal , “Inform the receipt of a proposal”), (Terminate communication , “Unilaterally terminate the communication process”)}

Table 2. Resulting classification of primitives

Issuer	Start	React	Modify	Finalize
A	CFP	Propose Withdraw Acknowledge	Modify	Accept Reject Terminate
B	RFQ Initial_Offer	Offer	Counter_offer	Accept Reject
C	Call for proposal	Propose proposal Withdraw proposal Inform proposal	Change proposal	Accept proposal Reject proposal Terminate communication

To establish semantic relations we need to identify the set of different communication links and the pairs of agents that may participate.

$$\begin{aligned}
 DCL &= \{ (a_1, a_2), (a_1, a_3), \dots, (a_i, a_j) \} \\
 DCL &= \{ (A, B), (A, C), (B, C) \}
 \end{aligned}
 \tag{2}$$

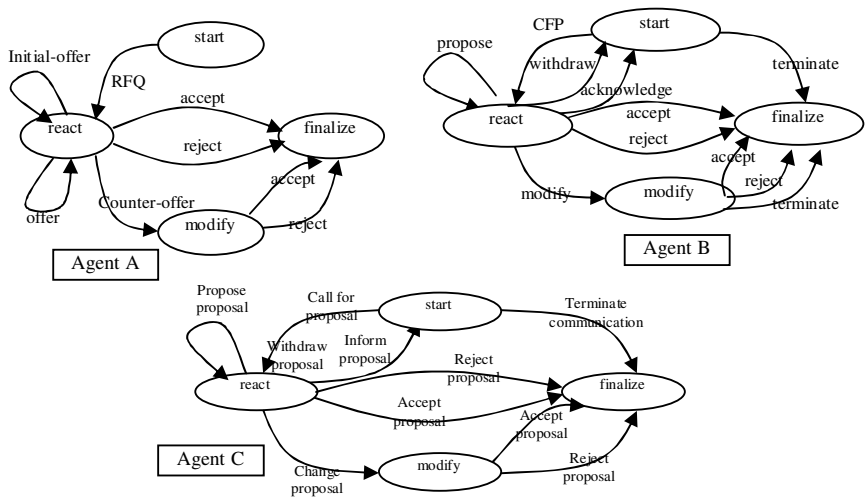


Fig. 2. FSM of communication protocol for agents A, B and C

Table 3. Semantic relations between primitives

CL(A, B)	CL(A, C)	CL(B, C)
<i>EQ</i> (A, CFP, B, RFQ)	<i>EQ</i> (A, CFP, C, Call for proposals)	<i>EQ</i> (B, RFQ, C, Call for proposals)
<i>EQ</i> (A, Propose, B, Offer)	<i>EQ</i> (A, Propose, C, Propose proposal)	<i>EQ</i> (B, Offer, C, Propose proposal)
<i>EQ</i> (A, Modify, B, Counter_offer)	<i>EQ</i> (A, Modify, C, Change proposal)	<i>EQ</i> (B, Counter_offer, C, Change proposal)
<i>SP</i> (A, Propose, B, Initial_Offer)	<i>EQ</i> (A, Withdraw, C, Withdraw proposal)	<i>EQ</i> (B, Accept, C, Accept proposal)
<i>SS</i> (A, CFP, B, Initial_Offer)	<i>EQ</i> (A, Acknowledge, C, Inform proposal)	<i>EQ</i> (B, Reject, C, Reject proposal)
	<i>EQ</i> (A, Accept, C, Accept proposal)	<i>SP</i> (B, Initial_Offer, C, Propose proposal)
	<i>EQ</i> (A, Reject, C, Reject proposal)	<i>SS</i> (B, Initial_Offer, C, Call for proposals)
	<i>EQ</i> (A, Terminate, C, Terminate communication)	

To compute pragmatic similarity we implemented an array-based algorithm. In this case we implemented three arrays, each with three columns which represent: the initial state, the input primitive and the final state. The algorithm is executed for each different communication link (a_i, a_j), where a_i represents the array of agent i .

```
For each transition function  $ft$  of  $a_i$ 
  For each transition function  $ft$  of  $a_j$ 
    If ( $a_i$ [initial-state] is equal to  $a_j$ [initial-state])
      and ( $a_i$ [final-state] is equal to  $a_j$ [final-state])
```

$a_i[\text{input-primitive}]$ is-similar-pragmatic to
 $a_j[\text{input primitive}]$

4. Establish Relationships

We finally established semantic relations between primitives following the set of rules presented in Section 3. We defined only equal and similar relations for primitives that are syntactically different. Results of this process are shown in Table 3. To define relations we used the form:

$$REL(A_i, P_i, A_j, P_j)$$

where
 A_i is the agent issuer of primitive P_i
 A_j is the agent issuer of primitive P_j

5 Communication Web-Based Environment

To evaluate our approach we implemented a Web-based communication environment, which consists of a **translator** module, invoked whenever is necessary; an **intermediary** program which is responsible for initialization of communication processes, sending and receiving messages form both agents, and recording the communication until an ending message is issued by any of the participants; and an protocol **ontology**, populated with communication primitives and relations (Figure 3).

Agents are the representative software entities used by their respective owners to program their preferences and communication strategies. The translator module was implemented using Jena¹, a framework for building Semantic Web applications. For the description and execution of communication processes, we used BPEL4WS. The interaction with each partner occurs through Web service interfaces, and the structure of the relationship at the interface level is encapsulated in what we call a partner link. The BPEL4WS process defines how multiple service interactions with these partners are coordinated, as well as the state and the logic necessary for this coordination. To code the ontology we decided to use OWL as the ontological language.

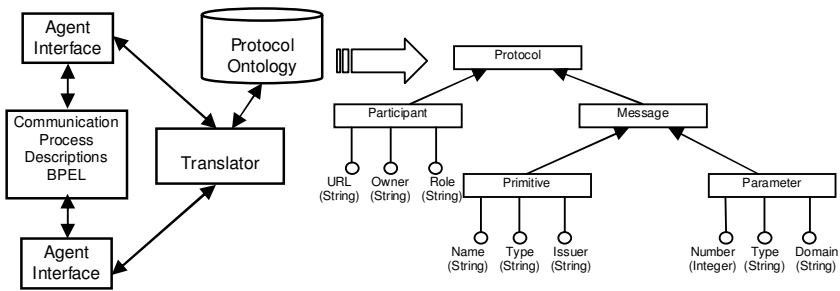


Fig. 3. Communication Web-based architecture

¹ <http://jena.sourceforge.net>

6 Experimentation Results

We executed a series of tests with these agents. Figure 4 shows the graphical results of this experiment. The first set of bars represents results of communications without translations, and the second set of bars represent results of communications invoking the translator. Results show that for the first set of executions many ended because of misunderstandings. For the second set of executions we can appreciate a reduction in the number of misunderstandings, although the problem remains, some communications are still ending due to this problem. For this case we suggest using a learning approach, in order to fully eliminate the problem. However, the result is good enough to evaluate the main contribution of this paper. The set of discovered relations were well defined by our semi-automatic combined approach. The Ontology was populated with these primitives and relations among them, and when integrated to the general environment it solved the main problem of our work: communication interoperability.

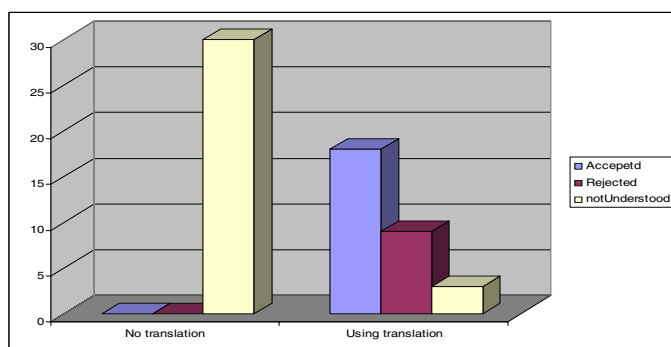


Fig. 4. Experimental results

7 Conclusions

As the Semantic Web has been evolving we are facing new requirements, such as discovering semantic relations from heterogeneous sources, in particular in this paper we have presented an approach for discovering relations between communication protocols. We have presented a combined approach for aligning communication primitives between multiple heterogeneous agents based on the use of FSM. This is a promising research area, because nowadays there is a tremendous amount of legacy software which in turn will require to be integrated transparently giving the idea of an integral solution independently of the inherent heterogeneity inside their logics or protocols. Our contribution consists of obtaining more information by comparing the classification of primitives with their use in FSM.

Our approach represents a new point of view, because traditional aligning approaches have been mainly developed for aligning data, vocabularies or ontologies.

Finally, we are sure that this ontology-based approach can be applied in other application areas such as Web service discovery, process engineering, and program

comparison or application integration, which have in common that they provide functionality information similar to protocols in communication scenarios.

References

1. Finning, T., Fritzson, R., McEntire, R.: KQML as an agent communication language. In: Proceedings of the 3rd International Conference on Information and Knowledge Management (November 1994)
2. FIPA – Foundation for Intelligent Physical Agents. FIPA Specifications (2003), <http://www.fipa.org/specifications/index.html>
3. Rahm, E., Bernstein, P.: A Survey of Approaches to Automatic Schema Matching. The VLDB Journal 10, 334–350 (2001)
4. Batini, C., Lenzerini, M., Navathe, S.: A Comparative Analysis of Methodologies for Database Schema Integration. ACM Computing Surveys 18, 323–364 (1986)
5. McGuinness, D., Fikes, R., Rice, J., And Wilder, S.: The Chimaera Ontology Environment. In: Proceedings of the 7th National Conference on Artificial Intelligence (AAAI 2000), Texas (2000)
6. Noy, F.N., Musen, M.A.: PROMPT, Algorithm and Tool for Automated Ontology Merging and Alignment. In: Proceedings of 17th National Conference on Artificial Intelligence (AAAI 2000), Texas (2000)
7. Bergamaschi, S., Castano, S., De Capitani di Vimercati, S., Montanari, S., Vincini, M.: A Semantic Approach to Information Integration: the MOMIS Project. In: Sesto Convegno della Associazione Italiana per l'Intelligenza Artificiale, AI*IA 1998, Padova IT (1998)
8. Doan, A., Madhavan, J., Domingos, P., Halevy, A.: Learning to Map Between Ontologies on the Semantic Web. In: Proceedings of the 11th International World Wide Web Conference (WWW 2002), Hawaii, USA (2002)
9. Müller, H.J.: Negotiation Principles, Foundations of Distributed Artificial Intelligence. In: O'Hare, G.M.P., Jennings, N.R. (eds.), John Wiley & Sons, New York

Reference Fusion and Flexible Querying

Fatiha Saïs¹ and Rallou Thomopoulos^{1,2}

¹ LIRMM (CNRS & Univ. Montpellier II), 161 rue Ada, F-34392 Montpellier cedex 5

² INRA, UMR1208, 2 place P. Viala, F-34060 Montpellier cedex 1

Fatiha.Sais@lirmm.fr, rallou@supagro.inra.fr

Abstract. This paper deals with the issue of data fusion, which arises once reconciliations between references have been determined. The objective of this task is to fusion the descriptions of references that refer to the same real world entity so as to obtain a unique representation. In order to deal with the problem of uncertainty in the values associated with the attributes, we have chosen to represent the results of the fusion of references in a formalism based on fuzzy sets. We indicate how the confidence degrees are computed. Finally we propose a representation in Fuzzy RDF, as well as its flexible querying by queries expressing users' preferences.

Keywords: Data integration, Data fusion, Fuzzy sets, Preferences, Flexible queries.

1 Introduction

Data reconciliation and data fusion are two of the main problems encountered when different heterogeneous data sources have to be integrated. These problems are independent from the kind of architecture that is used in the information integration system, whatsoever in a data warehouse or in a mediator architecture. Data reconciliation and data fusion problems are mainly related to the syntactic heterogeneity and to the semantic heterogeneity of the data sources. Data reconciliation consists in deciding whether different data descriptions refer to the same real world entity (e.g. the same person, the same museum, the same paper). Data fusion consists in the ability to obtain a single representation for the different descriptions of the reconciled data and then to give a clean and consistent result to the user.

Detecting reconciliations between different descriptions is insufficient to obtain an integrated representation of the reconciled data. Indeed, different conflicts and ambiguities between values can appear in data descriptions. For example, the same painting is named "*La Joconde*" in one description and "*Mona Lisa*" in another description. In this paper, we focus on the problem which occurs when different descriptions of reconciled data have to be fused. Different strategies can be used to select the best values to consider in the final representation of the real world entity: heuristic and statistic criteria like value frequency, data freshness, source confidence and so on.

Nevertheless, certainty in conflict resolution between values cannot be guaranteed. This uncertainty is all the more present since the data reconciliation decisions are themselves uncertain. To deal with this problem, we have chosen to keep all the values appearing in the data descriptions and to represent the fusion result by using the fuzzy

sets formalism (see [1,2]). We present a method to compute confidence degrees, associated to the values, by exploiting both value features and data source features. Then we propose a method for the flexible querying of the fusionned data based on fuzzy pattern matching [3]. The query processing is performed in two steps: selection of relevant results and ordering of these results.

The article is organized as follows: section 2 presents preliminary notions on fuzzy sets and fuzzy pattern matching. Section 3 states the fusion problem and gives our contributions concerning the reference fusion method. Section 4 presents our approach for the flexible querying of the data obtained by the fusion method. Finally, section 5 presents some of the related work and conclude the paper.

2 Basics of Fuzzy Sets

The two approaches, fuzzy sets [1] and possibility theory [2], constitute a homogeneous formalism in two different uses. In both uses, an order relation is defined on a domain of values.

Definition 1 (fuzzy set). A fuzzy set A on a domain X is defined by a membership function μ_A from X to $[0 ; 1]$ that associates with each element x of X the degree $\mu_A(x)$ to which x belongs to A . The domain X may be continuous or discrete.

For example, the fuzzy set *PaintingDate* is defined on the continuous domain $\{[1500, 1550], [1450, 1600]\}$. The fuzzy set *PaintingName* is defined on the discrete domain that is denoted $\{1/La Joconde, 0.5/Joconde\}$. The values 1 and 0.5 indicate the degree associated with each element. In the following, fuzzy sets are either user-defined, during the choice of the querying selection criteria, or computed as an ill-known datum resulting from the fusion of references.

We call *support* and *kernel* of a fuzzy set A respectively the sets:

$$support(A) = \{x \in X \mid \mu_A(x) > 0\} \text{ and } kernel(A) = \{x \in X \mid \mu_A(x) = 1\}$$

We now focus on fuzzy pattern matching, a comparison between fuzzy sets which allows to determine in a graduate way whether an ill-known datum somehow answers a flexible query.

2.1 Fuzzy Pattern Matching

Two scalar measures are classically used in fuzzy pattern matching [3] to evaluate the compatibility between an ill-known datum and a flexible query: (i) a possibility degree of matching [2]; (ii) a necessity degree of matching [4].

Definition 2 (possibility and necessity degrees of matching). Let Q and D be two fuzzy sets defined on a domain X and representing respectively a flexible query and an ill-known datum:

- the possibility degree of matching between Q and D , denoted $\Pi(Q; D)$, is an “optimistic” degree of overlapping that measures the maximum compatibility between Q and D , and is defined by $\Pi(Q; D) = \sup_{x \in X} \min(\mu_Q(x), \mu_D(x))$;

- the necessity degree of matching between Q and D , denoted $N(Q; D)$, is a “pessimistic” degree of inclusion that estimates the extent to which it is certain that D is compatible with Q , and is defined by $N(Q; D) = \inf_{x \in X} \max(\mu_Q(x), 1 - \mu_D(x))$.

3 Reference Fusion

3.1 Reference Fusion Problem

We suppose that we have a reconciliation method which infers reconciliation decisions for some pairs of references. We start from a set of reference reconciliation decisions. We consider that we have for each reconciled pair of references, the similarity score (a real value in $[0 ; 1]$) of their descriptions.

In Figure 1, we give an example of data sources to be reconciled.

