International Journal on Advances in Intelligent Systems

2012 vol. 5 nr. 1&2

The *International Journal on Advances in Intelligent Systems* is Published by IARIA.
ISSN: 1942-2679
journals site: http://www.iariajournals.org
contact: petre@iaria.org

Sponsored by IARIA
www.iaria.org

International Journal on Advances in Intelligent Systems
Volume 5, Number 1 & 2, 2012

Petr Berka, University of Economics, Czech Republic
Julita Bermejo-Alonso, Universidad Politécnica de Madrid, Spain
Aurelio Bermúdez Marín, Universidad de Castilla-La Mancha, Spain
Lasse Berntzen, Vestfold University College - Tønsberg, Norway
Michela Bertolotto, University College Dublin, Ireland
Ateet Bhalla, NRI Institute of Information Science and Technology, Bhopal, India
Freimut Bodendorf, Universität Erlangen-Nürnberg, Germany
Karsten Böhm, FH Kufstein Tirol - University of Applied Sciences, Austria
Pierre Borne, Ecole Centrale de Lille, France
Marko Bošković, Research Studios, Austria
Christos Bouras, University of Patras, Greece
Anne Boyer, LORIA - Nancy Université / KIWI Research team, France
Stainam Brandao, COPPE/Federal University of Rio de Janeiro, Brazil
Stefano Bromuri, University of Applied Sciences Western Switzerland, Switzerland
Vít Bršlica, University of Defence - Brno, Czech Republic
Dumitru Burdescu, University of Craiova, Romania
Diletta Romana Cacciagrano, University of Camerino, Italy
Kenneth P. Camilleri, University of Malta - Msida, Malta
Paolo Campegiani, University of Rome Tor Vergata , Italy
Marcelino Campos Oliveira Silva, Chemtech - A Siemens Business / Federal University of Rio de Janeiro, Brazil
Ozgu Can, Ege University, Turkey
José Manuel Cantera Fonseca, Telefónica Investigación y Desarrollo (R&D), Spain
Juan-Vicente Capella-Hernández, Universitat Politècnica de València, Spain
Bogdan Alexandru Caprarescu, West University of Timisoara, Romania
Miriam A. M. Capretz, The University of Western Ontario, Canada
Massimiliano Caramia, University of Rome "Tor Vergata", Italy
Davide Carboni, CRS4 Research Center - Sardinia, Italy
Mari Carmen Domingo, Barcelona Tech University, Spain
Luis Carriço, University of Lisbon, Portugal
Rafael Casado Gonzalez, Universidad de Castilla - La Mancha, Spain
Michelangelo Ceci, University of Bari, Italy
Fernando Cerdan, Polytechnic University of Cartagena, Spain
Alexandra Suzana Cernian, University "Politehnica" of Bucharest, Romania
Carlos Cetina, Technical Universidad San Jorge, Spain
Sukalpa Chanda, Gjøvik University College, Norway
David Chen, University Bordeaux 1, France
Luke Chen, University of Ulster @ Jordanstown, UK
Ping Chen, University of Houston-Downtown, USA
Kong Cheng, Telcordia Research, USA
Po-Hsun Cheng, National Kaohsiung Normal University, Taiwan
Dickson Chiu, Dickson Computer Systems, Hong Kong
Sunil Choenni, Research & Documentation Centre, Ministry of Security and Justice / Rotterdam University of Applied Sciences, The Netherlands
Smitashree Choudhury, Knowledge Media Institute, The UK Open University, UK
William Cheng-Chung Chu, Tunghai University, Taiwan
Christophe Claramunt, Naval Academy Research Institute, France

Cesar A. Collazos, Universidad del Cauca, Colombia
Phan Cong-Vinh, NTT University, Vietnam
Christophe Cruz, University of Bourgogne, France
Beata Czarnacka-Chrobot, Warsaw School of Economics, Department of Business Informatics, Poland
Claudia d'Amato, University of Bari, Italy
Sérgio Roberto P. da Silva, Universidade Estadual de Maringá - Paraná, Brazil
Mirela Danubianu, "Stefan cel Mare" University of Suceava, Romania
Dragos Datcu, Netherlands Defense Academy / Delft University of Technology , The Netherlands
Antonio De Nicola, ENEA, Italy
Claudio de Castro Monteiro, Federal Institute of Education, Science and Technology of Tocantins, Brazil
Noel De Palma, Joseph Fourier University, France
Jan Dedek, Charles University in Prague, Czech Republic
Zhi-Hong Deng, Peking University, China
Stojan Denic, Toshiba Research Europe Limited, UK
Sotirios Ch. Diamantas, Pusan National University, South Korea
Leandro Dias da Silva, Universidade Federal de Alagoas, Brazil
Jerome Dinet, Univeristé Paul Verlaine - Metz, France
Jianguo Ding, University of Luxembourg, Luxembourg
Yulin Ding, Defence Science & Technology Organisation Edinburgh, Australia
Alexiei Dingli, University of Malta, Malta
Mihaela Dinsoreanu, Technical University of Cluj-Napoca, Romania
Ioanna Dionysiou, University of Nicosia, Cyprus
Roland Dodd, CQUniversity, Australia
Nima Dokoohaki, Royal Institute of Technology (KTH)-Kista, Sweden
Suzana Dragicevic, Simon Fraser University- Burnaby, Canada
Mauro Dragone, University College Dublin (UCD), Ireland
Marek J. Druzdzel, University of Pittsburgh, USA
Carlos Duarte, University of Lisbon, Portugal
Raimund K. Ege, Northern Illinois University, USA
Jorge Ejarque, Barcelona Supercomputing Center, Spain
Larbi Esmahi, Athabasca University, Canada
Simon G. Fabri, University of Malta, Malta
Umar Farooq, Amazon.com, USA
Mehdi Farshbaf-Sahih-Sorkhabi, Azad University - Tehran / Fanavaran co., Tehran, Iran
Anna Fensel, Semantic Technology Institute (STI) Innsbruck and FTW Forschungszentrum Telekommunikation Wien, Austria
Stenio Fernandes, Federal University of Pernambuco (CIn/UFPE), Brazil
Oscar Ferrandez Escamez, University of Utah, USA
Florin Filip, Romanian Academy, Romania
Agata Filipowska, Poznan University of Economics, Poland
Ziny Flikop, Scientist, USA
Adina Magda Florea, University "Politehnica" of Bucharest, Romania
Francesco Fontanella, University of Cassino and Southern Lazio, Italy
Panagiotis Fotaris, University of Macedonia, Greece
Enrico Francesconi, ITTIG - CNR / Institute of Legal Information Theory and Techniques / Italian National Research Council, Italy

Rita Francese, Università di Salerno - Fisciano, Italy
Bernhard Freudenthaler, Software Competence Center Hagenberg GmbH, Austria
Sören Frey, University of Kiel, Germany
Steffen Fries, Siemens AG, Corporate Technology - Munich, Germany
Somchart Fugkeaw, Thai Digital ID Co., Ltd., Thailand
Naoki Fukuta, Shizuoka University, Japan
Mathias Funk, Eindhoven University of Technology, The Netherlands
Adam M. Gadomski**,** Università degli Studi di Roma La Sapienza, Italy
Alex Galis, University College London (UCL), UK
Crescenzio Gallo, Department of Clinical and Experimental Medicine - University of Foggia, Italy
Matjaz Gams, Jozef Stefan Institute-Ljubljana, Slovenia
Raúl García Castro, Universidad Politécnica de Madrid, Spain
Fabio Gasparetti, Roma Tre University - Artificial Intelligence Lab, Italy
Joseph A. Giampapa, Carnegie Mellon University, USA
George Giannakopoulos, NCSR Demokritos, Greece
David Gil, University of Alicante, Spain
Harald Gjermundrod, University of Nicosia, Cyprus
Angelantonio Gnazzo, Telecom Italia - Torino, Italy
Luis Gomes, Universidade Nova Lisboa, Portugal
Nan-Wei Gong, MIT Media Laboratory, USA
Francisco Alejandro Gonzale-Horta, National Institute for Astrophysics, Optics, and Electronics (INAOE), Mexico
Sotirios K. Goudos, Aristotle University of Thessaloniki, Greece
Victor Govindaswamy, Texas A&M University-Texarkana, USA
Gregor Grambow, Aalen University, Germany
Fabio Grandi, University of Bologna, Italy
Andrina Granić, University of Split, Croatia
Carmine Gravino, Università degli Studi di Salerno, Italy
Dominic Greenwood, Whitestein Technologies, Switzerland
Michael Grottke, University of Erlangen-Nuremberg, Germany
Vic Grout, Glyndŵr University, UK
Maik Günther, Stadtwerke München GmbH, Germany
Francesco Guerra, University of Modena and Reggio Emilia, Italy
Alessio Gugliotta, Innova SPA, Italy
Richard Gunstone, Bournemouth University, UK
Fikret Gurgen, Bogazici University, Turkey
Ivan Habernal, University of West Bohemia, Czech Republic
Maki Habib, The American University in Cairo, Egypt
Till Halbach Røssvoll, Norwegian Computing Center, Norway
Jameleddine Hassine, King Fahd University of Petroleum & Mineral (KFUPM), Saudi Arabia
Ourania Hatzi, Harokopio University of Athens, Greece
Yulan He, The Open University, UK
Kari Heikkinen, Lappeenranta University of Technology, Finland
Cory Henson, Wright State University / Kno.e.sis Center, USA
Arthur Herzog, Technische Universität Darmstadt, Germany
Rattikorn Hewett, Whitacre College of Engineering, Texas Tech University, USA
Celso Massaki Hirata, Instituto Tecnológico de Aeronáutica - São José dos Campos, Brazil

Jochen Hirth, University of Kaiserslautern, Germany
Bernhard Hollunder, Hochschule Furtwangen University, Germany
Thomas Holz, University College Dublin, Ireland
Władysław Homenda, Warsaw University of Technology, Poland
Carolina Howard Felicíssimo, Schlumberger Brazil Research and Geoengineering Center, Brazil
Jingwei Huang , University of Illinois at Urbana-Champaign, USA
Weidong (Tony) Huang, CSIRO ICT Centre, Australia
Xiaodi Huang, Charles Sturt University - Albury, Australia
Eduardo Huedo, Universidad Complutense de Madrid, Spain
Marc-Philippe Huget, University of Savoie, France
Chi Hung, Tsinghua University, China
Chih-Cheng Hung, Southern Polytechnic State University - Marietta, USA
Edward Hung, Hong Kong Polytechnic University, Hong Kong
Muhammad Iftikhar, Universiti Malaysia Sabah (UMS), Malaysia
Prateek Jain, Ohio Center of Excellence in Knowledge-enabled Computing, Kno.e.sis, USA
Wassim Jaziri, Miracl Laboratory, ISIM Sfax, Tunisia
Hoyoung Jeung, SAP Research Brisbane, Australia
Yiming Ji, University of South Carolina Beaufort, USA
Jinlei Jiang, Department of Computer Science and Technology, Tsinghua University, China
Weirong Jiang, Juniper Networks Inc., USA
Hanmin Jung, Korea Institute of Science & Technology Information, Korea
Ilya S. Kabak, "Stankin" Moscow State Technological University, Russia
Eleanna Kafeza, Athens University of Economics and Business, Greece
Hermann Kaindl, Vienna University of Technology, Austria
Ahmed Kamel, Concordia College, Moorhead, Minnesota, USA
Faouzi Kamoun, University of Dubai, UAE
Rajkumar Kannan, Bishop Heber College(Autonomous), India
Teemu Kanstrén, VTT, Finland
Fazal Wahab Karam, Norwegian University of Science and Technology (NTNU), Norway
Dimitrios A. Karras, Chalkis Institute of Technology, Hellas
Koji Kashihara, The University of Tokushima, Japan
Katia Kermanidis, Ionian University, Greece
Serge Kernbach, University of Stuttgart, Germany
Nhien An Le Khac, University College Dublin, Ireland
Malik Jahan Khan, Lahore University of Management Sciences (LUMS), Lahore, Pakistan
Reinhard Klemm, Avaya Labs Research, USA
Ah-Lian Kor, Leeds Metropolitan University, UK
Arne Koschel, Applied University of Sciences and Arts, Hannover, Germany
George Kousiouris, NTUA, Greece
Philipp Kremer, German Aerospace Center (DLR), Germany
Dalia Kriksciuniene, Vilnius University, Lithuania
Dariusz Król, AGH University of Science and Technology, ACC Cyfronet AGH, Poland
Roland Kübert, Höchstleistungsrechenzentrum Stuttgart, Germany
Markus Kunde, German Aerospace Center, Germany
Dharmender Singh Kushwaha, Motilal Nehru National Institute of Technology, India
Andrew Kusiak, The University of Iowa, USA

Dimosthenis Kyriazis, National Technical University of Athens, Greece
Vitaveska Lanfranchi, Research Fellow, OAK Group, University of Sheffield, UK
Mikel Larrea, University of the Basque Country UPV/EHU, Spain
Angelos Lazaris, University of Southern California, USA
Philippe Le Parc, University of Brest, France
Gyu Myoung Lee, Institut Telecom, Telecom SudParis, France
Kyu-Chul Lee, Chungnam National University, South Korea
Tracey Kah Mein Lee, Singapore Polytechnic, Republic of Singapore
Daniel Lemire, LICEF Research Center, Canada
Haim Levkowitz, University of Massachusetts Lowell, USA
Kuan-Ching Li, Providence University, Taiwan
Tsai-Yen Li, National Chengchi University, Taiwan
Yangmin Li, University of Macau, Macao SAR
Jian Liang, Nimbus Centre, Cork Institute of Technology, Ireland
Haibin Liu, China Aerospace Science and Technology Corporation, China
Lu Liu, University of Derby, UK
Qing Liu, The Commonwealth Scientific and Industrial Research Organisation (CSIRO), Australia
Shih-Hsi "Alex" Liu, California State University - Fresno, USA
Xiaoqing (Frank) Liu, Missouri University of Science and Technology, USA
David Lizcano, Universidad a Distancia de Madrid, Spain
Henrique Lopes Cardoso, LIACC / Faculty of Engineering, University of Porto, Portugal
Wassef Louati, University of Monastir, Tunisia
Sandra Lovrencic, University of Zagreb, Croatia
Jun Luo, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, China
Prabhat K. Mahanti, University of New Brunswick, Canada
Jacek Mandziuk, Warsaw University of Technology, Poland
Herwig Mannaert, University of Antwerp, Belgium
Yannis Manolopoulos, Aristotle University of Thessaloniki, Greece
Antonio Maria Rinaldi, Università di Napoli Federico II, Italy
Paolo Masci, University of London, UK
Ali Masoudi-Nejad, University of Tehran, Iran
Constandinos Mavromoustakis, University of Nicosia, Cyprus
Gerrit Meixner, German Research Center for Artificial Intelligence (DFKI) / Innovative Factory Systems (IFS) / Center for Human-Machine-Interaction (ZMMI), Germany
Zulfiqar Ali Memon, Sukkur Institute of Business Administration, Pakistan
Andreas Merentitis, AGT Group (R&D) GmbH, Germany
Jose Merseguer, Universidad de Zaragoza, Spain
Frederic Migeon, IRIT/Toulouse University, France
Harald Milchrahm, Technical University Graz, Institute for Software Technology, Austria
Fatma Mili, Oakland University, USA
Les Miller, Iowa State University, USA
Marius Minea, University POLITEHNICA of Bucharest, Romania
Yasser F. O. Mohammad, Assiut University, Egypt
Shahab Mokarizadeh, Royal Institute of Technology (KTH) - Stockholm, Sweden
Martin Molhanec, Czech Technical University in Prague, Czech Republic
Charalampos Moschopoulos, KU Leuven, Belgium

Mary Luz Mouronte López, Ericsson S.A., Spain
Henning Müller, University of Applied Sciences Western Switzerland - Sierre (HES SO), Switzerland
Susana Munoz Hernández, Universidad Politécnica de Madrid, Spain
Adrian Muscat, University of Malta, Malta
Peter Mutschke, GESIS - Leibniz Institute for the Social Sciences - Bonn, Germany
Bela Mutschler, Hochschule Ravensburg-Weingarten, Germany
Deok Hee Nam, Wilberforce University, USA
Fazel Naghdy, University of Wollongong, Australia
Joan Navarro, Research Group in Distributed Systems (La Salle - Ramon Llull University), Spain
Saša Nešić, University of Lugano, Switzerland
Günter Neumann, DFKI GmbH, Germany
Rui Neves Madeira, Instituto Politécnico de Setúbal / Universidade Nova de Lisboa, Portugal
Toàn Nguyên, INRIA Grenobel Rhone-Alpes/ Montbonnot, France
Andrzej Niesler, Institute of Business Informatics, Wroclaw University of Economics, Poland
Michael P. Oakes, University of Sunderland, UK
John O'Donovan, University of California - Santa Barbara, USA
Kouzou Ohara, Aoyama Gakuin University, Japan
Jonice Oliveira, Universidade Federal do Rio de Janeiro, Brazil
Ian Oliver, Nokia Location & Commerce, Finland / University of Brighton, UK
Michael Adeyeye Oluwasegun, University of Cape Town, South Africa
Sigeru Omatu, Osaka Institute of Technology, Japan
Sascha Opletal, University of Stuttgart, Germany
Flavio Oquendo, European University of Brittany/IRISA-UBS, France
Fakri Othman, Cardiff Metropolitan University, UK
Enn Õunapuu, Tallinn University of Technology, Estonia
Jeffrey Junfeng Pan, Facebook Inc., USA
Hervé Panetto, University of Lorraine, France
Malgorzata Pankowska, University of Economics, Poland
Harris Papadopoulos, Frederick University, Cyprus
Laura Papaleo, ICT Department - Province of Genoa & University of Genoa, Italy
Agis Papantoniou, National Technical University of Athens, Greece
Thanasis G. Papaioannou, École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
Andreas Papasalouros, University of the Aegean, Greece
Eric Paquet, National Research Council / University of Ottawa, Canada
Kunal Patel, Ingenuity Systems, USA
Carlos Pedrinaci, Knowledge Media Institute, The Open University, UK
Yoseba Penya, University of Deusto - DeustoTech (Basque Country), Spain
Cathryn Peoples, University of Ulster, UK
Asier Perallos, University of Deusto, Spain
Christian Percebois, Université Paul Sabatier - IRIT, France
Andrea Perego, European Commission, Joint Research Centre, Italy
Mark Perry, University of Western Ontario/Faculty of Law/ Faculty of Science - London, Canada
Willy Picard, Poznań University of Economics, Poland
Meikel Poess, Oracle, USA
Agostino Poggi, Università degli Studi di Parma, Italy
R. Ponnusamy, Madha Engineering College-Anna University, India

Dorin Popescu, University of Craiova, Romania
Stefan Poslad, Queen Mary University of London, UK
Wendy Powley, Queen's University, Canada
Radu-Emil Precup, "Politehnica" University of Timisoara, Romania
Jerzy Prekurat, Canadian Bank Note Co. Ltd., Canada
Didier Puzenat, Université des Antilles et de la Guyane, France
Sita Ramakrishnan, Monash University, Australia
Elmano Ramalho Cavalcanti, Federal University of Campina Grande, Brazil
Juwel Rana, Luleå University of Technology, Sweden
Martin Randles, School of Computing and Mathematical Sciences, Liverpool John Moores University, UK
Christoph Rasche, University of Paderborn, Germany
Ann Reddipogu, ManyWorlds UK Ltd, UK
Ramana Reddy, West Virginia University, USA
René Reiners, Fraunhofer FIT - Sankt Augustin, Germany
Paolo Remagnino, Kingston University - Surrey, UK
Sebastian Rieger, Karlsruher Institut für Technologie (KIT) / Steinbuch Centre for Computing (SCC), Germany
Andreas Riener, Johannes Kepler University Linz, Austria
Ivan Rodero, NSF Center for Autonomic Computing, Rutgers University - Piscataway, USA
Joel Rodrigues, Instituto de Telecomunicações / University of Beira Interior, Portugal
Alejandro Rodríguez González, University Carlos III of Madrid, Spain
Aitor Rodríguez-Alsina, University Autonoma of Barcelona (UAB), Spain
Paolo Romano, INESC-ID Lisbon, Portugal
Vicente-Arturo Romero-Zaldivar, Atos Origin SAE, Spain
Agostinho Rosa, Instituto de Sistemas e Robótica, Portugal
José Rouillard, University of Lille, France
Paweł Różycki, University of Information Technology and Management (UITM) in Rzeszów, Poland
Igor Ruiz-Agundez, DeustoTech, University of Deusto, Spain
Michele Ruta, Politecnico di Bari, Italy
Ryszard S. Choras, University of Technology & Life Sciences, Poland
Vivek S. Deshpande, MIT College of Engineering - Pune, India
Melike Sah, Trinity College Dublin, Ireland
Francesc Saigi Rubió, Universitat Oberta de Catalunya, Spain
Abdel-Badeeh M. Salem, Ain Shams University, Egypt
Yacine Sam, Université François-Rabelais Tours, France
Ismael Sanz, Universitat Jaume I, Spain
Ricardo Sanz, Universidad Politecnica de Madrid, Spain
Marcello Sarini, Università degli Studi Milano-Bicocca - Milano, Italy
Munehiko Sasajima, I.S.I.R., Osaka University, Japan
Minoru Sasaki, Ibaraki University, Japan
Hiroyuki Sato, University of Tokyo, Japan
Jürgen Sauer, Universität Oldenburg, Germany
Patrick Sayd, CEA List, France
Dominique Scapin, INRIA - Le Chesnay, France
Kenneth Scerri, University of Malta, Malta
Adriana Schiopoiu Burlea, University of Craiova, Romania
Rainer Schmidt, Austrian Institute of Technology, Austria

Bruno Schulze, National Laboratory for Scientific Computing - LNCC, Brazil
Wieland Schwinger, Johannes Kepler University Linz, Austria
Hans-Werner Sehring, T-Systems Multimedia Solutions GmbH, Germany
Paulo Jorge Sequeira Gonçalves, Polytechnic Institute of Castelo Branco, Portugal
Sandra Sendra Compte, Polytechnic University of Valencia, Spain
Kewei Sha, Oklahoma City University, USA
Hossein Sharif, University of Portsmouth, UK
Roman Y. Shtykh, Rakuten, Inc., Japan
Kwang Mong Sim, Gwangju Insitute of Science & Technology, South Korea
Robin JS Sloan, University of Abertay Dundee, UK
Vasco N. G. J. Soares, Instituto de Telecomunicações / University of Beira Interior / Polytechnic Institute of Castelo Branco, Portugal
Don Sofge, Naval Research Laboratory, USA
Christoph Sondermann-Woelke, Universitaet Paderborn, Germany
George Spanoudakis, City University London, UK
Vladimir Stantchev, SRH University Berlin, Germany
Claudius Stern, University of Paderborn, Germany
Mari Carmen Suárez-Figueroa, Universidad Politécnica de Madrid (UPM), Spain
Kåre Synnes, Luleå University of Technology, Sweden
Ryszard Tadeusiewicz, AGH University of Science and Technology, Poland
Yehia Taher, ERISS - Tilburg University, The Netherlands
Yutaka Takahashi, Senshu University, Japan
Azzelarabe Taleb-Bendiab, Liverpool John Moores University, UK
Dan Tamir, Texas State University, USA
Jinhui Tang, Nanjing University of Science and Technology, P.R. China
Yi Tang, Chinese Academy of Sciences, China
Said Tazi, LAAS-CNRS, Universite Toulouse 1, France
John Terzakis, Intel, USA
Sotirios Terzis, University of Strathclyde, UK
Vagan Terziyan, University of Jyvaskyla, Finland
Michael Tighe, University of Western Ontario, Canada
Ioan Toma, STI Innsbruck/University Innsbruck, Austria
Lucio Tommaso De Paolis, Department of Innovation Engineering - University of Salento, Italy
Davide Tosi, Università degli Studi dell'Insubria, Italy
Raquel Trillo Lado, University of Zaragoza, Spain
Tuan Anh Trinh, Budapest University of Technology and Economics, Hungary
Simon Tsang, Applied Communication Sciences, USA
Theodore Tsiligiridis, Agricultural University of Athens, Greece
Antonios Tsourdos, Cranfield University, UK
José Valente de Oliveira, University of Algarve, Portugal
Cristián Felipe Varas Schuda, Fraunhofer Institute for Open Communication Systems (FOKUS), Germany
Eugen Volk, University of Stuttgart, Germany
Mihaela Vranić, University of Zagreb, Croatia
Chieh-Yih Wan, Intel Labs, Intel Corporation, USA
Jue Wang, Washington University in St. Louis, USA
Shenghui Wang, OCLC Leiden, The Netherlands

Zhonglei Wang, Karlsruhe Institute of Technology (KIT), Germany
Laurent Wendling, University Descartes (Paris 5), France
Maarten Weyn, Artesis University College of Antwerp, Belgium
Nancy Wiegand, University of Wisconsin-Madison, USA
Alexander Wijesinha, Towson University, USA
Eric B. Wolf, US Geological Survey, Center for Excellence in GIScience, USA
Ouri Wolfson, University of Illinois at Chicago, USA
Yingcai Xiao, The University of Akron, USA
Reuven Yagel, The Jerusalem College of Engineering, Israel
Fan Yang, Nuance Communications, Inc., USA
Maribel Yasmina Santos, University of Minho, Portugal
Zhenzhen Ye, Systems & Technology Group, IBM, US A
Jong P. Yoon, MATH/CIS Dept, Mercy College, USA
Shigang Yue, School of Computer Science, University of Lincoln, UK
Constantin-Bala Zamfirescu, "Lucian Blaga" Univ. of Sibiu, Romania
Claudia Zapata, Pontificia Universidad Católica del Perú, Peru
Marek Zaremba, University of Quebec, Canada
Filip Zavoral, Charles University Prague, Czech Republic
Yuting Zhao, University of Aberdeen, UK
Hai-Tao Zheng, Graduate School at Shenzhen, Tsinghua University, China
Yu Zheng, Microsoft Research Asia, China
Zibin (Ben) Zheng, Shenzhen Research Institute, The Chinese University of Hong Kong, Hong Kong
Bin Zhou, University of Maryland, Baltimore County, USA
Alfred Zimmermann, Reutlingen University - Faculty of Informatics, Germany
Wolf Zimmermann, Martin-Luther-University Halle-Wittenberg, Germany

International Journal on Advances in Intelligent Systems
Volume 5, Numbers 1 & 2, 2012

CONTENTS

Atle Refsdal, SINTEF ICT, Norway
Ketil Stølen, SINTEF ICT and University of Oslo, Norway

Knowledge Base Approach for 3D Objects Detection in Point Clouds Using 3D Processing and Specialists Knowledge

Helmi Ben Hmida, Frank Boochs
Institut i3mainz, am Fachbereich Geoinformatik und Vermessung, Fachhochschule Mainz, Lucy-Hillebrand-Str. 255128 Mainz, Germany
e-mail: {helmi.benhmida, boochs}@geoinform.fh-mainz.de

Christophe Cruz, Christophe Nicolle
Laboratoire Le2i, UFR Sciences et Techniques
Université de Bourgogne
B.P. 47870, 21078 Dijon Cedex, France
e-mail: {christophe.cruz, cnicolle}@u-bourgogne.fr

***Abstract*—This paper presents a knowledge-based detection of objects approach using the OWL ontology language, the Semantic Web Rule Language, and 3D processing built-ins aiming at combining geometrical analysis of 3D point clouds and specialist's knowledge. Here, we share our experience regarding the creation of 3D semantic facility model out of unorganized 3D point clouds. Thus, a knowledge-based detection approach of objects using the OWL ontology language is presented. This knowledge is used to define SWRL detection rules. In addition, the combination of 3D processing built-ins and topological Built-Ins in SWRL rules allows a more flexible and intelligent detection, and the annotation of objects contained in 3D point clouds. The created WiDOP prototype takes a set of 3D point clouds as input, and produces as output a populated ontology corresponding to an indexed scene visualized within VRML language. The context of the study is the detection of railway objects materialized within the Deutsche Bahn scene such as signals, technical cupboards, electric poles, etc. Thus, the resulting enriched and populated ontology, that contains the annotations of objects in the point clouds, is used to feed a GIS system or an IFC file for architecture purposes.**

Keywords-Ontology; Semantic facility information model; Semantic VRML model; Geometric analysis; Topologic analysis; 3D processing algorithm, Semantic web; knowledge modeling; ontology; 3D scene reconstruction; object identification.

I. INTRODUCTION

Surveying with 3D scanners is spreading all domains. With every new scanner model on the market, the instruments become faster, more accurate and can scan objects at longer distances [1]. Such a technology presents a powerful tool for many applications and has partially replaced traditional surveying methods since it can speed up field work significantly. This method allows the creation of 3D point clouds from objects or landscapes.

From the other side, the technical survey of facility aims to build a digital model based on geometric analysis. Such a process becomes more and more tedious. Especially, with the new terrestrial laser scanners, where a huge amount of 3D point clouds are generated. Within such a scenario, new challenges have seen the light where the basic one is to make the reconstruction process automatic and more accurate. Thus, early works on 3D point clouds have investigated the reconstruction and the recognition of geometrical shapes [2] [3] to resolve this challenge. In fact, such a problematic was investigated as a topic of the computer graphic and the signal processing research, where most works focused on segmentation or visualization aspects. As most-recent works, the new tendency related to the use of semantic has been explored [4]. As a main operation, the technical survey relies fundamentally on the object reconstruction process where considerable effort has already been invested to reduce the impact of time consuming, manual activities and to substitute them by numerical algorithms.

Unfortunately, most of algorithmic conceptions are data-driven and concentrate on specific features of the objects, being accessible to numerical models. By these models, which normally describe the behavior of geometrical (flatness, roughness, for example) or physical features (color, texture), the data are classified and analyzed. Basically, such strategies are static and not to allow a dynamic adjustment to the object or initial processing results. In further scenarios, an algorithm will be applied to the data producing better or minor results, depending on several parameters like image or point cloud quality, the completeness of object representation, the viewpoints position, the complexity of object features, the use of control parameters and so on. Consequently, there is no feedback to the algorithmic part in order to choose a different algorithm or reuse the same algorithm with changed parameters. This interaction is mainly up to the user who has to decide by himself, which algorithms to apply for which kind of objects and data sets. Often good results can only be achieved by iterative processing controlled by a human interaction.