Source S1				
Ref.	MuseumName	MuseumAddress	MuseumContact	PaintingName
id11	Louvre	Palais Royal, Paris	info@louvre.fr	La Joconde
id12	Louvre	Palais Royal, Paris	0140205317	Joconde
id13	Orsay	Rive gauche de la seine, Paris		L'Européenne

Source S2				
Ref.	MuseumName	MuseumAddress	MuseumContact	PaintingName
id21	Louvre	99, rue Rivoli, 75001 Paris	info@louvre.fr	Mona Lisa
			0140205317	

Fig. 1. Example of references to be reconciled

In Figure 2, we show an example of a set of reconciliation decisions obtained by a numerical method of reference reconciliation (e.g. N2R method [5]), for the references shown in Figure 1 .

$((id11, id21), 0,6) ; ((id11, id12), 0,9) ; ((id12, id21), 0,7)$

Fig. 2. A set of reconciliation decisions between references

One of the most important difficulties that a fusion method has to face are the conflicts and the ambiguities between the different values of the same attribute. From a set of references pairwise reconciled, a method of fusion aims at providing a reference description such that there are no conflicts between attribute values.

Due to the frequency of syntactic variations, a fusion method cannot guarantee certainty in the attribute-value assignments. Consequently, instead of using pairs (*attribute*, *value*) to represent the descriptions of the fusionned references we will use triples (*attribute*, *value*, *confidence*), where *confidence* is a real value in $[0; 1]$.

We consider that a reference in one source can be reconciled with several references in another source. Indeed, in this case the fusion is applied on sets of reconciled references coming from all sources.

3.2 Reference Fusion Method Based on Fuzzy Sets

Here, we present a fusion method which allows providing a ranking of the values assigned to an attribute.

In order to define the ranking, for each attribute value, a confidence degree is computed by using various strategies, like those already used by [6,7]. These strategies exploit syntactic features of the values, such as value frequency, but also data sources features, such as freshness.

Ranking of attribute values

Objective. Starting from a set of n references $ref_1, \dots, ref_n$, such that :

- each reference ref_i is described by a set of attribute-facts $Desc(ref_i) = \{(A_1, v_{i1}), \dots, (A_p, v_{ip})\}$;
- these references are pairwise reconciled with a similarity score s_{ij} between ref_i and ref_j ;
- we denote S_i the data source from where the reference ref_i is coming ($S_1, \dots, S_n$ are not necessary distinct),

the objective of the ranking method is to provide, for each attribute A_k , the set of values v_{ik} that are assigned to this attribute, ranked by the degree $c_{ik} \in [0; 1]$. The degree c_{ik} measures the confidence in the statement that v_{ik} is the right value of the attribute A_k .

Criteria. In order to define the ranking on the attribute values, we consider several criteria and we propose measures which return normalized values (i.e. values in $[0; 1]$). The different criteria that are taken into account are:

Homogeneity. It concerns the set of values taken by the attribute in the descriptions of the reconciled references. In order to measure this homogeneity, we use the frequency of each value among the values taken by the attribute in the descriptions of the reconciled references. We define the homogeneity $hom(v_{ik})$ associated to the value v_{ik} as follows: $hom(v_{ik}) = \frac{Cardinal\{ref_j | \langle ref_j, A_k, v_{ik} \rangle \in Desc(ref_j)\}}{n}$ with $j \in [1; n]$.

Syntactic similarity. It also concerns the set of values taken by the attribute in the descriptions of the reconciled references. This criterion is based on the assumption that a value is all the more reliable since it is syntactically similar to the values taken by the attribute in the descriptions of the reconciled references. The syntactic similarity $Csim(v_{ik})$ between the value v_{ik} and the other values v_{jk} ($j \in [1; n], j \neq i$) is defined as follows: $Csim(v_{ik}) = \frac{\sum_j sim(v_{ik}, v_{jk})}{n-1}$ where sim represents a similarity measure between atomic values (for more details on the various similarity measures, see [8]).

“Global” similarity score between the reconciled references. This criterion replaces the precedent one, when the attribute value is missing in some reference descriptions.

In this case, we use the similarity of the whole descriptions of the reconciled reference pairs (i.e. by taking into account the values of the other attributes).

Freshness of the data source. This criterion can be viewed as an estimation of the reliability of the data source from which the value is coming. In the following, we propose a definition for the data source freshness. Let $MAJ(S_i)$ be the date of the last update of the data source S_i and let j be the current date. The freshness of S_i is given by the formula: $frch(S_i) = 1 - \frac{j - MAJ(S_i)}{\sum_{p \in [1; n]} (j - MAJ(S_p))}$.

This definition, based on the ratio of the “age” of the update of the considered source and the sum of the “ages” of the other sources, has the following interest: it converges towards 0 for a source which has not been updated for a long time and it converges towards 1 for a source which has been updated very recently. For example, if the last update of S_1 is of six months and the last update of S_2 is of two months, then we obtain $frch(S_1) = 1 - 6/8 = 1/4$ and $frch(S_2) = 1 - 2/8 = 3/4$;

Occurrence frequency. It concerns the values taken by the attribute in the reconciled reference descriptions among the set of all the values appearing in the data sources. Indeed, a value which is repeated several times inside the reference descriptions will be considered as more reliable, since it belongs to a common referencial and is less likely to contain typographical errors. The frequency $f(v_{ik})$ of the value v_{ik} is defined by : $f(v_{ik}) = \frac{Cardinal\{<ref \ A \ v_{ik}>\}}{\sum_{j \in [1; n]} Cardinal\{<ref \ A \ v_{jk}>\}}$ where ref denotes a reference which belongs to $S_1 \cup \dots \cup S_n$ and A an attribute.

Determining the confidence degrees. We propose a method which is based on the criteria defined above. They are combined in order to assign a confidence degree, in $[0 ; 1]$, for each value taken by the attribute in the set of descriptions of the reconciled references.

Definition 3 (confidence degree). Let A be an attribute and let $v_1, \dots, v_n$ be the values respectively taken by A in the descriptions of the references $ref_1, \dots, ref_n$ that are pairwise reconciled. The confidence degree $conf(v)$, where $v \in \{v_1, \dots, v_n\}$, measures the confidence in the fact that the right value of the attribute A is v . The confidence degree is obtained as follows:

- 1) if $hom(v) = 1$ then $conf(v) = 1$ (v is the value of A in all the reference descriptions);
- 2) if $hom(v) < 1$ (v is the value of A only in some reference descriptions), let I be the set of indexes $i \in [1; n]$ such that $v_i = v$, then: $conf(v) = \max_{i \in I} \frac{Csim(v_i) + frch(S_i) + f(v_i)}{3}$.

In the example described in Figures 1 and 2, we have as attribute *MuseumName* and its values “Louvre” and “Louvre”, with:

$$hom(\text{“Louvre”}) = 1/3, \text{ and } conf(\text{“Louvre”}) = (\frac{5}{6} + \frac{1}{4} + \frac{1}{3})/3 = 0.47;$$

$$hom(\text{“Louvre”}) = 2/3 \text{ and } conf(\text{“Louvre”}) = \max((\frac{11}{12} + \frac{1}{4} + \frac{2}{3})/3, (\frac{11}{12} + \frac{3}{4} + \frac{2}{3})/3) = \max(0.61, 0.78) = 0.78.$$

Thus, we obtain a set of possible values for the attribute A associated to a confidence degree in $[0 ; 1]$, i.e. a fuzzy set describing the value of A .

Fusion Result. In the following, we will denote ref_F the reference obtained by the fusion method where the attribute-value assignments are uncertain.

Definition 4 (fusionned datum). A fusionned datum is a reference $ref_F = \{ \langle A_1, V_1 \rangle, \dots, \langle A_p, V_p \rangle \}$, where $A_1, \dots, A_p$ are attributes and $V_1, \dots, V_p$ are fuzzy values. Each V_i is a set of pairs (v, c) composed of a value of A_i and its confidence degree.

Fusion Method Implementation. The result of the fusion method is represented in Fuzzy-RDF [9]. In Fuzzy-RDF, for each triple a real value α in $[0; 1]$ is added to represent the fuzzy truth value of the triple. To obtain the plain RDF representaion of the fuzzy one we use the RDF reification mechanism.

4 Flexible Querying of Fusionned Data

The objective of this section is to define the querying of fusion results, in which attribute values are fuzzy sets resulting from the fusion of references as defined in Definition 4, by flexible queries, where the users may express preferences in attribute values as well as in attribute importance.

4.1 Flexible Querying Language Definition

The queries are expressed in terms of a set of projection attributes and a set of conjunctive fuzzy selection criteria, ordered by preference, using the form $\langle \text{attribute, ordered set of values} \rangle$.

Definition 5 (query). A query Q is a set $\{ \{ A_{P1}, \dots, A_{Pn} \}, \{ \langle A_{S1}, V_{S1}, ord_{S1} \rangle, \dots, \langle A_{Sm}, V_{Sm}, ord_{Sm} \rangle \} \}$ where $\{ A_{P1}, \dots, A_{Pn} \}$ is the set of projection attributes, and $\{ \langle A_{S1}, V_{S1}, ord_{S1} \rangle, \dots, \langle A_{Sm}, V_{Sm}, ord_{Sm} \rangle \}$ is the set of triples defining the selection criteria. For all $i \in [1, m]$, these triples have the following meaning: (i) A_{Si} is a selection attribute; (ii) V_{Si} is an ordered set of values associated with the selection attribute A_{Si} . (iii) ord_{Si} is an integer giving the rank of the selection criterion defined on a_{Si} , by order of user's preference.

The ordering, defined by the user on the set of searched values, is then computed as a fuzzy set. There are two distinct cases:

1) A_{Si} is a symbolic attribute. V_{Si} is a set of pairs $(val, pref)$ composed of a value of A_{Si} and a preference degree between 0 and 1. This preference degree is computed using a rank r defined by the user for val (as an integer, with possible ex aequos). The preference degrees associated with the values val are regularly spaced: values at the first rank get the degree $pref = 1$, values at the last rank (l) get the degree $pref = 1/l$, values at the rank r ($r \in [1, l]$) get the degree $pref = r/l$;

2) A_{Si} is a numeric attribute. V_{Si} is a pair (K, S) composed of two intervals: (i) an interval of most preferred values, which is included in (ii) an interval of accepted values. These intervals respectively identify the kernel and the support of the fuzzy set representing user's preferences.

An example of query is the following:

$Q = \{\{MuseumName\}, \{< PaintingName, ((La\ Joconde, 1), (Mona\ Lisa, 0.5)), 1 >, < MuseumLocation, ((Paris, 1)), 2 >\}\}$.

The answer to a query is defined as follows.

Definition 6 (answer). The answer A to a query Q is an ordered set of fusionned data restricted to the selection attributes of Q .

For example, an answer to the query Q can be the following ordered set of museums:

1) Louvre $< 0.78 >$, Lovre $< 0.47 >$; 2) Orsay.

4.2 Query Processing

The principle of the query processing is based on two steps:

- 1) the identification of the set of relevant results, i.e. the fusionned data that satisfy at least one selection attribute with a strictly positive possibility degree;
- 2) the ordering of the results. An answer is considered as all the more relevant since: (i) it satisfies a selection criterion preferred by the user, according to the order defined by ord_{Si} (see Definition 5); (ii) the associated necessity degree is higher; (iii) the associated possibility degree is higher.

Identification of the set of relevant results

Definition 7 (relevant result). A fusionned datum $ref_F = \{< A_1, V_1 >, \dots, < A_p, V_p >\}$ is a relevant result to a query $Q = \{\{A_{P1}, \dots, A_{Pn}\}, \{< A_{S1}, V_{S1}, ord_{S1} >, \dots, < A_{Sm}, V_{Sm}, ord_{Sm} >\}\}$ if $\exists i \in [1; m], \Pi(V_{Si}, V_j) > 0$ with $j \in [1; p]$ such that $A_{Si} = A_j$.

Two cases can be distinguished:

- A_{Si} is a symbolic attribute. Then $V_{Si} = \{val_i, pref_i\}$ and $V_j = \{(v_j, c_j)\}$. $\Pi(V_{Si}, V_j)$ is strictly positive if $\{val_i \mid pref_i > 0\} \cap \{v_j \mid c_j > 0\} \neq \emptyset$;
- A_{Si} is a numeric attribute. Then $V_{Si} = (K, S)$ and $V_j = \{(v_j, c_j)\}$. $\Pi(V_{Si}, V_j)$ is strictly positive if $\exists v_j \in S$ and $c_j > 0$.

Ordering of the results. We propose the following algorithm to order the relevant results:

- we start with the attribute A_{Si} at the first rank in user's preferences: i is chosen such that $ord_i = 1$;
- for each relevant result, we compute the necessity and possibility degrees of matching for the attribute A_{Si} : $\Pi(V_{Si}, V_j)$ and $N(V_{Si}, V_j)$ (with $j \in [1; p]$ such that $A_{Si} = A_j$);
- relevant results are then ordered by decreasing necessity degrees, then in case of ex aequos, by decreasing possibility degrees, then in case of ex aequos the attribute at the next rank is considered: i is chosen such that $ord_i = ord_i + 1$.

The algorithm is thus recursively defined.

For instance, for the query $Q = \{\{MuseumName\}, \{< MuseumName, ((Louvre, 1)), 1 >\}\}$, the only relevant result is the fusionned datum resulting from i11, i12 and i21

(see Figures 1 and 2), with the possibility degree 0.78 and the necessity degree 0.53 (obtained for the value “Lovre”, as $1 - 0.47$).

The use of the necessity degree, then of the possibility degree in case of ex aequo, to order the results, is proposed in [3]. Another method is frequently used: the Pareto order, in which a result r_1 is ordered before a result r_2 if both its necessity and possibility degrees are higher. However this order is partial, since there are cases where r_1 has a higher possibility degree than r_2 , but a lower necessity degree.

Implementation of the Flexible Querying method. By using SPARQL, queries with user’s preferences can not be directly translated, because SPARQL does not provide mechanisms to express such preference values.

In 4.1, we have proposed a two-steps procedure for the flexible query processing. Its implementation corresponds to the mechanism indicated in [10] as “defuzzification”. In [10], the authors have proposed two kinds of “defuzzification” which depend on the nature of the selection attribute values: (i) *when the values are symbolic*, the “defuzzification” consists in just removing the user preference indexes and express the selection attribute in the FILTER clause with the union operator; (ii) *when the values are numeric*, the “defuzzification” consists in taking into account only the support interval and express the inclusion condition in the FILTER clause of the query.

Example 1. Let Q1 be a flexible query with user’s preferences expressed on a numeric attribute.

$$Q1 = \{ \{ \text{PaintingDate} \}, \{ < \text{PaintingDate}, ([1500, 1550], [1450, 1600]), 1 > \} \}.$$

The “defuzzified” representation of Q1 corresponds to :

$$DQ1 = \{ \{ \text{PaintingDate} \}, \{ < \text{PaintingDate}, ([1450, 1600]), 1 > \} \}.$$

Selection of relevant data according to user preferences. In the first step of the flexible query processing the more relevant answers are selected by evaluating the SPARQL query which translates the “defuzzified” representation of the initial query.

Query result ordering. This task is performed as a post-preprocessing step by using the algorithm given in Section 4.2.

5 Conclusion

In this paper, we have proposed a method for reference fusion and for flexible querying of the fusionned references.

The fusion method allows computing for each candidate value of a given attribute a confidence degree. It is obtained by a combination of different criteria that are related to the syntactic nature of the values but also to the features of the data sources. The sets of values that are assigned to attributes are expressed by fuzzy sets. By exploiting the value ranking, different fusion modalities can be specified for the presentation of the results to the users. The flexible querying method allows to query these fuzzy data by using fuzzy queries. These queries are fuzzy in the sense that they express users’ preferences by allowing fuzzy values in the selection criteria as well as the ranking of the selection criteria.

There are some studies on reference reconciliation that deal to a certain degree with reference fusion. In [7], a rule-based language is used by the administrator of the integration system to define different functions based on the reliability of the data sources. Thus, the fusion can be achieved without considering the values coming from the other sources. In [6], the authors propose a new operator, used in SQL queries, which takes as arguments a set of pre-defined functions (e.g. max, min). Compared to these studies on reference fusion, our fusion approach detects and suggests a resolution of the conflict between values by proposing a value ranking according to the computed confidence value. Unlike [6], our fusion method does not need any extension of the query language to be able to query the fusionned data.