These problems can be solved when further supplementary and guiding information is integrated into the algorithmic process chain for object detection and recognition, allowing to support the process of validation. Such an information might be derived from the context of the object itself and its behavior with respect to the data and/or other objects or from a systematic characterization of the parameterization and effectiveness of the algorithms to be used. As programming languages used in the context of numerical treatments are not dedicated to process knowledge, their condition of use is not flexible and makes the integration of semantic aspects difficult.

Ontologies are used to represent formally the knowledge of a domain. The basic ideas were to present knowledge using graphs and logical structure to make computers able to

understand and process knowledge [5]. As most recent works, the tendency related to the use of semantic has been explored [4][10][21]. In fact, the assumption that knowledge will help the improvement of the automation, the accuracy and the result quality is shared by specialists of the point cloud processing. However, many questions remain without answers. How the detection process can get support within different knowledge about the scene objects and what is the impact of this knowledge compared to classic approach. In such scenario, knowledge about such objects has to include detailed information about the objects' geometry, structure, 3D algorithms, etc.

The technical survey of facilities, as a long and costly process, aims at building a digital model based on geometric analysis since the modeling of a facility as a set of vectors is not sufficient in most cases. To resolve this problem a new standard was developed over ten years by the International Alliance for Interoperability (IAI).) It is named the IFC format (IFC - Industry Foundation Classes) [8]. The specification is a neutral data format to describe exchange and share information typically used within the building and facility management industry. This norm considers the building elements as independent objects where each object is characterized by a 3D representation and defined by a semantic normalized label. Consequently, the architects and the experts are not the only ones who are able to recognize the elements, but everyone will be able to do it, including the system itself. For instance, an IFC "Signal" is not just a simple collection of lines and geometric primitives recognized as a signal; it is an “intelligent” object signal which has attributes linked to a geometrical definition and function. IFC files are made of objects and connections between these objects. Object attributes describe the “business semantic” of the object. Connections between objects are represented by “relation elements” [1].

As a matter of fact, the WiDOP project (knowledge-based detection of objects in point clouds) aims at making a step forward. The goal is to develop efficient and intelligent methods for an automated processing of terrestrial laser scanner data, Figure **1**. The principle of the WiDop project is a knowledge-based detection of objects in point clouds for AEC (Architecture, Engineering and Construction) engineering applications using IFC format. In contrast to existing approaches, the project consists in using prior knowledge about the context and the objects. This knowledge is extracted from databases, CAD plans, Geographic Information Systems (GIS), technical reports or domain experts. Therefore, this knowledge is the basis for a selective knowledge-oriented detection and recognition of objects in point clouds.

The project WiDOP is Funded by the German government. However, the partners are the Fraport company (Frankfurt Airport manager), the German railway company (Deutsche Bahn), and the Metronom company which is specialized in 3D point cloud processing. Where the Deutsche Bahn main concerns are the management of the railway furniture. Actually, the environment of the railway is constantly changing. Where the cost of keeping these plans up to date is increasing. The present-time solution adopted by the Deutsche Bahn (DB) consists on fixing a 3D terrestrial laser scanner on the train and to survey the surrounding landscape (Railway, signals and green trees on the borders). Metronom automation is a DB subcontractor specialized in 3D data processing. This partner takes the survey point clouds as input and detects the different existent elements manually helped with some 3D process like spike detection. The main objective of Deutsch Bahn project consists in detecting automatically the objects in the 3D point clouds to feed the position and the semantic definition of objects into a GIS system.

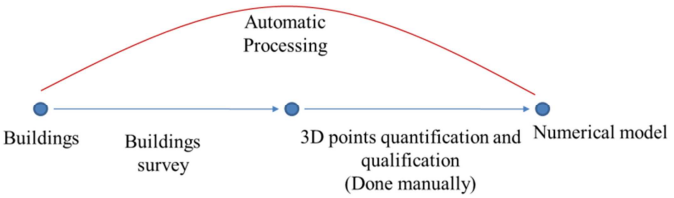


Figure 1. Automatic processing compared to the manual one

The present project aims at building a bridge between the semantic modeling and the numerical processing, to define strategies based on domain knowledge and 3D processing knowledge. The knowledge will be structured in ontologies containing a variety of elements like already existing information about objects of that scene. Like data sources (digital maps, geographical information systems, etc.), information about the objects' characteristics, the hierarchy of the sub-elements, the geometrical topology, the characteristics of processing algorithms, etc. In addition, all relevant information about the objects, geometries, inter and intra-relation and the 3D processing algorithms have been modeled inside the knowledge base, including characteristics such as positions, geometrics information, images textures, behavior and parameter of suitable algorithms, for example. The suggested system is materialized via WiDOP project [6]. Furthermore, the created WiDOP platform can generate an indexed scene from unorganized 3D point clouds visualized within the virtual reality modeling language. [7].

II. BACKGROUND CONCEPT AND METHODOLOGY

The problematic of 3D object detection and scene reconstruction, including semantic knowledge was recently treated within different domains. Basically, the photogrammetry one [9], the construction one, the robotics [10] and recently the knowledge engineering one [11]. Modeling a survey, in which low-level point cloud or surface representation is transformed into a semantically rich model is done through three main tasks. T first is the data collection, in which dense point measurements of the facility are collected using laser scans taken from key locations throughout the facility; Then data processing, in which the sets of point clouds from the collected scanners are processed. Finally, modeling the survey in which the low-level point cloud is transformed into a semantically rich model. This is done via modeling geometric knowledge, qualifying topological relations and finally assigning an object category to each geometry [12]. Concerning the

geometry modeling, we remind here that the goal is to create simplified representations of facility components by fitting geometric primitives to the point cloud data. The modeled components are labeled with an object category. Establishing relationships between components is important in a facility model and must also be established. In fact, relationships between objects in a facility model are useful in many scenarios. In addition, spatial relationships between objects provide contextual information to assist in object recognition [13]. Within the literature, three main strategies are described to rich such a model where the first one is based on human interaction with provided software's for point clouds classifications and annotations [14]. While the second strategy relies more on the automatic data processing without any human interaction by using different segmentation techniques for feature extraction [10]. Finally, new techniques presenting an improvement compared with the cited ones by integrating semantic networks to guide the reconstruction process are presented in [15].

A. Manual survey model creation

In current practice, the creation of a facility model is largely a manual process, performed by service providers who are contracted to scan and model a facility. In reality, a project may require several months to be achieved, depending on the complexity of the facility and the modeling requirements. Reverse engineering tools excel at geometric modeling of surfaces, but with the lack of volumetric representations, while such design systems cannot handle the massive data sets from laser scanners. As a result, modelers often shuttle intermediate results back and forth between different software packages during the modeling process, giving rise to the possibility of information loss due to limitations of data exchange standards or errors in the implementation of the standards within the software tools [16]. Prior knowledge about component geometry, such as the diameter of a column, can be used to constrain the modeling process, or the characteristics of known components may be kept in a standard component library. Finally, the class of the detected geometry is determined by the modeler once the object is created. In some cases, relationships between components are established either manually or in a semi-automated manner.

B. Semi-Automatic and Automatic methods

The manual process for constructing a survey model is time consuming, labor-intensive, tedious, subjective, and requires skilled workers. Even if modeling of individual geometric primitives can be fairly quick, modeling a facility may require thousands of primitives. The combined modeling time can be several months for an average-sized facility. Since the same types of primitives must be modeled throughout a facility, the steps are highly repetitive and tedious [17]. The above-mentioned observations and others illustrate the need for semi-automated and automated techniques for facility model creation. Ideally, a system could be developed that would take a point cloud of a facility as input and produce a fully annotated as-built model of the facility as output. The first step within the automatic process is the geometric modeling. It presents the process of constructing simplified representations of the 3D shape for survey components from point cloud data. In general, the shape representation is supported by CSG [18] or B-Rep [19] representation. The representation of geometric shapes has been studied extensively [20]. Once geometric elements are detected and stored via a specific presentation, the final task within a facility modeling is the object recognition. It presents the process of labeling a set of data points or geometric primitives extracted from the data with a named object or object class. Whereas the modeling task would find a set of points to be a vertical plane, the recognition task would label that plane as being a wall, for instance. Often, the knowledge describing the shapes to be recognized is encoded in a set of descriptors that implicitly capture object shape. Research on recognition of facilities specific components is still in its early stages. Methods in this category typically perform an initial shape-based segmentation of the scene, into planar regions, for example, and then use features derived from the segments to recognize objects. This approach is exemplified by Rusu et al. who use heuristics to detect walls, floors, ceilings, and cabinets in a kitchen environment [10]. A similar approach was proposed by Pu and Vosselman to model facility façades [21].

To reduce the search space of object recognition algorithms, the use of knowledge related to a specific facility can be a fundamental solution. For instance, Yue et al. overlay a design model of a facility with the as-built point cloud to guide the process of identifying which points clouds data belong to specific objects and to detect differences between the as-built and as-designed conditions [22]. In such cases, object recognition problem is simplified to be a matching problem between the scene model entities and the data points. Another similar approach is presented in [23]. Other promising approaches have only been tested on limited and very simple examples, and it is equally difficult to predict how they would fare when faced with more complex and realistic data sets. For example, the semantic network methods for recognizing components using context work well for simple examples of hallways and barren, rectangular rooms [13], but how would they handle spaces with complex geometries and clutter.

C. Discussion

The presented methods for survey modeling and object recognition rely on knowledge about the domain. Concepts like "Signals are vertical" and "Signals intersect with the ground" are encoded explicitly through a set of rules. Such rule based approaches tend to be brittle and break down when they are tested in new and slightly different environments. Additionally, regarding the literature, people models the context by specifying the concepts and the relationships of objects to describe the world. However, no one mentions the knowledge about the 3D processing algorithms and the associated results such as the geometry and the topology.

Based on these observations, flexible representations of facility objects and more sophisticated guidance based algorithms for object detection by modeling *algorithmic*, *geometric* and *topological knowledge* within an ontology

structure the way of a significant improvement. Actually, it will allow the process to create a dynamic sequence of 3D processing algorithms for object detections and to guarantee an automatic detection and recognition of objects in 3D point clouds, materialized via the semantic annotation process.

III. OVERVIEW OF THE WIDOP GENERAL MODEL

In general, mathematical algorithms contain different data processing steps, which are combined with internal decisions based on numerical results. This makes the processing inflexible and error prone, especially when the data does not behave as the model behind the algorithm expects. One of the purposes behind this contribution is to put these implicit decisions outside, make a semantic layer out of it and combine it with the object model. This approach is more flexible and can be easily extended, since knowledge and data processing are separated.

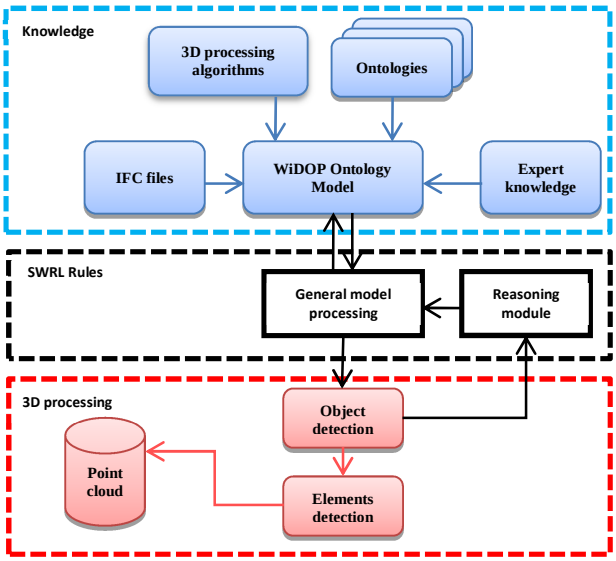


Figure 2. WiDOP: Overview system

Figure 2 presents the general architecture for the WiDOP project. It is composed of three parts: the knowledge model, the 3D processing algorithms execution, and the interaction management and control part labeled WiDOP processing materialized within rules and extensions, ensuring the interaction between the above sited parts. In contrast with existing approaches, we aim at the utilization of previous knowledge on objects. This knowledge can be contained in databases, construction plans, as-built plans or Geographic Information Systems (GIS).

A. The knowledge model

The term "Semantic Web" has been defined numerous time. Though there is no formal definition of Semantic Web, some of its most used definitions are "The Semantic Web is not a separate Web but an extension of the current one, in which information is given well-defined meaning, better enabling computers and people to work in cooperation" [28]. It is a source to retrieve information from the Web (using the Web spiders from RDF files) and access the data through Semantic Web Agents or Semantic Web Services. Simply, Semantic Web is data about data or metadata. "A Semantic Web is a Web where the focus is placed on the meaning of words, rather than on the words themselves, where information becomes knowledge after semantic analysis is performed. For this reason, a Semantic Web is a network of knowledge, compared with what we have today, that can be defined as a network of information [29]. The Semantic Web provides a common framework that allows data to be shared and reused across application, enterprise and community boundaries [30]. In fact, description logics provide a formalization for knowledge representation of real-world situations. This provides the logical replies to the queries of real-world situations. The results are highly sophisticated reasoning engines, which utilize the expressiveness capabilities of DLs to manipulate the knowledge. A Knowledge Representation system is a formal representation of a knowledge described through different technologies. When it is described through DLs, they set up a Knowledge Base (KB), the contents of which could be reasoned or infer to manipulate them. A knowledge base could be considered as a complete package of knowledge content. It is, however, only a subset of a Knowledge Representation system that contains additional components.

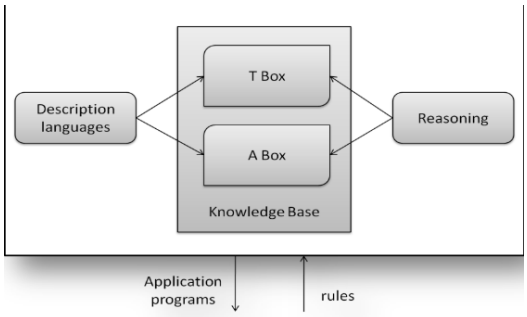


Figure 3. The Architecture of a knowledge representation system

As seen in Figure 3, the author [31] sketches the architecture of any Knowledge Representation system based on DLs. It could be seen the central theme of such a system is a Knowledge Base (KB). It is composed by two components: the TBox and the ABox. TBox statements are the terms or the terminologies that are used within the system domain. In general, they are statements describing the domain through the controlled vocabularies. For example, in terms of a Deutsche Bahn domain the TBox statements are the set of concepts as Signal, Fourniture, ProcessingAlgorithm, etc. or the set of roles as hasCharacteristics, isDeseignedFor, hasGeometry etc. ABox in contains assertions to the TBox statements. For example, Wall1 is an ABox presents the TBox Wall.

Our approach is intended to use semantics based on OWL technology [44] for knowledge modeling and processing. Knowledge has to be structured and formalized based on IFC schema, XML files and particularly on Deutsche Bahn and 3D processing domain experts, etc., using classes, instances,

relations and rules. An object in the ontology can be modeled as presented; a room has elements composed of walls, a ceiling and a floor. The sited elements are basic objects. They are defined by their geometry (plane, boundary, etc.), features (roughness, appearance, etc.), and also the qualified relations between them (adjacent, perpendicular, etc.). The object "room" gets its geometry from its elements, where further characteristics may be added, such as functions in order to estimate the existent sub elements. For instance, a "classroom" will contain "tables", "chairs", "a blackboard", etc. The research of the object "room" will be based on an algorithmic strategy which will look for the different objects contained in the point cloud. This means, using different detection algorithms for each element, based on the above mentioned characteristics, will allow us to classify most of the point region in the different element categories. It corresponds to the spatial structure of any facility, and it is an instance of semantic knowledge defined in the ontology. This instance defines the rough geometry and the semantics of the building elements without any real measurement. This model contains also knowledge extracted from the technical literature of the domain and knowledge from experts of the domain also. In addition, the ontology is, as well enriched with knowledge about 3D processing algorithms and populated with the results of experiences undertaken on 3D point clouds, which define the empirical knowledge extracted from point clouds regarding a specific domain of application.

B. The 3D processing algorithms

Numerical processing includes a number of algorithms or their combination to process the spatial data. Strategies include geometric element detection (straight line, plane, surface, etc.), projection-based, region estimation, histogram matrices, etc. All of these strategies are either under the guidance of knowledge, or use the previous knowledge to estimate the object intelligently and optimally. Alongside with 3D point clouds, various types of input data sets can be used such as images, range images, point clouds with intensity or color values, point clouds with individual images oriented to them or even stereo images without a point cloud. All sources are exploited for application to particular strategies. Knowledge not only describes the information of the objects, but also gives a framework for the control of the selected strategies. The success rate of detection algorithms using RANSAC [24], Iterative Closest Point [25] and Least Squares Fitting [26] should significantly increase by making use of the knowledge background. However, we are planning not only to process point data sets, but also surface and volume representation like mesh, voxels and bounding Boxes. These methods and others will be selected in a flexible way, depending on the semantic context.

C. The WiDOP processing

In order to manage the interaction between the knowledge part and the 3D processing part, a new layer labeled WiDOP processing materialized within rules is created. This layer ensures the control and the management of the knowledge transaction and the decision taken based on SWRL languages, and its extensions through several steps explained in the next section. The semantic within the ontologies expressed through OWL can be used inside the ontologies, and the knowledge bases themselves for inference purposes. However, in order to express the rules, the Semantic Web Rule Language (SWRL) is emerged [45]. The SWRL has the form antecedent → consequent, where both antecedent and consequent are conjunctions of atoms written $a_1 \wedge ... \wedge a_n$. Atoms in swrl rules can be of the form C(x), P(x,y), Q(x,z), sameAs(x,y), differentFrom(x,y), or builtIn (pred, z_1, …, z_n), where C is an OWL description, P is an OWL individual-valued property, Q is an OWL data-valued property, pred is a datatype predicate, x and y are either individual-valued variables or OWL individuals, and z, z_1, … z_n are either data-valued variables or OWL data literals. An OWL data literal is either a typed literal or a plain literal. Variables are indicated by using the standard convention of prefixing them with a question mark (e.g., ?x). URI references (URIrefs) are used to identify ontology elements such as classes, individual-valued properties and data-valued properties. For instance, the following rule asserts that one's parents' brothers are one's uncles where parent, brother and uncle are all individual-valued properties.

$$parent(?x, ?p) \wedge brother(?p, ?u) \rightarrow uncle(?x, ?u) \qquad (1)$$

The set of built-ins for SWRL are motivated by a modular approach allowing further extensions in future releases within a taxonomy. SWRL's built-ins approach is also based on the reuse of existing built-ins in XQuery or XPath, which are themselves based on XML Schema by using Datatypes. The system of built-ins should as well help in the interoperation of SWRL with other Web formalisms, by providing an extensible, modular built-ins infrastructure for Semantic Web Languages, Web Services, and Web applications. Many built-ins are defined. These built-ins are keys for any external integration. This project takes advantages of this extensional mechanism to integrate new Built-ins for 3D processing and topological processing.

D. Interaction process

To focus on our method for the combination of the Semantic Web technologies and the 3D processing algorithms, Figure 4 illustrates an UML sequence diagram that represents the general design of the proposed solution. Hence, the purpose is to create a more flexible, easily extended approach where algorithms will be executed reasonably and adaptively on particular situations following an interaction process.

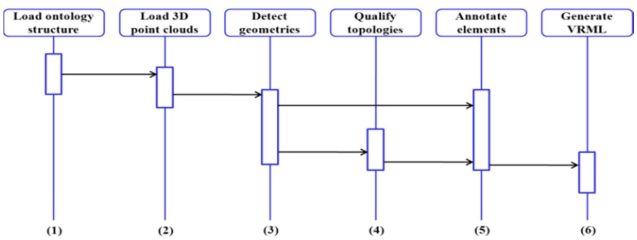


Figure 4. The sequence diagram of interactions between the laser scanner, the 3D processing, the knowledge processing and the knowledge base.

The processing steps can be detailed where three main steps aim at detecting and identifying objects.

(3) From 3D point clouds to geometric elements.
(4) From geometry to topological relations.
(5) From geometric and/or topological relations to semantic elements annotated.

As intermediate steps, the different geometries within specific 3D point clouds are detected and stored in the ontology structure. Once done, the existent topological relations between the detected geometries are qualified and then populated within the prior knowledge. Finally, detected geometries are annotated semantically, based on existing knowledge's related to the geometric characteristics and topological relations. The input ontology contains knowledge about the Deutsche Bahn railway objects and knowledge about 3D processing algorithms.

IV. DESCRIPTION OF THE WIDOP KNOWLEDGE BASE

This section discusses the different aspects related to the Deutsche Bahn scene ontology structure installed behind the WiDOP Deutsche Bahn prototype [11]. The domain ontology presents the core of WiDOP project and provides a knowledge base to the created application. The global schema of the modeled ontology structure offers a suitable framework to characterize the different Deutsche Bahn elements from the 3D processing point of view. The created ontology is used basically for two purposes:

- To guide the processing algorithm sequence creation based on the target object characteristics.
- To facilitate the semantic annotation of the different detected objects inside the target scene.

The current ontology, following to above considerations and with respect to technological possibilities, will be modeled in various levels. In principle, we have to distinguish between object-related knowledge and algorithmic related knowledge. In addition, the same distinction has to be done on the *layer of the object knowledge* and the *layer of the algorithmic knowledge* containing the respective semantic information. In fact, the ontology is managed through different components of description logics where we find five main classes within other data and objects properties able to characterize the scene in question.

- Algorithm
- Geometry
- DomainConcept
- Characteristics
- Scene

The *DomainConcept* class can be considered as the main class in the ontology as it is the class where the target objects are modeled. However, the importance of other classes cannot be ignored. They are used to either describe the object geometry, through the *Geometry* class by defining its geometric component or the bounding rectangle of the object that indicate its coordinates, or to either describe its characteristics through the *Characteristics* class. Additionally, the suitable algorithms are automatically selected based on its compatibility within the object geometry and characteristics. Add to that, other classes are equally significant but play their roles in the backend. The connection between the basic mentioned classes is carried out through object and data properties. There exist object properties for each mentioned activities. Besides, the object properties are also used to relate an object to other objects via topological relations. In general, there are five general object properties in the ontology which have their specialized properties for the specialized activities, Figure 5. They are:

- hasTopologicRelation
- IsDeseignedFor
- hasGeometry
- hasCharacteristics

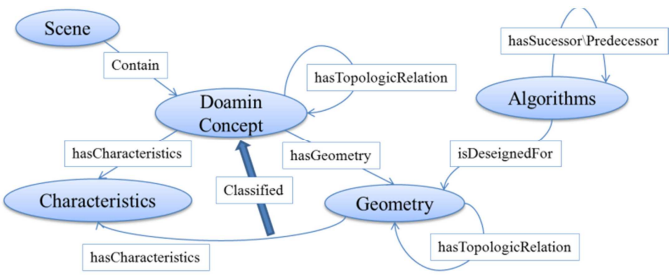


Figure 5. Ontology general schema overview

The next sections focus on the layers, the object and the algorithmic knowledge definition.

A. Layers of object knowledge

The object knowledge layer will be classified in three categories: geometric, topological and semantic knowledge representing a certain scenario [35]. Therefore we distinguish between:

- Deutsche Bahn Scene knowledge
- Geometric knowledge
- Topological knowledge

1) Layer of the Deutsche Bahn Scene knowledge

The layer of object knowledge contains all relevant information about objects and elements which might be found within a Deutsch Bahn scene. This might comprise a list such as: {Signals, Mast, Schalanlage, etc.}. They are used to fix either the main scene within its point clouds file and its size through attributes related to the scene class, or even to characterize detected element with different semantic and geometric characteristics.

The created knowledge base related to the Deutsche Bahn scene has been inspired next to our discussion with the domain expert and next to our study based on the official Web site for the German rail way specification [46]. An overview of the targeted elements, the most useful and discriminant characteristics to detect it and their inter-relationship is presented.

Table 1. EXAMPLE OF THE DB SCENE OBJECTS

Class	Sub Class	Subsub Class	Height	Correspondent image
Signals	Basic Signals	Main Signal	Between 4 and 6 m	
		Distant Signal	Between 4 and 6 m	
	Secondary signal	Vorsignalbake	between 1,5 and 2.5 m	
		Breakpoint_table	between 1 and 2 m	
		Chess_board	between 1 and 1,5 m	
Mast	BigMast	More than 6m		
	NormalMast	Between 5 and 6		
Schaltanlage	Schalthause	Less than 1m		
	SchaltSchrank	Less than 0,5m		

Table **1** shows a possible collection of scene elements in case of a Deutsche Bahn scene. They may be additionally structured in a hierarchical order as might be seen convenient for a scene, while Figure 6 shows the suggested structure to model them within the OWL language.

Basically, a railway signal is one of the most important elements within the Deutsche Bahn scene, where we find main signals and secondary ones. The main signals are classified onto *the primary signal* and the *distant ones*. In fact, the primary signal is a railway signal. It indicates whether the subsequent section of track may be driven on. A primary signal is usually announced through a distant signal. The last one indicates which image signal to be expected, that will be associated to the main signal in a distance of 1 km. Big variety of secondary signals exists like the Vorsignalbake, the Haltepunkt and others. From the other side, the other discriminant elements within the same scene are the Masts presenting electricity born for the energy alimentation. Usually, masts are distant from 50 m to each other. Finally, the Schaltanlage elements present small electric born connected to the ground.

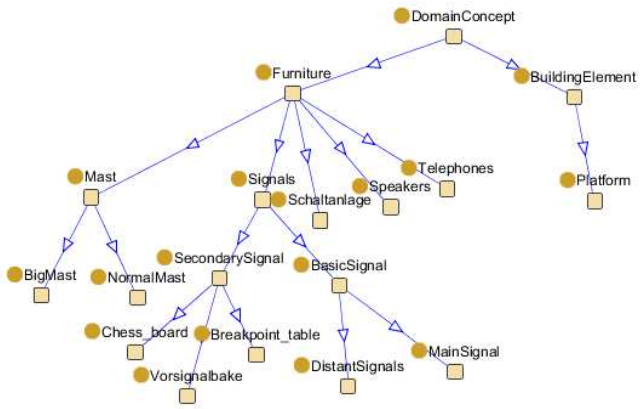


Figure 6. Example of the DB scene objects modeling

Additionally, the above cited concepts are extended by relations to other classes or data. As an example, the data property "*has_Bounding_Box*" aims to store the placement of the detected object in a bounding box defined by its eight 3D points (each 3D point is defined by three values x, y and z).
To specify its semantic characteristics, new classes are created, aiming to characterize a semantic object by a set of characteristics like color, size, visibility, texture, orientation and its position in the point cloud after detection. To do so, new object properties like "*has_Color*", "*has_Size*", "*has_Orientation*", "*has_Visibility*" and "*has_Texture*" are created linking the *Semantic_Object* class to the "*color*", "*size*", "*Orientation*", "*Visibility*" and "*Texture*" classes respectively.

2) Layer of the geometric knowledge

Geometrical knowledge formulates geometrical characteristics to the physical properties of scene elements. In the simplest case, this information might be limited to few coordinates expressing a bounding box containing the object. However, for elements being accessible to functional descriptions, additional knowledge will be mentioned. A signal, for example, has vertical lines, which needs to be described by a line equation, its values and completed by width and height. In fact, we think that such knowledge can present a discriminant feature able to improve the automatic annotation process. For this reason, we opt to study the different geometric features related to the cited semantic elements, then, use only the discriminant one as basic features for a given object. The following table gathers the object characteristics together regarding the properties of a bounding box,

Table **2**, Figure 6. This table is extended with algorithm characteristics, but it is not presented here.

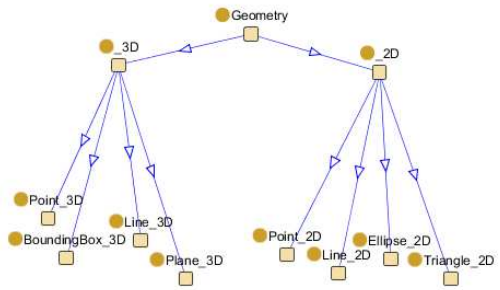


Figure 7. The geometry class hierarchy

Table 2. Geometric characteristics overview

Class	SubClass	Subsub Class	Restriction on Line number	Restriction on Planes number
Signals	Basic Signals	Main Signal	1 or 2 Vertical line	0
		Distant Signal	1 or 2 Vertical line	0
	Secondary signal	Vorsignalbake	1 Vertical line	1 Vertical plane
		Breakpoint_table	2 Vertical lines	1 Vertical Plan
		Chess_board	1 Vertical line	1 Vertical plane
Mast	BigMast	More than 6m	2 or 4 vertical lines	0
	NormalMast	Between 5 and 6	2 or 4 vertical lines	0
Schaltanlage	Schalthause	Less than 1m		1 Vertical plane 1 Horizontal plane
	SchaltSchrank	Less than 0,5m		1 vertical plane

3) Layer of the topological knowledge

While exploring the railway domain, lots of standard topological rules are imposed; such rules are used to help the driver and to ensure the passengers' security. From our point of view, these are helpful also to verify and to guide the annotation process. In fact, topological knowledge represents adjacency relationships between scene elements. For instance, and in case of the Deutsche Bahn scene, the distance between the distant signal and the main one corresponds to the stopping distance that the trains require. The stopping distance shall be set on specific route and is in the main lines often 1000 m or in a rare case, 700 m. Add to that, three to five Vorsignalbake are distant from 75m while then the last one is distant from 100m to the distant signal, Figure 8.

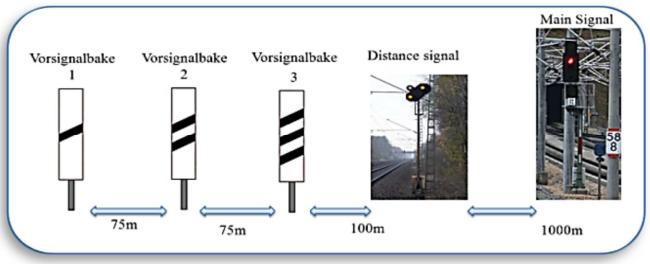


Figure 8: Topologic rules

The purpose of such object properties is to spatially connect *Things* presented in the scene. At semantic view, topological properties describe adjacency relations between classes. For example, the property isParallelTo allows characterizing two geometric concepts by the feature of parallelism. Similarly relations like isPerpendicularTo and isConnectedTo will help to characterize and exploit certain spatial relations and make them accessible to reasoning steps.

B. Layer of processing knowledge

The 3D processing algorithmic layer contains all relevant aspects related to the 3D processing algorithms. It´s integration into the semantic framework is done by special Built-Ins called “Processing Built-Ins”. They manage the interaction between above mentioned layers. In addition, it contains algorithm definitions, properties, and geometries related to each defined algorithms. An importance achievement is the detection and the identification of objects, which has a linear structure such as signal, indicator column, and electric pole, etc., through utilizing their geometric properties. Since the information in point cloud data sometimes is unclear and insufficient, the various methods to RANSAC [24] are combined and upgraded. This combination is able to robustly detect the best fitting lines in 3D point clouds for example.

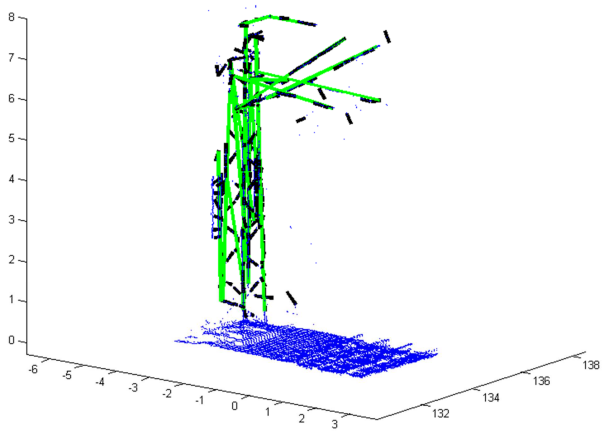


Figure 9. Mast detection

Figure 9 contains the Mast object constructed by linear elements, ambiguously represented in point cloud as blue points. Green lines are results of possible fitting lines and clearly show the shape of the object that is defined in the ontology. The object generated from this part is a bounding box that includes all inside geometries of the object and a concept label.

Next to the 3D expert recommendation, knowledge within the Table 3 is created, where a set of 3D processing algorithms within the target detected geometry are structured; the input and output are created.

Table 3. 3D Algorithms and experts observations

Algorithm name	has Input	hasOutput	isDesignedfor	hasSuccessor
Vertical Objects Detection	PointCloud	Point_2D	Vertical gemetry	None
Segmentation2D	Point_2D PointCloud	SubPointCloud	Vertical gemetry	VerticalObjectsDetection
BoundingBox	SubPointCloud	Point_3D	Vertical gemetry	Segmentation in2D
Approximate Height	SubPointCloud	number	Geometry height	Segmentation in2D
RANSAC Line Detection	SubPointCloud	Line_3D	3D Lines	Segmentation in2D)
FrontFaceDetection	SubPointCloud	Boolean	Geometry with front face	Segmentation in2D
CheckPerpendicular	Line_3D	Boolean angle	Geometry containing Perpendicular elements	LinesDetectionin3DbyRANSAC
CheckParallel	Line_3D	Boolean angle	Geometry containing Parallel elements	LinesDetectionin3DbyRANSAC

The subclasses of the *Algorithm* class, Figure 10, are representing all the algorithms developed in the 3D processing layer. They are related to several properties which they are able to detect. These properties (Geometric and semantic) are shared with the *DomainConcept* and the *Geometry* classes. By this way, a sequence of algorithms can detect all the characteristics of an element.

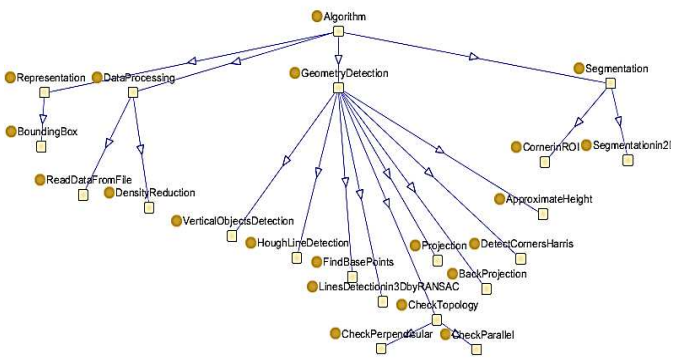


Figure 10. Hierarchical structure of the Algorithm class

The next section introduces an overview of the approach undertaken in the WiDOP project to detect and annotate semantically the different Deutsch Bahn objects.

V. INTELLIGENT PROCESS

The basic strength of formal ontology is their ability to reason in a logical way based on Descriptive Logic language DL [36]. The last one presents a form of logic to reason on objects. Lots of reasoners exist nowadays like Pellet [37], and KAON [38]. Actually, despite the richness of OWL's set of relational properties, the axioms does not cover the full range of expressive possibilities for object relationships that we might find, since it is useful to declare relationship in term of conditions or even rules. These rules are used through different rules languages to enhance the knowledge possess in an ontology.

Within the WiDOP project, the domain ontologies are used to define the concepts, and the necessary and sufficient conditions that describe the concepts. These conditions are of value, because they are used to populate new concepts. For instance, the concept "Vertical_BoudinBox" can be specialized into "Signal" if it contains a "VerticalLines". Consequently, the concept "Signal" will be populated with all "Vertical_BoudinBox" if they are linked to a "VerticalLines" with certain parameters. In addition, the rules are used to compute more complex results such as the topological relationships between objects. For instance, the relations between two objects are used to get new efficient knowledge about the object. The ontology is than enriched with this new relationship. The topological relation built-ins are not defined in the SWRL language. Consequently, the language was extended.

To support the defined use cases, two basic further layers to the semantic one are added to ontology in order to ensure the geometry detection and annotation process tasks. These operations are the 3D processing and topological relations qualification respectively.

A. Integration of 3D processing operations

The 3D processing layer contains all relevant aspects related to the 3D processing algorithms. Its integration into the WiDOP semantic framework is done by special Built-Ins. They manage the interaction between processing layers and the semantic one. In addition, it contains the different algorithm definitions, properties, and the related geometries to the each defined algorithms. An importance achievement is the detection and the identification of objects with specific characteristics such as a signal, indicator columns, and electric pole, etc. through utilizing their geometric properties. Since the information in point cloud data sometimes is unclear and insufficient.

The Semantic Web Rule Language within extended built-ins is used to execute a real 3D processing algorithm, and to populate the provided knowledge within the ontology (e.g. Table 4). The "3D_swrlb_Processing: VerticalElementDetection" built-ins for example, was created, it aims at the detection of geometry with vertical orientation. The prototype of the designed Built-in is:

3D_swrlb_Processing:VerticalElementDetection(?Vert, ?Dir)

where the first parameter presents the target object class, and the last one presents the point clouds' directory defined within the created scene in the ontology structure. At this point, the detection process has as a result a set of bounding boxes, representing a rough position and orientation of the detected object. Table 4 shows the mapping between the 3D processing built-ins, which is computer and translated to predicate, and the corresponding class.

TABLE 4. 3D PROCESSING BUILT-INS MAPPING

3D Processing Built-Ins	Correspondent Simple class
3D_swrlb_Processing: VerticalElementDetection (?Vert,?Dir)	Vertical_BoundingBox(?x)
3D_swrlb_Processing: HorizentalElementDetection (?Vert,?Dir)	Horizental_BoundingBox(?y)

B. *Integration of Topologic operations*

The layer of the topological knowledge represents topological relationships between scene elements since the object properties are also used to link an object to others by a topological relation. For instance, a topological relation between a distant signal and a main one can be defined, as both have to be distant from one kilometer. The qualification of topological relations into the semantic framework is done by new topological Built-Ins.

This step aims at verifying certain topology properties between detected geometries. Thus, 3D_Topologic built-ins have been added in order to extend the SWRL language. Topological rules are used to define constrains between different elements. After parsing the topological built-ins and its execution, the result is used to enrich the ontology with relationships between individuals that verify the rules. Similarly to the 3D processing built-ins, our engine translates the rules with topological built-ins to standard rules, Table 5.

TABLE 5. EXAMPLE OF TOPOLOGICAL BUILT-INS

Processing Built-Ins	Correspondent object property
3D_swrlb_Topology:Upper(?x, ?y)	Upper(?x,?y)
3D_swrlb_Topology:Intersect(?x, ?y)	Intersect (?x,?y)

C. *Guiding 3D processing algorithms*

Actually, the created knowledge base aims to satisfy to basic purposes which are:

- Guiding the processing algorithm sequence creation based on the target object characteristics.
- Facilitate the semantic annotation of the different detected objects inside the target scene.

Let's remember that the one of the main ideas behind this project is to direct, adapt and select the most suitable algorithms based on the object's characteristics. In fact, one algorithm could not detect and recognize different existent objects in the 3D point clouds, since they are distinguished by different shapes, size and capture condition. The role of knowledge is to provide not only the object's characteristics (shape, size, color...) but also object's status (visibility, correlation) to algorithmic part, in order to adjust its parameters to adapt with a current situation. Based on these observations, we issue a link from algorithms to objects based on the similar characteristics as Figure 11 shows.

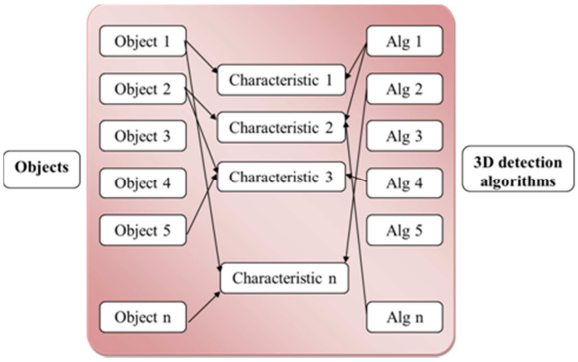


Figure 11. Algorithms selection based on object's characteristics

In fact, knowledge controls one or more algorithms for detecting object. To do this, we try to find a match between the object's characteristics and characteristics that a certain algorithm can be used for. For example, object O has characteristics: C_1, C_2, C_3; and algorithm A_i can detect characteristic C_1, C_3, C_4, while algorithm A_j can detect characteristic C_2, C_5. Then, decision algorithm will select A_i and A_j since these algorithms have capability detecting the characteristics of object O. The set of characteristics are determined by the object's properties such as geometrical features and appearance. Once done, selected algorithms will be executed and target characteristics will be detected.

The whole process takes as input the 3D point clouds scenes, an ontology structure presenting a knowledge base to manipulate objects, geometries, topologies and relations (Object and data property) and produces as an output, an annotated scene within the same ontology structure. As intermediate steps, the different geometries within a specific 3D point cloud scene are detected and stored in the ontology. Once knowledge about geometries and the topologies are experienced, SWRL rules aim at qualifying and annotating the different detected geometries. The following simple example shows how a SWRL rule can specify the class of a VerticalBoundingBox which is of type Mast regarding its altitude. The altitude is highly relevant only for this element.

```
3DProcessing_swrlb:VerticalElementDetection(?
Vert, ?dir) ^ altitude (?x, ?alt)
^swrlb:moreThan (?alt, 6) → Mast (?Vert)
```

In other cases, geometric knowledge is not sufficient for the previous process. The topological relationships between detected geometries are helpful to manage the annotation

process. The following example shows how semantic information about existing objects is used conjunctly with topological relationships in order to define the class of another object.

```
Mast (?vert1) ^ VerticalBB (?Vert2) ^
hasDistanceFrom (?vert1,?vert2, 50) →
Mast(?vert2)
```

VI. WiDOP PROTOTYPE

WiDOP prototype takes in consideration the adjustment of the old methods and, in the meantime, profit from the advantages of the emerging cutting-edge technology. From the principal point of view, our system still retains the storing mechanism within the existent 3D processing algorithms; in addition, suggest a new field of detection and annotation, where we are getting a real-time support from the target scene knowledge. Add to that, we suggest a collaborative Java Platform based on semantic web technology (OWL, RDF, and SWRL) and knowledge engineering in order to handle the information provided from the knowledge base and the 3D packages results.

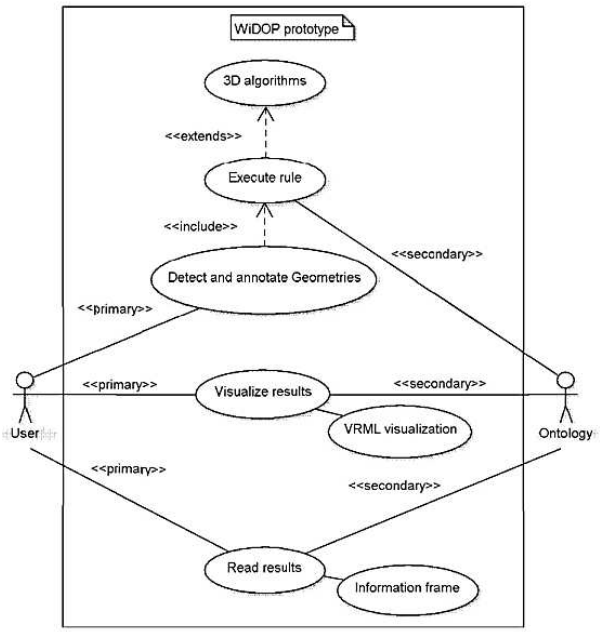


Figure 12. the WiDOP use case diagram

The process enriches and populates the ontology with new individuals and relationships between them. In order to graphically represent these objects within the scene point clouds, a VRML model file [7] is generated and visualized within the prototype where the color of objects in the VRML file represents its semantic definition. The resulting ontology contains enough knowledge to feed a GIS system, and to generate IFC file [37] for CAD software. As seen in Figure 12, the created system is composed of three parts.

- Generation of a set of geometries from a point could file based on the target object characteristics.
- Computation of business rules with geometry, semantic and topological constrains in order to annotate the different detected geometries.
- Generation of a VRML model related to the scene within the detected and annotated elements.

In addition, the created WiDOP platform offers the opportunity to materialize the annotation process by the generation and the visualization based on a VRML structure alimented from the knowledge base. It ensures an interactive visualization of the resulted annotation beginning from the initial state, to a set of intermediate states, coming finally to an ending state, Figure 13 where the set of rules are totally executed.

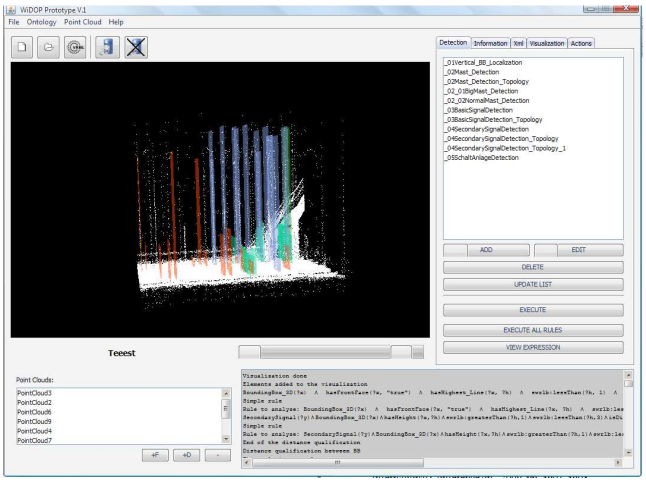


Figure 13. Snapshot of the WiDOP prototype

As a first impression, the system responds to the target requirement since it would take a point cloud of a facility as input and produce a fully annotated as-built model of the facility as output.

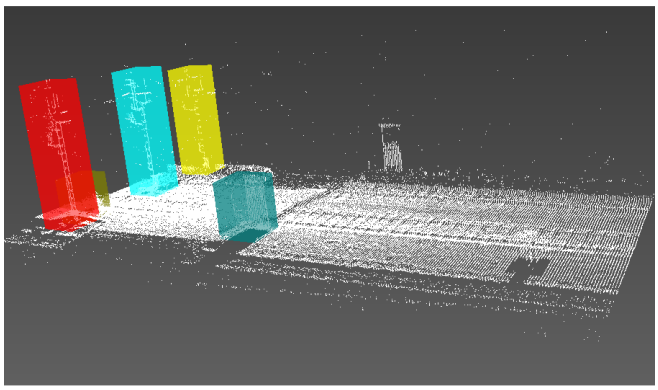


Figure 14. Detected and annotated elements visaliazation within VRML language

VII. BENCHMARKS

A. Annotation process summerized

For the demonstration of our prototype, two different sections from the whole scanned point clouds related to Deutsch Bahn scene in the city of *Nürnberg* was extracted. While the last one measure 87 km, we have just taken two small scenes of 500 m each one. Each one of the kept scenes contains a variety of the target objects. The whole scene has been scanned using a terrestrial laser scanner fixed within a train, resulting in a large point cloud representing the surfaces of the scene objects. Within the created prototype, different SWRL rules are processed, e.g. Figure 15. First, geometrical elements will be searched in the area of interest based on dynamic 3D processing algorithm sequence created based on semantic object properties.

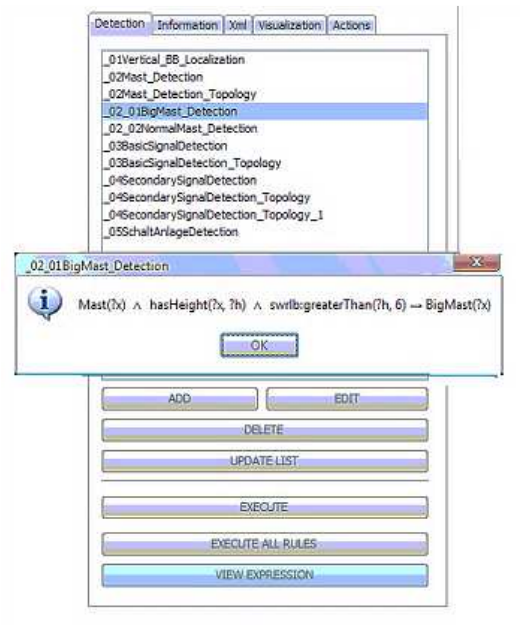


Figure 15. Example of executed rules

Once done, the second step within our approach aims to identify existing topologies between the detected geometries. Thus, useful topologies for geometry annotation are tested. Topological Built-Ins like `isConnected`, `touch`, `Perpendicular`, `isDistantfrom` are created. As a result, relations found between geometric elements are propagated into the ontology, serving as an improved knowledge base for further processing and decision steps.

The last step consists in annotating the different geometries. Vertical elements of certain characteristics can be annotated directly. Subsequently, further annotation may be relayed on aspects expressing facts to orientation or size of elements, which may be sufficient to finalize a decision upon the semantic of an object or, in more sophisticated cases, our prototype allows the combination of semantic information and topological ones that can deduce more robust results by minimizing the false acceptation rate, Figure **16**. By this way, and based on a list of SWRL rules, most of the detected geometries are annotate as seen within Table 6, Table 7, Table8.

B. System evaluation

Our testes had been made on two different data bases with 500 m long extracted from the whole scanned point clouds data. Where the first scene contains just 37 elements, and the second one contain 128 elements. As a first impression, it´s totally reasonable that the number of elements varies from a scene to another, because we are near from the rail way station, more the scene is rich and vice versus. It's also clear from the above-mentioned tables, how our knowledge base could recognize which geometry represents a real element from those which are noise, Table 6.

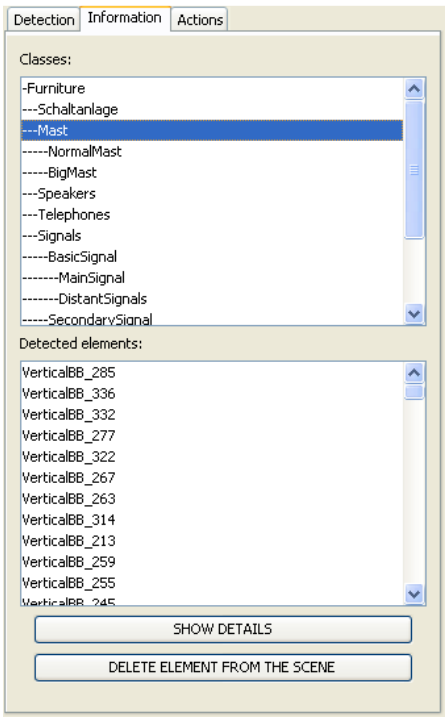


Figure 16. Annotated Bounding Box as Masts

As well, in most cases, our annotation process is able to affect the right label to the detected Bounding box based on knowledge on its component, its internal and external topology. In the first example, Table 7, among 13 elements are classified as Masts, three as a SchaltAnlage and 18 signals. While in the second scene, among 67 elements are classified as Masts, 55 signals and finally 155 Schaltanlage, Table 8.

TABLE 6. DETECTED ELEMENT WITHIN THE SCENE AND ANNOTATED ONES

	Scene Size	**Detected Bounding Box**	**Annotated elements**	**Truth data**
Scene1	500m	105	34	37
Scene2	500m	344	277	128

TABLE 7. DETECTED AND ANNOTATED ELEMENTS WITHIN THE SCENE1

	Masts	signal	Schaltanlage
Annotated	13	18	3
Truth data	12	20	5

TABLE 8. DETECTED AND ANNOTATED ELEMENTS WITHIN THE SCENE2

	Masts	signal	Schaltanlage
Annotated	67	55	155
Truth data	65	50	13

Some clear limits are detected within the Table 8. Where lots of false Schaltanlage are detected and annotated. Before explaining the reason behind this false detection, let's recall that the Schaltanlage present very small electronic boxes installed on the ground. In the case of scene 2 which is near the rail station, the level of the ground is a higher compared to the other scenes. For this reason, lots of bounding boxes are detected where a high average of them presents small noise on the ground. The reason for the false annotation is the lack of semantic characteristics related to such elements because, until now; there is no real internal or external topology neither internal geometric characteristic that discriminate such an element compared to others.

VIII. DISCUSSION AND CONCLUSION

We have presented an automatic system for survey information model creation based on semantic knowledge modeling. Our solution aims to perform the detection of objects from a technical survey within the laser scanner technology by using available knowledge about a specific domain (DB). This prior knowledge is modeled within an ontology structure. SWRL rules are used to control the 3D processing execution, the topological qualification and finally to annotate the detected elements in order to enrich the ontology and to drive the detection of new objects.

The designed prototype takes 3D point clouds of a facility, and produce fully annotated scene within a VRML model file. The suggested solution for this challenging problem has proven its efficiency through real tests within the Deutsche Bahn scene. The creation of processing and topological Built-Ins has presented a robust solution to resolve our problematic and to prove the ability of the semantic web language to intervene in any domain and create the difference.

Future work will include the integration of new knowledge's that can intervene within the annotation process like the number of detected lines within each bounding box and the update of the general platform architecture, by ensure more communication between the scene knowledge within the 3D processing one. It will also include a more robust identification and annotation process of objects based on individual object characteristics. Finally, further knowledge related to the algorithm parameterization that can intervene within the detection and annotation process will be studied to make the process more flexible and intelligent.

IX. ACKNOWLEDGMENT

This paper presents work performed in the framework of the research project funded by the German ministry of research and education under contract No. 1758X09. The authors cordially thank for this funding. Special thinks also for Andreas Marbs, Ashish Karmacharya, and Hung Truong for their contribution.

REFERENCES

[1] W. Boehler, M. Bordas Vicent, and A. Marbs, "Investigating laser scanner accuracy," The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. 34, pp. 696-701, 2003.

[2] R. Wessel, R. Wahl, R. Klein and R Schnabel, "Shape recognition in 3D point clouds," in Proc. Conf. in Central Europe on Computer Graphics, Visualization and Computer Vision., 2008, vol. 2.

[3] A. Kim, V G. Funkhouser and T. Golovinskiy, "Shape-based recognition of 3d point clouds in urban environments," in 12th International Conference on Computer Vision, IEEE, 2009, pp. 2154-2161.

[4] C. Cruz, Y. Duan and C. Nicolle, "Architectural Reconstruction of 3D Building Objects through Semantic Knowledge Management," in 11th ACIS International Conference on Software Engineering Artificial Intelligence Networking and Parallel/Distributed Computing (SNPD), 2010, pp. 261-266.

[5] W.N. Borst, J.M. Akkermans, and J.L. Top, "Engineering Ontologies," International Journal of Human-Computer Studies, vol. 46, pp. 365- 406, 1997.

[6] H. Ben Hmida, C. Cruz, C. Nicolle and F. Boochs, "Semantic-based Technique for the Automation the 3D Reconstruction Process," in SEMAPRO 2010, The Fourth International Conference on Advances in Semantic Processing, Florence, Italy, 2010, pp. 191-198.

[7] VRML Virtual Reality Modeling Language. (1995, Apr.) W3C. [Online]. http://www.w3.org/MarkUp/VRML/

[8] J. Seo and I. Kim, "Industry Foundation Classes-Based Approach for Managing and Using the Design Model and Planning Information in the Architectural Design," Journal of Asian Architecture and Building Engineering, vol. 8, pp. 431-438, 2009.

[9] S. Vosselman and G. Pu, "Extracting windows from terrestrial laser scanning," Intl Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. 36, pp. 12-14, 2007.

[10] R B. Marton, Z C. Blodow, N. Holzbach, A. Beetz and M Rusu, "Model-based and learned semantic object labeling," in IEEE/RSJ International Conference on 3D point cloud maps of kitchen environments," in Intelligent Robots and Systems, 2009. IROS 2009, pp. 3601-3608.

[11] H. Ben Hmida, C. Cru, C. Nicolle and F. Boochs, "From 3D point clouds to semantic object" in KEOD 2011, the International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management.

[12] H. Ben Hmida, A. Marbs, H. Truong, A. Karmacharya, C. Cruz, A. Habed, C. Nicolle, Y. Voisin and Frank Boochs, "Integration of knowledge to support automatic object reconstruction from images and 3D data," in International Multi-Conference on Systems, Signals & Devices, Sousse Tunisia, March 22-25, 2011.

[13] H. Cantzler, "Improving architectural 3D reconstruction by constrained modelling," College of Science and Engineering. School of Informatics, 2003.

[14] Leica Cyclone. (2011) [Online]. http://hds.leica-geosystems.com/en/Leica-Cyclone_6515.htm

[15] N. Andreas, "Automatic Model Refinement for 3D Reconstruction with Mobile Robots," 3DIM Fourth International Conference on 3-D Digital Imaging and Modeling, 2003., pp. 394-401, 2003.

[16] H.E Goldberg, "State of the AEC industry: BIM implementation slow, but inevitable," Revista CAD alystmaio, 2005.

[17] H. Becerik-Gerber and B. Hajian, "A Research Outlook for Real-Time Project Information Management by Integrating Advanced Field Data", In ASCE Acquisition Systems and Building Information Modeling", 2009.

[18] Leadwerks Corporation. (2006) What is Constructive Solid Geometry? [Online]. http://www.leadwerks.com/files/csg.pdf

[19] OPEN CASCADE. (2000) OpenCascade - an open source library for BRep solid modeling. [Online]. http://www.opencascade.org/

[20] R.J. Flynn and P.J. Campbell, "A survey of free-form object representation and recognition techniques," Computer Vision and Image Understanding, vol. 81, pp. 166-210, 2001.

[21] G. Vosselman and S. Pu, "Knowledge based reconstruction of building models from terrestrial laser scanning data," ISPRS Journal of Photogrammetry and Remote Sensing, vol. 64, pp. 575-584.

[22] K. Huber, D. Akinci, B. Krishnamurti and R. Yue, "The ASDMCon project: The challenge of detecting defects on construction sites," International Symposium on 3D Data Processing Visualization and Transmission, vol. 0, pp. 1048-1055, 2006.

[23] F. Haas and CT. Bosche, "Automated retrieval of 3D CAD model objects in construction range images," Automation in Construction, vol. 17, pp. 499-512, 2008.

[24] F. Tarsha-Kurdi, T. Landes, and P. Grussenmeyer, "Hough-transform and extended RANSAC algorithms for automatic detection of 3D building roof planes from Lidar data," International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, ISPRS, Volume 3, Issue Part 3/W52, p.407-412, 2007.

[25] A. Milella and R. Siegwart, "Stereo-based ego-motion estimation using pixel tracking and iterative closest point," in IEEE International Conference on Computer Vision Systems 2006, pp. 21-21.

[26] C. A. Cantrell, "Technical Note: Review of methods for linear least-squares fitting of data and application to atmospheric chemistry problems," Atmospheric Chemistry and Physics Discussions, vol. 8, pp. 6409-6436, 2008.

[27] building SMART International Ltd. (2008) Industry Foundation Classes (IFC) — BuildingSmart, International Alliance for interoperability. [Online]. http://buildingsmart-tech.org/

[28] T. Berners-Lee. J. Hendler and O. Lassila, The Semantic Web," Scientific American, 2001.

[29] D. Huynh, S. Mazzocchi and D. Karger, "Piggy bank: Experience the semantic web inside your web browser," The Semantic Web-ISWC 2005, pp. 413-430, 2005.

[30] G. Wang, "Methodology Research of Ontology Building in Semantic Web," Computer and Information Science, vol. 3, p. p236, 2010.

[31] F. Baader and W. Nutt, "Basic Description Logics". In the Description Logic Handbook, edited by F. Baader, D. Calvanese, DL McGuinness, D. Nardi, PF Patel-Schneider, 2002.

[32] C. Cruz, F. Marzani, and F. Boochs, "Ontology-driven 3D reconstruction of architectural objects," VISAPP (Special Sessions), pp. 47-54, 2007.

[33] T. Gruber. (2005) www-ksl.stanford.edu/kst/what-is-an-ontology.html.

[34] D. L. McGuinness and F. v. Harmelen. (2004, February) W3C Recommendation. [Online]. http://www.w3.org/TR/owl-features/

[35] E. J. Whiting, "Geometric, Topological & Semantic Analysis of Multi-Building Floor Plan Data," phdthesis 2006.