In the context of soft querying [11], fuzzy sets, which are more generally used to represent concepts whose borders are not strictly delimited, can be used to define flexible selection criteria, by associating a preference degree with every candidate value. The proposed approach of flexible querying combines both, i.e. queries with preferences to query ill-known data, by using fuzzy pattern matching [3] which is a novelty in the framework of reference fusion.

In this paper we have shown the applicability of our methods, but we plan in a short term to experiment them on larger scale data to evaluate their efficiency. Another perspective is to extend the approach to handle hierarchically organized concepts and data, taking into account the domain ontology. Finally, we are interested in exploiting the history of users' queries to manage users' profiles for personalization purposes.

References

1. Zadeh, L.: Fuzzy sets. *Information and Control* 8, 338–353 (1965)
2. Zadeh, L.: Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems* 1, 3–28 (1978)
3. Dubois, D., Prade, H.: Tolerant fuzzy pattern matching: an introduction. In: Bosc, P., Kacprzyk, J. (eds.) *Fuzziness in Database Management Systems*, pp. 42–58. Physica, Heidelberg (1995)
4. Dubois, D., Prade, H.: *Possibility Theory - An Approach to Computerized Processing of Uncertainty*. Plenum Press, New York (1988)
5. Saïs, F.: *Semantic Data Integration guided by an Ontology*. Ph.D thesis, université de Paris-Sud (2007)
6. Bleiholder, J., Naumann, F.: Declarative data fusion – Syntax, semantics, and implementation. In: *Proc. of the 9th East European Conference on Advances in Databases and Information Systems* (2005)
7. Papakonstantinou, Y., Abiteboul, S., Garcia-Molina, H.: Object fusion in mediator systems. In: *VLDB, San Francisco, CA, USA*, pp. 413–424 (1996)
8. Cohen, W.W., Ravikumar, P., Fienberg, S.E.: A comparison of string distance metrics for name-matching tasks. In: *IIWeb*, pp. 73–78 (2003)
9. Mazzieri, M.: A fuzzy rdf semantics to represent trust metadata. In: *1st Workshop on Semantic Web Applications and Perspectives* (2004)
10. Haemmerlé, O., Buche, P., Thomopoulos, R.: The miel system: Uniform interrogation of structured and weakly-structured imprecise data. *J. Intell. Inf. Syst.* 29(3), 279–304 (2007)
11. Bosc, P., Liétard, L., Pivert, O.: Soft querying, a new feature for database management system. In: Karagiannis, D. (ed.) *DEXA 1994. LNCS*, vol. 856, pp. 631–640. Springer, Heidelberg (1994)

Mediation-Based XML Query Answerability

Hong-Quang Nguyen¹, Wenny J. Rahayu¹, David Taniar², and Kinh Nguyen¹

¹ Department of Computer Science and Computer Engineering,

La Trobe University, VIC 3086, Australia

{h20nguyen,wenny,kinh.nguyen}@cs.latrobe.edu.au

² School of Business Systems

Monash University, Clayton, VIC 3800, Australia

david.taniar@infotech.monash.edu.au

Abstract. This paper presents a novel mediation-based query answering approach which allows users (1) to reuse their own predefined queries to retrieve information properly from their local data source, and (2) to reformulate those queries in terms of remote data sources in order to obtain additional relevant information. The problem of structural diversity in XML design (e.g. nesting discrepancy and backward paths) makes it difficult to reformulate the queries. Therefore, we highlight the importance of precise query rewriting using *composite concepts* and *relations* of the mediated schema. Our experimental evaluations on real application datasets show that our approach effectively obtains correct answers over a broad diversity of schemas.

1 Introduction

Traditional mediation-based XML query answering approaches allow users to directly pose queries over the global mediated schema; then, the queries are reformulated in terms of each local schema to retrieve answers. We refer it as a *centralized query model*, which suffers several disadvantages. The users at the local data source have to double their effort in learning and working on two different systems: their own local system and the integrated system. Posing the queries over the mediated schema, the users may not be able to find some kinds of data which are very specific to their local data source and are absent on the mediated schema. Further, only a limited number of queries built on the mediated schema are available for the users to obtain their desired answers.

To overcome such disadvantages, we propose a *decentralized query model* in Fig. 1. In this model, the users should still be able to work on their own defined queries on their local schema, increasing the reuse of their queries and reducing the effort of learning the new integrated system. At the same time, they obtain more additional relevant answers from other remote data sources. The users of organization S can use their own familiar *existing* queries Q_S to pose on their own data source S . At the same time, the query is also propagated to the integrated system to ask for extra information from other remote sources. The final answer $A_S \cup (A_{R_1} \cup A_{R_2})$ obtains more answers $(A_{R_1} \cup A_{R_2})$ from R_1 and

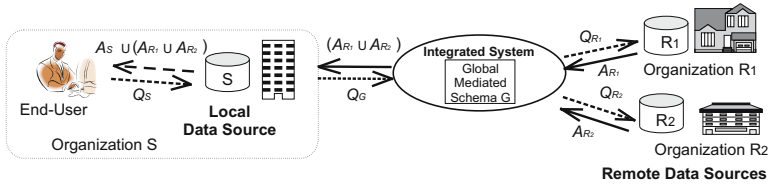


Fig. 1. Decentralized Query Requests

R_2 , and combines with desired answer from S . The queries are automatically reformulated, freeing users from the complexity of using multiple systems and making transparent the complex process of query answering.

2 Related Work and Motivation

Several query answering approaches have been proposed in peer data management systems, such as Hyperion [1], PeerDB [2] and Piazza [3]. Peer-based query answering approach can benefit from straightforward pair-wise mappings between two data sources. However, it becomes more complicated when dealing with a large collection of heterogeneous data sources. It may require much human intervention [2] and make the query reformulation process more complicated due to the directional mappings [3]. Mediation-based query rewriting approaches [4,5] have been developed based upon the global schema and mediation mapping rules. However, the global schema is not rich enough to capture semantically hierarchical structure, such as backward paths [5]. To the best of our knowledge, existing approaches mainly use queries available on the global schema rather than reusing the queries available at local data sources. Thus, our approach is novel in the way users pose their queries over the local data source.

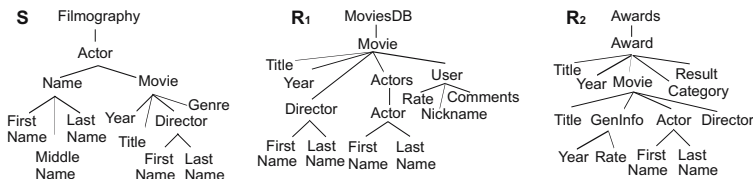


Fig. 2. Case Study: Simplified Schema Repository

Fig. 2 presents a simplified schema repository consisting of three data sources represented by schema trees $D = \{S, R_1, R_2\}$ on *movies* domain. Suppose users from organization S want to search for information from their local data source S , and from remote data sources R_1 and R_2 . Consider the following user queries:

Query 1: Find title and year of movies in which Jackie Chan casts as an actor since 2001.

Query 2: Find movies in which Chan is an actor. The detail of the movies include information on directors.

Query 3: Find title of movies in which Chan participates.

With the knowledge of the structure of their local data source S , users may find it simple to write four corresponding XPath queries as follows:

```

Q1S = for $m in /*/Actor[Name/FirstName/text()="Jackie" and
    */LastName/text()="Chan"]/Movie[Year >= 2001]
    return <result>$m/Title $m/Year</result>
Q2S = //Actor[*/LastName/text()="Chan"]/Movie
Q3S = //Actor[//LastName/text()="Chan"]/Movie/Title

```

The queries defined above may face a typical problem of *label conflicts* if the remote sources R_1 and R_2 contain different labels conveying the same meaning; e.g. *film* vs. *movie*. The label conflict may arise from label abbreviation or different naming conventions (e.g. *FirstName* or *first-name*). Another problem is *nesting discrepancy* which happens in $Q1_S$ when a node can be represented as either a child or as a descendant of another node but have the same meaning. For example, *Actor* has *FirstName* as a child in R_1 but as a descendant in S . The discrepancy may also result from *backward paths*, such as *Actor/Movie* in S vs. *Movie//Actor* in R_1 and *Movie/Actor* in R_2 . Answering query $Q2_S$ will return the target node *Movie* and all of its subtrees; that is, the subtree with root *Director* is implicitly included in the result. Such *implicit inclusion* of a subtree may lead to two main problems: (i) the answer may not include desired data in other remote schemas (e.g. *Director* designed as ancestor or sibling of *Movie*); (ii) the answer may include redundant or irrelevant information such as subtree with root *User* from R_1 . Query $Q3_S$ returns movie's title in which *Chan* participates as either an *actor* or a *director*. *Actor* and *Director* have ancestor/descendant relationship in S but have sibling/cousin relationship in R_1 and R_2 , making the query answering more difficult. The query reformulation has to identify such contextual meaning for the correct rewriting. Our novel query reformulation approach is proposed to solve all of the problems above.

3 Generating Mediated Schemas

This paper focuses on how to effectively rewrite queries using the global mediated schema, as opposed to the process of creating the global schema. Yet it is important to briefly describe our schema mediation method to make clearer our proposed query answering approach. Frequent subtree mining approach [6] is used to generate the mediated schema. We define *frequency* of a subtree T' of in D , denoted by $f(T')$, is the percentage of the number of trees T in D containing at least one subtree T' . Subtree T' is called *frequent* in D if its frequency is more than or equal to a user-defined minimum support threshold: $f(T') \geq \text{minsup}$.

A labeled node in a schema tree represents a concept in the real world. A concept which appears frequently in D is considered to be important and of interest in that domain. Frequent concepts being selected as candidate nodes

ensure that only important concepts are retained in the final mediated schema. Omitting infrequent concepts allows us to early remove unimportant or irrelevant concepts. At this step, mediation mappings between constituent schemas and the global schema are retained for the subsequent query reformulation process. We resolve the problem of label conflicts by tokenizing the labels, expanding their abbreviations, and eliminating unimportant parts (such as hyphens and prepositions) [7]. Although frequent concepts found above are of interest, they do not carry much information because they are just single nodes in isolation. They will become more meaningful when they are semantically connected with each other, forming a larger concept, called *composite concept* (CC for short).

Definition 1 (Composite/Elementary Concept). A composite concept $X(x_1, x_2, \dots, x_p)$ is a tree of height 1, which consists of X as the root and $\{x_1, x_2, \dots, x_p\}$ as a set of X 's children. When standing alone, X is referred to as CC-name, and x_i ($i = 1..p$) is called elementary concept (EC). The following conditions hold: $(X \text{ is a non-leaf in } D) \wedge (x_i \text{ is a leaf in } D) \wedge f(X) \geq \text{minsup} \wedge f(x_i) \geq \text{minsup} \wedge f(X//x_i) \geq \text{minsup}$.

The purpose of a CC $X(x_1, x_2, \dots, x_p)$ is to bear a coherent semantics representing a real world entity in the domain of interest. The definition on CC also describes what a CC is and how to mine such substructure from our schema sources. For example, **Movie(Title, Year)**, as a whole, is a CC, in which **Movie** is CC-name, and **Title** and **Year** are two ECs. Examining ancestor-descendant path between X and x_i (i.e. embedded subtrees) allows us to solve the problem of nesting discrepancy. It is the structural context of x_i under X that helps clarify the meaning of both X and x_i in the hierarchy. Only such meaningful frequent relationships between two frequent concepts are kept for the construction of the mediated schemas. The mined CCs are currently disconnected from each other. In the real world, the existence of CCs becomes clearer if they interact with each other. The most popular hierarchical path is the forward path because the design of XML structure is generally based on top-down design approach. However, we observe that the same meaning can be represented by $X//Y$ or $Y//X$, which are referred to as *forward* and *backward* paths. Ignoring such semantic similarity will cause information loss. Thus, it is critical to mine both forward and backward paths so that the final mediated schemas become more comprehensive.

Definition 4 (Relation). Given two CCs $X, Y \in \mathcal{C}$. A relation from X to Y , denoted as $X \rightarrow Y$, is defined as a frequent ancestor-descendant path from X to Y . A relation between X and Y can be bidirectional, denoted as $X \leftrightarrow Y$. Let $\mathcal{C}, \mathcal{R}$ denote a set of CCs and a set of relations mined from D , respectively. A relation has to satisfy two rules as follows: (1) Forward paths only: $f(X//Y) \geq \text{minsup} \wedge f(Y//X) = 0 \wedge X \rightarrow Y \in \mathcal{R}$; (2) Backward paths: $f(X//Y) > 0 \wedge f(Y//X) > 0 \wedge (X \rightarrow Y \in \mathcal{R} \wedge Y \rightarrow X \in \mathcal{R})$.

To extract relations, we mine frequent ancestor-descendant $X//Y$ (or $Y//X$) which will be included into the final mediated schema (e.g. **Movie** $\rightarrow$ **Director**). A relation can be either forward pathMining relations based on both forward and backward paths gives a less strict condition by using the sum of the two

kinds of paths. This allows more chance of an ancestor-descendant path between X and Y to become frequent; e.g. $\text{Actor} \leftrightarrow \text{Movie}$.

Definition 5 (Mediated Schema). Let $\mathcal{C}$ and $\mathcal{R}$ denote a set of mined CCs and a set of mined relations, respectively. A global mediated schema is a triple $G = \langle \text{ROOT}, \mathcal{C}, \mathcal{R} \rangle$, where ROOT is the root of G .

We build the mediated schema $G = \langle \mathcal{C}, \mathcal{R} \rangle$, where $\mathcal{C} = \{\text{Movie}(\text{Title}, \text{Year}, \text{Rate}), \text{Actor}(\text{FirstName}, \text{LastName}), \text{Director}(\text{FirstName}, \text{LastName})\}$, and a set of relations $\mathcal{R} = \{\text{Actor} \leftrightarrow \text{Movie}, \text{Movie} \rightarrow \text{Director}\}$, with $\text{minsup} = 0.6$. The containment relationship between a CC-name and an CC (such as between **Movie** and **Title**) can be represented by parent-child or ancestor-descendant path in source schemas. Semantic correspondence between constituent schemas and the global schema are captured in the mediation mappings.

4 Query Reformulation Using Mediated Schema

Our approach uses XPath 2.0 syntax [8], a subset language of XQuery, to define the XPath queries. Basically, a XPath expression is defined as a list of nodes and location paths: $p = o_1 n_1 o_2 n_2 \dots o_k n_k$, where o_i is a location path operator in $\{/, //, *\}$, and $n_i \in \text{TagSet}$ is a node in a set of tags. Given an XPath query, we replace variables in predicates and target nodes with values of binding variables in **for** expression. A multi-path query Q_S can be decomposed into different single sub-queries, each of which is reformulated as the normal single-path query.

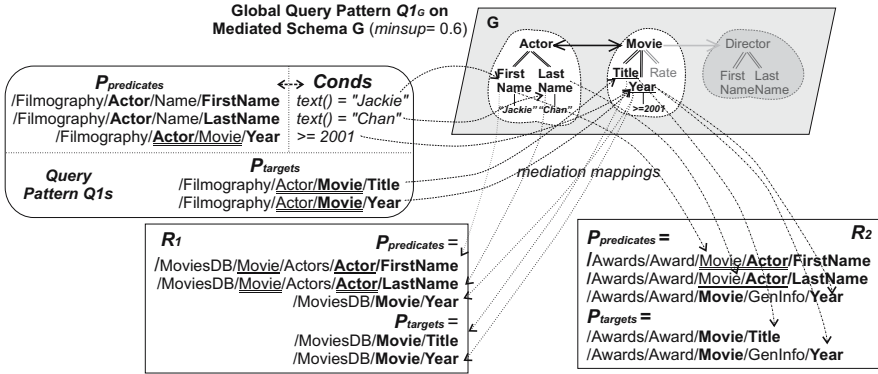
Next, we decompose all paths in predicates into path expressions and selection conditions. Let Targ_S be a set of paths locating target nodes to be retrieved, Pred_S be a set of predicate paths, and Cond_S contains be a set of conditions in predicates. Path expressions of predicates of source query posed over S are included in Pred_S , their corresponding selection conditions are included in Cond_S , and path expressions of target nodes are kept in Targ_S . Then, we expand all of the abbreviated paths with ‘*’ and ‘//’ in source query Q_S into its equivalent unabbreviated forms. We define a *query pattern* [9] of Q_S as a triple $\langle \text{Targ}_S, \text{Pred}_S, \text{Cond}_S \rangle$. For example, the query pattern of Q_{1S} : $\text{Targ}_S = \{ / * / \text{Actor} / \text{Movie} / \text{Title}, / * / \text{Actor} / \text{Movie} / \text{Year} \}$, $\text{Pred}_S = \{ / * / \text{Actor} / \text{Name} / \text{FirstName}, / * / \text{Actor} [\text{Name} / \text{LastName}, / * / \text{Actor} / \text{Movie} / \text{Year}] \}$, $\text{Cond}_S = \{ \text{text}() = \text{“Jackie”}, \text{text}() = \text{“Chan”}, >= 2001 \}$.