[36] F. Baader, I. Horrocks, and U. Sattler, "Description logics," Foundations of Artificial Intelligence, vol. 3, pp. 135-179, 2008.

[37] E. Sirin, B. Parsia, B C. Grau, A. Kalyanpur and Y. Katz, "Pellet: A practical owl-dl reasoner," Web Semantics: science, services and agents on the World Wide Web, vol. 5, pp. 51-53, 2007.

[38] B. Motik, U. Sattler and U. Hustadt. (2010) KAON2. [Online]. http://kaon2.semanticweb.org/

[39] J J. Carroll et al., "Jena: implementing the semantic web recommendations," in Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters, 2004, pp. 74-83.

[40] I. Horrocks et al., "SWRL: A semantic web rule language combining OWL and RuleML," W3C Member submission, vol. 21, 2004.

[41] R. Vanland, C. Cruz and C. Nicolle, "IFC and building lifecycle management," Automation in Construction, vol. 18, pp. 70-78, 2008.

[42] Jena – A Semantic Web Framework for Java. [Online]. http://jena.sourceforge.net/

[43] I. Horrocks et al., "SWRL: A semantic web rule language combining OWL and RuleML," W3C Member submission, vol. 21, 2004.

[44] OWL Web Ontology Language Guide. (February 2004) W3C. [Online]. http://www.w3.org/TR/owl-guide/. The last access date: 01-2012.

[45] SWRL: A Semantic Web Rule Language Combining OWL and RuleML. (May 2004) W3C. [Online]. http://www.w3.org/Submission/SWRL/. The last access date: 01-2012.

[46] Alles über Stellwerke. (July 2007) Stellwerke. [Online]. http://www.stellwerke.de. The last access date: 01-2012.

Integrating Web-Enabled Energy-Aware Smart Homes to the Smart Grid

Andreas Kamilaris*, Yiannis Tofis†, Chakib Bekara‡, Andreas Pitsillides* and Elias Kyriakides†
* Department of Computer Science
Networks Research Laboratory
University of Cyprus
P.O. Box 20537, Nicosia, CY 1678, Cyprus
Email: kami@cs.ucy.ac.cy, andreas.pitsillides@ucy.ac.cy
†KIOS Center for Intelligent Systems and Networks
Department of Electrical and Computer Engineering
University of Cyprus
P.O. Box 20537, Nicosia, CY 1678, Cyprus
Email: tofis.n.yiannis@ucy.ac.cy, elias@ucy.ac.cy
‡ Fraunhofer FOKUS Institute
Kaiserin-Augusta-Allee 31, D-10589
Berlin, Germany
Email: chakib.bekara@fokus-extern.fraunhofer.de

***Abstract*—Energy conservation is a global issue with great implications. High energy demands and environmental concerns force the transformation of electricity grids into smart grids, towards more rational utilization of energy.**

Embedded computing and smart metering transform houses into energy-aware environments, allowing residents to make informed choices about electricity. Web technologies are successfully used for managing heterogeneous home devices, facilitating the remote management of the house through Web APIs. Hence, the Web, as an ubiquitous and scalable platform, is suitable for interconnecting energy-aware smart homes and the smart grid.

In this paper, we investigate the possibilities created when energy-aware smart homes communicate in near real-time with the smart grid and we propose an architecture for their flexible integration to the grid, through the Web. A proof of concept deployment is performed and general security aspects are discussed. The potential of this Web-based architecture is demonstrated by developing two applications that exploit these new capabilities of smart homes, towards an intelligent grid. Demand response is harnessed to schedule electricity-related tasks for future execution and load shedding is employed to reduce the total load for avoiding outages. Finally, issues such as peak leveling, fault tolerance, billing and a market for energy are briefly discussed.

Keywords-Smart Home; Smart Grid; Web; Web of Things; REST; Smart Power Outlets; Demand Response; Load Shedding; Security.

I. Introduction

Increasing energy demands, depletion of natural resources and rising costs make energy conservation a universal problem with tremendous environmental, political and social implications. Predictions denote that by the year 2030, the global energy demand will double, rising up the energy-related green gas emissions by 55% [2].

This high energy demand cannot be accommodated by current electricity grids. Most of the electricity grids around the world have been built many decades ago, to meet the energy requirements of the society at that time. They are being incrementally upgraded, but these upgrades may not be completely adequate for the future grid, in addition to the green concerns.

Increasing demand and environmental concerns influenced initiatives towards a more rational utilization of electrical energy. This goal can be best achieved when the electric utilities are fully aware in real-time about the electrical consumption and the demands of their customers. The grid becomes intelligent when it manages to deliver electricity from suppliers to consumers, supported by two-way digital communications and a smart metering system, in a fault-tolerant, secure and more reliable manner. This vision is believed to convert traditional electricity grids into modern *smart grids*. A smart grid[1] is a network of networks that has come to describe the future electricity grid, enhanced with Information and Communication Technology (ICT), applied to generation, delivery and consumption of electric power.

Electricity smart metering involves measuring the consumption of electrical energy in frequent intervals and communicating that information at least daily back to the utility for monitoring and billing purposes. Smart metering does not only affect the future development of the smart grid, but also motivates the rational management of the electrical consumption in houses and buildings. Buildings consume a large proportion of the world's total electrical energy [3]. This fact has a significant environmental impact, as more than 30% of all greenhouse gas emissions is attributed to buildings.

Lately, residential smart meters have been introduced in our lives as sensor devices that measure in small time intervals

[1] http://smartgrid.ieee.org/ieee-smart-grid

the energy consumption of a house. In the near future, smart appliances would take advantage of the smart grid's functionality to synchronize their operations with its current state. For example, they may respond to pricing signals and decide when it is most economical to operate. An intermediate step before utilizing smart appliances could be the use of smart power outlets, which are devices that measure the consumption of electrical appliances and control their operation in real-time. In general, the practice of equipping home area networks (HAN) with smart meters, smart appliances and smart power outlets, enables the development of *energy-aware smart homes*.

Undoubtedly, these new technological advancements of HAN offer new possibilities for effective energy management and conservation. The capabilities of being informed about the domestic electrical consumption in (near) real-time and being able to control the electrical appliances of a house remotely, enable novel applications to be developed for saving energy. Especially, when combined with the operations of the smart grid, these capabilities could offer great potential towards a coordinated, large-scale plan for energy efficiency.

Nowadays, smart home solutions are vendor-specific and heterogeneous, employing various hardware and software technologies to achieve home automation. This trend is expected to continue in the future. Hence, in order to enable the smart grid vision, a common ground needs to be specified, a common language understood by all HAN, facilitating in-home and home-to-grid communication. We anticipate that the Web, as a highly scalable, pervasive and flexible platform, is an appropriate solution for such a wide-scale interconnection.

Reusing well-accepted and understood Web principles to interconnect heterogeneous embedded devices, built into everyday smart things, is the vision of the *Web of Things* (WoT) [4], [5]. It is about using the Web as a standard, to assure interoperability in resource-constrained, pervasive spaces. The concepts of the WoT have also penetrated in smart home environments, in which the performance of Web-enabled home devices is considered acceptable [6], [7].

The main contribution of this paper is to provide the architecture for the integration of energy-aware smart homes to the smart grid via the Web, emphasizing on the smart home environment, and also to demonstrate potential applications based on the proposed platform. An energy-aware smart home is deployed using smart power outlets and its remote management is enabled through the Web. The emulation of a smart grid scenario allows the (near) real-time Web-based communication between the HAN and the grid. As security is a crucial topic in this initiative, a section is dedicated on explaining in general how the whole system can be secured.

Through this proof of concept deployment, some interesting applications related to the smart grid are investigated, from an energy-aware smart home perspective. More specifically, the dynamic pricing program of the grid is exploited to schedule electricity-related tasks for the future and load shedding is studied, as a technique to reduce total consumption for avoiding outages. Furthermore, other possible applications such as peak leveling, fault tolerance, automatic billing and a distributed electricity market are discussed.

We note that in this paper the focus is on the ICT infrastructure that could be used for enabling flexible, reliable and efficient integration of smart homes to the grid. The proposed architecture targets the non-critical systems of the power grid. It is recommended that the Supervisory Control and Data Acquisition (SCADA) system of the smart grid should be on a separate architecture/network. A SCADA system is an industrial control system used to monitor and control electrical power transmission and distribution.

The rest of the paper is organized as follows: Section II presents related work concerning mainly projects dealing with the integration of smart homes to the smart grid, along with background information about the smart grid and energy-aware smart homes. After, Section III explains the reasoning why the Web could constitute a suitable platform for this integration. Then, in Section IV the development of an energy-aware smart home using Web techniques is described and in Section V our approach for connecting HAN to the smart grid through the Web is discussed. Afterwards, Section VI considers end-to-end security in the whole Web-based infrastructure while Sections VII and VIII investigate potential applications that can be developed when enabling (near) real-time interaction between smart homes and the grid. Finally, Section IX concludes the paper and defines future work directions.

II. Related Work

In this section, the state of the art is presented regarding projects that aim to interconnect energy-aware smart homes and the smart grid. In addition, some background information is provided about the smart grid and energy-aware smart homes in general.

A. Building the Smart Grid

The smart grid is expected to modernize current electricity grids by providing advanced functionalities such as advanced management and control, high power quality, immediate failure alarms, fault localization and response to disturbance (self-healing), reliability, security, resilience to natural disasters and improved customer service.

An important characteristic of the smart grid is timely pricing, which is a smart energy pricing scheme that is set for a specific time period on an advance basis, and may change according to load demands or price changes in the market. Prices paid for electricity consumed during these periods are known to consumers a priori, based on a short-term demand forecasting, allowing them to vary their energy use in response to these prices and manage their energy costs by shifting the operation of some electrical appliances to a lower tariff period. This mechanism is mainly known as demand response (DR).

Numerous pilot projects that implement the smart grid in an experimental basis, taking into account the domestic environment, have appeared lately. We list below some of these projects, emphasizing on aspects that concern the interaction of the grid with energy-aware smart homes.

A popular project is the SmartGridCity[2], performed by Xcel Energy utility supplier in the area of Boulder, Colorado. During this project, Xcel Energy has installed approximately 23,000 smart electric meters at the customer premises, managing to collect energy usage data wirelessly and inform the customers in 15-minute intervals about their electrical consumption.

Masdar City[3] aims to be the world's first zero-carbon city, powered entirely by renewable energy sources. Pilot residences are equipped with smart meters, DR-enabled smart appliances and building management systems. By means of this infrastructure, an integrated citywide distributed management system would be created, which manages the electrical load on the grid. As an example, smart appliances are expected to customize their operation by signals received from the grid, in order to reduce the total grid's energy demand. Currently, Masdar City operates with six buildings. The city is expected to have 40,000 residents and 50,000 commuters by 2025.

BeyWatch[4] is a European project aiming to design, develop and evaluate an energy-aware, flexible and user-centric smart home solution, able to provide interactive energy monitoring for white goods, intelligent control and power demand balancing at home, block and neighbour level. ZigBee-enabled smart plugs are used for communication between home agents and the home appliances. A home agent is a middleware, implemented using OSGi service bundles, which allows seamless device/service discovery and is used mainly for energy monitoring and device control.

The Smart-A project[5] intends to consider to what extent it is possible for smart appliances to adapt their operation to variations in the energy supply. The focus is on common household appliances and, for each appliance, its operation and energy demand are modeled and its options for load shifting are analyzed. In this way, the impact on appliance design and potential service is assessed. This project offers useful results that may be used for more analytic approaches regarding operations of the smart grid such as load shedding.

The work in [8] presents various security and privacy issues arising from a smart home/smart grid interaction, the vulnerabilities of the advanced metering infrastructure (AMI) and the employed ontologies (sensors, smart meters, telecommunication protocols) as well as requirements and potential solutions to the underlined challenges. This work discusses security issues in general, while the security aspects we consider in Section VI focus mainly on a Web-based environment.

Finally, the SmartHouse/SmartGrid project[6] focuses on the interconnection of smart homes and the smart grid, proposing an Internet-based interconnection by means of big Web services (WS-*) [9]. It is suggested that service-oriented architectures (SOA) are suited for integrating smart houses to the grid [10]. The role and architecture of smart meters as well as their security and business implications are additionally discussed [11].

Similarly to [10], we argue that Web services are suitable for this integration and we move one step further by developing an experimental energy-aware smart home that is synchronized with the smart grid through the Web. We believe that REST [12] constitutes a more appropriate technique in embedded computing and for smart home solutions [13], and it is nowadays mature enough, also to be employed for the communication needs with the smart grid.

Our work differs from related work by proposing a RESTful, truly Web-based architecture for integrating smart homes to the smart grid. Our proof of concept smart home deployment, using smart power outlets and a reliable application framework for smart homes, along with the development of two energy-related applications, demonstrate the potential of interconnecting smart homes and the grid through the Web. The proposed architecture offers advanced flexibility and interoperability among heterogeneous smart home solutions, respects the privacy of customers by giving them the opportunity to actively participate in the smart grid operations while security aspects may be effectively addressed by using the Web as a platform.

B. Towards Enabling Energy-Aware Smart Homes

Residential smart metering has the potential to transform home environments into energy-aware smart spaces. There exist two broad categories for household energy monitoring and control: whole-home and device-specific.

Whole-home approaches place residential smart meters where the home connects to the power grid. Such products include Wattson[7] and Current Cost[8]. Numerous efforts tried to analyze smart metering data to identify the energy consumption of household appliances. As an example, Marchiori et al. [14] used circuit-level power measurements to separate aggregated data into device-level estimates, with an accuracy of more than 90%. Additionally, ViridiScope [15] placed inexpensive sensors near electrical appliances to estimate their power consumption with less than 10% error.

Traditional smart meters offer a house-level granularity, where only the whole-home energy consumption can be visualized. As the technology advances, monitoring the energy consumption of each electrical appliance and controlling its operation becomes possible. Device-specific techniques plug smart power outlets in individual electrical appliances. Some power outlets even offer wireless networking capabilities, extending the residential smart metering infrastructure into a robust wireless network.

ACme [16] is a high-fidelity AC metering network that uses wireless sensor nodes, equipped with digital energy meters to provide accurate energy measurements of single devices. Energie Visible[9] visualizes in real-time the energy consumption of electrical appliances in a Web-based interface. In the Energy

[2] http://smartgridcity.xcelenergy.com/
[3] http://www.masdarcity.ae/en/
[4] http://www.beywatch.eu
[5] http://www.smart-a.org
[6] http://www.smarthouse-smartgrid.eu
[7] http://www.diykyoto.com/uk
[8] http://www.currentcost.com/
[9] http://www.webofthings.com/energievisible/

Aware Smart Home [17], users can use their mobile phones as ”magic lenses” to view the energy consumption of their appliances, just by pointing on them with the phone's camera.

A big challenge for energy-aware smart homes, taking into account the existence of the smart grid, is to provide to the home environment visibility of grid conditions and dynamic prices, to take local decisions and intelligently control the use of household electrical appliances, in order to save energy and money [1]. Towards this direction, an appliance scheduling approach to allow appliances to coordinate power use so that the total demand for the home is kept below a target value is investigated in [18]. In addition, an optimization technique to reduce the share of the appliances in the energy bills and to reduce their contribution to the peak load is presented in [19].

In this paper, an experimental energy-aware smart home is enabled, using smart power outlets to manage the operation of the domestic electrical appliances. Our work differs from related approaches by employing a RESTful, Web-based application framework for smart homes [6], [7], which guarantees the reliable and efficient performance of the power outlets, offering even support for prioritized requests (e.g., from the smart grid). The smart power outlets become enabled to the Web through the framework and multiple family members may interact with them in real-time.

III. Using the Web as an Integration Platform

The Web could constitute a suitable platform for bridging energy-aware smart homes and the smart grid. Through the Web, smart homes can be fully synchronized to the grid.

The Web is highly ubiquitous, flexible and it scales particularly well. A Web-based approach would guarantee high interoperability between heterogeneous smart grid technologies and components, and also between different smart home solutions and embedded sensor and actuation devices.

Most houses offer Internet connectivity nowadays, while technological advancements in mobile telecommunications such as LTE, 3G and WiMAX, permit the Internet to penetrate almost everywhere.

A. Web-based Smart Homes

Designing smart homes based on the Web principles is a recent practice. Web-based smart homes build upon the notion of the WoT [4], [5], which is about employing well-accepted Web practices to interconnect the quickly expanding ecosystem of sensors, actuators and smart physical devices.

Web-based interaction with household appliances is achieved following the REpresentational State Transfer (REST) [12], which is a lightweight architectural style that defines how to use HTTP as an application protocol. REST advocates in providing Web services modeled as *resources*. Resources may be manipulated by the methods specified in the HTTP standard (e.g., GET, PUT, POST, DELETE), under a uniform interface. REST guarantees interoperability and a smooth transition from the Web to home environments.

REST can be appropriate for enabling a Web-based smart home, as it is a flexible, loose-coupled approach that promotes using the Web as the *actual* application layer of the system. Besides, it can be easily applied for enabling resource-constrained devices such as smart appliances and smart power outlets to the Web.

An alternative design would be by employing WS-*, as proposed in [20]. WS-* [9] are a set of complex standards and specifications for enterprise application integration. They are more standardized and they could provide enhanced security. However, since they use technologies such as SOAP/XML, they are not very efficient in home environments, especially in terms of response time and energy consumption [21]. A comparison between REST and WS-* is provided in [13].

Web integration of embedded devices can be performed either through embedding Web servers directly on them or by employing gateways. Directly embedding Web servers on physical devices is a recent development [22], [23]. Web-enabled embedded devices expose their services under a RESTful application programming interface (API), while communication is based on HTTP calls.

Enabling home devices to the Web, permits the extension of Web mashups into *physical mashups* [24]. Physical mashups take advantage of real-world services offered by physical devices and combine them using the same tools and techniques of classic Web mashups. In this way, physical devices can be blended with Web content and services, without much effort.

To demonstrate the flexibility obtained by using physical mashups, we list below a PHP script that implements a physical mashup in only six lines of code, combining electrical appliances and RESTful Web services provided by an electric utility. We assume in this example that the utility exposes, as a Web API, information about its current tariffs. Offering real-time information is not infeasible for electric utilities. Recently, three utilities in California announced that they allowed their consumers to access their utility data through the Web. This initiative was called the Green Button [25].

This script checks the current *hometariff* and starts charging automatically a plug-in hybrid electric vehicle (*PHEV*) as soon as the tariff falls below the defined *LOW_TARIFF* limit.

```
<?php
$tariff=http_get("UtilityAddr/hometariff/");

if($tariff <= LOW_TARIFF){
 $req=new HttpRequest("HomeAddr/PHEV/Switch/");
 $req->setOptions(array(state=>"ON"));
 $req->send();
 $response = $http_req->getResponseBody();
}
?>
```

As a more general example, a reliable Web-based weather forecast service can be combined with smart appliances, e.g., to turn off the electric heating automatically, in case the temperature is about to increase in the next few hours.

Through the Web, residents may pull easily the data they need from an open API offered by their electric utility, and use them right away in their own applications, in any programming language that supports HTTP. Similarly, the smart home could offer its functionality as a Web API, allowing the utility to interact with it almost in real-time, for exchanging information

and remote administration and control. These simple examples indicate that advanced home automation, high flexibility, seamless integration to the smart grid and energy conservation may be achieved, when using the Web as a platform.

B. The Web and the Smart Grid

Using the Web as the ICT platform for various smart grid operations offers numerous benefits also to the power grid. These benefits concern not only the communication with energy-aware smart homes but also some internal, non-critical, ICT-related functionalities of the grid.

A smart grid implementation that exploits cloud computing, using infrastructure as a service (IaaS) from the cloud, constitutes a cost-efficient practice for enhancing the functionalities of the grid incrementally as energy demands arise. The flexibility of cloud computing enables new capabilities to be implemented on the Web, in parallel to existing operations and systems, minimizing the impact of ongoing operations.

Existing systems can be securely integrated with new components and further be connected to users and customers, by means of the Web (see Section VI). The Web constitutes a pervasive and scalable platform for incorporating third-party and partner techniques.

Furthermore, utilization of the Web would promote the Web service model, minimizing expenses for additional infrastructure and overall implementation time. Web services are core parts of cloud computing, providing a wealth of proven methods for systems integration.

Most importantly, a future cloud-based smart grid strategy using Web services would allow the seamless integration of Web-based, energy-aware smart homes to the grid. RESTful Web server may interconnect smart homes and the grid in a flexible, loose-coupled, interoperable manner. Since REST is a lightweight protocol, it could be used for efficient and scalable, near real-time interaction with hundreds or even millions of houses. As the Internet offers only best-effort services, real-time guarantees can not be provided. However, the current Internet infrastructure may support interactions through REST/HTTP close to real-time.

We note that WS-* could be well applied to achieve this interconnection [10]. Even though it constitutes a heavy protocol, it offers better security features, enables service contracts through WSDL and is more suitable for business applications. It is a matter of the electric utility to select the scheme it would adopt, or even if both should be offered. We argue that REST is becoming mature enough to be effectively utilized.

IV. Creating a Web-Based Energy-Aware Smart Home

A lightweight, Web-oriented application framework for smart homes, providing uniform access to heterogeneous embedded devices via standard HTTP calls, was developed in [6], [7]. Central principles of the modern Web architecture were used to integrate home devices to the Web, in order to build an interoperable smart home that supports multiple home residents concurrently. By using the Web as the application layer at the home environment, following REST principles, flexible applications on top of heterogeneous embedded devices can be built with a few lines of code, facilitating home automation.

In this work, this application framework was extended to support interaction with smart power outlets. Thus, by means of the power outlets, remote control of the electrical appliances of a house could be achieved, through the Web.

Ploggs[10] were utilized as the smart outlets of our experimental smart home. Ploggs are ZigBee-based devices that incorporate wireless transceivers, based on the IEEE802.15.4[11] standard, forming a wireless smart metering network inside the smart home. Since Ploggs are programmed with a firmware that can not be easily changed, they can not be enabled directly to the Web. Thus, they are enabled indirectly through the application framework, by means of Java drivers that allow the communication with them through a RESTful interface.

Each Plogg is associated with some specific electrical appliance, for monitoring its electricity footprint and control its operation. To derive the house's total electrical consumption, a Plogg equipped with an external current transformer for loads up to 100 A was attached to the mains meter of the house.

Figure 1 depicts the general architecture of the extended smart home application framework. It follows a layered model and is composed of three principal layers: *Device Layer*, which is responsible for the management and control of the smart power outlets, *Control Layer*, which is the central processing unit of the system and *Presentation Layer*, which represents the access point to the framework from the Web, enabling the uniform interaction with the electrical appliances of the smart home over a RESTful interface.

Each time a new smart outlet is discovered, a new thread dedicated to the corresponding electrical appliance is created. A *Resource Registry* maintains the services offered by the power outlet, as well as information how to properly invoke them. A *Request Queue* is attached to each thread, to enqueue concurrent requests to it. Requests are stored in a FIFO manner and are transmitted sequentially to the device. Whenever a transmission failure occurs, the failed request is retrieved from the request queue and retransmitted. In this way, transmission failures are masked effectively and reliability is assured.

Driver module holds the technology-specific drivers for enabling communication with the smart power outlets, by sending/receiving requests to/from them. A *Web Server* allows Web-based interaction between users and the home devices. A *REST Engine*, implemented by means of Restlet[12], ensures a RESTful system behaviour.

In Figure 2, a typical deployment of Ploggs inside an energy-aware smart home is shown. These smart meters use their multi-hop communication abilities to inform the residents about the electricity footprint of each appliance. In the figure, five hops are needed from the meter that acts as the base station to reach the meter which monitors and controls the

[10]Energy Optimizers Ltd has stopped producing Ploggs.
[11]http://www.ieee802.org/15/pub/TG4.html
[12]http://www.restlet.org/

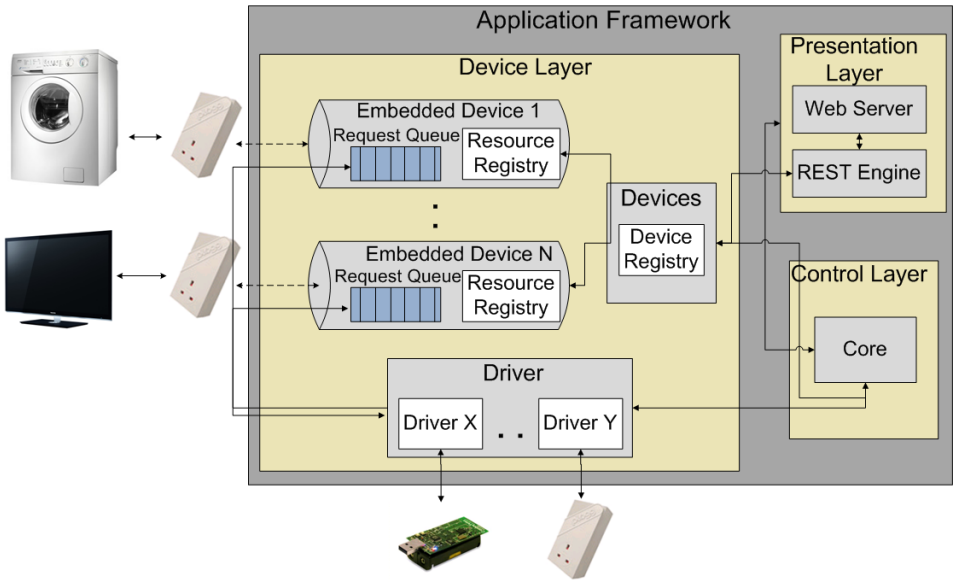


Fig. 1. The general architecture of the application framework including smart power outlets.

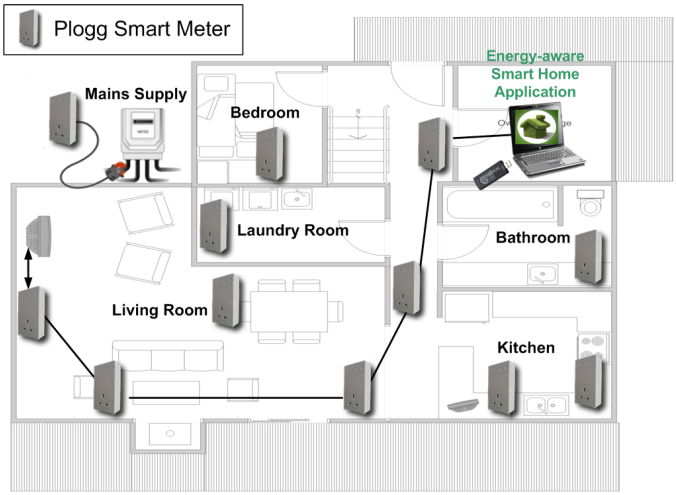


Fig. 2. A deployment of smart power outlets in an energy-aware smart home.

television. Plogg discovery is automatic, based on the ZigBee specifications. The application framework queries the wireless network of Ploggs for new devices in frequent intervals.

As shown in Figure 2, the framework has been installed on some computing device of the smart home and communicates with the Ploggs by means of a Telegesis USB stick. We must note that the framework could have been installed on the main smart meter of the house, which communicates directly with the electric utility (in a smart grid scenario) through AMI. Probably this could be the case in the near future, when main smart meters would be powerful enough to support also smart home applications and embedded Web servers.

Moreover, the functionality of the application framework was exposed as a RESTful interface and a client application was developed in JavaScript, using the Google Web Toolkit (GWT). This client application offers a Web-based, interactive graphical user interface (GUI), in order to help residents to visualize their energy consumption and fully manage their electrical appliances through the Web. Detailed, real-time consumption data from each electrical appliance and the aggregation of historical data about energy into graphs, facilitate the extraction of informed knowledge about the home's energy performance, encouraging the habitant towards a more rational use of electricity. Some psychological studies indicate that timely electrical consumption feedback is believed to assist in reducing electrical consumption by 5-20% [26], [27].

A typical snapshot of the client application can be observed in Figure 3, in which the electricity footprint of household electrical appliances is provided on a daily basis for the current week. Through detailed energy monitoring, electricity-wasting actions may be avoided and energy-inefficient devices can be managed better or be replaced.

Through the Web, each appliance can be individually controlled. For example, residents may switch off the television remotely from work, in case they forgot it on, when they hastily left the house. Residents can associate the energy consumption of their electrical appliances to the actual tariffs from their electric utility, translating kilowatt hours (kWh) into money. Based on these tariffs, the electricity cost consumed by each appliance is automatically calculated.

V. Connecting Smart Homes to the Smart Grid

Connecting energy-aware smart homes to the smart grid creates a new potential for saving energy and money. Household appliances account for 50-90% of the residential consumption and their rational management is crucial for any energy conservation initiative.

The Web-based interaction between smart homes and the smart grid could be facilitated by utilizing intermediary devices called *smart grid controllers*. Each controller would be responsible for some houses or neighborhoods. More powerful

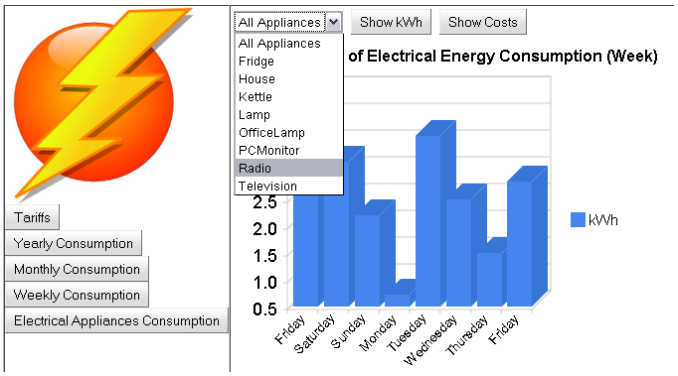


Fig. 3. Detailed electrical consumption of the electrical appliances of the energy-aware smart home.

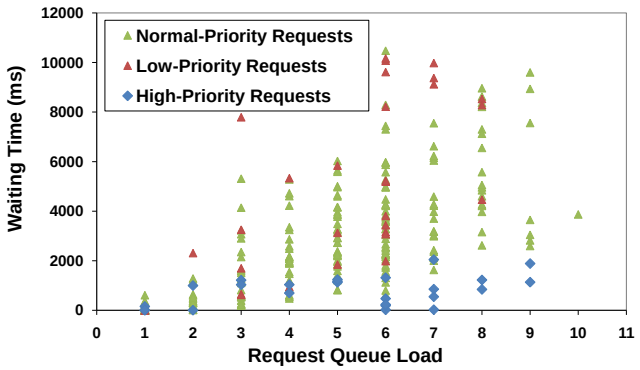


Fig. 5. The priority mechanism at the smart home application framework, showing the waiting times for requests with different priorities and heap load conditions.

grid controllers could manage larger areas such as villages and small towns. Smart grid controllers could maintain a hierarchical structure for fault tolerance and scalability. For example, controllers at higher levels (i.e., closer to the smart grid system) could be used for administering controllers at lower levels (i.e., closer to the customer premises). Low-level controllers would be able to communicate in a timely manner with each house while high-level controllers could interact with the main grid through a SCADA system. The overall system architecture is depicted in Figure 4.

We need to note that these smart grid controllers represent domestic controllers and should be separated from the main electricity infrastructure. It is also important to note that our focus is on integrating energy-aware smart homes to the smart grid, enabling flexible interaction patterns between them, and not on the operation of the smart grid and its controllers. The controllers could have specialized software for performing the grid's operations. An example implementation of a smart grid controller for emulating the behaviour of the smart grid in a scenario involving load shedding is described in Section VII-B.

As mentioned in Section III-A, the functionality of a smart home would be exposed as a Web API. Therefore, controllers can only interact with the house through the functions specified by this API. A simple Web API for Web-based smart homes, targeted to enable remote management and control by electric utilities is presented in Table I.

This API would allow the utility to get informed about the total electrical consumption of the house (*GET electricity*), ask the smart home to reduce its consumption because an outage is possible (*POST reduceconsumption*) or allow the house to increase its consumption when the total load is in safe margins (*POST increaseconsumption*). The targeted quantity of reduced consumption is specified by the parameter *reduction* and it is defined in Watts. Similarly, the maximum allowed quantity of (increased) consumption is specified by the parameter *maxincrease* and it is also defined in Watts.

The responses from these HTTP requests are in standardized formats. The POST requests for reducing/increasing consumption are satisfied through a plain-text response, indicating a ACK/NACK, while the GET request triggers a JSON response, including information such as consumption in kWh, instant consumption in Watt and a timestamp.

It is crucial for the healthy operation of the smart grid to be assured that all the HTTP calls to the energy-aware smart homes would be executed reliably and on time. Our smart home application framework guarantees the successful execution of all Web requests by using a request queue for each smart power outlet and a fast retransmission mechanism, triggered in case some transmission failure occurs in the smart home environment.

To ensure the urgent execution of the requests that are made by the smart grid, a priority mechanism may be easily included to the framework by transforming the request queues into priority heaps. Hence, requests coming from the smart grid could be labeled as "high-priority requests", obtaining a prioritized execution, regardless of the current load at the heaps of the smart power outlets. The waiting times for requests with different priorities in varied load conditions are displayed in Figure 5, for a typical operation of the priority mechanism in increased traffic. According to the figure, high-priority requests are executed first, with average waiting times less than a second, reaching two seconds only when the heap has a size equal or more than seven (i.e., seven or more requests waited already at the heap when the high-priority request arrived).

The functionalities of the application framework regarding mainly the fast retransmission mechanism and the support for prioritized requests are described in [28].

Respecting the privacy of the consumers, the smart home may act as a "black box", allowing the smart grid to make requests to assure its proper operation but, at the same time, leaving the full control and responsibility on how to satisfy these requests to the residents, aiming to maintain a high comfort level at a reasonable expense. In such a way, people's privacy and the healthy operation of the grid can be balanced.

The important question then becomes how to handle a request from the utility for reducing the overall electrical consumption. The most obvious approach would be to rely on the residents to assign priorities to their electrical devices.

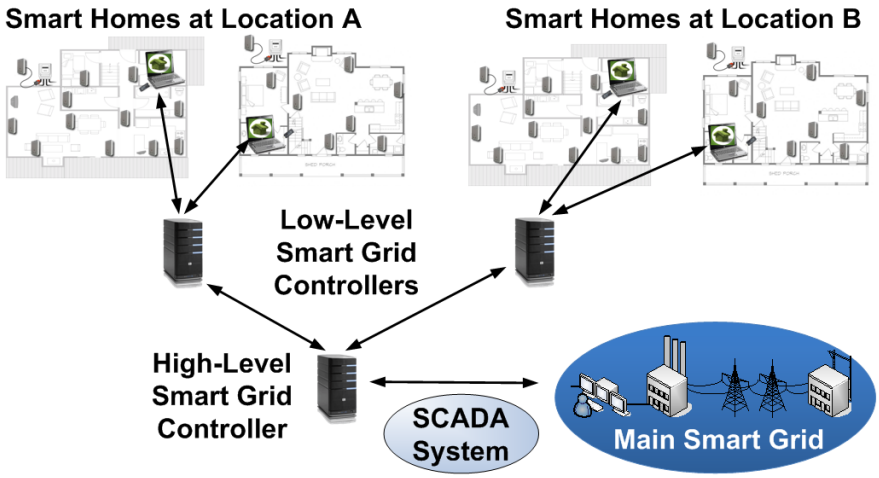


Fig. 4. System architecture for integrating energy-aware smart homes to the smart grid through the Web.

No.	Resource URL	REST Verb	MIME (Return) Type	Parameter (Type)
1	HouseName/electricity	GET	JSON	-
2	HouseName/reduceconsumption	POST	text/plain	reduction (Integer)
3	HouseName/increaseconsumption	POST	text/plain	maxincrease (Integer)

TABLE I
WEB API OF A WEB-BASED SMART HOME FOR INTERACTION WITH THE SMART GRID.

However, this may become inflexible and would complicate the whole procedure for the customers, who may have changing priorities and may not be willing to participate in such smart grid programs. An alternative approach would be to categorize devices according to their use patterns. Hence, we separate household devices into three broad categories. *Permanent devices*, which should never be turned off such as a fridge, *on-demand devices*, which are utilized by home residents spontaneously, in order to accomplish a momentary task such as a toaster and *schedulable devices*, which are devices that are supposed to accomplish some specific task, but their operation is not momentarily urgent and can be postponed for a future time such as a dishwasher.

We focus mainly on schedulable devices since their operation can be postponed for low-demand and respectively low-tariff periods of the day. These appliances could be immediately turned off, in case there is a prompt call from the utility to reduce urgently the domestic energy consumption.

Thus, customers are just required to identify which of their home devices are considered schedulable. Then, the application framework targets this device category to postpone use, in case there is a necessity. In the scenario when no schedulable devices consume energy and there is an urgent need for reduction, then on-demand devices would be selected.

It is trivial for the application framework to consider which devices are used on-demand, by observing their consumption patterns. Concerning permanent devices, the fridge is a special case as it could be turned off for some minutes without a problem. In addition, air conditioners and the electric heater could be special cases of on-demand devices whose operation may be postponed for some minutes.

In a complete smart grid scenario, different policies could have effect concerning the remote management of smart homes from the grid. Service-level agreements (SLA) could be used for assuring smooth supply of electricity and different customer pricing schemes could be applied. For example, ”gold customers” could pay a small extra fee for avoiding possible reduction of electrical supply in peak periods.

VI. ASSURING SECURITY IN SMART GRID-ENABLED SMART HOMES

The bidirectional Web-based communication between the smart grid and each smart home requires a trustworthy communication environment, where each party trusts the other communicating party, as well as the correctness, integrity and freshness of the received data. For instance, upon reception of pricing messages from the utility, the smart home application framework is assumed to take some actions (e.g., to charge an electric vehicle). If the pricing messages were changed on-route, the application will possibly take wrong actions. This could cause financial losses for the consumers, and even lead to a power outage (e.g., by sending fake low-cost tariffs during peak-periods).

Moreover, since the operation of smart appliances/smart power outlets must be managed by an energy-aware smart home application (e.g., to turn on energy-consuming electrical appliances during off-peak periods), it needs to be ensured that these smart devices are managed by the appropriate home application and not the application running on a neighboring home or by an attacker.

To provide secure communications for smart grid-enabled smart homes, the following basic security services need to be guaranteed:

- Authentication. Ensure the identity that another party claims to be. For instance, the smart home framework

needs to be sure that it gets pricing information only from the utility.

- Integrity. Ensure that stored or received data were not modified on-route. For instance, the utility needs to ensure the integrity of metering data received from smart homes.
- Authorization. This service allows one party to verify that another authenticated party has the right to do some actions or access some resources. For instance, the smart home framework needs to be sure that a tenant requiring access to some electrical appliance has the necessary rights to do that.
- Confidentiality. Ensure that data are illegible to non-authorized parties. For instance, energy consumption sent by the smart home to the utility needs to be encrypted, such as only the utility is able to access it.
- Non-repudiation. This service prevents one party to deny sending a message or doing some action. For example. assuming that the utility sends a pricing message with a low price *Y* but applies a high price *Z*, it could not deny the fact that the low-price message was really sent by it. Furthermore, the utility needs to be sure that a consumer could not contest a bill by denying the sending of the corresponding energy consumption measurement.
- Freshness. This service protects from replay attacks, where a valid message sent at time *t*, is also sent in the future by the attacker. For instance, an attacker could replay low-tariff pricing messages during peak-periods.

Unfortunately, the Web (HTTP) does not provide a trusted communication environment, since it offers poor built-in security mechanisms and, as a consequence, needs to rely on some extra mechanisms to provide stronger security. For instance, HTTP is target to several attacks that could harm the proposed Web-based architecture, such as:

- Man-In-The-Middle (MITM) attack: An attacker successfully impersonates each communicating party (e.g., smart home framework and smart grid controller) to the other party, and injects, modifies or drops packets. This attack could target functionalities of the smart grid such as demand response, load shedding and billing, causing financial losses to both parties, and even leading to power outages.
- Man-In-The-Browser (MITB) attack: This attach involves a malicious program (e.g., a Trojan) that infects a Web browser and takes control of data entered by the user or data retrieved from the Web server and displayed by the browser. This attack could harm the user by displaying false statistics about his consumption, and not the real statistics provided by the actual smart meter of the house.
- Denial of Service (DoS) Attack on HTTP: This kind of attack aims to make a service or resource (e.g., a Web server) unavailable. For instance, this attack could target making a real-time tariff service unavailable, thus forcing the DR functionality to stop. A similar attack could be launched against the smart home framework, making grid-related operations unavailable at the victim smart homes.

A. Securing the Smart Home-Smart Grid Interaction

Securing this interaction is mandatory for making the Web-based integration of energy-aware smart homes to the smart grid in a secure, feasible and acceptable way, both for the utility and the consumers. This interaction includes exposing the functionalities/services offered by smart homes (see Table I) as Web resources, accessed and utilized by the utility. In addition, the utility exposes a set of information (e.g., energy tariffs) as resources, accessed by smart homes via the Web.

By leveraging the existing Web security mechanisms, several security issues inherent to the smart home-smart grid interaction can be addressed. Using HTTP Secure (HTTPS) [29] is one way of protecting the communications between the two parties of our architecture. HTTPS is HTTP layered over the Transport Layer Security (TLS) protocol.

The TLS protocol [30] fits between the application and the transport layer (mainly TCP), and provides a plenty of security services over the Internet, such as cryptographic key-exchange and per-session key establishment, mutual authentication, integrity, confidentiality, non-repudiation and freshness. TLS allows the establishment of a secure communication channel between a TLS-enabled client and a TLS-enabled server, in which the server is first authenticated through a certificate (optionally also the client), and then secret session keys are established using some key management protocol. Once the communication channel is established, HTTP request/response messages may be securely sent between the smart home application framework and the (low-level) smart grid controller.

In the context of a smart grid scenario, since the smart home framework and the grid controller play both the role of HTTP client/server, mutual authentication through public-key certificate is mandatory for TLS use. Each smart home obtains a pair of certified private/public keys from a trusted Certificate Authority (CA), with a strong advice to keep the private key on a smart card, onto which all public-key cryptographic operations are done, in order to avoid the private key disclosure. For reinforcing smart home security, the equipment on which the smart home framework is running should not be visible or directly accessible from the outside. Furthermore, smart homes should be protected by a separate firewall. In this way, the firewall will be the first line of defense for the smart home, while the access to the resources provided by the application framework could be easily done through redirection rules implemented at the firewall.

Finally, the smart grid controllers also obtain certified private/public keys, and securely disseminate their certificates to the respective smart homes. Similar to the smart home case, in order to protect the non-critical smart grid controllers and also the critical SCADA systems, at least a firewall protection should be employed. Nevertheless, we suggest that SCADA systems are interconnected through their own dedicated network, connected to the remaining smart grid network through

gateways (integrated to or separated from the firewalls). In this way, DoS attacks may be prevented.

Since the smart home framework and the grid controllers are assumed to be hosted in powerful machines, the overheads induced by the TLS protocol (e.g., computation, transmission) could be affordable. However, if the home framework is running on the smart meter of the house, assuming that smart meters are currently resource-constrained devices, then the implementation of the whole TLS protocol could not be very efficient, because of increased memory demands and expensive TLS cryptographic operations, especially during the hand-shake phase for key-establishment.

B. Security Inside the Smart Home

Assuming all the in-home interactions are done through the Web, resource-constrained devices such as smart power outlets and smart appliances shall run an embedded Web server, in order to expose their capabilities as Web resources, accessible by the smart home application framework. The main challenge here is how to secure this Web-based interaction against attacks, in such a resource-constrained environment.

Generally, porting the IP stack on embedded devices is a recent achievement [31], [32]. Furthermore, Web-enabled sensor network systems in which sensor devices offer their functionalities as RESTful Web services have recently appeared [7], [22], [23]. Although these recent achievements, the WoT is not yet very common in embedded systems and constrained devices. Additionally, using standard security protocols, such as TLS or IPsec [33], does not yet constitute a popular practice in embedded computing, due to the heavy induced costs on the device operation. However, some efforts have been made recently to make these security protocols feasible for resource-constrained devices.

For instance, the Constrained Application Protocol (CoAP) [34] is a lightweight protocol for Web transfer in resource-constrained networks, sharing several similarities with HTTP. CoAP security is based on the use of Datagram TLS (DTLS) [35] or IPsec for securing the communications between the client (e.g., the smart home framework) and server (e.g., the home devices), at the transport and network layers respectively [36]. DTLS is a variant of TLS which operates over UPD (TLS operates over TCP), and which allows the establishment of a secure communication channel over which CoAP messages could transit. Due to the constrained environment of physical devices, only a subset of encryption and hash functions is assumed, in addition to the use of Elliptic Curve Cryptography (ECC) [37] instead of classical public-key cryptography (e.g., RSA, DSA), due to its low overhead in computation, storage and bandwidth.

While DTLS protects end-to-end communication (i.e., CoAP client and CoAP server), a hop-to-hop protection is also required inside the smart home, since requests and responses will probably travel through several hops to reach their destination. Since the majority of smart devices inside the home environment typically use IEEE 802.15.4 at the PHY and MAC layers, they could use the cryptographic facilities provided by the MAC layer, and in particular the AES algorithm to provide both encryption and data integrity. However, the key provisioning and key management is still an open issue, and needs to be determined at higher layers.

An additional requirement for inside-home security is the pairing. We need to guarantee that communications involve only authorized smart meters, smart power outlets and smart appliances belonging to a smart home A, and not those of a neighboring Smart Home B. In addition, smart devices of home A must be managed solely by the smart home application deployed at smart home A. Also, we need to guarantee that the smart home framework in home A controls only the home smart devices, and not those of a neighboring home B. The pairing will definitely require some level of user interaction, depending on the I/O and computational capabilities of the paired devices.

To sum up, the communication inside the smart home environment may be performed through CoAP and secured by DTLS. The CoAP request/response messages are exchanged in a way similar to HTTP, in addition to an application-level acknowledgment, since TCP is not used at the transport level for reliable data transfer. Unfortunately, as the smart power outlets included in our experimental smart home (Ploggs) were programmed with a closed firmware, they could not be enhanced with CoAP and its security mechanism. However, we strongly recommended that future real-life deployments of energy-aware smart homes need to consider protocols that provide secure communications, such as CoAP.

Finally, we need to note that we only covered some general aspects regarding assuring security in Web-based smart homes that are enabled to the smart grid. A full study, which is beyond the scope of the current paper, is required to study in detail all security issues that result in our proposed architecture.

VII. Applications of Smart Grid-Enabled Smart Homes

Here we outline some interesting applications that may be enabled when energy-aware smart homes are connected to the smart grid, through the Web. Our objective through these applications is to demonstrate the potential benefits of enabling smart homes to the grid, using the Web as the interconnection platform. Many more applications may occur in the future, exploiting more effectively the capabilities of the smart grid. This is a matter of further research.

A. Exploiting Demand Response

A significant feature of the grid is demand response. DR would assist in offering dynamic tariffs, according to supply conditions. Dynamic tariffs can be received almost in real-time, when utilities provide Web APIs to automatically disseminate them to the homes of the consumers. The DR capability would allow users to cut their energy bills by telling low priority devices to harness energy only when it is cheapest.

A DR-based task scheduler may also have a psychological factor. Energy-aware smart homes and the introduction of the smart grid in the residents' daily lives could engage them in

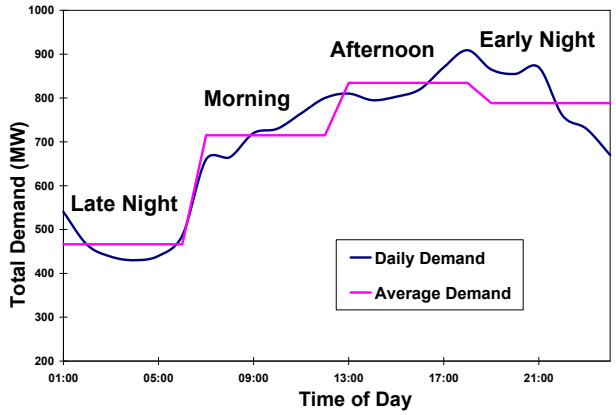


Fig. 6. Total electricity demand in a typical winter day.

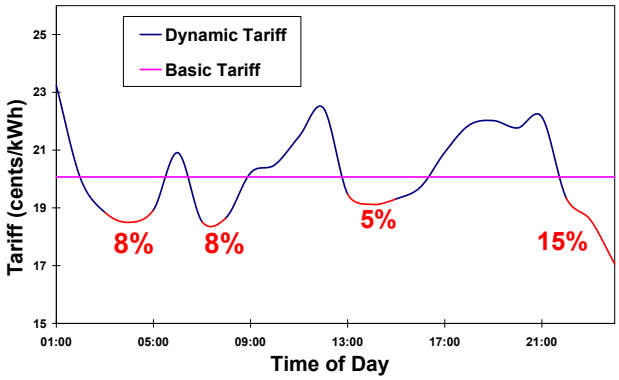


Fig. 7. Real-time tariffs based on electricity demand.

more sustainable lifestyles and energy-efficient practices [38]. The potential for saving energy and money can cultivate informed, actively involved, environmentally-aware consumers.

To demonstrate DR in energy-aware smart homes, a simulation of an electric utility offering timely tariffs was developed, interacting with a task scheduling mechanism, which was created to exploit the DR capability of electric utilities. The implementation efforts and a proof of concept evaluation procedure are described in the following subsections.

1) Implementation: We enhanced the client application, built on top of the application framework, with a task scheduling mechanism adapted to DR from electric utilities, following the physical mashup paradigm. The residents are able to program actions to be executed automatically in low-tariff hours. A low tariff is specified as a lower percentage from the basic tariff, which is offered by the utility. As an example, the resident can program the electric water heater to heat water for a shower, when the tariff is 10% less than normally.

The residents are able to further adjust the task scheduling procedure, according to their own preferences. They can define a maximum amount of waiting time, in case tariff does not fall below the specified limit in that time window. In this case, the task can start right after. The residents can also specify the execution of a task to be performed in the morning, afternoon or night. Finally, they can set the duration of each task, forcing the application to switch the corresponding electrical appliance off, as soon as the task completes.

We developed a Web server that simulates a (low-level) smart grid controller, supporting DR functionality for the Electricity Authority of Cyprus (EAC), which is the only utility in Cyprus. A RESTful Web service informs customers about the utility's current tariffs using RSS Web feeds.

These tariffs, although simulated, aim to reflect the actual energy loads and demands in our country. Figure 6 presents the total electricity demand in Cyprus, at a typical winter day. We assume that the power plants of EAC are able to operate in four different modes for generating electric power. These modes reflect the average electricity demands when dividing a winter day into morning, afternoon, early night and late night. These four modes can be observed in the figure.

To produce dynamic tariffs, correlated to electricity demand patterns, we used the simple equation shown in (1):

$$Tariff = \alpha \cdot BasicTariff \cdot (\frac{InstantDemand}{AverageDemand}) \quad (1)$$

where α is a coefficient used to weight the prices according to differences in demands. Using this equation with $\alpha = 1$, dynamic tariffs are produced that give incentives to consumers to utilize their electrical appliances not in peak hours. These tariffs fluctuate around the basic home tariff, as shown in Figure 7.

2) Evaluation: To test the performance of our system, we considered a typical real-life scenario. Most washing machines allow a user to define a preferred operation mode and start the washing in a future time. We programmed such a washing machine through the task scheduling mechanism, to start the washing when the tariff from the electric utility is 5% less than its normal price. According to Figure 7, this would happen at 4:00, 7:00, 14:00 and 23:00 in a typical winter day. We also set some parameters such as the duration of the task to be one hour and 30 minutes and the maximum waiting time to be eight hours. We measured the execution times of this task, placing the washing machine and its corresponding Plogg, in different hops from the application framework (base station).

Figure 8 illustrates the results of this experiment. In all five multi-hop scenarios, we created the task at 12:00, it started executing exactly at 14:00 and it finished execution at 15:30. Less than two seconds are needed, from the time the application is informed about the tariff change, until the washing machine starts working, even in five-hop distance. Switching off the device needs a bit longer, approximately 1-3 seconds. This difference is due to the specific operation of the Ploggs' firmware.

In this experiment, we utilized Ploggs with firmware version 2.00. Comparing with the same experiment, performed in [1], where we employed Ploggs with firmware version 1.67, response time in this experiment is reduced significantly, especially for switching off some electrical appliance. This

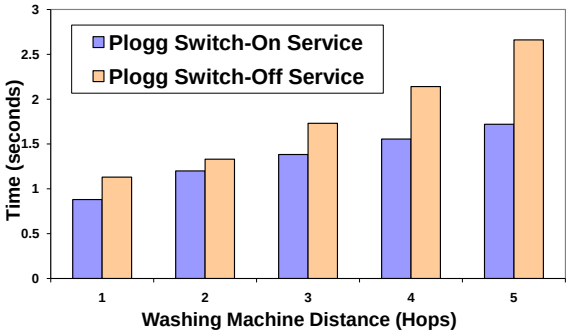


Fig. 8. Task scheduling performance.

difference occurs because data is transmitted in binary instead of ASCII form in the new version of the firmware.

Since the task scheduling mechanism will operate for control scenarios with low workload, the results in regard to task execution times are considered satisfactory.

A small-scale, telephone-based survey was performed in [1] to identify schedulable electrical appliances and their usage patterns by housewives in Cyprus. After performing some basic calculations for money savings, taking into account the average energy consumption of these appliances and the typical home tariff offered by EAC (20,07 cent/kWh), it was considered that monthly savings can be summed around €6 in 10% tariff reduction and up to €19 in case of 30% reduction. Considering the fact that the average monthly cost for electricity in houses around Cyprus is €175, possible saving of €19 gives 10,85% reduction at the bill of a typical home, when our task scheduling mechanism is applied for all schedulable electrical devices. This is definitely a significant saving amount.

Finally, since many schedulable devices are used only sometimes per month or some hours per day, it is not necessary to buy a smart power outlet for each different device. By purchasing 5-10 power outlets, it may be enough to cover the daily needs of the family and schedule the operations of the schedulable devices through the task scheduling mechanism in low-tariff hours of the day. In this way, this application would constitute a low-cost investment.

B. Load Shedding

Load shedding [39] is an action taken to prevent frequency abnormal operation and is the last resort to maintain frequency stability (i.e., it retains frequency within the operational and statutory limits), in case of contingency scenarios or autonomous-islanded operation. Such a scenario could include the non-scheduled outage of a generation unit or a main transformer. In this case, the non-served load previously served by the generator that currently experiences an outage will be allocated to other online units, given that their loading can be extended, remaining within the limits indicated by the manufacturer.

However, there are two unfortunate cases that might happen. First, the online generators may not be able to accommodate/undertake the extra load because they are already highly loaded. Secondly, online generation units might be able to accommodate the extra load (because they are not fully loaded) but, depending on the magnitude of the loss of generation, the response rate of their prime movers will not be in position to accommodate such a sudden increase in load, within the time slot indicated in transmission system regulations.

In such cases, in order to avoid an under-frequency abnormal operation of the power system, the operator will be forced to apply a low frequency demand control action, removing intentionally loads from service in order to prevent the total collapse of the system due to cascading events. This procedure is the definition of load shedding and it lasts until the frequency magnitude recovers at the desired levels, when the rest of the online units are able to fully compensate the non-served load.

Load shedding is a procedure undertaken from the electric utility or the power system operator in a centralized way. The approach followed in this work is to exploit the proposed architecture and the functionalities provided by the Web APIs of Web-enabled energy-aware smart homes (see Table I), to achieve selective load shedding that can be performed in a distributed manner, providing to the grid the capability of directly controlling domestic loads. This is a major characteristic of the future grid not yet standardized and its implementation has not yet been decided.

1) Implementation: The architecture described in Section V illustrates the path of the control messages and how the control objective will be achieved, i.e., the frequency stability of the power system. Concretely, the electric utility monitors the frequency of the power system almost in real time. In the case of a critical variation based on the frequency value and its rate of change, control messages are issued from the utility control center to the high-level smart grid controllers, which order the low-level grid controllers to reduce the power consumption of the area that they are responsible for (i.e., a neighborhood). Then, the low-level controllers use the Web API of smart homes in a best-effort manner, asking the houses to reduce their consumption based on their current electricity demand and the condition of the grid.

Harnessing a smart home for these purposes has not yet been thoroughly explored by researchers and it is expected to revolutionize the future grid's structure and control. To demonstrate this potential application of the smart grid, an emulated scenario of selective load shedding has been implemented, employing three residential units in which our Web-based application framework for smart homes (see Section IV) has been deployed, along with 4-5 Ploggs at each house, associated with various electrical appliances of the house, mostly schedulable devices. The experimental setup is displayed in Figure 9.

In the scenario under consideration, it is assumed that the residential untis are located in an islanded power system. A set of generators are committed and serve the total load of the system. Without loss of generality, the system is modeled with

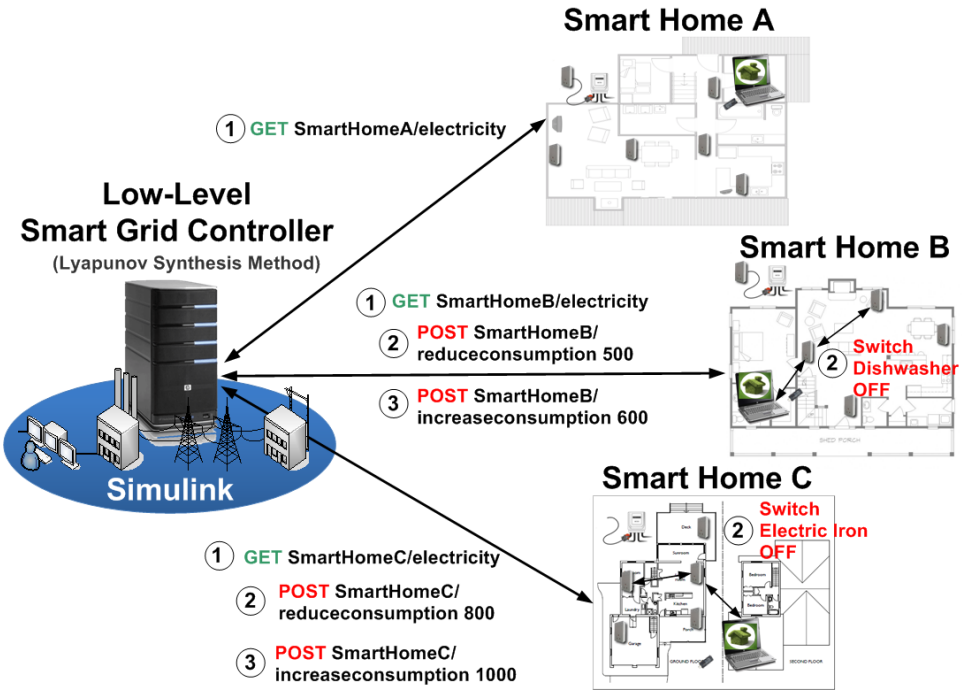


Fig. 9. The experimental setup used in the scenario of load shedding.

a generator, a transmission line and a variable load. The aim is to monitor the frequency stability of the system.

Virtual Phasor Measurement Units (PMUs) [40] have been used, which are capable of measuring both the frequency and its first derivative online in real-time.

System parameters were identified by applying the Lyapunov Synthesis Method [41], based on a simple linear second-order system frequency response (SFR) model [42]. Lyapunov Synthesis Method enables to identify the parameters of the plant by employing a suitable Lyapunov function, in terms of state variables and time, forcing this function to be at least negative semi-definite in order to obtain the desirable stability.

Further, a (low-level) smart grid controller was simulated on Simulink[13], and its main task was to maintain the stability of the system by determining the optimal (per unit) amount of electric load that should be shed to achieve frequency stability. The basic parameters of the simulation are listed in Table II.

Parameter	Value
Simulator Time Step	0.27424 msec
Total Simulation time	150 sec
Sampling Rate for Domestic Consumption	350 msec
Total Number of HTTP requests	428

TABLE II
PARAMETERS AT THE SIMULATION OF A SMART GRID CONTROLLER.