Given the query pattern of Q_S , we generate the query pattern of Q_G on the mediated schema G by extracting CCs and relations from Q_S . The query pattern of Q_G will be used for the query reformulation on the remote sources in Fig. 3.

a) Identifying Composite Concepts in Source Queries:

We define a function $\text{findCC}: \text{Pred}_S \cup \text{Targ}_S \rightarrow \mathcal{C}$ which extracts all CCs from an XPath query by mapping every path in sets of predicates and target nodes into a set of CCs $\mathcal{C}$. Let $p = n_1 / n_2 / \dots / n_k$ be a path of k nodes in $(\text{Pred}_S \cup \text{Targ}_S)$. The last node n_k in path p may belong to one of three cases:

i) n_k is an EC of a CC $n_i(n_k)$: Function findCC first seeks the CC-name n_i to associate with the EC n_k (where $i < k$), and returns the CC $n_i(n_k)$. For example,



the CC Movie(Title) is extracted from the target node $\text{/Filmography/Actor/Movie/Title}$ in the query $Q1_S$.

ii) n_k is a CC-name of a CC in $\mathcal{C}$. If n_k is a predicate node in Pred_S , findCC simply returns n_k as a CC-name of a CC in G without returning its ECs. The meaning of the predicate n_k is to check the existence of n_k for returning the answer. Node n_k is actually a non-leaf node which contains a substructure and does not store any value.

If n_k is a target node of the query Q_S , the whole subtree rooted at n_k will be included in the answer. It is important to note that that subtree may include zero or more subtrees (say $n_{k'}$) via implied inclusion, in which $n_{k'}$ corresponds to a CC on G . Thus, findCC will perform two tasks. First, it seeks CC n_k and all of its ECs from G . An EC on G may or may not corresponds to any node in S and/or R_i . If the EC does not belong to G , it is considered not of interest to users because it is unpopular and too specific to the local schema S . For example, $\text{/Filmography/Actor/Movie}$ in $Q2_S$ has the last node Movie as a target node which is the CC-name of $\text{Movie(Title, Year, Rate)}$; Rate is an additional EC that does not appear in S but is included in the answer. Further, findCC extracts the subtree $n_{k'}$ if it exists on the mediated schema G and other remote schemas. In addition to Movie found for $Q2_S$, findCC also includes CC $\text{Director(FirstName, LastName)}$ in the target nodes. Such consideration resolves problem of implicit inclusion.

iii) n_k does not belong to any CC: n_k is neither an EC nor CC-name. This happens when n_k is too specific to the local schema S and is not common among other remote schemas. If n_k is a target node, it cannot be retrieved from other remote sources due to its absence in the global schema G . The query reformulation process will stop without further querying other remote sources. If n_k belongs to a predicate, the query rewritten at G does not contain that predicate. Therefore, the answer returned from the mediated schema (and hence, from remote schemas) is either superset or subset of the desired answer. The identification of CCs in source queries allows non-CC-name nodes to occur between a CC and an

EC. In other words, an EC can be a child node or a descendant node of a CC, addressing the problem of nesting discrepancy.

b) Identifying Relations in Source Queries:

In this section, we identify all relations between CCs from source query Q_S . The relations in Q_S are important to construct path patterns between CCs in queries from remote sources. We define function $findRel : (Pred_S \cup Targ_S) \rightarrow \mathcal{R}$ which maps each path $p \in Pred_S \cup Targ_S$ into a relation $r \in \mathcal{R}$. Function $findRel$ derives relations r between CCs rooted at n_i and n_j from path p based upon the following conditions: (i) $\exists n_i, n_j \in roots(\mathcal{C})$ such that $n_i/n_j \in p$. (ii) $\nexists n_t \in p, n_i, n_j, n_t \in roots(\mathcal{C})$ such that $n_i/n_t/n_j \in p$.

Path p contains more than one node indicating CC (such as n_i and n_j) separated by location paths and non-CC nodes. n_i and n_j forms one relation on p such that there exist no other CC n_t between their path. Relations extracted from source queries establish query pattern Q_G over multiple CCs.

c) Query Reformulation:

Query reformulation involves in using mediated schema to map all paths of source query patterns into their corresponding paths of remote query patterns. Generating mediated schema (section 3) provides mediation mappings between mediated schema and each constituent schema. In other words, we map each element of source query pattern $(Pred_S, Targ_S, Cond_S)$ into a correspondence in remote query pattern $(Pred_{R_i}, Targ_{R_i}, Cond_{R_i})$, respectively. Consider query $Q1_S$. The predicate `/Filmography/Actor/Movie/Year>=2001` of $Q1_S$ is mapped into a conditional EC `Movie(Year)>=2001` of G , which corresponds to one or more remote query patterns: $Q1_{R_1} = /MoviesDB/Movie/Year>=2001$ and $Q1_{R_2} = /Awards/Award/Movie/GenInfo/Year>=2001$. The same process is applied to other paths of source query patterns.

Next, we compose remote query from its query pattern (represented by $Pred_{R_i}, Targ_{R_i}, Cond_{R_i}$). To do this, we merge all of the paths and conditions from its query pattern by defining two generic functions: *prefix* and *suffix*. Function *prefix* is defined to map a set of paths $P = \{p_1, p_2, \dots, p_k\}$ into a single path p' such that p' is the common path (from the root) shared among elements of P . Function *suffix* returns the suffix of path p_i which is *not shared* with other paths in P . In other words, $suffix(p_i, P) = p_i - prefix(P)$. For example, suppose $P = \{p_1, p_2\} = \{/n_1/n_2/n_3, /n_1/n_3\}$, function *prefix*(P) returns shared paths between p_1 and p_2 , i.e. $prefix(P) = /n_1$ whereas $suffix(p_1, P) = n_2/n_3$ and $suffix(p_2, P) = n_3$.

Definition(Query Answerability). Given a set of XML documents D , query answerability $\mathcal{Q}$ is a measure to determine the ability to find correct answers for a query posed over D , and is defined as the proportion of number of correct answers found to the number of correct answers: $\mathcal{Q} = \frac{\#Correct\ Answers\ Retrieved}{\#Correct\ Answers}$

The number of correct answers found from D is less than or equal to the number of real answers; thus, $\mathcal{Q}$ is a real number between 0 and 1. The higher value of $\mathcal{Q}$ is, the more correct answers are found.

5 Experimental Evaluation

We perform several experiments on *Movies*¹ – a collection of real applications data collected from Yahoo! Movies (*movies.yahoo.com*) and Internet Movies Databases (*imdb.com*). The Movies dataset contains 1,312 documents of diverse structural designs and a total of 64,706 nodes. The smallest tree has 14 nodes while the largest one contains 91 nodes; on average, there are 49.32 nodes per tree. The height of the trees ranges from 4 to 10 with an average of 6.81.

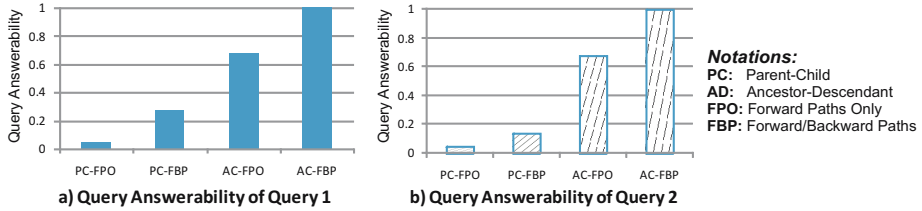


Fig. 4. Query Answerability measures the ability to find correct answers for Query 1-2

We use query answerability to evaluate the quality of different mediation-based query answering approaches. Four different approaches are classified based on the combination of tree traversal types: parent-child (PC) vs. ancestor-descendant (AD), and forward-paths only (FPO) vs. both forward and backward paths (FBP). Fig. 4 presents the query answerability of four kinds of experiments. Among the four experiments, our approach which is based on AD-FBP provides the best query answerability. It obtains complete answers for Query 1 ($Q1_S$) from the dataset with 1.5, 3.67, and 20.9 times improvement compared to PC-FPO, PC-FBP and AD-FPO, respectively. PC-FPO approach returns the worst result with least correct answers for Query 1 and Query 2 (query answerability ≈ 0.048) because it limits its search within parent-child paths and forward paths only. When the same concept is expressed as a descendant of a node, the mediated schema based on PC-FPO (e.g. PORSCHE [7]) cannot discover such semantic matching. Hence, its corresponding queries cannot find the correct answers due to the problem of both nesting discrepancy and backward paths. Other traditional schema matching approaches (such as COMA++ [10], Similarity Flooding [11]) only perform pair-wise element matching between two schemas without examining structural context. They do not produce mediated schema for semi-structured documents; thus, they do not belong to any of these approaches. Dealing with nesting discrepancy gives AD-FPO approach an improvement of 2.5 times and 4.8 times the query answerability of Query 1 and Query 2, respectively, based on PC-FBP approach. As a result, query answering based on mediation approaches which resolve both nesting discrepancy and backward paths provides the most comprehensive answers from large collection of heterogeneous XML documents.

¹ <http://homepage.cs.latrobe.edu.au/h20nguyen/research>

6 Conclusion

In this paper, a mediation-based query reformulation approach is proposed to reuse the existing XPath queries to retrieve more information from other remote sources. It also frees users from the complexity of using multiple systems at the same time. The mediated schema enables us to efficiently reformulate remote queries. Further, it helps prevent source queries with too specific selection conditions from being propagated to remote sources. We resolve the problem of semantic conflicts in labels. Our approach supports users to query and obtains information from heterogeneous data sources of different structures, including nesting discrepancy and backward paths. There are several opportunities for future work. We plan to extend our work to cover other components in structured query languages such as XQuery and XQueryX. Also, we are going to work on the evolution of queries when the global schema evolves.

References

1. Arenas, M., Kantere, V., Kementsietsidis, A., Kiringa, I., Miller, R.J., Mylopoulos, J.: The hyperion project: from data integration to data coordination, vol. 32, pp. 53–58 (2003)
2. Ooi, B.C., Shu, Y., Tan, K.L.: Relational data sharing in peer-based data management systems. *SIGMOD Record* 32(3), 59–64 (2003)
3. Tatarinov, I., Halevy, A.: Efficient Query Reformulation in Peer Data Management Systems. In: *SIGMOD* (2004)
4. Halevy, A.Y., Etzioni, O., Doan, A., Ives, Z.G., Madhavan, J., McDowell, L., Tatarinov, I.: Crossing the structure chasm. In: *CIDR* (2003)
5. Madria, S.K., Passi, K., Bhowmick, S.S.: An xml schema integration and query mechanism system. *Data Knowl. Eng.* 65(2), 266–303 (2008)
6. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules in large databases. In: *VLDB*, pp. 487–499 (1994)
7. Saleem, K., Bellahsene, Z., Hunt, E.: PORSCHE: Performance ORiented SCHEma mediation. *Information Systems* (2008)
8. Melton, J., Buxton, S.: *Querying XML: XQuery, XPath, and SQL/XML in Context*. Elsevier, Amsterdam (2006)
9. Yang, L.H., Lee, M.L., Hsu, W., Acharya, S.: Mining Frequent Query Patterns from XML Queries. In: *DASFAA 2003* (2003)
10. Do, H.H.: *Schema Matching and Mapping-based Data Integration*. Ph.D thesis, Dept of Computer Science, University of Leipzig, Germany (2006)
11. Melnik, D., Rahm, E., Bernstein, P.A.: Rondo: A programming platform for generic model management. In: *SIGMOD* (2003)

Ontology Matching Supported by Query Answering in a P2P System

François-Élie Calvier and Chantal Reynaud

LRI, Univ Paris-Sud & INRIA Saclay - Île-de-France
4, rue Jacques Monod - Bât. G
91893 Orsay Cedex, France
{francois.calvier, chantal.reynaud}@lri.fr
<http://www.lri.fr/iasi>

Abstract. In this article we propose methods to automate the extraction of alignments or mappings between ontologies by using query answering in the peer data management system (PDMS) SomeRDFS which uses a data model based on RDF(S). Query answering is composed of a rewriting and an evaluation phase. We show that query answering provides information that offer automated support for discovering new mappings. It is used to generate either mapping shortcuts corresponding to mapping compositions or mappings which can not be inferred from the network but yet relevant. The strategy followed by a peer and filtering criteria defined by the administrator of a peer are used to select the most relevant mappings to be represented.

Keywords: ontology matching, peer-to-peer, data management systems.

1 Introduction

The use of peer-to-peer systems consists of querying a network of peers for information. Queries are asked to one of the peers. The peers communicate to each other to answer queries in a collective way.

We focus on the ontology matching process in the peer data management system (PDMS) SomeRDFS [1] in the setting of the MediaD project¹. Ontologies are the description of peers data. Peers in SomeRDFS interconnect through alignments or mappings which are semantic correspondences between their own ontologies. Thanks to its mappings a peer may interact with the others in order to answer a query. No peer has a global view of the data management system. Each peer has its own ontology, its own mappings and its own data. It ignores the ontology, the mappings and the data of the other peers. In this setting, our work aims at increasing the mappings of the peers in SomeRDFS in order to increase the quantity and the quality of the answers of the whole data management system. We are interested in identifying two kinds of mappings: mapping shortcuts corresponding to a composition of pre-existent mappings and mappings which can not be inferred from the network but yet relevant. In both cases, the idea is to make the generation of mappings automatically supported by query answering. We

¹ Research project funded by France Telecom R&D.

take also into account the strategy followed by a peer which is important to select the most relevant mappings to be represented.

A survey of usual methods for automating the generation of mappings is presented in [2], [3] and [4] but very little work has been done on this problem in a P2P environment taking into account its distinguishing features, in particular the distributed global schema or ontology. For example, in Piazza [5], tools and techniques developed to simplify and assist mapping creation [6] assume that the whole ontology of a peer can be known by any other peer. In [7], shared ontologies are used. Unlike these approaches we propose techniques to generate mappings between entirely decentralized ontologies. Each peer has its own ontology and ignores the ontology of the others. Furthermore, queries are not specific and are not routed to all peers over the network as in [8]. They are usual queries submitted to SomeRDFS and usual SomeRDFS reasoning mechanisms are reused. These mechanisms are exploited in two ways. We generate mapping shortcuts corresponding to mapping compositions. These mappings are produced from the routing of the queries over the network. The problem we are interested in is then to select the mappings useful to be represented. This differs from works whose goal is to produce mapping composition algorithms [9]. Furthermore, relevant elements to be mapped are selected. They share a common interpretation context making the alignment process easier by avoiding misinterpretations. The identification of context used for focusing the matching process is also a solution provided in [10] based on the analysis of the interactions between agents that are assumed to follow conventions and pattern. We share this idea but differ in its accomplishment. Finally, our work can be seen as a complement of [11] whose goal is to identify methods to establish global forms of agreement from a graph of local mappings among schemas. This work assumes that skilled experts supported by appropriate mapping tools provide the mappings. Our approach provides automated techniques to generate these mappings.