At the scenario under discussion, the grid experiences a sudden, non-scheduled increase in load. This increase has been modeled as a step function change in demand. As a result, the frequency of the power system starts to decline continuously. Three different cases are considered:

[13]http://www.mathworks.com/products/simulink/

I. No load shedding takes place.
II. Load shedding is performed following the conventional practice that is applied by the vast majority of power system operators worldwide.
III. An intelligent selective/soft load shedding is performed, exploiting the proposed system architecture.

In the first case, the frequency decline is just monitored without taking any action. In the second case, circuit breakers are activated, shedding load based on an approximate rule of thumb, which indicates that the connected load magnitude should be decreased linearly in relation to the frequency decline. For example, if the frequency decreases by 1%, then load magnitude would be decreased by 2% [43]. Thus, when frequency declines 1% becoming 49.5 Hz, 2% of the load is shed. At the frequency level of 49Hz, a 4% of the initial load is shed and so on.

Finally, in the third case, the proposed algorithm was applied to determine when load shedding should be performed and the amount of load that should be shed. The current consumption is monitored in near real-time by the operator of the grid through the relevant smart grid controller, which is responsible to aggregate the consumption of the houses it controls. The grid controller obtains the current consumption by issuing an HTTP *GET electricity* command to each smart home, in regular time intervals of 100 ms. The total consumption, input to the simulated power system as a load, has been derived from the current consumption of the three domestic units, multiplied by a factor of 10,000. In this way, the emulation scenario has been scaled to an islanded grid.

Based on the total consumption, the current value of the grid's electrical frequency and its rate of change, the operator

asks from the grid controller to shed the required amount of load. Then, the controller asks from each house to shed a specific amount of load by issuing an HTTP *POST reduceconsumption* command. The targeted amount of load to be shed at each home depends on its current consumption. After, the application framework decides which device(s) should be switched off based on the policy employed (e.g., according to device category and its current consumption). In this implementation, schedulable devices whose consumptions were closest to the targeted reduction were preferred to be switched off.

This is an iterative conversational procedure that takes place until the targeted total amount of load is shed. As soon as the command is successfully executed (i.e., an ACK has been received and the corresponding devices have been switched off), the simulator is fed with the scaled amount of shed load.

Finally, when the grid is in a "safe" condition again (i.e., the frequency has recovered in normal/desired limits), the controller starts progressively to issue HTTP *POST increaseconsumption* commands to the smart homes, to gradually add the load that has been previously curtailed, allowing a maximum restoration in load at each house. Hence, the application framework switches on the schedulable devices that had been previously switched off, allowing them to finish their task.

The three different phases of the proposed algorithm can be observed in Figure 9. In this specific scenario, the grid controller monitors the electrical consumption of all the three houses and, at some time, needs to perform load shedding in order to maintain frequency stability of the system. Hence, it decides to issue HTTP POST commands to smart homes B and C to reduce their instant consumption by 500 and 800 Watts respectively. Smart Home B responds to this command by switching off the dishwasher and smart home C by switching off the electric iron. These are schedulable devices whose operation may be postponed for a future time. Then, when the system is stabilized again, the grid controller issues another POST command to smart homes B and C, allowing them to increase their consumption by 600 and 1000 Watts respectively.

2) Evaluation: Figure 10 depicts the performance of the three different schemes in regard to the time response of the frequency variation. When no load shedding is applied, the frequency declines exceeding the permitted limits (of +/- 3 Hz), causing under-frequency abnormal operation. In this case, the power grid will experience instability and cascading events will possibly follow, causing a total blackout. Furthermore, devices such as induction motors can be damaged or even burned out at low/under-frequency operation.

In the second case, when conventional practices regarding load shedding are performed, the frequency exceeds the desired levels for four seconds and needs 35 seconds in total in order to "absorb" the disturbance. After this critical time, frequency oscillations still exist, being reduced with a small step. In this case, a set of customers would experience a total outage causing major discomfort to them.

Finally, when our intelligent selective load shedding algorithm is applied, the frequency remains at the desired levels during the whole emulation time. It recovers fast in the first 35 seconds of the emulation and totally absorbs the disturbance after 50 seconds.

Moreover, because the intelligent algorithm is sensitive even at small frequency variations and acts proactively, based on the frequency rate of change, it finally achieves a minimum total amount of load to be shed. This is in contrast to the conventional practice in which a larger amount of load needs to be shed in order to achieve the same control objective, i.e., maintaining the system frequency in the desired levels.

In addition, load restoration can be applied smoothly as soon as the frequency experiences an upward trend, exceeding a certain threshold with certain momentum (i.e., certain rise rate given by the positive time derivative of the frequency). In this way, the maximum load is restored in minimal time.

Our findings indicate that this application contributes to a more robust power grid that controls frequency oscillations in a more effective and efficient way, preventing power system instabilities and total outages of utility services. At the same time, it minimizes unexpected outages and customer discomfort by minimizing the load to be shed and by restoring it faster than the conventional practices do.

VIII. Other Potential Applications

In the following subsections, some other potential applications that could be enabled using the proposed architecture are briefly described. These applications include peak leveling, fault tolerance, billing and a distributed market.

A. Peak Leveling/Shaving

Peak leveling/shaving is a process that aims to eliminate the demand in peak hours and to shift it in non-peak demand periods. In this way, the demand curve is leveled, providing maximum exploitation of the current utility infrastructure while the need for excessive spinning reserves is reduced.

For example, in the case of an expected peak hour (e.g., a world cup final), expensive generators are supposed to be employed that are able to switch on fast and follow the load changes quickly. However, this is very costly for the utility, both in operations and expenses.

The proposed end-to-end smart grid architecture will mitigate this situation by applying peak shaving, shifting the low-priority and reschedulable domestic loads to non-peak demand hours, in the way of distributing evenly the produced energy throughout the day.

The approach described in Section VII-B for load shedding could also be employed for peak shaving. In addition, demand response programs could be utilized (see Section VII-A), using special tariffs to influence consumer behavior.

B. Fault Tolerance

An important characteristic of the smart grid is the timely detection and localization of faults. The proposed architecture has been designed in order to facilitate this task through the hierarchical, distributed structure of the controllers.

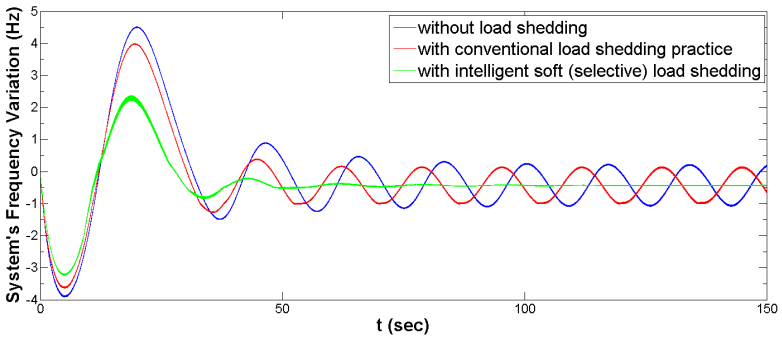


Fig. 10. Load shedding using different practices.

The established end-to-end connections between the smart grid controller and each of the smart home application frameworks provide in (near) real-time the customers' consumption. In addition, the HTTP POST requests issued by the controller, destined to the customer premises, reveal information about the status of the house, i.e., if a command has been successfully executed concerning shedding some amount of load.

In this way, the controllers can identify any abnormal behavior of the customers' consumption patterns and find out quickly which customers experience malfunctions or outages. Further, if the smart home framework does not respond to the issued commands multiple times (the customer-specific commands are re-sent until an ACK is received), this implies that the specific user experiences a failure or violates the SLA contracted with the operator of the grid.

In such cases, the smart grid controller may send some additional requests to the home framework in order to get informed about the status of the house and increase its awareness about the situation. Hence, the customers who experience unavailability of some or all services will be automatically detected and will be associated with the faulty feeders. Then, the utility crews will locate the fault and rush in immediately, reducing the duration of service unavailability (outage).

Their fast response would increase the reliability index of the company and generally improve its severity-based indices. Indices including expected unserved demand per year and expected unserved energy per year would be improved because of the faster faults restoration and the early actions taken as a result of the enhanced situational awareness of the grid.

Of course, there exist situations that could complicate fault localization. For example, a massive DoS attack on the customer premises and the smart grid controllers could hinder any efforts for fault identification and solving. Security countermeasures such as firewalls (see Section VI-A) could be employed to prevent such attacks. Nonetheless, since the proposed architecture relies on the Internet/Web, fault tolerance can not be guaranteed in general.

C. Billing

Billing is a major source of expenses by electric utilities since dedicated personnel must be employed for reading the home meters manually, requiring the physical presence of an employee at each house. The smart grid can provide long-term savings to the electric utility by providing automated meter reading via frequent interactions with energy-aware smart homes, through the Web.

Considering the Web API offered by the smart home as provided in Table I, it could be enough for the utility to issue GET requests in frequent intervals (in magnitude of seconds/minutes), to get informed about the instant domestic electrical consumption. However, this pull-based technique would not scale for millions of houses. Therefore, a Web-based push technique, such as the RESTful Message System (RMS) [44], would be more appropriate as it is based on a publish/subscribe model.

Availability of timely billing information would help residents to become more aware about their electricity footprint, giving them financial incentives to reduce their consumption.

D. A Market for Generation/Consumption of Electricity

The use of domestic renewable electricity generators could become a trend in the future. These generators would form decentralized energetic islands or *microgrids*. A microgrid consists of many small, distributed energy resources, located near each other in the low-voltage distribution system, connected to each other through some network.

Through this decentralized network architecture, smart homes may be capable of trading energy for money by means of market agents. These agents would represent electric devices, either generators (e.g., photovoltaics, wind turbines), or loads (e.g., television, fridge).

Hence, real-time auctions about electricity could be developed among energy-aware smart homes and the smart grid. According to the current generation and demand, smart homes would sell/buy electrical energy in competing prices.

The proposed architecture defined in Section V could be employed for enabling a real-time market for energy. The Web could become the platform for handling the message exchange between houses and the grid. More specifically, the Web API offered by smart homes, as shown in Table I, could be extended to support also market-related functionalities. For example, assuming that a smart home needs energy in some sunny day, its home market agent could query another smart home that generates electricity from photovoltaics, asking

about the price of produced energy. This could be achieved by issuing a *GET energyprice* request. In case a deal is accomplished, this home agent could issue a *POST energy-demand* command declaring the needed amount of power. Of course, this is only a simple example, since a complete solution would require money transactions between the two parties, intelligent algorithms for automating the purchase of energy in competing prices and an infrastructure for offering the distributed generated electricity to other houses.

IX. Conclusion and Future Work

In this paper, we examined the interconnection possibilities of energy-aware smart homes with the forth-coming smart grid of electricity, using the Web as the application protocol. A Web-based architecture is suggested for bridging the gap between home environments and the smart electricity grid. The Web is considered suitable for this interconnection, as it can address the heterogeneity of smart home technologies and smart grid components. Hence, advanced flexibility and interoperability is achieved by using the Web as the actual platform for the smart home-smart grid communication.

A Web-based application framework for smart homes was adapted to support smart power outlets, for monitoring and controlling the electrical appliances of the smart home. Using these smart outlets, an energy-aware smart home was deployed, allowing the remote management of home devices almost in real-time, through the Web. Reliability and prioritized requests coming from the smart grid were supported by employing request queues and priority heaps, for better handling the in-house communication with the power outlets. Moreover, a smart grid scenario was emulated, allowing the near real-time Web-based communication of the grid with smart homes by means of smart grid controllers.

The practice of connecting smart homes to the grid through the Web allows the flexible creation of various energy-related applications that target the healthy, efficient and effective operation of the grid. We demonstrated two such applications: the demand response program of the grid, for scheduling electricity-related tasks for the future, when the price for electrical energy will be cheaper; and load shedding, as a technique to reduce total consumption when danger for outages exists.

Of course, there still exist many issues that need to be addressed, before using massively the Web for this purpose. Technical issues include Web applications for smart homes that operate behind firewalls and home IP addresses that are rapidly changing. Smart home applications must conform to the Web API specifications, defined by the electric utilities. Reliability, especially inside the HAN, must be ensured. Our deployment using Ploggs showed that smart home hardware technologies are not yet mature to be used in such large-scale scenarios. We experienced regularly temporary device failures and service unavailability. These reliability issues would not be acceptable in a smart grid integration of the smart home.

Respecting the privacy of the customers is a crucial parameter for enabling a smart grid that is fully synchronized with energy-aware smart homes. Customer privacy in his home environment is reinforced through our proposed architecture by offering only a Web API to the grid, restricting the control of smart homes to the functionalities offered by this API. Considering our Web-based architecture, solely the smart home application framework can take the appropriate actions to handle grid commands, maintaining the comfort of the residents. The residents just need to specify which of their electrical appliances are considered schedulable, permitting the postponement of their operation for future time.

Security is another important factor that must be carefully considered. A satisfactory trustworthy Web-based communication environment between the smart grid and the smart homes can be assured by employing HTTPS. However, inside the home environment in which resource-constrained devices are involved, compromises need to be made to balance between acceptable performance and security. DTLS is a candidate technology for ensuring in-home security.

Our current proposal of a Web-based, RESTful architecture ensures only syntactic interoperability between heterogeneous smart home technologies and the smart grid. However, the semantic meaning of the exchanged messages is not effectively addressed by the current system. To obtain also semantic interoperability, semantic Web services may be the solution. Semantic Web services [45] are built around universal standards for the interchange of semantic data between machines. They can address issues such as advanced interoperability, automatic reasoning and knowledge inference on the smart home-smart grid ecosystem.

For future work, we plan to examine other potential applications that could be enabled when energy-aware smart homes connect to the smart grid. We also wish to validate the findings of our current applications through more detailed evaluations, including a larger number of houses and smart power outlets.

Finally, in collaboration with EAC, we plan to emulate the operation of the smart grid in a small neighborhood of energy-aware smart homes, involving also the local residents, to consider the feasibility of our proposed approach in real life and measure the actual impact of the system on the grid. Feedback from residents would be important to assess if the proposed architecture and technologies are suitable to them.

Few people predicted the revolutionary advancements the Internet has brought to the world. Even fewer have predicted that the Web would affect so many aspects of our lives. Energy-aware smart homes and smart grid controllers may represent the extension of this trend to power consumption.

X. Acknowledgments

This work was supported in part by the Electricity Authority of Cyprus (EAC). We would like to thank Mr Antonis Valanides, the director of the IT Department of EAC, for his valuable remarks and advice.

References

[1] A. Kamilaris and A. Pitsillides. Exploiting Demand Response in Web-based Energy-aware Smart Homes. In *the First International Conference on Smart Grids, Green Communications and IT Energy-aware Technologies (Energy 2011)*, Venice, Italy, May 2011.

[2] International Energy Agency. World Energy Outlook 2007. 2007.

[3] Europa Press Release. Communication from the European Commission. Energy Efficiency: Delivering the 20% target. November 2008.

[4] E. Wilde. Putting things to REST. Technical Report UCB iSchool Report 2007-015, School of Information, UC Berkeley, 2007.

[5] Dominique Guinard, Vlad Trifa, and Erik Wilde. Architecting a Mashable Open World Wide Web of Things. Technical Report No. 663, Dept. of Computer Science, ETH Zurich, February 2010.

[6] A. Kamilaris, V. Trifa, and A. Pitsillides. The Smart Home meets the Web of Things. *International Journal of Ad Hoc and Ubiquitous Computing (IJAHUC), Special issue on The Smart Digital Home*, 7(3):145–154, 2011.

[7] A. Kamilaris, V. Trifa, and A. Pitsillides. HomeWeb: An Application Framework for Web-based Smart Homes. In *Proceedings of the 18th International Conference on Telecommunications (ICT 2011)*, Ayia Napa, Cyprus, May 2011.

[8] Himanshu Khurana, Mark Hadley, Ning Lu, and Deborah A. Frincke. Smart-Grid Security Issues. *IEEE Security and Privacy*, pages 81–85, 2010.

[9] G. Alonso, F. Casati, H. Kuno, and V. Machiraju. *Web Services: Concepts, Architectures*. Springer, 2004.

[10] C. Warmer, K. Kok, S. Karnouskos, A. Weidlich, D. Nestle, P. Selzam, J. Ringelstein, A. Dimeas, and S. Drenkard. Web services for integration of smart houses in the smart grid. In *Proceedings of the Grid-Interop Conference*, Denver, USA, November 2009.

[11] Stamatis Karnouskos, Orestis Terzidis, and Panagiotis Karnouskos. An Advanced Metering Infrastructure for Future Energy Networks. In *IFIP/IEEE first International Conference on New Technologies, Mobility and Security (NTMS 2007), Paris, France*, pages 597–606, May 2007.

[12] R. T. Fielding. *Architectural Styles and the Design of Network-based Software Architectures*. PhD thesis, University of California, Irvine, California, 2000.

[13] D. Guinard, I. Ion, and S. Mayer. In Search of an Internet of Things Service Architecture: REST or WS-*? A Developers Perspective. In *Proceedings of the 8th International ICST Conference on Mobile and Ubiquitous Systems (Mobiquitous)*, Copenhagen, Denmark, December 2011.

[14] A. Marchiori and Q. Han. Using Circuit-Level Power Measurements in Household Energy Management Systems. In *First ACM Workshop On Embedded Sensing Systems For Energy-Efficiency In Buildings (BuildSys)*, pages 7–12, Berkeley, California, November 2009.

[15] Y. Kim, T. Schmid, Z. M. Charbiwala, and M. B. Srivastava. ViridiScope: design and implementation of a fine grained power monitoring system for homes. In *Proceedings of the 11th International Conference on Ubiquitous computing (Ubicomp)*, pages 245–254, Orlando, Florida, USA, 2009.

[16] X. Jiang, S. Dawson-Haggerty, P. Dutta, and D. Culler. Design and implementation of a high-fidelity AC metering network. In *Proceedings of the International Conference on Information Processing in Sensor Networks (IPSN)*, pages 253–264, Washington, DC, USA, April 2009.

[17] M. Jahn, M. Jentsch, C. R. Prause, F. Pramudianto, A. Al-Akkad, and R. Reiners. The Energy Aware Smart Home. In *Proceedings of the 5th International Conference on Future Information Technology (FutureTech)*, pages 1–8, Busan, Korea, May 2010.

[18] G. Xiong, C. Chen, S. Kishore, and A. Yener. Smart (in-home) power scheduling for demand response on the smart grid. In *IEEE PES Innovative Smart Grid Technologies (ISGT)*, pages 1–7, Manchester, UK, January 2011.

[19] M. Erol-Kantarci and H.T. Mouftah. Wireless Sensor Networks for Cost-Efficient Residential Energy Management in the Smart Grid. *IEEE Transactions on Smart Grid*, 2(2), June 2011.

[20] N. B. Priyantha, A. Kansal, M. Goraczko, and F. Zhao. Tiny web services: design and implementation of interoperable and evolvable sensor networks. In *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems (SenSys)*, pages 253–266, Raleigh, USA, 2008.

[21] C. Groba and S. Clarke. Web services on embedded systems - a performance study. In *IEEE Workshop on the Web of Things, in Proc. of Pervasive Conference*, pages 726–731, Mannheim, Germany, March 2010.

[22] L. Schor, P. Sommer, and R. Wattenhofer. Towards a Zero-Configuration Wireless Sensor Network Architecture for Smart Buildings. In *First ACM Workshop On Embedded Sensing Systems For Energy-Efficiency In Buildings (BuildSys)*, Berkeley, California, November 2009.

[23] D. Yazar and A. Dunkels. Efficient Application Integration in IP-based Sensor Networks. In *First ACM Workshop On Embedded Sensing Systems For Energy-Efficiency In Buildings (BuildSys)*, Berkeley, California, November 2009.

[24] D. Guinard and V. Trifa. Towards the Web of Things: Web Mashups for Embedded Devices. In *Workshop on Mashups, Enterprise Mashups and Lightweight Composition on the Web, in Proc. of WWW Conference*, Madrid, Spain, April 2009.

[25] The White House. Office of Science and Technology Policy. Empowering Customers With a Green Button. Online at: http://www.whitehouse.gov/blog/2011/11/21/empowering-customers-green-button.

[26] S. Darby. The effectiveness of feedback on energy consumption A review for defra of the literature on metering, billing and direct displays. *Environmental Change Institute, University of Oxford*, 2006.

[27] G. Wood and M. Newborough. Dynamic energy-consumption indicators for domestic appliances: environment, behaviour and design. *Energy and Buildings*, 35(8):821–841, 2003.

[28] Andreas Kamilaris and Andreas Pitsillides. A Restful Architecture for Web-based Smart Homes using Request Queues. Technical Report No. TR-12-5, Dept. of Computer Science, University of Cyprus, June 2012. Online at: http://www.cs.ucy.ac.cy/ResearchLabs/netrl/papers/files/kamilaris-TR-12-5.pdf.

[29] E. Rescorla. HTTP over TLS. RFC 2818.

[30] T. Dierks and E. Rescorla. The Transport Layer Security (TLS) protocol Version 1.2. RFC 5246.

[31] A. Dunkels, T. Voigt, and J. Alonso. Making TCP/IP Viable for Wireless Sensor Networks. In *Proceedings of the first European Workshop for Wireless Sensor Networks (EWSN)*, Berlin, Germany, January 2004.

[32] J. W. Hui and D. E. Culler. IP is dead, long live IP for wireless sensor networks. In *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems (SenSys)*, pages 15–28, Raleigh, USA, 2008.

[33] K. Kent and K. Seo. Security Architecture for the Internet Protocol. RFC 4301.

[34] Zach Shelby, Klaus Hartke, Carsten Bormann, and Brian Frank. Constrained Application Protocol (CoAP), March 2012. IETF Draft, draft-ietf-core-coap-09.

[35] E. Rescorla and N. Modadug. Datagram Transport Layer Security. RFC 4347.

[36] J. Arko and A. Keranen. CoAP Security Architecture, July 2011. IETF Draft, draft-arkko-core-security-arch-00.

[37] Darrel Hankerson, Alfred J. Menezes, and Scott Vanstone. *Guide to Elliptic Curve Cryptography*. Springer-Verlag, Secaucus, USA, 2003.

[38] W. F. Van Raaij and T. M. M. Verhallen. A behavioral model of residential energy use. *Journal of Economic Psychology*, 3(1):39–63, 1983.

[39] D. Craciun, S. Ichim, and Y. Besanger. A new soft load shedding: Power system stability with contribution from consumers. In *IEEE PES PowerTech*, pages 1–6, Bucharest, Romania, July 2009.

[40] A.G. Phadke and B. Kasztenny. Synchronized Phasor and Frequency Measurement Under Transient Conditions. *IEEE Transactions on Power Delivery*, 24(1):89–95, January 2009.

[41] J. Farrell and M. Polycarpou. *Adaptive approximation based control: unifying neural, fuzzy and traditional adaptive approximation approaches*. Adaptive and learning systems for signal processing, communications, and control. Wiley-Interscience, 2006.

[42] P.M. Anderson and M. Mirheydar. A low-order system frequency response model. *IEEE Transactions on Power Systems*, 5(3):720–729, aug 1990.

[43] P.M. Anderson, A.A.A. Fouad, Institute of Electrical, and Electronics Engineers. *Power system control and stability*. IEEE Press power engineering. 2003.

[44] V. Trifa, D. Guinard, V. Davidovski, A. Kamilaris, and I. Delchev. Web messaging for open and scalable distributed sensing applications. In *Proceedings of the 10th International Conference on Web engineering (ICWE)*, pages 129–143, Vienna, Austria, July 2010.

[45] Jorge Cardoso and Amit P. Sheth. *Semantic Web Services, Processes and Applications*, volume 3 of *Semantic Web And Beyond Computing for Human Experience*. Springer, 2006.

[46] J. Franks, P. Hallam-Baker, J. Hostetler, S. Lawrence, P. Leach, A. Luotonen, and L. Stewart. HTTP Authentication: Basic and Digest Access Authentication. RFC 2617.

Using Components to Provide a Flexible Adaptation Loop to Component-based SOA Applications

Cristian Ruz, Françoise Baude, Bastien Sauvan
INRIA Sophia Antipolis Méditerranée
CNRS, I3S, Université de Nice Sophia Antipolis
France
{*Cristian.Ruz, Francoise.Baude, Bastien.Sauvan*}*@inria.fr*

***Abstract*—The Service Oriented Architecture (SOA) model fosters dynamic interactions of heteregeneous and loosely-coupled service providers and consumers. Specifications like the Service Component Architecture (SCA) have been used to tackle the complexity of developing such applications; however, concerns like runtime management and adaptation are left as platform specific matters. Though several solutions have been proposed, they have rarely been designed in an integrated way and with the capability to evolve the adaptation logic itself. This work presents a component based framework that allows the insertion of monitoring and management tasks, providing flexible autonomic behaviour to component-based SOA applications. Each phase of the autonomic control loop is implemented by a different component, in such a way that different implementations can be developed for each phase and they can be replaced at runtime, providing support for evolving non-functional requirements. We present an illustrative scenario that is dynamically augmented with components to tackle non-functional concerns and support adaptation. We use an SCA compliant platform that allows distribution and architectural reconfiguration of components. Micro-benchmarks and a use case are presented to show the feasibility of our proposed implementation, and illustrate the practicality of the approach. Overall, we show that a component-based approach is suitable to provide autonomic and adaptable behaviour to component-based SOA applications.**

***Keywords-Monitoring*; *Autonomic Management*; *SLA Monitoring*; *Reconfiguration*; *Component-based Software Engineering*.**

I. Introduction

According to the principles of Service Oriented Architecture (SOA), applications built using this model comprise loosely-coupled services that may come from different heterogeneous providers. At the same time, a provided service may be composed of, and consume other services, in a situation where service providers are also consumers. Moreover, SOA principles like abstraction, loosely-coupling and reusability foster dynamicity, and applications should be able to dynamically replace a service in a composition, or adapt the composition to meet certain imposed requirements.

Requirements over service based applications usually include metrics about Quality of Service (QoS) like availability, latency, response time, price, energy consumption, and others, and are expressed as Service Level Objectives (SLO) terms in a contract between the service consumer and the provider, called Service Level Agreement (SLA). However, SLAs are also subject to evolution due to different providers, environmental changes, failures, unavailabilities, or other situations that cannot be foreseen at design time. The complexity of managing changes under such dynamic requirements is a major task that pushes the need for flexible and self-adaptable approaches for service composition. Self-adaptability requires monitoring and management features that are transversal to most of the involved heterogeneous services, and may need to be implemented in different ways for each one of them.

Several approaches have been proposed for tackling the complexity, dynamicity, heterogeneity and loosely-coupling of SOA-based compositions. Notably, the Service Component Architecture (SCA) is a technologically agnostic specification that brings features from Component-Based Software Engineering (CBSE) like abstraction and composability to ease the construction of complex SOA applications. Non-functional concerns can be attached using the SCA Policy Framework. However, monitoring and management tasks are usually left out of the specifications and must be handled by each SCA platform implementation, mainly because SCA is design-time and not runtime focused.

In our previous work [1] we have proposed a component-based approach to ease the implementation of flexible adaptation in component-based service-oriented applications. Our solution implements the different phases of the widely used MAPE (Monitor, Analyze, Plan, and Execute) autonomic control loop [2] as separate components that can interact and support multiple sets of monitoring sources, conditions, strategies and distributed actions.

Our approach gives two kinds of flexibility: (1) we can dynamically inject or remove conditions, sensors, planning strategies, or adaptation actions in the MAPE loop in order to modify the way the autonomic behaviour is implemented in the application; and (2) we can insert or remove elements of the MAPE loop, modifying the composition of the autonomic control loop itself, and making the application more or less autonomic as needed.

In this work we extend the presentation of our component-

based framework detailing the design considerations for each phase of our autonomic control loop and how they provide the flexibility that we expect. We present a concrete use case of an application that is dynamically augmented with autonomic behaviour.

The rest of the paper is organized as follows. Section II presents the example that we use to motivate and illustrate the practicality of our work, and provides a general overview of our contribution. Section III describes the design of our framework from a technologically independent point of view. Section IV presents our implementation over a concrete middleware and component model. Section V shows a practical example of use of our framework and the evaluations we have carried on. Section VI describes related work and differentiations with our solution. Finally, Section VII concludes the paper.

II. Motivating example and Overview of Our Contribution

Consider a tourism office that has composed a smart service to assist visitors who request information from the city and provides suggestions of activities. The application uses a local database of touristic events and a set of providers who sell tickets to museums, tours, etc. A weather service can be used to complement the proposition of activities, and a mapping service creates a map with directions. A payment service is used to process online sells in some cases. Once all information is gathered, a local engine composes a PDF document and optionally prints it. The composed design of the application is shown in Figure 1 using the SCA [3] diagram notation.

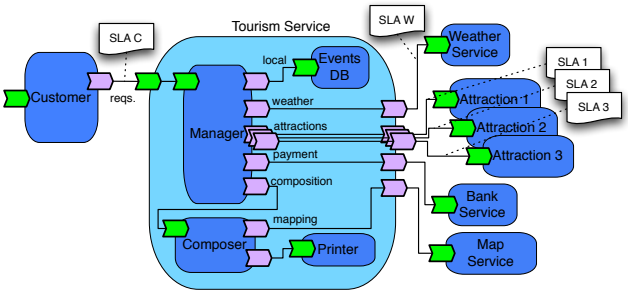


Figure 1. The SCA description of the application for tourism planning scenarios.

Such a composition involves some terms for service provisioning. For example, the Tourism Service agrees to provide a touristic plan within 30 sec.; the Weather Service charges a fee for each forecast depending on the level of detail; the Mapping Service is a free service but has no guarantees on response time or availability; the Payment Service ensures 99% of availability. All these conditions are formally established in several SLAs.

The runtime compliance to the SLAs may influence certain decisions on the composed service. For instance, if the Mapping Service is not reachable at a certain moment or if it takes too much time to deliver a response, then the Tourism Service may provide a touristic plan without maps in order to meet the agreed response time at the expense, however, of a lower quality response (workflow modification). Another situation may happen if the Weather Service increases its costs, thus violating the agreement, then the Tourism Service may decide to replace it for another equivalent cheaper service (service replacement). Finally, if the Printer service is running short on color cartridge, then the Tourism Service may decide to use only black and white printing (parameter modification).