The paper is organized as follows. Section 2 describes the fragment of RDF(S) that we consider as data model for SomeRDFS and query answering. Section 3 shows how the query answering process can be used to discover new mappings according to a given strategy of a peer. We conclude and outline remaining research issues in Section 4.

2 Data Model and Query Answering in a SomeRDFS PDMS

In SomeRDFS ontologies and mappings are expressed in RDF(S) and data are represented in RDF. (Sub)classes, (sub)properties can be defined. Domain and range of properties can be typed. Classes inclusion, properties inclusion, domain and range typing of a property are the only authorized constructors. This language, denoted core-RDFS, has a clear and intuitive semantics. It is constructed on unary relations that represent classes and binary relations that represent properties. The logical semantics of core-RDFS, expressed in description logic (DL notation) and its translation in first-order logic (FOL), is given in Table 1.

Peers ontologies are made of core-RDFS statements involving the vocabulary of only one peer. We use the notation $\mathcal{P}:R$ for identifying the relation (class or property) R of the ontology of the peer $\mathcal{P}$. A mapping is an inclusion statement between classes or properties of two distinct peers (cf. Table 2 (a) and (b)) or a typing statement of a

Table 1. Core-RDF(S) operators

Operator	DL Notation	FOL translation
Class inclusion	$C_1 \sqsubseteq C_2$	$\forall X, C_1(X) \Rightarrow C_2(X)$
Property inclusion	$P_1 \sqsubseteq P_2$	$\forall X \forall Y R_1(X, Y) \Rightarrow R_2(X, Y)$
Domain typing of a property	$\exists P \sqsubseteq C$	$\forall X \forall Y, P(X, Y) \Rightarrow C(X)$
Range typing of a property	$\exists P^- \sqsubseteq C$	$\forall X \forall Y, P(X, Y) \Rightarrow C(Y)$

Table 2. Mappings

Mappings	DL Notation	FOL translation
(a) Class inclusion	$\mathcal{P}_1:C_1 \sqsubseteq \mathcal{P}_2:C_2$	$\forall X, \mathcal{P}_1:C_1(X) \Rightarrow \mathcal{P}_2:C_2(X)$
(b) Property inclusion	$\mathcal{P}_1:P_1 \sqsubseteq \mathcal{P}_2:P_2$	$\forall X \forall Y, \mathcal{P}_1(X, Y) \Rightarrow \mathcal{P}_2(X, Y)$
(c) Domain typing of a property	$\exists \mathcal{P}_1:P \sqsubseteq \mathcal{P}_2:C$	$\forall X \forall Y, \mathcal{P}_1:(X, Y) \Rightarrow \mathcal{P}_2:C(X)$
(d) Range typing of a property	$\exists \mathcal{P}_1:P^- \sqsubseteq \mathcal{P}_2:C$	$\forall X \forall Y, \mathcal{P}_1:(X, Y) \Rightarrow \mathcal{P}_2:C(Y)$

property of a given peer with a class of another peer (cf. Table 2 (c) and (d)). Mappings are defined as core-RDFS statements involving vocabularies of different peers.

The specification of the data stored in a peer is done through the declaration of assertion statements relating data of a peer to relations of its vocabulary. The DL notation and the FOL translation of assertion statements are $C(a)$ and $P(a, b)$ where a and b are constants.

Query answering is a two-step process: first, queries are rewritten in a set of more specific queries. The set of all the rewritings of a query can be obtained from the conjunctions of the rewritings of each relation (property or class) of the query. Then, every rewriting is evaluated to get corresponding data. Users can pose unary, conjunctive or disjunctive queries. In case of conjunctive queries, rewritings are obtained from the conjunctions of the rewritings of each relation of the original query. In case of disjunctive queries, each disjunction is managed as a unary query.

3 Exploiting SomeRDFS Reasoning

SomeRDFS reasoning, in particular, query answering can offer an automated support for discovering new mappings. We propose in this section a method to guide the ontology matching process based on query answering. Query answering is used to generate mapping shortcuts and to identify relations, denoted target relations, which are starting points in the mapping discovering process. These relations allow identifying relevant mapping candidates limiting that way the matching process to a restricted set of elements. Discovered mappings can be relevant or not according to the strategy involved in the PDMS. Thus, in a first sub-section we present the different strategies that can be followed by a peer. In the next sub-section, we present how mapping shortcuts and target relations can be identified using query answering. In the last sub-section, we describe the techniques used to obtain a set of relevant mapping candidates from a set of target relations and corresponding to a given strategy.

3.1 Strategies of a Peer

A PDMS can be seen as a very large data management system with a schema and data distributed through, respectively, the union of the peer ontologies and mappings, and

the union of the peer storage description. It can also be viewed as a system where many peers each with its own ontology, mappings and data, choose to share data. In any case each peer has to access knowledge of the other peers. Having more mappings can be beneficial for three reasons. It is a way to access new data sources and so obtaining richer answers. It is a way to allow more precise queries assuming users are able to pose queries using the relations in mappings belonging to the vocabulary of distant peers. Finally, it is a way to make the PDMS steadier because less dependant of the comings and leavings of the peers in the network. Thus, any peer may decide to increase the number of its mappings. It can decide to look for new mappings whatever they are (the *default* strategy denoted S_1) or to look for particular mappings: either new mappings involving peers not yet logically connected to it (the *not yet connected peers* oriented strategy denoted S_2) or mappings involving peers already logically connected to it (the *connected peers* oriented strategy denoted S_3). Two peers are logically connected if there exists a mapping between their two ontologies. The choice of one of these strategies depends on the number of already connected peers and on the number of mappings involving a given peer.

3.2 Using Query Answering

Mappings Shortcuts Discovery

A mapping shortcut is a composition of mappings. Mapping shortcuts consolidate PDMSs by creating direct links between indirectly connected peers. We could imagine automatically combining mappings in order to obtain shortcuts. Indeed, given a peer $\mathcal{P}$ systematic queries corresponding to each of its relation allows to identify mapping shortcuts involving each of them. However, this method generates a lot of traffic on the network and all the mappings obtained this way are not always useful. Mapping shortcuts are useful when some peers disappear from the PDMS. However, they do not lead to more answers and they add caching in the rewriting process.

We propose a two-step automatic selection process. We first identify potentially useful mappings shortcuts exploiting query answering. In this step, the goal is to retain only mappings which would be useful in the rewriting process with regard to the queries really posed by users to the peer $\mathcal{P}$. Then we propose a second selection step based on filtering criteria.

To achieve the first step we need to distinguish the rewriting and evaluation phases of query answering. Query answering will not be a unique and global process anymore but two connected processes which can be separated if needed. Indeed, users do not always find the right needed relations in the ontology of the queried peer. In that case, they choose other relations among the relations of the queried peer. However, if a more specific relation is queried all the required data will not be obtained. On the other hand, if a more general relation is asked, all the required data will be obtained but these data will be mixed to others. For example, a user may need asking $\mathcal{P}_1$ for instances of *SteelSculptor*. Such a query can not be posed because of the lack of the *SteelSculptor* relation in $\mathcal{P}_1$. The user could decide to ask for a more general relation, for example $\mathcal{P}_1:Artist(X)$. Rewritings obtained involve $\mathcal{P}_2:SteelSculptor(X)$ which is the relation he is interested in, but also $\mathcal{P}_2:WoodSculptor(X)$ and $\mathcal{P}_2:GlassSculptor(X)$. The evaluations of these two later

relations are not needed with respect to the user's expectations. Considering rewriting as a process different from evaluation allows the user to examine the results of the rewriting phase in order to select which rewritings have to be evaluated. The fact that the user selects rewritings that have to be evaluated is a good indicator of the relations he is really interested in. Thus, we propose to analyze the interactions between users and peers and to add mappings that are direct specialization links between the (more general) queried relation and the one the user has chosen to be evaluated. In this example, it would be $\mathcal{P}_2:SteelSculptor(X) \Rightarrow \mathcal{P}_1:Artist(X)$ added to $\mathcal{P}_1$. We consider that this mapping is a useful mapping shortcut. Note that if the user asks for the evaluation of several relations several mapping shortcuts will be proposed. Furthermore, in this article we do not describe discovering of mapping shortcuts based on conjunctive queries because of space limitations.

The second selection step is based on the strategy of the peer and potentially exploits filtering criteria defined by the administrator of this peer. Indeed, according to the strategy S_2 or S_3 chosen by a peer $\mathcal{P}$ only a subset of the mapping shortcuts will be considered. Then a peer may want to operate a finer selection using additional filtering criteria. The usable criteria are specific to each peer but are limited. They concern either the kind of user (member of a particular group or of a given category: permanent users, temporary users, users making an intensive use of the peer, ... assuming that the group and the category are given when a user registers) who posed the query which originated the mapping (user-criterion) or the kind of relation belonging to $\mathcal{P}$ involved in the mapping (relation-criterion). The favored relations can be indicated one by one or according to their level in the hierarchy. We can, for instance, favor mappings establishing a connection with the n last levels in the class or property hierarchy of the ontology. A value is associated to each criterion, 1, 0.5 or 0, depending on whether the involved mapping has to be more or less favored. Let us note that the same mapping can be obtained several times from different queries potentially posed by different users. The weight of the user-criterion may be different from one mapping to another but the weight of the relation-criterion will always be the same. Thus, we propose a relevance measure for the mapping shortcuts which takes into account the weight of each additional filtering criterion but also the number of times that the mapping was obtained. Table 3 gives the value of the relevance measure of a mapping shortcut m_j when this mapping has been obtained n times given a sample of studied shortcut mappings composed of M elements. $W(U_{i,j})$ is the weight of the user-criterion for the occurrence i of the mapping m_j . $W(R_j)$ is the weight of the relation-criterion for the relation R_j .

Table 3. Relevance measure of the mapping m_j with n occurrences

n : # occurrences of m_j	relevance measure
$n \geq 80\% \times M$	1
$50\% \times M \leq n < 80\% \times M$	$\text{Max}\left(0.5, \frac{\sum_{i=1}^n W(U_{i,j}) + n \times W(R_j)}{2n}\right)$
$n < 50\% \times M$	$\frac{\sum_{i=1}^n W(U_{i,j}) + n \times W(R_j)}{2n}$

Identification of Target Relations

In SomeRDFS mappings are the key notion for establishing semantic connections between the peers ontologies. They are used to rewrite queries posed in terms of a local ontology and rewritings are then run over the ontology of logically connected peers. That way, distant peers may contribute to answers. However, when a user interrogates the PDMS through a peer of his choice answers may be unsatisfactory because of a lack of specialization mappings. The problem may originate from relations called target relations, which are blocking points in query answering because they are an obstacle in achieving the strategy its peer has chosen to implement. Our objective is to identify them and to consider them as starting points in the ontology matching process. As the different strategies that we consider (cf. Section 3.1) rely on the number of logically connected peers and on the number of specialization mappings the definition of a target relation will be based on a counting function. That function will differ according to the strategy of the peer and also according to the method used to count. Indeed, given a relation R of a peer $\mathcal{P}$, the number of distant relations involved together with R in RDF(S) statements, either specialization mappings of $\mathcal{P}$ or locally inferred statements, can be calculated either with regard to the knowledge of the peer $\mathcal{P}$ or with regard to rewritings obtained from queries. This is also true when given a relation R of a peer $\mathcal{P}$ we want to compute the number of distant peers corresponding to relations involved in specialization mappings of R or locally inferred statements. The result of the counting function will be compared to a threshold that will be fixed by the administrator of the peer. When the value of the function is lower than the threshold the relation will be a target relation. We first give a general definition of a target relation. We will then precise the general definition to handle all the different cases.

Definition 1 (Target Relation). $\mathcal{P}_1:R_1$ is a target relation iff $f(\mathcal{P}_1:R_1) < t$, f being a counting function and t a threshold.

In Table 4, we precise the definition of the function f for the relation R_1 of the peer $\mathcal{P}_1$ according to the strategy chosen by the peer and according to the method used to count (cf Section 3.1).

Note that the target relations obtained using the counting function C_2 will be different from the target relations obtained using C_1 . Indeed C_2 takes into account distant relations which belong to rewritings produced by connected distant peers but not distant relations coming from disconnected peers and C_1 does the opposite.

If the strategy of $\mathcal{P}_1$ is the *default* strategy S_1 and if C_1 is used the result of $f(\mathcal{P}_1:R_1)$ is the number of distant relations specializing R_1 according to the mappings of $\mathcal{P}_1$ or specializing another relation R_k of $\mathcal{P}_1$ with $R_k \Rightarrow R_1$ locally inferred. Using C_2 the result of $f(\mathcal{P}_1:R_1)$ will be the number of distant relations belonging to the rewritings of R_1 . If this number of distant relations is lower than the threshold t then R_1 will be a target relation.

If the strategy of $\mathcal{P}_1$ is the *not yet connected peers* oriented strategy S_2 and if C_1 is used the result of $f(\mathcal{P}_1:R_1)$ is the number of distant peers involved in specialization mappings of R_1 or in statements specializing another relation R_k of $\mathcal{P}_1$ with $R_k \Rightarrow R_1$ locally inferred. If this number of distant peers is lower than the threshold t then R_1

Table 4. Definition of $f(\mathcal{P}_1:R_1)$

Method used to Strategy of a peer	C_1 (with regard to the knowledge of $\mathcal{P}_1$)	C_2 (based on rewritings)
S_1 (default strategy)	$ \{\mathcal{P}_i:R_j / [\mathcal{P}_i:R_j \Rightarrow \mathcal{P}_1:R_1]$ or $[\exists \mathcal{P}_1:R_k \text{ such that } \mathcal{P}_1:R_k \Rightarrow \mathcal{P}_1:R_1 \text{ can be}$ inferred and $\mathcal{P}_i:R_j \Rightarrow \mathcal{P}_1:R_k]\} $	$ \{\mathcal{P}_i:R_j \in RW\} $ where RW is the query rewriting set of $Q \equiv \mathcal{P}_1:R_1$
S_2 (not yet connected peers oriented strategy)	$ \{\mathcal{P}_i:R_j / \exists \mathcal{P}_i:R_j \text{ such that } [\mathcal{P}_i:R_j \Rightarrow \mathcal{P}_1:R_1]$ or $[\exists \mathcal{P}_1:R_k \text{ such that } \mathcal{P}_1:R_k \Rightarrow \mathcal{P}_1:R_1 \text{ can be}$ inferred and $\mathcal{P}_i:R_j \Rightarrow \mathcal{P}_1:R_k]\} $	$ \{\mathcal{P}_i:R_j \in RW\} $ where RW is the query rewriting set of $Q \equiv \mathcal{P}_1:R_1$
S_3 (connected peers oriented strategy)	$\min_i (\{\mathcal{P}_i:R_j / \mathcal{P}_i:R_j \Rightarrow \mathcal{P}_1:R_1$ or $[\exists \mathcal{P}_1:R_k \neq \mathcal{P}_1 \text{ such that } \mathcal{P}_1:R_k \Rightarrow \mathcal{P}_1:R_1 \text{ can}$ be inferred and $\mathcal{P}_i:R_j \Rightarrow \mathcal{P}_1:R_k]\})$	$\min_i \{\mathcal{P}_i:R_j \in RW\} $ where RW is the query rewriting set of $Q \equiv \mathcal{P}_1:R_1$

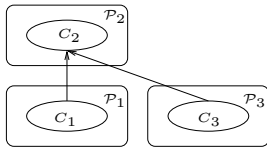
will be a target relation. Using C_2 the result of $f(\mathcal{P}_1:R_1)$ will be the number of distant peers involved in the rewritings of R_1 .

If the strategy of $\mathcal{P}_1$ is the *connected peers* oriented strategy S_3 , R_1 will be a target relation if there is at least one peer involved in a low number of specialization statements of R_1 . Thus, if C_1 is used, $f(\mathcal{P}_1:R_1)$ provides the minimum number of relations of a given distant peer specializing R_1 according to the mappings of $\mathcal{P}_1$ or specializing another relation R_k of $\mathcal{P}_1$ with $R_k \Rightarrow R_1$ locally inferred. If C_2 is used, $f(\mathcal{P}_1:R_1)$ will provide the minimum number of distant relations which belong to the rewritings of R_1 and which are involved in the mappings of $\mathcal{P}_1$.