In all these cases the decisions should, ideally, be taken in an autonomic way. This requires to constantly monitor certain parameters of the application and, in order to timely react, an efficient analysis and decision taking process. However, it should not be a task of the programmer of each service to code all these autonomic behaviours. Instead, it is more desirable to compose the autonomic behaviour in a separate way and insert it or remove it from the service activity as needed. Moreover, if an autonomic behaviour requires to collect information from different services, then forcing each service to be explicitly aware of the details of other services would increase the coupling of the services.

Also because of the heterogeneity of the services, the monitoring requirements may be different for each service; for example, in the case of the printer it is important to measure the amount of paper or ink; in the case of the touristic plan composer it is important to know the time it takes to create a document; some of the external services may provide their own monitoring metrics and, as they are not locally hosted and only accesible through a predefined API, it may not be possible to add specific monitoring on their side. So, in any case the monitoring capabilities will be limited by the monitoring features available from each service. This situation imposes a requirement for supporting heterogeneous services and adaptable monitoring.

A. Concerns

As it can be seen from the example, concerns about SLA and QoS can be manifold. A monitoring system may be interested in indicators for performance, energy consumption, price, robustness, security, availability, etc., and the range of acceptable values may be different for each monitored service. Moreover, not only the values of these indicators may change at runtime, but also the set of required indicators. Also, heterogeneity plays a role at the moment of programming the access to the required values.

In general, the evolution of the SLA and the required indicators can not be foreseen at design time, and it is not feasible to prepare a system where all possible monitorable conditions are ready to be monitored. Instead, it is desirable to have a flexible system where only the required set of monitoring metrics are inserted and the required conditions

checked, but as the application evolves, new metrics and conditions may be added and others removed minimizing the intrusion of the monitoring system in the application.

B. Contribution

We argue that a component-based approach can tackle the dynamic monitoring and management requirements of a composed service application while also providing the capability to make the application self-adaptable. We propose a component-based framework to add flexible monitoring and management concerns to a running component-based application.

In this proposition we separate the concerns involved in a classical autonomic control loop (MAPE) [2] and implement those concerns as separate components. These components are attached to each managed service, in order to provide a custom and composable monitoring and management framework. The framework allows distributed monitoring and management architectures to be built in a way that they are clearly associated to the actual functional components. The framework leverages the monitoring and management features of each service to provide a common ground in which monitoring, SLA checking/analysis, decisions, and actions can be carried on by different components, and they can be added or replaced separately.

We believe that the dynamic inclusion and removal of monitoring and management concerns allows (1) to add only the needed monitoring operations, minimizing the overhead, and (2) to better adapt to evolving monitoring needs, without enforcing a redeployment and redesign of the application, and increasing separation of concerns.

III. Design of the Component-Based Solution

Our solution relies on the separation of the phases of the classical MAPE autonomic control loop. Namely, we envision separate components for monitoring, analysis, planning, and execution of actions. These components are attached to each managed service.

From an external point of view, a regular service A is augmented at design time with a set of additional interfaces. These interfaces define the entry points to the management framework for each service A, which is transformed into *managed service* A, as shown in Figure 2. The management interfaces allow the service to interact with other managed services and take part in the framework; however the services are not forced to provide an implementation of all these management interfaces. Instead, these implementations can be dynamically added.

The general structure of our design is shown for an individual service A in Figure 3. Service A is extended with one component for each phase of the MAPE loop and converted into a *Managed Service* A, indicated by dashed lines. The original "service" and "reference" interfaces of service A are promoted to the corresponding interface of

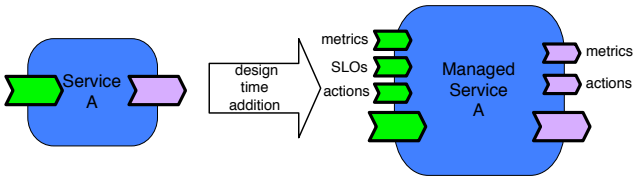


Figure 2. SCA component A extended at design time with management interfaces

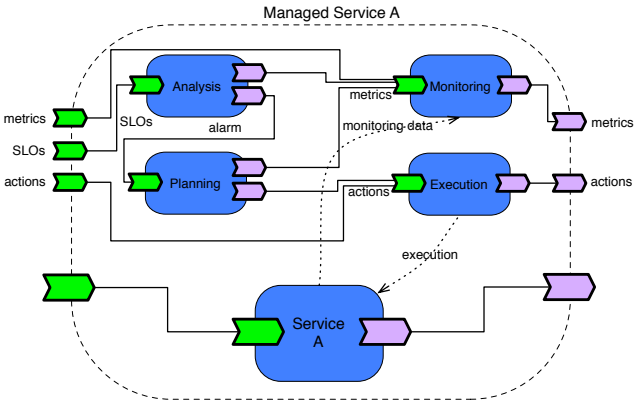


Figure 3. SCA component A with all its attached monitoring and management components

Managed Service A so that, from a functional point of view, the *Managed Service* A can be used in the same way as the original Service A.

The general functioning of the framework is as follows. The *Monitoring* component collects monitoring data from service A using the specific means that A may provide. Using the collected monitoring data, the *Monitoring* component provides access to a set of metrics through the *metrics* interface. The computation of metrics may involve communication with the *metrics* interface of other managed services. The *Analysis* component provides an interface for receiving and storing SLOs expressed as conditions. At runtime, the *Analysis* component checks the SLOs using the metrics that it obtains from the *Monitoring* component. Whenever an SLO is not fulfilled (a faulting condition), the *Analysis* component sends an alarm signal that activates the *Planning* component. The *Planning* component uses a pre-stored strategy to create an adaptation plan, described as a sequence of actions, that will be the response of the autonomic system to the faulting condition. If the adaptation strategy requires additional monitoring information, it can be obtained from the *Monitoring* component. The sequence of actions created by the *Planning* component are sent to the *Execution* component, which executes the actions on the service using the specific means that the service allows and, if needed, it can delegate the execution to the *Execution* component of other services. This way, the loop is completed.

Although simple, this component view of the autonomic control loop has several advantages.

- First, by separating the control loop from the component implementation, we obtain a clear **separation of concerns** between functional content and non-functional activities; meaning that the programmer of the application does not need to explicitly deal with management activities or with autonomic behaviour.
- Second, the component-based approach allows separate implementations to be provided for each phase of the loop. As each phase may require complex tasks, we abstract from their implementation, that may be specific for each service, and allow them to interact only through predefined interfaces, so that each phase may be implemented by different experts.
- Third, as each phase can be implemented in a separate way, we may consider components that include, for example, multiple sensors, condition evaluators, planning strategies, and connections to concrete effectors as required. This way we allow multiple autonomic control loops running over the same system, taking care of different concerns.

Regarding the genericity or the approach we have described it in a way as technology-independent as possible. However, every implementation that intends to manage a concrete service has, at some point, to use the specific means that the service admits either for obtaining information from it, or for modifying it. Our design is generic until the point that we must define the concrete sensors and actuators that must interact with the managed service. Actually, the amount of information that we can collect from the service and the kind of actions that we can execute over it, will be limited by the methods that the service makes available. We consider, however, that this limitation is given by the technology that provides access to the services (in this case, a component middleware) instead of the service programmer itself. In Figure 3, the service implementation dependent parts are indicated by the dashed arrows between the Service A and its respective *Monitoring* and *Execution* components.

The framework allows the addition and removal at runtime of different components of the loop, which means that, for example, a service that does not need monitoring information extracted, does not need to have a *Monitoring* component and may only have an *Execution* component to modify some parameter of the service. Later, if needed, it is possible to add other components of the framework to this service. This way, a service may be modified at runtime to have a major or minor level of autonomicity according to the needs.

As a simple example, consider a component that represents a storage service, and provides some basic operations to read, write, search and delete files. In order to get information about the performance of the storage service, a *Monitoring* component can be added and expose metrics about the average response time for each operation, and the amount of free space. As an evolution, some non-functional maintenance actions can be exposed to compress, index, or tune the periodicity of backups. These actions can be exposed by adding an *Execution* component that can execute them over the storage service. Now the managed storage service exposes some metrics, and exposes an interface for executing maintenance actions. However, the storage service is still not autonomic and the reading of metrics and execution of maintenance actions are invoked by external entities. A next evolution can consider adding an autonomic behaviour to avoid filling the capacity of the storage service. An *Analysis* component can be added and include a condition that checks the amount of free space, and in case it is less than, for example, 2%, it triggers an action oriented to increase the amount of free space. The decision about what action to take can be delegated to a *Planning* component, which will create the list of actions to be carried on by the *Execution* component.

Depending on the management needs, any evolution of the storage service can be used. If the autonomic behaviour described is not needed anymore, then the *Analysis* and *Planning* components can be removed and return to the simple version of the storage service. The three versions mentioned of the storage service are shown in Figure 4.

In the following, we describe the components considered in the monitoring and management framework, their function and some design decisions that have been taken into account.

A. Monitoring

The *Monitoring* task consists of collecting information from a service, and computing a set of indicators or *metrics* from it. The *Monitoring* component includes sensors specific for a service or, alternatively, supports the communication with sensors provided by the target service. This way, the *Monitoring* component can be effectively attached to the service.

In the presence of a high number of services, the computing and storage of metrics can be a high-demanding task, specially if it is done in a centralized manner. Consequently, the monitoring task must be as decentralized and low-intrusive as possible. For this, our design considers one *Monitoring* component attached to each monitored service, that collects information from it, and exposes an interface to provide the computed metrics. This approach is decentralized and specialized with respect to the monitored service. On the other side, some metrics may require additional information from other services: for example, to compute the cost of running a composition, the *Monitoring* component would require to know the cost of all the services used while serving some request. To address this situation in a decentralized way, the *Monitoring* component is capable

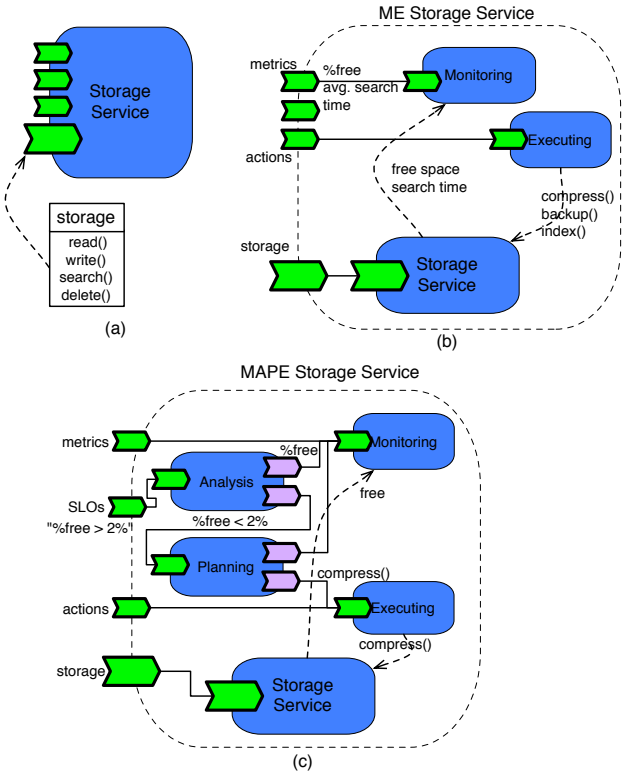


Figure 4. (a) Storage service in its basic version, (b) with Monitoring and Executing components, (c) with all the MAPE components and providing an autonomic behaviour

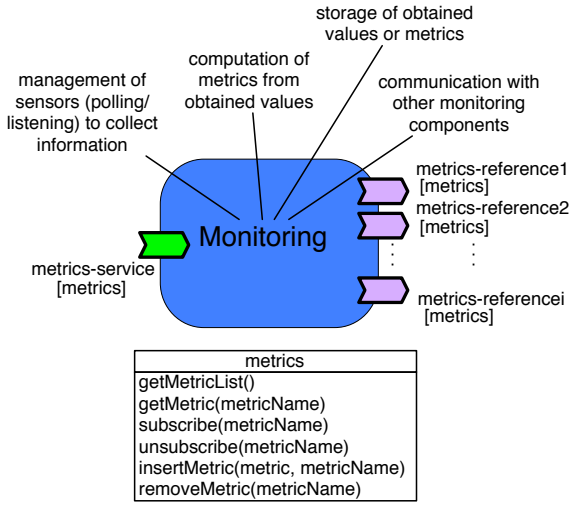


Figure 5. The *metrics* interface of the *Monitoring* component

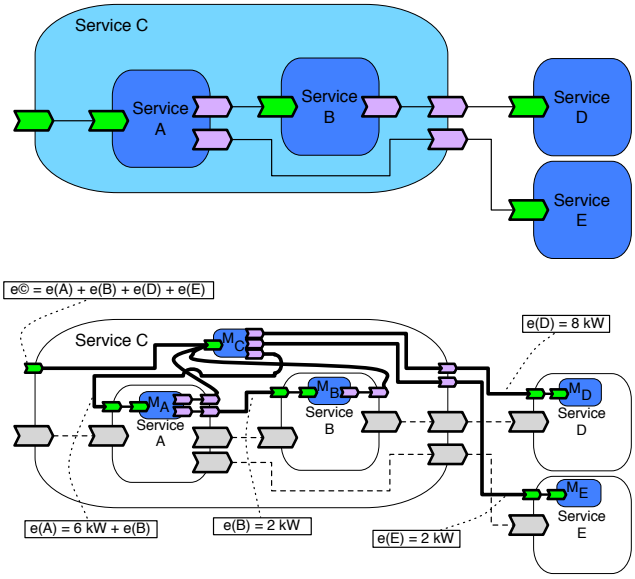


Figure 6. An SCA application, and the inner "monitoring backbone"

of connecting to the *Monitoring* components of other services. The set of *Monitoring* components are inter-connected forming an architecture that reflects the composition of the monitored service and forming a "monitoring backbone" as shown in Figure 6.

Figure 5 shows the methods of the *metrics* interface. Metrics are referenced by a *metricName* string. The method *getMetric(metricName)* is used by another component, or by an external tool to fetch the current value of the metric *metricName* in a *pull* mode. It is also possible to read the values in a *push* mode by using the *subscribe(metricName)* and *unsubscribe(metricName)* methods, so that the *Monitoring* component notifies the receptor of any changes in the value. The method *getMetricList()* allows the caller to verify which metrics are available from the *Monitoring* component, and the *insertMetric(metric, metricName)* and *removeMetric(metricName)* methods allow the caller to manipulate the available metrics by inserting or removing the code that actually computes the values. An actual implementation of this interface is permitted to extend it as needed.

Figure 6 shows an example of a *metric* named "energy consumption" ($e(i)$) for each component i. Each *Monitoring* component M_i is in charge of computing its value $e(i)$ as the sum of its own energy metric, and those of its references. In the case of the composite service C, the value $e(C)$ is the sum of the values of both internal components, $e(A)$ and $e(B)$, and of its references $e(D)$ and $e(E)$. Using the connection between the different *Monitoring* components, the total value $e(C)$ is computed by M_C and exposed through its *metrics* interface. Note that the means for computing the energy metric for each component may be different, depending on the characteristics of the implementation; however, once the value is computed in the corresponding *Monitoring* component, it becomes accessible in a uniform way by the other *Monitoring* components.

Figure 6 also shows a characteristic of our design with respect to the number of monitoring interfaces. In order to connect to monitoring interfaces of other components, each *Monitoring* component includes one reference to the

Monitoring component of each component to which the managed component is bound. This is done so that we can properly identify the monitoring information coming from each managed component. It is possible to see, for example, that M_C includes three references: one for communicating with M_A because the service interface of Service C is bound to Service A; and two reference interfaces for M_D and M_E because Service D and Service E are referenced by Service C. In this particular case, M_C is not bound to M_B because its service interface is not bound to any service interface of Service C.

B. Analysis

The *Analysis* component checks the compliance to a previously defined SLA. An SLA is defined as a set of simpler terms called SLOs, which are represented by conditions that must be verified at runtime.

One of the challenges of the *Analysis* component is to be able to understand the conditions that need to be checked. There exist several languages proposed for representing SLOs and the metrics they require [4], [5], [6], [7]. Using a component-based approach inside the *Analysis* component it should be possible to embed an interpreter for these languages into the *Analysis* component.

For illustrative purposes, we can consider a very simple description of conditions using triples $\langle metric, comparator, value \rangle$ expressing, for instance, "$respTime \leq 30sec$"; or more complex expressions involving other metrics or operations on them like "$cost(weatherService) < 2 \times cost(mappingService)$", where the metrics used by different services are required.

The *Analysis* component obtains the values of the metrics it needs from the *Monitoring* component and, thanks to the interconnected *Monitoring* components, it can obtain metrics from other services as well.

The *Analysis* component receives a set of conditions (SLOs) to monitor through the *SLOs* interface, and it checks the compliance of all the stored SLOs according to the metrics reported by the *Monitoring* component. In case some SLO is not fulfilled, the *Analysis* component sends an alarm notification through a reference *alarm* interface. The consequences of this alarm are out of the scope of the *Analysis* component and will be mentioned in the next section.

The *Analysis* component can also be configured in a proactive way to detect SLA violations not only after they happened, but instead to generate the alarm before the violation happens (with a certain probability). This predictive capability may be useful in many contexts, as it can avoid incurring into penalties as a consequence of the ocurrence of the violation [8]. Of course, a tradeoff between the precision of the prediction and the cost of the prevention must be made.

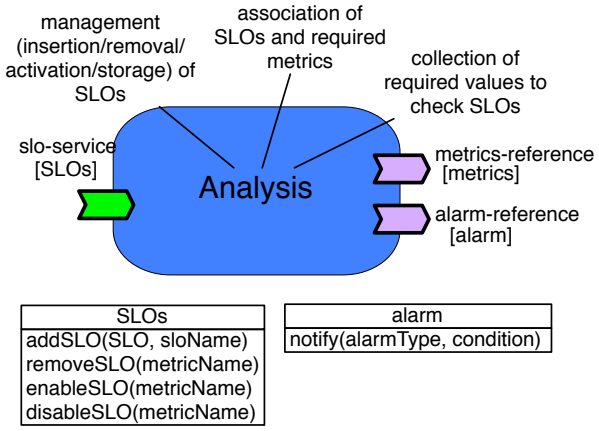


Figure 7. The SLOs interface of the *Analysis* component

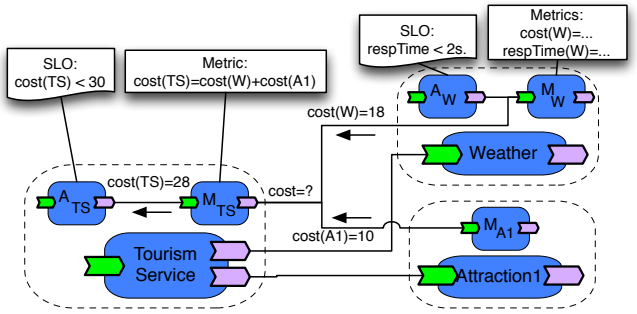


Figure 8. SCA components with Analysis (A_i) and Monitor (M_i) components. *Tourism Service* and *Weather* have different SLAs. The metric *cost* is computed in *Tourism Service* by calling the monitors of *Weather* and *Attraction1*.

By having the *Analysis* component attached to each service, the conditions can be checked closely to the monitored service and benefit of the hierarchical composition. This way, the services do not need to take care of SLAs in which they are not involved.

Figure 7 shows the methods of the *SLOs* interface. The methods allow the caller to manipulate the list of SLOs that are checked by the *Analysis* component by inserting or removing the object that contains the SLO description and referencing it through the *sloName* string. The enable/disable methods permit the caller to enable or disable the verification of a particular SLO. The precise manner in which the *Analysis* component reads and stores the SLO objects, checks the compliance of the SLOs, and obtains the information from the *Monitoring* component are left as an implementation concern. One way to implement it is described in Section IV-D.

Figure 8 shows an example where Service *TourismService (TS)* has an *Analysis* component A_{TS}, and a *Monitoring* component M_{TS}; services *Weather (W)* and *Attraction1 (A1)* are referenced by *TS*. Service *W* includes an *Analysis* component A_W and a *Monitoring* component M_W; service

A1 only includes a *Monitoring* component M_{A1}.

The *Analysis* component of *TS* must check the SLO "$\langle cost, <, 30\rangle$" over Service *TS*. For checking that condition, it requires the value of the metric *cost* from M_{TS}. In M_{TS}, the computation of the metric *cost* requires the value of the metric *cost* from both services W and $A1$. M_{TS} obtains this information from the corresponding *Monitoring* components M_W and M_{A1} and is able to deliver the response to A_{TS}. It is worth noting that A_{TS} is not aware that the computation of M_{TS} actually required additional requests to M_W and M_{A1}, as this logic is hidden into M_{TS}. At the same, the *Analysis* component A_W works independently to check a condition related to the response time (*respTime*) metric from service W, which requires to read the appropriate metric from M_W.

C. Planning

The objective of the *Planning* phase is to generate a sequence of actions, called *plan*, that can modify the state of the service in order to restore some desired condition. In general, we want to restore the condition (the *SLO*) that has been violated.

The computation of a plan is triggered when a notification is received indicating that a condition is not being fulfilled, through the *alarm* interface. For creating such a plan, the *Planning* component must execute a planning algorithm that can determine that sequence of actions. This logic can be implemented in a number of ways. On the more simple side, a strategy may be a notification to a human agent (email, SMS, etc.) who would be responsible of taking any further action; another alternative could rely on a table of predefined actions, like ECA (Event-Condition-Action) triggers, such that if some conditions hold, then the corresponding action is generated. On a more complex side, numerous strategies and heuristics, in particular from the artificial intelligence area have been proposed for planning a composition or recomposition of services that complies with certain desired QoS characteristics. The aim of our *Planning* component is to be capable of supporting the implementation of such existing strategies.

The *alarm* interface is shown in Figure 9. It only considers one method *notify(alarmType, condition)* that includes the condition that is triggering the reaction, and optionally a level indicator called *alarmType* that permits the caller to assign priorities or levels of gravity of the notification.

Given the wide range of different solutions for generating a plan, it does not seem easy to find an interface that is uniform across all the possible strategies. However, most of the strategies require as input information the current state of the service in order to guide the possible solutions. Consequently, our *Planning* component considers one interface for obtaining information about the state of the service, connected to the *Monitoring* component.

Although a simple implementation would embed only one specific strategy, our approach considers that several

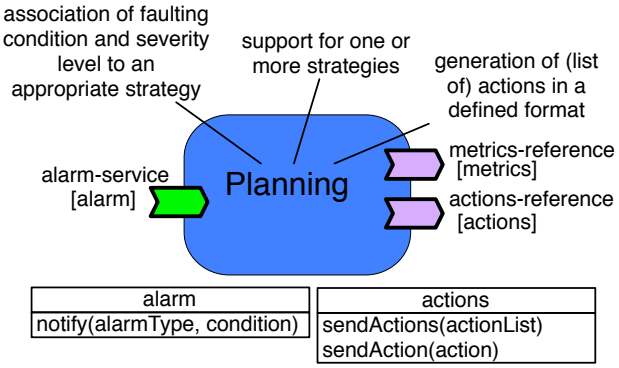


Figure 9. The *alarm* interface of the *Planning* component

conditions may be supported by the *Analysis* component. Consequently, several conditions may need to be checked and, if it is necessary to take some actions, different strategies may be applied upon each case. That is why we think that a component-based approach applied to the *Planning* component should be able to support different planning strategies that would be activated depending on the condition that needs to be restored.

It is also a concern that these strategies may be replaced at runtime. For example, an application may be driven by a cost-saving strategy and, at some point the administrator may need to change the requirements and enforce an energy-saving strategy. In that case, a replacement of the corresponding strategy should be performed inside the *Planning* component. However, this task is not an autonomic task of the framework itself and is, instead, driven by an administrator of the management layer.

Figure 10 shows an example where service *Tourism-Service (TS)* uses two services *Weather (W)* and *Mapping (MP)*. The *Planning* component of *TS*, P_{TS} receives an alarm from the *Analysis* component A_{TS} indicating that the condition $\langle cost, <, 30\rangle$ has been violated, and that an action should be taken. P_{TS} executes a very simple strategy, which intends to replace the component with the higher cost. For obtaining the *cost* of both components *W* and *MP*, P_{TS} uses the *Monitoring* component M_{TS}, which communicates with M_W and M_{MP} to obtain the required values. As *MP* has the higher cost, the strategy determines that this component must be replaced. P_{TS} uses an embedded reference to a discovery service, to obtain an alternative service, called *MX*, which provides the same functionality as *MP* (this is neccesary to not interfere with the functional task of the application) and whose *cost* is expected to satisfy the condition $\langle cost, <, 30\rangle$. With all this information, P_{TS} is able to produce a single action $replace(MP, MX)$ as output.

It is worth to notice that all the logic of the planning algorithm is encapsulated inside P_{TS}, and that M_{TS} is only used to obtain the values of the metrics that the strategy may need.

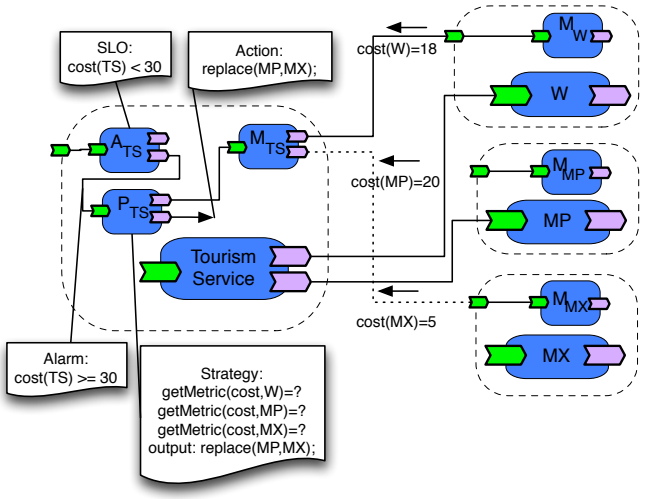


Figure 10. Example for the Planning component.

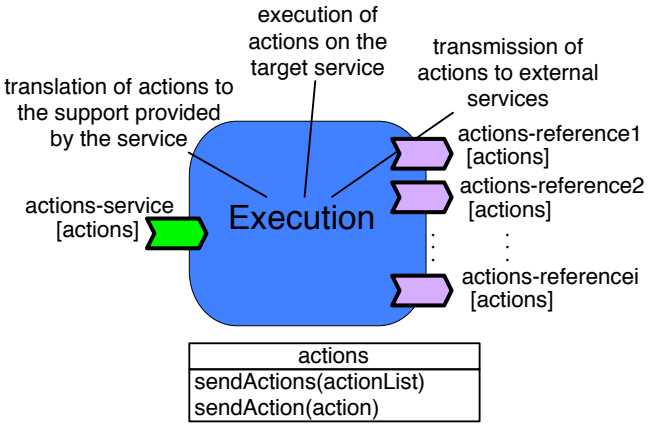


Figure 11. Example for the Execution component.

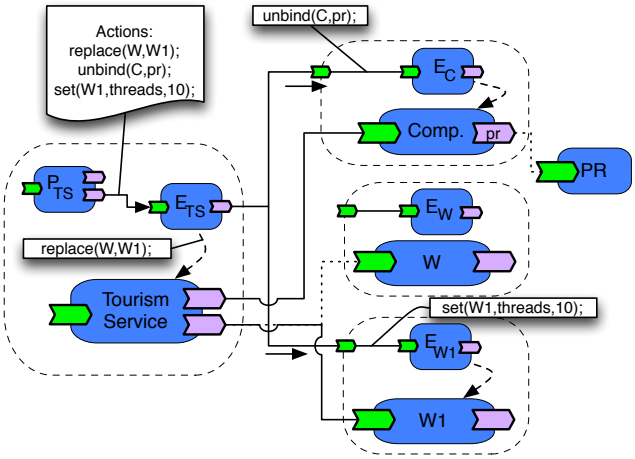


Figure 12. Example of propagation of actions through *Execution* components

D. Execution

The *Execution* component carries out the sequence of actions that have been determined by the *Planning* component.

Although it seems reasonable that once the actions have been decided, those be executed immediately, the *Execution* component has more importance than just executing actions. One of the reasons for having a different component is to separate the description of the actions from the specific way to execute them. In the same sense that the *Monitoring* component abstracts the way to retrieve information from the target service and provides a common interface to access the metrics it collects, the *Execution* component abstracts the communication with the target service to provide a uniform way to execute actions on the service. This also implies that, like the *Monitoring* component, the *Execution* component must be implemented according to the specific characteristics of the service on which the actions must be executed.

The set of actions demanded may involve not only the managed service, but also different services. For this reason, the *Execution* component is also able to communicate with the *Execution* components attached to some other components and send actions to them as part of the main reconfiguration action. The set of connected *Execution* components forms an "execution backbone" that propagates the actions from the component where the actions have been generated to each of the specific components where some part of the actions must take place, possibly hierarchically down to their respective inner components. This approach allows to distribute the execution of the actions.

The *Execution* component receives the sequence of actions to execute from the *actions* interface, which is shown in Figure 11. The interface has two methods that permit the caller to send either a list of *action*s, or a single *action* to the *Execution* component. The proper definition of the *action* object will depend on the implementation. In any case, the *Execution* must be able to read this object and interpret it as an action that can be executed on the service.

Figure 12 shows an example where three actions are generated by the *Planning* component of *TourismService (TS)*: one to replace the service *Weather (W)*, one to unbind the service *Printer (PR)*, and the third one to set a parameter on the reference to service *Mapping (MP)*. In the example, the *Planning* component P_{TS} has sent the list of actions to the *Execution* component E_{TS}. The action of replacing component *W* by W_1 is executed locally at *TS*. However, the unbinding of reference *pr* on service *Composer (C)* must be executed by E_{TS}; and the setting of the parameter "threads" on service W_1 must be executed by E_{W_1}. By using the connections between the different *Execution* components, the actions can be delegated to the appropriate place.