3.3 Obtaining a Set of Relevant Mapping Candidates

Our objective is to use target relations in order to identify relevant mapping candidates, limiting that way the matching process to a restricted set of elements. In this section, we propose methods to discover new mappings from target relations. These methods are performed by a given peer given its target relations.

Target relations can allow discovering relevant mapping candidates according to two scenarios. In the first scenario (cf Figure 1), let us consider $\mathcal{P}_1$, $\mathcal{P}_2$ and $\mathcal{P}_3$ three peers with C_1 , C_2 and C_3 three classes and the following mappings: $\mathcal{P}_1:C_1(X) \Rightarrow \mathcal{P}_2:C_2(X)$ and $\mathcal{P}_3:C_3(X) \Rightarrow \mathcal{P}_2:C_2(X)$, each known by the two involved peers.


Fig. 1. Scenario 1

From the point of view of $\mathcal{P}_1$ $C_1(X)$ is a target relation because there is no distant relation specializing $C_1(X)$. The query $Q_4(X) \equiv \mathcal{P}_1:C_1(X)$ has no rewriting. That target relation is interesting since $\mathcal{P}_1:C_1(X) \Rightarrow \mathcal{P}_2:C_2(X)$ is a mapping in $\mathcal{P}_1$, $Q_5(X) \equiv \mathcal{P}_2:C_2(X)$ could be a query posed to $\mathcal{P}_2$ by $\mathcal{P}_1$. The obtained rewritings would be $\mathcal{P}_1:C_1(X)$ and $\mathcal{P}_3:C_3(X)$ and looking for mappings between all the relations belonging to this set of rewritings is relevant. Indeed, it could allow to discover the mapping $\mathcal{P}_3:C_3(X) \Rightarrow \mathcal{P}_1:C_1(X)$ making that way a connection between $\mathcal{P}_3$ and $\mathcal{P}_1$. Note that, according to this scenario 1, the peers $\mathcal{P}_1$ and $\mathcal{P}_2$ can be the same, and $\mathcal{P}_2$ and $\mathcal{P}_3$ too.

In the second scenario (cf Figure 2) let us consider $\mathcal{P}_1$ and $\mathcal{P}_2$ two peers, $\mathcal{P}_1:C_1$, $\mathcal{P}_2:C_2$ and $\mathcal{P}_2:C_3$ three classes. $\mathcal{P}_2:C_2(X) \Rightarrow \mathcal{P}_2:C_3(X)$ is a statement in $\mathcal{P}_2$. $\mathcal{P}_2:C_2(X) \Rightarrow \mathcal{P}_1:C_1(X)$ is a mapping in $\mathcal{P}_2$ and $\mathcal{P}_1$.

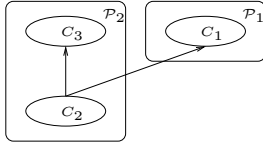


Fig. 2. Scenario 2

From the point of view of $\mathcal{P}_2$ $C_2(X)$ and $C_3(X)$ are target relations because there is no distant relation specializing $C_2(X)$ nor $C_3(X)$. The query $Q_6(X) \equiv \mathcal{P}_2:C_3(X)$ has only one local rewriting which is $\mathcal{P}_2:C_2(X)$. No distant relations belong to the rewritings. This scenario is also interesting since $\mathcal{P}_2:C_2(X) \Rightarrow \mathcal{P}_1:C_1(X)$ is a mapping in $\mathcal{P}_2$, it could be relevant to look for mappings between $C_1(X)$ and $C_3(X)$, two relations which subsume $C_2(X)$. It could allow to discover the mapping $\mathcal{P}_1:C_1(X) \Rightarrow \mathcal{P}_2:C_3(X)$ establishing a connection between $\mathcal{P}_2$ and $\mathcal{P}_1$ usable to rewrite $\mathcal{P}_2:C_3(X)$.

Let us note that the $\mathcal{P}_1:C_1(X) \Rightarrow \mathcal{P}_2:C_2(X)$ mapping in scenario 1 and the $\mathcal{P}_2:C_2(X) \Rightarrow \mathcal{P}_2:C_3(X)$ mapping in scenario 2 can be locally inferred in $\mathcal{P}_1$ and $\mathcal{P}_2$, respectively. Furthermore, these two scenarios are elementary and could be combined in order to deal with more complex ones. These two scenarios use those target relations as starting points for the identification of relevant mapping candidates. However, all target relations will not allow finding relevant mapping candidates. Thus, we just consider target relations with regard to the two scenarios described above.

For each target relation we look for sets of mapping candidates, denoted MC . Our approach is based on the idea that it is relevant to look for connections between relations if they have common points. In our setting the common point that we are going to consider is a common relation, either more general or more specific. The construction of the set of mapping candidates can be achieved according to two processes.

Specific Candidates Algorithm: This algorithm is performed for target relations with one or more general relations, R_g , according to the knowledge of its peer (according to the ontology or to the mappings). This scenario is represented in Figure 1 with C_2 in the place of R_g . In that case, we propose to pose the query $Q(X) \equiv R_g(X)$ in order to obtain its rewritings. The set of the rewritings is MC . It is composed of relations that are more specific than R_g .

General Candidates Algorithm: This algorithm is performed for target relations R_s with several (at least two) more general relations according to the knowledge of its peer (according to the ontology or to the mappings). This scenario is represented in Figure 2 with C_2 in the place of R_s . In that case, all the more general relations are members of the set MC .

4 Conclusion and Perspectives

In this paper we have presented how SomeRDFS query answering can offer an automated support for discovering new mappings. In particular, we have shown that query answering in a decentralized setting can be used to select elements which are relevant to be matched when the number of elements to be matched is a priori huge and when no peer has a global view of the ontologies in the network. Our approach is based on

query answering and filtering criteria. It applies to any system with large amounts of data organized according to local RDF schemas and which provides a communication infrastructure based on query rewriting: PDMSs but also, in a more general way, other decentralized systems such as networks of existing websites or local databases.

Currently, we implemented the identification process of potentially useful mapping shortcuts according to the default strategy S_1 . We also implemented the identification process of target relations and mapping candidates according to each of the strategies and counting methods introduced in this paper. We have a running prototype, Spy-Where, providing mapping candidates in SomeRDFS peers. The first experiments show the relevance of our approach. In a near future, we plan to integrate suitable alignment techniques. In a previous work, we addressed the problem of taxonomy alignment when the structures of the taxonomies are heterogeneous and dissymmetric, one taxonomy being deep whereas the other is flat [12]. We are going to explore the suitability of these techniques to our new context in order to propose extensions or adaptations really suited to the SomeRDFS PDMSs setting. Then we plan to evaluate our approach more completely. Future research includes also considering coherence issues due to the integration of discovered mappings among older ones.

References

1. Adjiman, P., Goasdoué, F., Rousset, M.C.: SomeRDFS in the semantic web. *Journal on Data Semantics*, 158–181 (2006)
2. Rahm, E., Bernstein, P.A.: A survey of approaches to automatic schema matching. *VLDB J.* 10(4), 334–350 (2001)
3. Euzenat, J., Shvaiko, P.: *Ontology matching*. Springer, Heidelberg (2007)
4. Kalfoglou, Y., Schorlemmer, M.: Ontology mapping: the state of the art. *The Knowledge Engineering Review* 18, 1–31 (2003)
5. Halevy, Y., Ives, G., Suciu, D., Tatarinov, I.: Schema mediation for large-scale semantic data sharing. *The VLDB Journal* 14(1), 68–83 (2005)
6. Tatarinov, I., Ives, Z.G., Madhavan, J., Halevy, A.Y., Suciu, D., Dalvi, N.N., Dong, X., Kadiyska, Y., Miklau, G., Mork, P.: The piazza peer data management project. *SIGMOD Record* 32(3), 47–52 (2003)
7. Herschel, S., Heese, R.: Humboldt discoverer: A semantic p2p index for pdms. In: *DISWeb 2005* (June 2005)
8. Castano, S., Ferrara, A., Montanelli, S.: H-match: an algorithm for dynamically matching ontologies in peer-based systems. In: *SWDB 2003*, Berlin, Germany (September 2003)
9. Bernstein, P.A., Green, T.J., Melnik, S., Nash, A.: Implementing mapping composition. In: *VLDB*, pp. 55–66 (2006)
10. Besana, P., Robertson, D.: How service choreography statistics reduce the ontology mapping problem. In: *ISWC/ASWC*, pp. 44–57 (2007)
11. Aberer, K., Cudré-Mauroux, P., Hauswirth, M.: The chatty web: emergent semantics through gossiping. In: *WWW*, pp. 197–206 (2003)
12. Reynaud, C., Safar, B.: When usual structural alignment techniques don't apply. In: *The ISWC 2006 workshop on Ontology matching (OM 2006)* (2006)

Using the Ontology Maturing Process Model for Searching, Managing and Retrieving Resources with Semantic Technologies

Simone Braun, Andreas Schmidt, Andreas Walter, and Valentin Zacharias

FZI Research Center for Information Technologies
Information Process Engineering
Haid-und-Neu-Straße 10-14, 76131 Karlsruhe, Germany
{Simone.Braun, Andreas.Schmidt, Andreas.Walter,
Valentin.Zacharias}@fzi.de

Abstract. Semantic technologies are very helpful in improving existing systems for searching, managing and retrieving of resources, e.g. image search, book-marking or expert finder systems. They enhance these systems through background knowledge stored in ontologies. However, in most cases, resources in these systems change very fast. In consequence, they require a dynamic and agile change of underlying ontologies. Also, the formality of these ontologies must fit the users needs and capabilities and must be appropriate and usable. Therefore, a continuous, collaborative and work or task integrated development of these ontologies is required. In this paper, we present how these requirements occur in real world applications and how they are solved and implemented using our Ontology Maturing Process Model.

1 Introduction

So far the potential of semantic annotation approaches has not been realized in practice; semantic annotation systems are still restricted to academia. At the same time very simple and limited tag based annotation approaches have emerged on the internet and found wide usage: proving both the users need for annotation approaches and their principal willingness to perform manual annotations.

Our work and this paper is based on the assumption that the failure of semantic annotation approaches can be traced to the misunderstanding of ontologies in the system as relatively fixed, expert maintained artifacts. We believe that semantic annotation approaches can only show their true potential by understanding the ontologies as artifacts that are permanently, rapidly and simply adapted by the users of the system to their task and changing domain; indeed we believe that understanding the model of the system as object of user's action will emerge as *the* defining criteria for semantic applications in general. To realize this vision, we extended our *Ontology Maturing Process Model* [1] and implemented two applications (SOBOLEO and ImageNotion) that support (parts of) this model for the domains of web and image annotation. We have already conducted multiple evaluations that were also used to refine the ontology maturing process model.

The next section of this paper describes the Ontology Maturing process model within two use cases, focusing on the need to view the artifact, knowledge, and social dimension of ontology maturing as separate. The third section, then, gives an overview of four evaluations and the lessons learned. Finally related work is introduced before the paper concludes.

2 Collaborative and Work-Integrated Ontology Development

With the Web 2.0 social tagging applications for managing, searching, and finding resources by dint of arbitrary tags found wide usage. However, problems such as homonyms, synonyms, multilinguality, typos or different ways to write words, and tags on different levels of abstraction hamper search and retrieval in these applications [1,2,3]. On the other hand, current Semantic annotation approaches avoid these problems, but usually don't allow to quickly and continuously adapt the ontology, often resulting in unsatisfied users being confronted with out-of-date, incomplete, inaccurate and incomprehensive ontologies that they cannot easily use for annotation [4,5]. To a large extent because the annotation process, i.e. the usage of the ontology, and the creation of the ontology are two separate processes, performed by a different set of people [6].

The goal of our work, then, is the combination of the benefits of social tagging with those of semantic annotation in order to address their respective weaknesses. Starting with simple tags, each user shall contribute to the collaborative development of ontologies. For this purpose, we integrate the creation process of ontologies into their usage process, e.g. search and annotation processes. Each community member can contribute new ideas (tags) emerging from the usage to the development of ontologies. The community picks them up, consolidates them, refines them, and formalizes them with semantic relations towards lightweight ontologies.

2.1 The Ontology Maturing Process Model

To operationalize this view, we have developed the ontology maturing process model that structures the ontology engineering process into four phases (for details of the complete model, please refer to [1]). An overview of this process model process is shown in figure 1.

In phase 1 *Emergence of ideas*, new ideas emerge and are introduced by individuals as new concept ideas or informal tags. These are ad-hoc and not well-defined, rather descriptive, e.g. with a text label. They are individually used and informally communicated. Phase 2 is the *Consolidation in Communities*, through the collaborative (re-) usage of the concept symbols (tags) within the community, a common vocabulary (or folksonomy) develops. The emerging vocabulary, which is shared among the community members, is still without formal semantics. *Formalization* happens within the third phase, when the community begins to organize the concepts into relations. This results in lightweight ontologies that rely primarily on inferencing based on subconcept relations. In the fourth, the *Axiomatization* phase, the ontologies are extended with axioms to allow for more powerful inferences.

It is important to note that ontology maturing does not assume that ontologies are built from scratch, but can be equally applied to already existent core ontologies used

for community seeding. Likewise, this model must not be misunderstood as a strictly linear process. Usually individually used tags, common but not yet formal terminologies as well as formally defined concepts coexist at any moment.

2.2 The Artifact, Knowledge and Social Dimensions of Ontology Maturing

Our evaluation sessions (which are described in section 3) have shown, that concentrating on the development of the ontology as a mediating artifact is not sufficient to prepare for sustainable community-driven semantic applications. Beyond the mere construction of an artifact, we have to consider that users have different levels of understanding of parts of the domain (e.g. identified by interest in background knowledge to improve their own understanding, asking for help, or taking the lead within a group) and that this understanding also evolves within usage processes. Furthermore, the social dimension of community-driven sites has to be addressed, e.g., which instruments are needed to support a growing community. As a consequence, we need to describe ontology maturing in three different dimensions, the artifacts, knowledge and social dimensions (see Fig. 1).

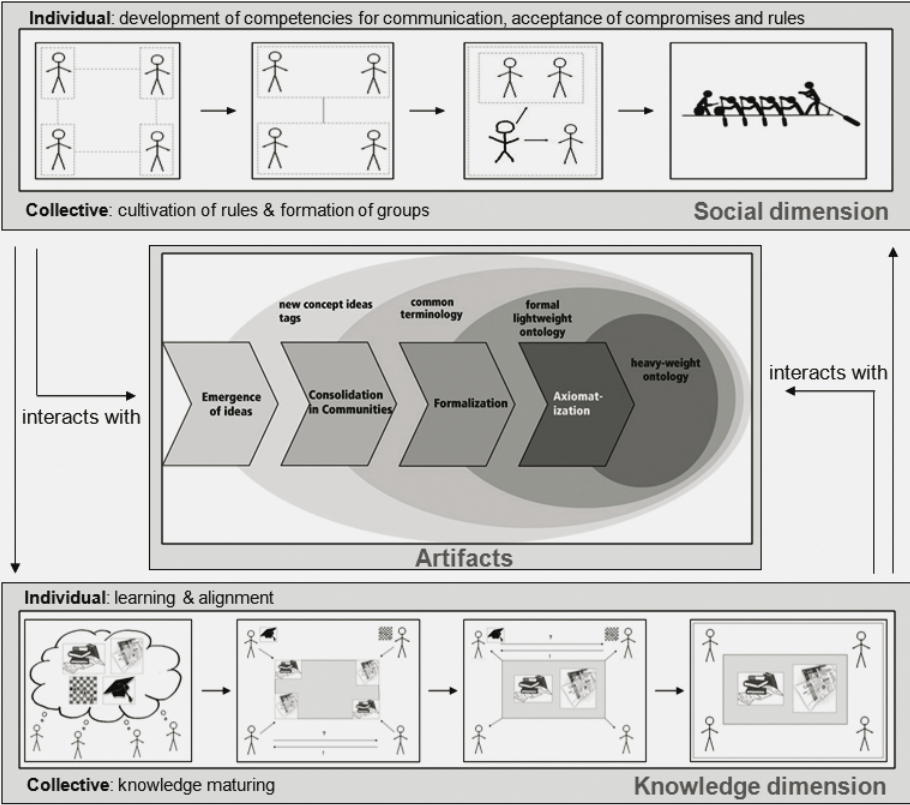


Fig. 1. View on the extended ontology maturing process model

The artifact dimension is concerned with the created ontology elements, the knowledge dimension with the maturing and alignment of knowledge, and the social dimension with the development of competencies and social structures.

Artifacts are “something viewed as a product of human conception”. In folksonomies, tags are the product of human conception. In semantic applications, ontologies are considered as a product of formalized human conception. Using our ontology maturing model, artifacts mature from simple tags to formalized or even axiomatized ontology elements as described in the previous section. Thus, the *artifact dimension* identifies the available ontology elements and their relations. This dimension has (naturally) been the focus of semantic technology research so far.

Users can only model appropriately what they have sufficiently understood, and the process of modeling usually involves a deepening of the understanding of the real-world topic. Within the *knowledge dimension*, we need to distinguish between individual knowledge and the abstraction of collective knowledge. On the level of the individual, we need to consider alignment processes that bring forth a sufficient level of shared understanding of the domain and learning processes on the methods to create artifacts (modeling competencies). On the collective level, this is about the development of an understanding as such.

Viewing ontology development as collaborative learning processes, e.g. interaction, communication and coordination among the individuals, we have to consider the social structures and processes in the *social dimension*. Users can only build a shared understanding, shared artifacts and methods to create these if they learn to collaborate on the individual as well as on the collective level. Learning on the individual level comprise a general willingness and competencies to interact with others, communicate, negotiate, compromise and accept rules.

2.3 Use Cases and Tool Support

In the following, we present two use cases for the ontology maturing process and their support with our tools ImageNotion [6,7] and SOBOLEO [8].

Semantic Image Annotation and Search. This use case concerns the management and retrieval of images with the use of semantic annotations.

Users, e.g. of an image archive, uploading images can use available elements from the ontology (e.g. via an ontology browser) to create semantic image annotations. In cases where a user is missing elements from the ontology, however, she can also create them directly integrated into the image uploading process. These newly created ontology elements may also be created with vague information that is then later refined by the community. Then, other users, e.g. image buyers can benefit from these semantic annotations when they perform semantic image search request, e.g. by searching for “all French generals who participated in the WWI” in an image archive with historical images.

During the use of semantic annotations in this domain a number of interesting phenomena occurred that go beyond the simple creation of artifacts in an ontology, but nevertheless influence it. We identified these phenomena during previous evaluations, e.g. [6]. Image annotators have a big interest improving their background knowledge about

a domain, e.g. by reading Wikipedia articles. In addition, they take the image searcher as their main focus – and try to homogenize the created artifacts with the knowledge of a user. For instance, when they expect searchers to be experts, they annotate images using very specific or even scientific annotations; but they use very easy annotations when the targeted users are private users. Another effect we identified in this use case was that individual users, who are experts of a given topic, take the lead in images concerning this domain. Then, in discussions about image annotations for these images, this user is asked to solve conflicts.

ImageNotion: An imagenotion (formed from the words image and notion) graphically represents a semantic notion with the help of an image. The associated methodology (based on section 2) consists of three different steps. Step 1 is the creation of new imagenotions, Step 2 is the consolidation of imagenotions in communities and Step 3 is the formalization of imagenotions by defining creation rules (such as naming conventions) and relations. Imagenotions from each maturing grade may be used for semantic image annotations. In the ImageNotion application, imagenotions are used for the semantic image annotation instead of textual tags as in traditional image archives.

For instance, for creating the semantic element representing the current president of the European Commission “Manuel Barroso” with that to annotate then images showing Manuel Barroso, one user may have created the imagenotion “Manuel Barroso” and selected an image showing him as representing image. In addition, she gave this imagenotion a main label. Some other member of the group added an alternative label text, the full name of Barroso which is “José Manuel Durão Barroso”, as well as his birthday, 1956-03-23, and another member added relations to the other imagenotions “European Commission” and “Portugal”. All in all, they created and matured the descriptive and visual information of this imagenotion.

Semantic Annotation and Search of Web Pages. This use case is taken from the German research project “Im Wissensnetz”¹ (“In the Knowledge Web – linked information processes in research networks”), which aims to support researcher from various disciplines within e-Science. One major problem is searching and retrieving adequate up-to-date resources in the internet. The dynamic of the domain is a particular challenge in this project, e.g. the area of plastics new materials or new forms of existing ones frequently enter the market; brand names and manufacturers are permanently changing and hardly traceable – attributes of a chemical substance retrievable using its brand name today, are very hard to find once it’s sold under a different label.

In this use case, the users want to have a tool to collaboratively collect and semantically annotate web pages. Thus, when one user finds a web page, e.g. the manufacturer’s website for a specific plastic or an article about a new material, she wants to pick it up into a shared bookmark collection and semantically annotate it, e.g. with the specific plastic. If the needed concept does not exist in the ontology (e.g. in the case of a new material) or is not suitable (e.g. when the brand name changed), the user wants to immediately modify an existing concept (e.g. extending a concept with the new brand name) or add arbitrary tags (e.g. a new material) while annotating the web page. Sometimes, users start with vague information (e.g. because it is a new method or technique) that

¹ <http://www.im-wissensnetz.de>

is then later consolidated and refined within the community. When searching for resources, e.g. with the former brand name, the search engine should make use of the underlying ontology and further provide search relaxation or refinement in order to reduce irrelevant results and to guide the user. Inadequacies of the ontology or the annotations can also be corrected right during the search process.

SOBOLEO: SOBOLEO is a web-based system that supports people working in a certain domain in the collaborative development of a shared bookmark collection and of a shared ontology that is used to organize the bookmarks. That means, collected bookmarks can be annotated with concepts from the ontology and the ontology can be changed at the same time. If users encounter a web resource, they can add it to the bookmark collection and annotate it with concepts from the SKOS ontology [9] for better later retrieval. If a needed concept does not exist in the underlying ontology or is not suitable, the users can modify an existing concept or use arbitrary tags, which are automatically added as "prototypical concepts" to the ontology. In this way, new concept ideas are seamlessly gathered when occurring (maturing phase 1) and existing ones are refined or corrected (maturing phase 2). The users can structure the concepts with hierarchical relations (broader and narrower) or indicate that they are "related". These relations are also considered by the semantic search engine and for navigation support within the bookmark collection. That means, the users can improve the retrieval and exploration of their annotated web pages by adding and refining ontology structures (maturing phase 3).

3 Evaluation

The goal of the evaluation was to show that (a) our showcase semantic applications are accepted by end users and that (b) key assumptions of our ontology maturing model (and the derived applications) are true. For (a), we need to show that these applications are perceived as useful and usable, both the annotation and search as well as the ontology editing part. For (b), we want to show that ontology maturing actually occurs in collaboration between different users of the system and that we can observe the proposed phases. For conducting the evaluations, we have applied a formative usability evaluation methodology that is also geared towards eliciting new requirements.

3.1 Evaluation Sessions

The first evaluation (*S1*) of SOBOLEO was an online evaluation held during the the Workshop on Social and Collaborative Construction of Structured Knowledge held at the 16th International World Wide Web Conference. The participants added in total 202 new concepts and 393 concept relations to the ontology. Further, they collected 155 web resources, which they annotated with 3 concepts per resource on average. The second evaluation of SOBOLEO (*S2*) took place within the scope of the project "Im Wissensnetz". Within two one-hour sessions, four users had to carry out specific tasks simulating the usage of SOBOLEO within their daily work activities. Half of the users were researchers of the rapid prototyping domain and half of them patent experts for German research. All of them were unexperienced in ontology development. We

provided a basic ontology with 31 concepts to start with that was thematically tailored to the rapid prototyping domain. The tasks were tailored to gain orientation within the ontology by letting the users place or add synonyms to existing concepts. Thus, the users added 6 concepts to the ontology, 11 synonyms and 21 concept relations. For the ImageNotion application we conducted an online survey (I1) with 137 participants. We were interested how individual users would create an initial version of a semantic element for Manuel Barroso, the current president of the European Commission. Users may then use this imagenotion for the semantic annotation of images showing Barroso. In addition, we executed a workshop I2 where three groups of six people participated. The groups were recruited from different communities: from Wikipedia users, from employees of French image agencies and from Italian history students. They had to perform tasks for the semantic annotation of images. They started with a small ontology; in its core based on CIDOC-CRM [10].

3.2 User Acceptance and Usefulness

The evaluations with users from different background showed that users appreciate both applications (according to DIN EN ISO 9241 & 13407):

- S1/I2 The users liked the ease of use of the ontology editor (in comparison to other, more heavy-weight applications) and particularly enjoyed the simple way of annotating resources with concepts or tags, which are then automatically added. Thus, to have the possibility to integrate not yet well defined concepts but something like *"starter concepts"* and, in this way, to *"get the ontology building almost for free"*.
- S2 Although the users came from a non-IT background, they appreciated SOBOLEO for its ease of use. Some of the users had some problems at the beginning due to their very basic knowledge in ontologies, but were able to obtain the necessary skills within the evaluation sessions.
- I1 The rate of success for completing the tasks in the online survey for the ImageNotion application has shown that people from a variety of backgrounds are able to understand and interact with a semantic image search and annotation application without prior training.

Furthermore, it turned out that the applications were actually used as collaborative applications by contributing to the construction of a shared ontology. Due to the more open setting of the evaluation in (S1) and (S2), this was more visible in the evaluation of SOBOLEO. Particularly, in (S2), the chat turned out to be an essential utility for simultaneous working. For instance, two users had problems in placing concepts in the given ontology because they had only basic knowledge of the rapid prototyping domain. In consequence, they began to ask their colleagues for help via the integrated chat functionality. Nevertheless, the chat appeared to be too simple. For improvement, the users wished to have a better integration of what is discussed and where the changes are done.

3.3 Validation of the Ontology Maturing Model and Its Implications

The validation of the ontology maturing model was the particular focus of the evaluation (I1) of the ImageNotion application.

Emergence of Ideas. Users were asked to state descriptive information for this politician. The most frequently mentioned labels were two different versions of his name: “Manuel Barroso” and “Barroso”. In addition, further version of his name and his profession “politician” were entered. For the alternative label, most people chose “politician”. In terms of semantics, this may already be seen as specifying a semantic element. “Barroso” was the second most frequent alternative label, while on the third place we got the full name of Manuel Barroso, “Jose Manuel Durao Barroso”. I.e., the mostly used tags for searching for Manuel Barroso are his name and his profession, followed by different spellings of the name and finally semantic elements such as “EU” or “person”. This is a very motivating result for us, because it shows that people in general not only think in terms of tags but also consider semantically relevant aspects. In case, users had written all these information directly in the ImageNotion application, the community would have created collaboratively a semantic element for Manuel Barroso with very detailed information. Since most users were not ontology experts, this is a very promising result.

Consolidation of Artifacts. The next task concerned the consolidation of artifacts. We have already shown that consolidation of artifacts in communities in other evaluations [6] and that training sessions helps in deepen the knowledge of users in understanding the general meanings of ontologies. The focus of this task was, how artifacts mature by considering the knowledge dimension. Therefore, users were first asked to deepen their knowledge about an artifact before maturing it. Then, they were asked, what kind of additional information they would state about Manuel Barroso. Therefore, they were asked to read the Wikipedia article about Manuel Barroso. The users added many more detailed information about Manuel Barroso, e.g. that he was born in Portugal, was Prime Minister of Portugal, or studied law at the University of Lisbon. All together, this shows that the consideration of the knowledge dimension improves the creation of artifacts with a high quality, because of a matured background knowledge of a domain.

Formalization. With the scope on the maturing of ontologies, we finally evaluated whether users would like to create relations beyond broader, narrower, and unnamed relations (referring to the formalization phase). Therefore, we asked the participants, what kind of named relations they would use for the relations they created. Users suggested specific names for relations such as “is president of” (24%), “works for” (8%), and “has nationality” (6%). With 84%, most of the participants thought that relations are important for semantic image search. Since more than 60% of the users stated that they had low or little knowledge about semantic technologies, this is a promising result for creating semantic systems for the managing of resources. Users not only understood the meaning of semantic elements, they also requested the creation of named relations.

3.4 Evaluation of the Artifact, Knowledge and Social Dimensions

The following observations base on the evaluations S1, S2 and I2. They show and explain, how the extended ontology model and the modeled artifact, knowledge, and social dimensions occur and work together.

Mutual Support. Some users did not know at the beginning how to use semantic elements, although they had an introduction before the evaluation started. That means, their personal expertise and knowledge was too immature in order to act on the ontology. In consequence, these users began to ask for help. In turn, one of the other users stated and shared his expertise and answered questions that allowed the other users for participating in the annotation work.

Explanation: Mutual support starts in the social dimension with the general willingness of an individual to participate. Because of her incomplete knowledge in the knowledge dimension, a user recognizes that she can not fulfill her desired tasks in a collaborative application. As a consequence, she communicates with other participants that are willing to share their knowledge with her. With matured knowledge, the user can better participate on the communities collaborative work tasks.

Homogenization. One very interesting point were homogenization processes. In the evaluation I2, one part of the participants had the role of image annotators (because of their daily work in professional image archives) who are interested in creating annotations so that image searchers can retrieve them as easy as possible. In contrast, the other group's role was that of image searchers who directly identified the ontology artifacts they would use for searching. In communication, they became aware of and adapted to the "other side's" likely use of the ontology elements. This means that even when a participant had complex knowledge about a domain, he cared about the usage of artifacts matching the need of others. In consequence, all participant homogenized their view on commonly shared knowledge to optimize the retrieval quality for the annotated resources in collaborative work.

Explanation: Starting point for the homogenization phenomenon is the personal willingness of each participant in the social dimension to integrate himself into the community. This interacts with the knowledge dimension, because all participants need to align their knowledge to achieve a commonly accepted, shared understanding.

Interest in Background Knowledge. In all three evaluations, users read external resources (mostly Wikipedia articles) and used the new background information for their artifacts and added descriptive information (e.g., birthday of a person) or relations (e.g. relations to specific events).

Explanation: The interest in background knowledge is a proof for the willingness of an individual in the knowledge dimension that bases on the general willing to integrate itself in the community in the social dimension. As a consequence, this first influences the knowledge dimension because of improved support for each users' interest to learn and alignment of knowledge, but also has an impact on the maturing of the ontology elements that reflect the new background knowledge.

4 Related Work

Our related work section is focused on ontology development methodologies and tools which allow for a collaborative and work integrated ontologies engineering processes.

With the Human-Centered Ontology Engineering Methodology (HCOME) Kotis et al. [11] view ontology development as a dynamic, human-centered process and focus particularly on ontology evolution. They assume a decentralized engineering model where everyone first formalizes her own ontology and shares it in a further step within the community. There, the individual ontologies are merged or further developed. However, findings in [12] (based on action theory) suggest that collaboration plays a more important role *before* we have formalized (individual) ontologies. So we think that the HCOME methodology can benefit from incorporating the notion of different maturity levels. Gibson et al. [13] involve domain experts in an early, less-formal stage of the ontology development process and support the communication with a Web 2.0 user interface. However, they also assume knowledge engineers to do the modeling task and not the domain experts doing it by themselves. A similar approach to ours is proposed by Siorpaes et al. [14]. They also conceive ontology building as a community-driven evolution process and use a wiki system as enabling technology. Wiki systems consider the aspects of collaboration and can support the early phases of ontology construction. Semantic wiki systems² try to extend the traditional wikis with semantic web technologies. These systems help users in creating definitions, e.g. beginning with informal texts. Because of discussion pages and versioning for each article they are suitable for complex coordination and consolidation processes. All of these methodologies and tools lack in possibilities for an integration of the ontology development in work processes.

5 Conclusions and Future Work

We have argued that the lack of acceptance of semantic applications in the large is due to the static and expert-based view of ontology engineering as separated from the use of the ontology (e.g. for annotation and search). In order to overcome these problems, we built on the success of Web 2.0 tagging approaches and combine these with ontology-based approaches, with the help of the ontology maturing model. To show the usefulness of this model, we have created two applications, SOBOLEO and ImageNotion. In four evaluation sessions we have shown that such tools are perceived useful and usable by users from a variety of backgrounds without prior training. Furthermore, these evaluations sessions have provided evidence that ordinary users are willing and able to engage in maturing activities for an ontology and that the development of a shared vocabulary takes place according to the ontology maturing theory.

Further, more long-term evaluations will have to take place to show that such applications allow for overcoming the time lag problem of controlled vocabularies/ontologies.

On the methodological side, we will try to derive a methodological framework for engineering maturing-aware applications beyond the two showcases from the ontology maturing model. This framework will be realized and evaluated in the next generation of SOBOLEO and ImageNotion. Another route of development is to investigate more advanced support tools that take into account the different dimensions; like (visual)

² e.g. SemanticWiki Interest Group
(http://semanticweb.org/wiki/Semantic_Wiki_State_Of_The_Art)

analysis tools of activities, or suggestions for consolidation that further ease the ontology construction task, particularly in larger user groups. This will take place within the context of the MATURE project³.

References

1. Braun, S., Schmidt, A., Walter, A., Nagypal, G., Zacharias, V.: Ontology Maturing: a Collaborative Web 2.0 Approach to Ontology Engineering. In: Proc. of the Workshop on Social & Collaborative Construction of Structured Knowledge, CEUR Workshop Proc., vol. 273 (2007)
2. Golder, S., Huberman, B.A.: The Structure of Collaborative Tagging Systems. *Journal of Information Sciences* 32, 198–208 (2006)
3. Guy, M., Tonkin, E.: Folksonomies: Tidying up tags? *D-Lib Magazine* 12 (2006)
4. Hepp, M.: Possible Ontologies: How Reality Constraints Building Relevant Ontologies. *IEEE Internet Computing* 11, 90–96 (2007)
5. Barker, K., Chaudhri, V.K., Chaw, S.Y., Clark, P., Fan, J., Israel, D., Mishra, S., Porter, B.W., Romero, P., Tecuci, D., Yeh, P.Z.: A Question-Answering System for AP Chemistry: Assessing KR&R Technologies. In: Proc. of the Int. Conf. on Principles of Knowledge Representation and Reasoning, pp. 488–497 (2004)
6. Walter, A., Nagypal, G.: ImageNotion - Methodology, Tool Support and Evaluation. In: Meersman, R., Tari, Z. (eds.) OTM 2007, Part I. LNCS, vol. 4803, pp. 1007–1024. Springer, Heidelberg (2007)
7. Walter, A., Nagypal, G.: IMAGENOTION - Collaborative Semantic Annotation of Images and Image Parts and Work Integrated Creation of Ontologies. In: Proc. of 1st Conference on Social Semantic Web. LNCS. Springer, Heidelberg (2007)
8. Zacharias, V., Braun, S.: SOBOLEO – Social Bookmarking and Lightweight Engineering of Ontologies. In: Proc. of the Workshop on Social & Collaborative Construction of Structured Knowledge, CEUR Workshop Proc., vol. 273 (2007)
9. Miles, A., Bechhofer, S.: SKOS Simple Knowledge Organization System Reference. W3C Working Draft 25 January 2008, W3C (2008)
10. Crofts, N., Doerr, M., Gill, T., Stead, S., Stiff, M.: Definition of the cidoc conceptual reference model version 4.2. In: CIDOC CRM Special Interest Group (2005)
11. Kotis, K., Vouros, G.A., Alonso, J.P.: HCOME: A Tool-Supported Methodology for Engineering Living Ontologies. In: 2nd Int. Workshop on Semantic Web and Databases. LNCS, pp. 155–166. Springer, Heidelberg (2004)
12. Allert, H., Markannen, H., Richter, C.: Rethinking the Use of Ontologies in Learning. In: Proc. of the 2nd Int. Workshop on Learner-Oriented Knowledge Management and KM-Oriented Learning, pp. 115–125 (2006)
13. Gibson, A., Wolstencroft, K., Stevens, R.: Promotion of ontological comprehension: Exposing terms and metadata with web 2.0. In: Proc. of the Workshop on Social & Collaborative Construction of Structured Knowledge, CEUR Workshop Proc., vol. 273 (2007)
14. Siorpaes, K., Hepp, M.: Myontology: The marriage of ontology engineering and collective intelligence. In: Bridging the Gap between Semantic Web and Web 2.0 (SemNet 2007), pp. 127–138 (2007)

³ <http://mature-ip.eu>

Author Index

- Ahmad, Imran I-802
 Aimé, Xavier II-1423
 Alencar, Antonio Juarez I-409
 Alia, Mourad I-497
 Arcieri, Franco II-1131
 Armendáriz-Íñigo, J.E. I-607
 Asiki, Athanasia I-212
 Atencia, Manuel I-445
 Atkins, Daniel E. I-2

 Bai, Xi II-1304
 Baïna, Karim I-392
 Barone, Paolo I-553, I-571
 Bellahsene, Zohra I-249
 Ben Ahmed, Mohamed I-589
 Benali, Khalid I-392
 Ben Hamida, Amira I-589
 Berbers, Y. I-571
 Besana, Paolo II-1217
 Bevilacqua, Luca I-822
 Bhowmick, Sourav S. I-484
 Blanco, Carlos II-1052
 Blomqvist, Eva II-1235
 Bolzoni, Damiano II-938
 Borgida, Alexander II-1440
 Borissov, Nikolay I-744
 Bosque, José Luis I-833
 Bottaro, Andre I-497
 Bouamrane, Matt-Mouley II-1458
 Boukhelef, Djelloul I-193
 Branco, Miguel I-775
 Bratosin, Carmen I-898
 Braun, Simone II-1568
 Bravo, Maricela II-1532
 Brezany, Peter I-762
 Bringay, Sandra II-1385
 Buyya, Rajkumar I-62

 Cabaj, Krzysztof II-1019
 Calvanese, Diego II-1440
 Calvier, François-Élie II-1559
 Camara, Fatoumata I-497
 Caminero, Agustín I-726
 Caminero, Blanca I-726
 Capuano, Luisa I-822

 Cárdenas-Montes, Miguel I-793
 Carrión, Carmen I-726
 Castañon, Gerardo I-812
 Castellanos, Malu I-8
 Castillo-Perez, Sergio II-987
 Chen, Qiming I-175
 Chen, Shiping I-301
 Chen, Xin II-969
 Chen, Wei I-675
 Coletta, Remi I-249
 Correia Jr., A. I-691
 Cortes, Toni I-914
 Crooy, R.A. I-319
 Curbera, Francisco I-100
 Cuzzocrea, Alfredo I-852

 Dayal, Umeshwar I-8
 Decker, Gero I-265
 Decker, H. I-607
 Delbru, Renaud II-1304
 Della Penna, Giuseppe II-1131
 de Mendivil, J.R. González I-607
 Deng, Lingli II-1069
 de Oliveira Victorio, Rui I-409
 de Roure, David I-775
 Devlic, A. I-571
 Di-Jorio, Lisa II-1385
 Dickens, Phillip I-870
 Dimitri, Andrea II-1131
 Ding, Ying II-1355
 Doganata, Yurdaer I-100
 Doi, Norihisa II-956
 Doka, Katerina I-212
 Domaszewicz, Jaroslaw II-1471
 Dragone, Luigi I-157
 Duchateau, Fabien I-249
 Duong, Maggie II-1183

 Eliassen, Frank I-553
 Etalle, Sandro II-938
 Euzenat, Jérôme II-1164

 Fernández-Medina, Eduardo II-1052
 Ferreira, Armando Leite I-409

- Fiot, Céline II-1385
 Formica, Anna II-1289
 Frà, C. I-571
 Frénot, Stéphane I-589
 Fried, Michael II-1355
 Fukuno, Naoya II-956
 Furst, Frédéric II-1423
- Gal, Avigdor I-283
 Garbinato, Benoît I-625
 García-Alfaro, Joaquín II-987
 García-Bañuelos, Luciano I-436
 García-Molina, Hector I-7
 García-Rodríguez de Guzmán, Ignacio II-1052
 Garonne, Vincent I-775
 Georgakopoulos, Dimitrios II-1215
 Giunchiglia, Fausto II-1217
 Godart, Claude I-44
 Gokhale, Aniruddha I-515
 Gómez-Iglesias, Antonio I-793
 Gómez-Pulido, Juan A. I-793
 Gonzalez, J.L. I-914
 Gottschalk, Florian I-418
 Großkopf, Alexander I-265
 Grześkowiak, Maciej II-1140
 Guermouche, Nawal I-44
- Hallé, Sylvain I-337
 Hao, Ming I-8
 Hardouin, Briac I-497
 He, Yeping II-1069
 Herborn, Timo I-475
 Herrero, Pilar I-886
 Hidders, Jan I-374
 Hoffert, Joe I-515
 Holzer, Adrian I-625
 Hsu, Meichun I-175
 Hu, Chunming I-231
 Hu, HuaPing II-969
 Hu, X. I-571
 Huai, Jinpeng I-231
 Huedo, Eduardo I-711
 Hull, Richard II-1152
 Hurrell, Martin II-1458
- Ignat, Claudia-Lavinia I-456
 Intrigila, Benedetto II-1131
- Jansen-Vullers, Monique H. I-418
 Joshi, James B.D. II-1104
- Jurczyk, Paweł I-465
 Jürgenson, Aivo II-1036
- Kakousis, Konstantinos I-657
 Kang, Sin-Jae II-1355
 Kantere, Verena II-1367
 Kapitza, Rüdiger I-535
 Kikuchi, Hiroaki II-956
 Kirsch-Pinheiro, M. I-571
 Kitagawa, Hiroyuki I-193
 Kowalkiewicz, Marek I-82
 Koziris, Nectarios I-212
 Koziuk, Michał II-1471
 Kumar, Akhil I-120
 Kuntz, Pascale II-1423
- Lassnig, Mario I-775
 Laurent, Anne II-1385
 Leal, Katia I-711
 Lee, Deirdre II-1517
 Le Mouël, Frédéric I-589
 Li, Xianxian I-231
 Lin, Li I-231
 Lin, Wilfred W.K. II-1200
 Liu, Bo II-969
 Liu, Chengfei I-82
 Llorente, Ignacio M. I-711
 Logan, Jeremy I-870
 Loutas, Nikos II-1517
- Madria, Sanjay K. I-484
 Magaldi, Massimo I-822
 Magazzeni, Daniele II-1131
 Majumdar, Shikharesh I-802
 Mamelli, Alessandro I-553, I-571
 Margasiński, Igor II-1019
 Marie, Anan I-283
 Martens, Axel I-100
 Martinez-Elizalde, Lorena I-812
 Martinez Gonzalez, Cristina I-1
 Masoumzadeh, Amirreza II-1104
 Massacci, Fabio II-1087
 Mazurczyk, Wojciech II-1001, II-1019
 McNeill, Fiona II-1217
 Missikoff, Michele II-1289
 Mlýnková, Irena II-1253
 Mondorf, Ansgar I-475
 Monfroy, Eric I-26
 Montes, Jesús I-886
 Morales-Ramos, Enrique I-793
 Mougouie, Babak I-475

- Muñoz-Escof, F.D. I-607
 Mukhi, Nirmal K. I-100
 Nasirifard, Peyman II-1122
 Nemes, Gabor I-354
 Nepal, Surya I-301
 Nguyen, Hong-Quang II-1550
 Nguyen, Kinh II-1550
 Oliveira, R. I-691
 Oro, Ermelinda II-1482
 Oster, Gérald I-456
 Panagiotou, G. I-571
 Pane, Juan II-1217
 Papadopoulos, George A. I-553, I-657
 Paspallis, Nearchos I-553, I-657
 Patel, Chandrakant I-8
 Pathan, Mukaddim I-62
 Pedone, Fernando I-639
 Pereira, J. I-691
 Pérez, María S. I-886
 Peristeras, Vassilios II-1122, II-1517
 Perrin, Olivier I-26
 Petrov, Ilia I-354
 Piattini, Mario II-1052
 Pierno, Stefania I-822
 Pirró, Giuseppe II-1271
 Politou, Maria-Eirini II-1367
 Polyvyanyy, Artem I-427
 Pourabbas, Elaheh II-1289
 Prasanna, Viktor K. II-1500
 Qiao, Xiaoqiang I-675
 Radziszewski, Paweł II-1019
 Rahayu, Wenny J. II-1550
 Ramirez-Velarde, Raul I-812
 Rector, Alan II-1458
 Reynaud, Chantal II-1559
 Ricklefs, Michael II-1235
 Ringeissen, Christophe I-26
 Robles, Oscar D. I-833
 Rocha, Ricardo I-775
 Rodriguez-Muro, Mariano II-1440
 Romano, Luigi I-822
 Rouvoy, Romain I-553
 Ruffolo, Massimo II-1482
 Ruiz-Fuertes, M.I. I-607
 Sadiq, Wasim I-82
 Saïs, Fatiha II-1541
 Salgado, Pedro I-775
 Sánchez, Alberto I-833, I-886
 Sánchez-Bajo, Florentino I-793
 Schmidt, Andreas II-1568
 Schmidt, Douglas I-515
 Schmitz, Eber Assis I-409
 Schober, Sven I-535
 Schoeneich, Radoslaw Olgierd II-1471
 Schorlemmer, Marco I-445
 Schröder-Preikschat, Wolfgang I-535
 Seco, Nuno II-1271
 Sellis, Timos II-1367
 Shan, Zhe I-120
 Sharma, Ratnesh I-8
 Shvaiko, Pavel II-1164, II-1217
 Sidorova, Natalia I-898
 Slominski, Aleksander I-100
 Söldner, Guido I-535
 Soma, Ramakrishna II-1500
 Spyns, Peter II-1404
 Sroka, Jacek I-374
 Strasunskas, Darijus II-1319
 Szczypiorski, Krzysztof II-1001, II-1019
 Taglino, Francesco II-1289
 Talamo, Maurizio II-1131
 Taniar, David II-1550
 Teisseire, Maguelonne II-1385
 Terada, Masato II-956
 Thomopoulos, Rallou II-1541
 Toharia, Pablo I-833
 Toma, Ioan II-1355
 Tomás, Luis I-726
 Tomassen, Stein L. II-1319
 Touré, Fodé I-392
 Tovar, Elsa Liliana II-1338
 Trčka, Nikola I-898
 Trichet, Francky II-1423
 Trujillo, Juan II-1052
 Tsoumakos, Dimitrios I-212
 Tummarello, Giovanni II-1304
 Urban, Susan D. I-139
 Valdés, Julio J. I-886
 Valla, M. I-571
 van der Aalst, Wil M.P. I-319,
 I-418, I-898
 van Dongen, B.F. I-319
 Vanrompay, Y. I-571

Vargas, Cesar I-812
 Vecchiola, Christian I-62
 Vega-Rodríguez, Miguel A. I-793
 Velázquez, José II-1532
 Vennelakanti, Ravigopal I-8
 Vessaz, François I-625
 Victor, K. I-571
 Vidal, María-Esther II-1338
 Villemaire, Roger I-337
 Viyanon, Waraporn I-484

 Walter, Andreas II-1568
 Wang, Chen I-301
 Wei, Jun I-675
 Weidlich, Matthias I-265
 Weske, Mathias I-265, I-427
 Willemson, Jan II-1036
 Wimmer, Maria A. I-475
 Wirström, Niklas I-744
 Wöhrer, Alexander I-762

Wong, Allan K.Y. II-1200
 Wong, Jackei H.K. II-1200
 Wu, Guoquan I-675

 Xiao, Fengtao II-969
 Xiao, Yang I-139
 Xiong, Li I-465
 Xu, Ziyao II-1069

 Yatskevich, Mikalai II-1217

 Zacharias, Valentin II-1568
 Zaluska, Ed I-775
 Zannone, Nicola II-1087
 Zhang, Yanchun II-1183
 Zhang, Zhixiong II-1355
 Zhao, Xiaohui I-82
 Zhu, Yanmin I-231
 Zic, John I-301
 Zuikevičiūtė, Vaidė I-639