IV. Implementation

This section describes our prototype implementation over a middleware that implements a particular component model. We describe the pieces of the framework that have been implemented according to the design guidelines presented in Section III and exemplify how they can be used to provide self-adaptability in the context of the scenario described in Section II.

A. Background: GCM/ProActive

The ProActive Grid Middleware [9] is a Java middleware, which aims to achieve seamless programming for concurrent, parallel and distributed computing, by offering an uniform active object programming model, where these objects are remotely accessible via asynchronous method invocations and futures. Active Objects are instrumented with MBeans, which provide notifications about events at the implementation level, like the reception of a request, and the start and end of a service. The notification of such events to interested third parties is provided by an asynchronous and grid enabled JMX connector [10].

The Grid Component Model (GCM) [11] is a component model for applications to be run on computing grids, that extends the Fractal component model [12]. Fractal defines a component model where components can be hierarchically organized, reconfigured, and controlled offering functional server interfaces and requiring client interfaces (as shown in Figure 13). GCM extends that model providing to the components the possibility to be remotely located, distributed, parallel, and deployed in a grid environment, and adding collective communications (multicast and gathercast interfaces). In GCM it is possible to have a componentized membrane [13] that allows the existence of non-functional (NF) components, also called *component controllers* that take care of non-functional concerns. NF components can be accessed through NF server interfaces, and components can make requests to NF services using NF client interfaces (shown respectively on top and bottom of A in Figure 13).

The use of NF components instead of simple object controllers as in the Fractal reference implementation, allows a more flexible control of NF concerns and to develop more complex implementations, as the NF components can be bound to other NF components within a regular component application. This notion of defining a componentized membrane has been used in previous works to manage an define structural reconfigurations [13], [14]. In this work we use these notions to address self-adaptability concerns in service-oriented contexts.

GCM/ProActive is the reference implementation of GCM, within the ProActive middleware, where components are implemented by Active Objects, which can be used to implement new services using Java, or wrap existent legacy applications like C/Fortran MPI code, or a BPEL code.

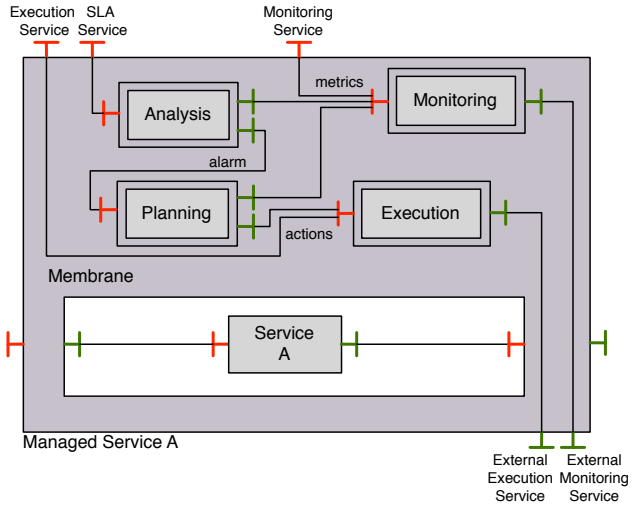


Figure 13. Framework implementation weaved to a primitive GCM component A. The MAPE components are isolated from the functional part in the *membrane* of the component.

The GCM/ProActive platform provides asynchronous communications with futures between bound components through GCM bindings. GCM bindings are used to provide asynchronous communication between GCM components, and can also be used to connect to other technologies and communications protocols, like Web Services, by implementing the compliance to these protocols via specific controllers in the membrane. These controllers have been used to allow GCM to act as an SCA compliant platform, in a similar way as achieved by the SCA FraSCAti [15] platform, which however bases upon non distributed components (Fractal/Julia).

B. Framework Implementation

The framework is implemented in the GCM/ProActive middleware as a set of NF components that can be added or removed at runtime to or from the membrane of any GCM component, which becomes a managed service of the application.

We have designed a set of predefined components that implement each one of the elements we have described in Section III. This is just one of possible implementations, and particularly this has been designed to provide self-adaptable capabilities to the composition.

The general implementation view for a single GCM component is shown in Figure 13 (using the GCM graphical notation [11]), and resembles the design presented in Figure 3, however now the components that implement the MAPE control loop are inserted in the membrane and they are structurally isolated from the functional part. The framework is weaved in the GCM component A by inserting NF components in its membrane. Monitoring and management features are exposed through the NF server interfaces *Mon-*

itoring Service, *SLA Service* and *Execution Service* (top of Figure 13). NF components can communicate with the NF components of other GCM components through the NF client interfaces *External Monitoring Service* and *External Execution Service* (bottom of Figure 13). The sequence diagram of the self-adaptability loop is shown in Figure 14.

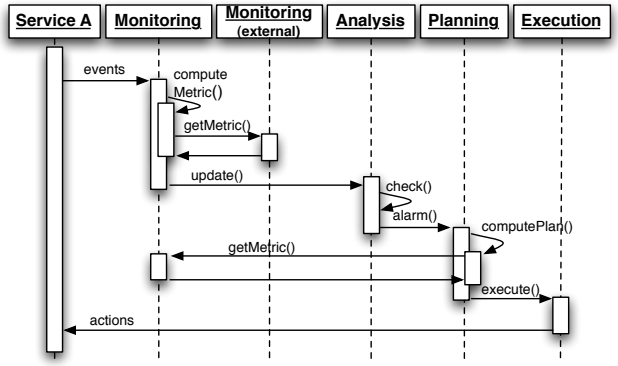


Figure 14. Sequence diagram for the autonomic control loop

C. Monitoring

We have designed a set of probes for CPU load and memory use, and incorporated them along with the events produced by the GCM/ProActive platform. Over them, we provide a *Monitoring* component, shown on Figure 15, which includes (1) an *Event Listener* that receives events from a GCM component and provides a common ground to access them; (2) a *Record Store* to store records of monitored data that can be used for later analysis; (3) a *Metric Store* that stores objects that we call *Metrics*, which actually compute the desired metrics using the records stored, or the events caught; and (4) a *Monitor Manager*, which provides the interface to access the stored metrics, and add/remove them to/from the Metrics Store.

The *Monitor Manager* receives a Metric that, in our implementation, is a Java object with a *compute* method, and inserts it in the *Metric Store*. The *Metric Store* provides to the Metrics the connection to the sources that they may need; namely, the *Record Store* to get already sensed information, the *Event Listener* to receive sensed information directly, or the *Monitoring* component of other external components, allowing access to the distributed set of monitors (i.e., to the monitoring backbone). For example, a simple *respTime* metric to compute the response time of requests, requires to acess the *Record Store* for retrieving the events related to the start and finish times of the service of a request.

Consider, for instance, that the Tourism Service needs to know the decomposition of the time spent while serving a specific request r_0. For this, a metric called *requestPath* for a given request r_0 can ask the *requestPath* to the *Monitoring* components of all the services involved while serving r_0, which can repeat the process themselves; when no more calls

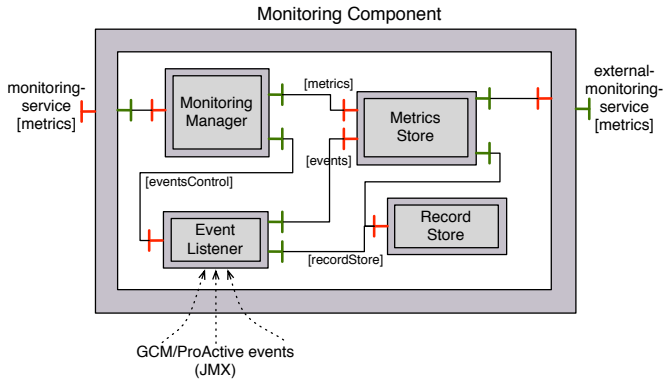


Figure 15. Internal Composition of the Monitoring component

are found, the composed path is returned with the value of the *respTime* metric for each one of the services involved in the path. Once the information is gathered in the *Monitoring* component of the Tourism Service, the complete path is built and it is possible to identify the time spent in each service.

D. SLA Analyzer

The *SLA Analyzer* is implemented as a component that queries the *Monitoring* component. The *SLA Analyzer* consists in (1) an *SLO Analyzer*, which transforms the SLO description to a common internal representation, (2) an *SLO Store* that maintains the list of SLOs, (3) an *SLO Verifier* that collects the required information from the *Monitoring* interface and generates alarms, and (4) an *SLA Manager* that manages all the process.

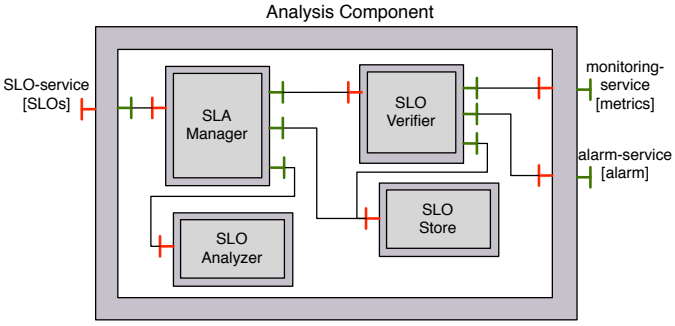


Figure 16. Internal Composition of the Analysis component

In this implementation, an SLO is described as a triple $\langle metricN, comparator, value \rangle$, where *metricN* is the name of a metric. The SLA Monitor subscribes to the *metricN* from the *Monitoring* component to get the updated values and check the compliance of the SLO.

For example, the Tourism Service includes the SLO: "All requests must be served in less than 30 secs", described as $\langle respTime, <, 30 \rangle$. The *SLA Manager* receives this description and sends a request to the *Monitoring* component for subscription to the *respTime* metric. The condition is then

stored in the *SLO Store*. Each time an update on the metric is received, the *SLA Manager* checks all the SLOs associated to that metric. In case one of them is not fulfilled, a notification is sent, through the *alarm* interface including the description of the faulting SLO.

E. Planning

The *Planning* component, shown on Figure 17, includes a *Strategy Manager* that receives an alarm message and, depending on the content of the alarm, it triggers one of several bound *Planner* components. Each one of the *Planner* components implements a planning algorithm that can create a plan to modify the state of the application. Each *Planner* component can access the *Monitoring* components to retrieve any additional information, they may need; the output is expressed as a list of actions in a predefined language.

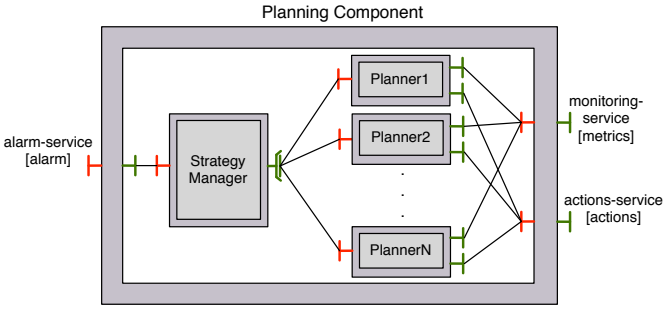


Figure 17. Internal Composition of the Planning component

In our implementation we profit by the selective 1-to-N communications provided by GCM to decide the *Planner* component that will be triggered. For example, if the SLO violated is related to response time, we may trigger a planner that generates a performance-oriented recomposition; or if a given cost has been surpassed, we may trigger a cost-saving algorithm. The decision of what planner to use is taken in the *Strategy Manager* component. However, the possibility of having multiple strategies might be a source for conflicting decisions; while we do not provide a method to solve these kind of conflicts, we assume that the conflict resolution behaviour, if required, is provided by the *Strategy Manager*.

We have implemented a simple planning strategy that, given a particular request, asks to compute the *requestPath* for that request, then finds the component most likely responsible for having broken the SLO, and then creates a plan that, when executed, will replace that component for another component from a set of possible candidates. Applied to the Tourism Service, suppose a request has violated the SLO $\langle respTime, <, 30 \rangle$. The *Strategy Manager* activates the *Planner* component that obtains the *requestPath* for that request along with the corresponding response time, selects the component that has taken the highest time, then obtains a set of possible replacements for that component, and obtains for each of them the *avgRespTime* metric. The output is a plan expressed in a predefined language that aims to replace the slowest component by the chosen one.

Clearly this strategy does not intend to be general, and does not guarantee an optimal response in several cases. Even, in some situations, it may fail to find a replacement and, in that case, the output is an empty set of actions. However, this example describes a planning strategy that can be added to implement an adaptation for self-optimizing and that uses monitoring information to create a list of actions.

F. Execution

The *Execution* component, shown on Figure 18, includes a *Reconfiguration Engine*. This engine uses a domain specific language called PAGCMScript, an extension of the FScript [16] language (designed for Fractal), which supports GCM specific features like distributed location, collective communications, and remote instantiation of components.

The *Execution* component receives actions from the *Planning* component. As many strategies may express actions using different formats, a component called *Execution Manager* may require a transformation to express the actions in an appropriate language for the *Reconfiguration Engine*, using a *Translation* component. The *Execution Manager* may also discriminate between actions that can be executed by the local component, or those that must be delegated to external *Execution* components.

For example, if a planner determines that the *Weather* service must be removed from the composition, it can be unbound from the *Tourism Service* by using a PAGCMScript command like the following

```
unbind($tourism/interface::"weather")
```

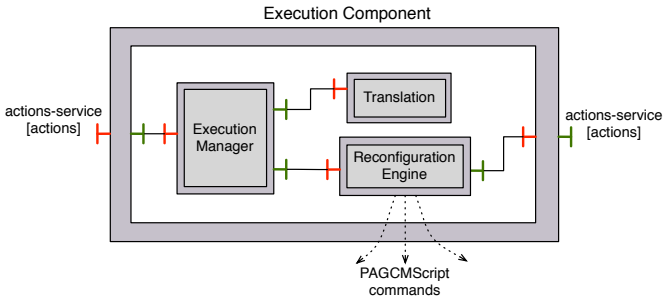


Figure 18. Internal Composition of the Execution component

G. Generalization and Dynamic Insertion

The GCM-based framework shown in Figure 13 has been presented as an instantiation of the SCA version shown in Figure 3. Indeed, the SCA design of Figure 3, presented only in terms of SCA elements, can be realized for any SCA runtime platform. The deployment of the framework may be done by injecting the required SCA description in

the SCA ADL file. This way, the application is deployed with all the needed elements of the framework attached.

In our implementation, however, we allow the insertion of the components that provide the autonomic behaviour to occur at runtime. We have provided a console application that can use the standard NF API of GCM components to insert or remove at runtime the required components of the framework.

The console, while not being itself a part of the framework, shows that an external application can be built and connected to the NF interfaces of the running application and handle at runtime the composition and any subsequent reconfiguration, if needed, of the monitoring and management framework itself. In the use case that we present in Section V-B we use this console application, for instance, to interact with the *Monitoring* interface and obtain the value of certain metrics.

V. Use Case and Evaluation

This section shows the experimentation we have made with the implementation of our framework over the GCM/ProActive middleware. The experimentation is divided in two parts. First we execute some micro-benchmarks to analyze the overhead incurred by the execution of the MAPE components concurrently with the functional application in our particular implementation. Then, we describe from a working point of view the use of the framework to insert and modify a set of MAPE components into a concrete application, showing the practicality of our proposition.

A. Performance

We have built a sample application with several components that interchange messages. Each execution performs a distributed computation through all the components to compose a return message, so that each execution generates a communication that ultimately reaches every other component.

1) MAPE Execution Overhead: We run a repetition of n messages in two versions of the application: one with no MAPE components inserted, and another with a version of each MAPE component inserted in all the membranes. This is, a complete MAPE cycle in each component. The *Monitoring* component computes metrics related to response time; the *Analysis* component checks an *SLO* that compares the response time in a *push* mode (subscription) upon each update of the *respTime* metric and, in case it is bigger than 1 second, it sends an alarm to a *planner* component. The *planner* only checks the last value obtained for the *respTime* metric from the *Monitoring* component, but does not generate actions. In order to isolate the execution of the application respect to network communication, in this experiment all the components are deployed in a single node.

The times obtained for each execution depending on the number of requests, and the overhead obtained for the total execution is shown in Table I. The "Base" column shows the execution time without any MAPE component inserted, and the "w/MAPE" columns shows the execution with all the MAPE components inserted and running in the membranes of each functional component.

#msgs	Base (sec)	w/MAPE (sec)	Diff.	%Overhead
1000	6.98	8.00	1.02	14.6
2500	17.20	19.29	2.09	12.2
5000	34.39	39.18	4.79	13.9
10000	68.57	77.55	8.98	13.1
20000	140.38	158.91	18.53	13.2

Table I
Execution Overhead in non-distributed application with MAPE components executing in the membrane of functional components

We observe that the overhead incurred stabilizes around 13% of the initial time. Although it seems important, we must highlight that this case represents one of the worst cases of an execution, as the only thing that this application does is to send requests to other components, while little functional work is done by each individual service. In a more general situation, an application would be expected to do some other activity than only sending requests. However, this experiment allows us to test the behaviour of our framework implementation under a high load and still obtaining acceptable results.

2) MAPE Execution and Communication Overhead: In this experiment we use a distributed version of the application, where each component is deployed in a different node in a grid environment. In this case, in addition to the overhead caused by the execution of the MAPE components, we expect to have and additional overhead caused by the communication between the membranes of the different functional components.

The results are shown in Table II. The "Base" column shows the execution time of the distributed application without any MAPE component inserted, and the "w/MAPE" columns shows the execution with all the MAPE components inserted and running in all the membranes, and in the same node of their corresponding managed functional component.

In this case, the overhead reaches around 15% of the "Base" execution time. This is not a big increment with respect to the previous situation, while the amount of network communication is bigger. Once again, we must mention that this particular experiment reflects a situation where the components spent most of the time sending and receiving requests, which consequently triggers reactions over the application. The node where each component runs must support the execution of both the original functional node, and the activity of the additional NF components.

#msgs	Base (sec)	w/MAPE (sec)	Diff.	%Overhead
1000	29.66	33.69	4.03	13.6
2500	72.20	82.18	9.98	13.8
5000	138.72	156.74	18.02	13.0
10000	271.45	314.20	42.75	15.7
20000	539.26	624.27	85.01	15.8

Table II
EXECUTION OVERHEAD IN A DISTRIBUTED APPLICATION WITH MAPE COMPONENTS EXECUTING IN THE MEMBRANE OF FUNCTIONAL COMPONENTS

Overall, the insertion of the MAPE components in this implementation implies a bigger load in the execution of the managed component, which is natural. In a worst-case scenario, the overhead incurred does not account for more than 15% of the not-managed execution. This measure however, is not completely accurate, as the actual overhead incurred by the MAPE components may depend on many additional factors. For one, the specific logic applied to the *metrics* implementation, and to the *planner* strategies may require much more additional processing. Moreover, the *planner* strategy may require to (it is not forbidden to) temporarily stop the functional execution of the component if some computation needs to be performed in an isolated way, introducing more overhead in the execution. However, we must remember that the planning activity should be executed mainly for resolving undesired situations and not become the main activity of the application.

Another factor is the supporting implementation. In our case we have conducted our experiments over a distributed environment supported by the GCM/ProActive middleware. This particular implementation profits of asynchronism to allow the concurrent execution of the MAPE components. Each implementation of the framework, however, may profit of their particular characteristics and optimize the implementation.

B. Use Case

We implement the application described in Section II using the GCM/ProActive implementation of our framework. The application is presented as an example of use of the framework to add progressively autonomic behaviour to an application.

The application is initially designed without any monitoring or management activity. However, in order to be able to insert some MAPE components later, it is necessary that the required interfaces be previously declared. In the context of our implementation, this is achieved by introspecting the functional interfaces defined for the component and, before instantiating the component, declaring the monitoring and management interfaces. This extension of the originally declared interfaces is done in an automatic way by our implementation prior to deploy the components. The components are, thus, deployed without any MAPE components inserted, however they are prepared to receive them and gradually support autonomic behaviour.

The design from Figure 1 is shown using the GCM notation in Figure 19 for the *Tourism Service* composite.

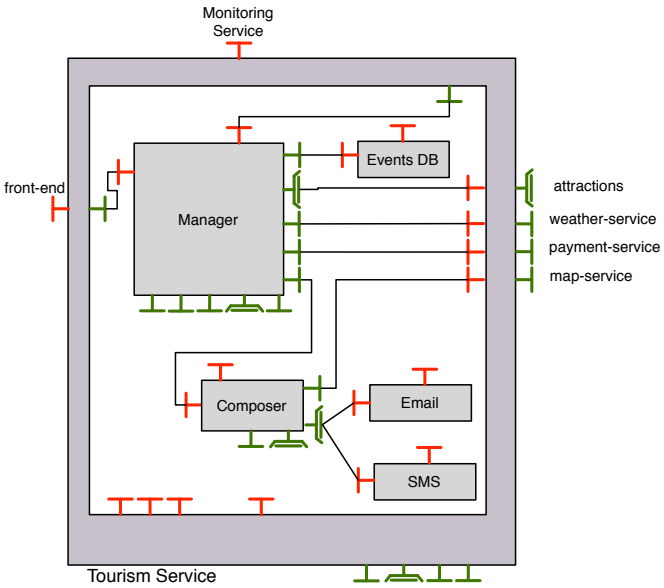


Figure 19. GCM description of the Tourism Service composite. NF interfaces are available but no NF Component is in the membrane

1) Inserting Monitoring activity: In order to monitor the application, it is possible to insert a *Monitoring* component as the one described in Section IV-C. Figure 20 shows the *Tourism Service* composite once the *Monitoring* component has been inserted in its membrane, and in each one of its subcomponents. The NF bindings are shown as solid lines inside the membrane, and as dashed lines in the functional part.

Using this configuration, it is now possible to connect to the *Monitoring* interfaces of each component and insert, query, or remove some metrics. Among others, we have implemented a metric called *respTime*, which computes the response time on the server side of a binding, a metric called *avgRespTime* that keeps an average of response time on each interface, and another one called *requestPath* that uses the previous one to trace the tree of calls generated by a request including the response time on each component. Our console application includes commands to connect and interact with the monitoring interfaces, providing an interaction like it is shown in Listing 1.

Listing 1. Request Path computation by invoking a metric from the console. Numbers in parenthesis are unique request identifiers

```
> addMetric TourismServ requestPath rp
Metric rp (type: requestPath) added to TourismServ
...
> addMetric MappingServ requestPath rp
Metric rp (type: requestPath) added to MappingServ
```

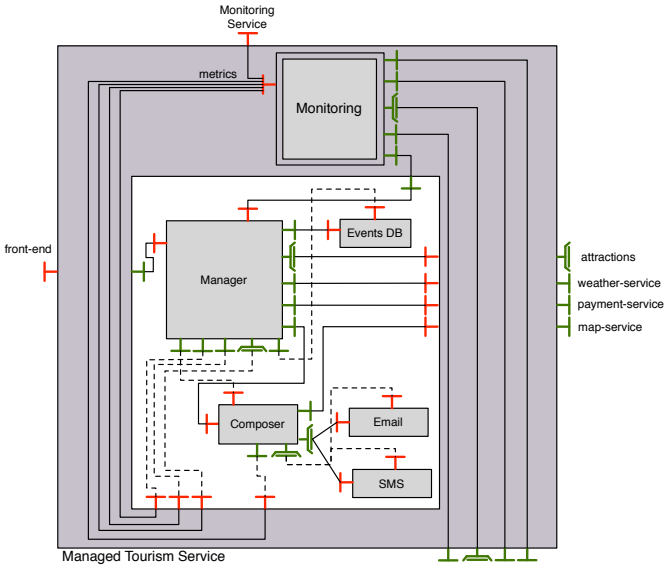


Figure 20. GCM description of the Tourism Service composite, once the *Monitoring* component has been inserted in the membrane of all components and its NF Interfaces are bound

```
> runMetric TourismServ rp 1131284383
Path from TourismServ, for request 1131284383
Request Path from request 1131284383
* (1131284383) TourismServ.reqs.buildDoc:
client: 7943 server: 7646
*     (-516789329) Manager.events.getEvent:
client:  410 server:  398
*     (-516789328) Manager.weather.getWeather:
client: 2224 server: 2118
*        (1131284384) TourismServ.weather.getWeather:
client: 2011 server: 1841
*     (-516789327) Manager.attr3.getTicktData:
client: 3019 server: 2867
*        (1131284385) TourismServ.attr3.getTicktData:
client: 2860 server:  702
*     (-516789326) Manager.composer.buildDoc:
client: 5066 server: 5002
*        (1278875256) Composer.mapping.getLocn:
client: 3200 server: 3109
*           (1131284385) TourismServ.mapping.getLocn:
client: 3006 server: 2955
*        (1278875257) Composer.email.send:
client: 1434 server: 1137
>
```

2) Automating the monitoring: By connecting to the *Monitoring* interface, it is possible to introduce metrics and request their values. However, this still requires to explicity ask for the values and interpret them in an external way from the application as shown on Listing 1.

A next level of autonomic behaviour is achieved by automating the monitoring. The *Analysis* component can be dynamically inserted in the membrane and bound to the *Monitoring* component to check periodically certain metrics. In the example, an *Analysis* component like that described in Section IV-D is inserted in the *Tourism Service* component, and made available through the *SLA Service* interface. This interface allows to insert SLOs according to the format described in Section IV-D and associate them to the metrics provided through the *Monitoring* interface. In the example shown in Figure 21, the *Analysis* component uses the *avgRespTime* metric to check the average response time on the *frontEnd* interface.

In case a condition is not met, the *Analysis* component is expected to throw an alarm through its *alarm* interface. This notification must be logged and produce some action in order to be useful. A simple way to handle it is to insert a *Planning* component like that described in Section IV-E that implements a *planner* whose only action is to send a notification email about the faulty condition. This simple activity is described in Figure 21.

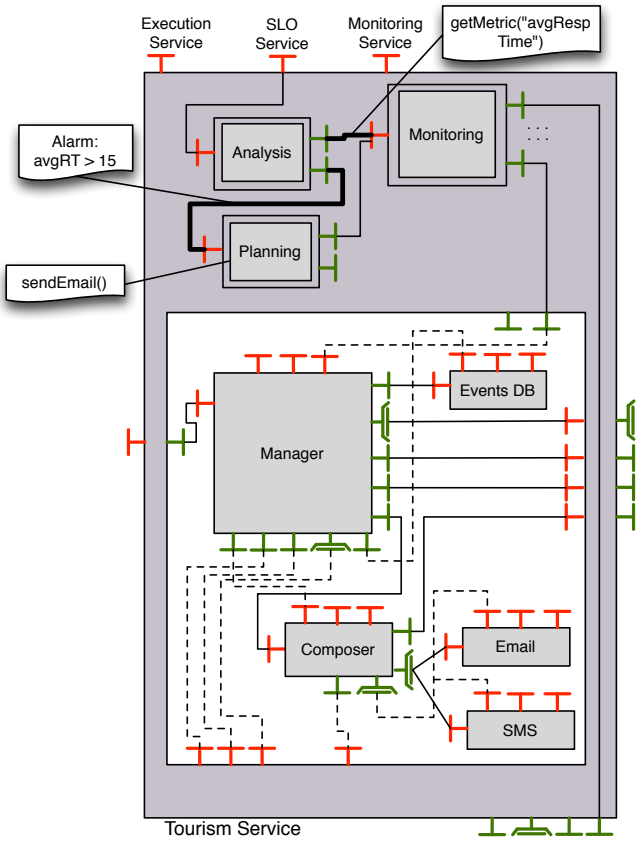


Figure 21. *TourismService* with *Monitoring*, *Analysis*, and simple *Planning* inserted. The basic action is to check a metric and notify in case a threshold is reached. The complete set of monitoring bindings is not shown for clarity.

3) Providing a self-optimizing autonomic loop: At this moment the autonomic control loop is not complete, as the final action is still dependent on a human administrator. In order to provide a complete autonomic behaviour, a more complex *planner* can be added to the *Planning* component and associated to the SLO that checks the *avgRespTime* metric, and an *Execution* component must be inserted in each component where an action may be carried on.

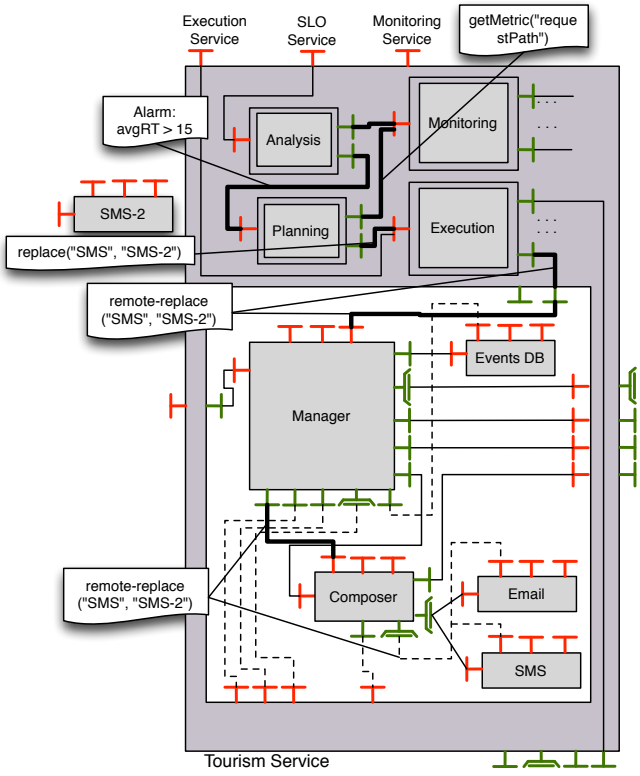


Figure 22. *TourismService* with all MAPE components, providing a self-optimizing behaviour. If the average response time is not met, the slowest component is identified and replace by an equivalent one. The complete set of monitoring bindings is not shown for clarity.

A simple self-optimizing autonomic behaviour may consist of reacting when the desired average response time is not obtained in the *frontEnd* interface, and replacing the component that takes the most time to execute by an equivalent quicker component.

To implement this kind of action, the new *planner* component must implement a behaviour slightly more complex that the old component. The *planner* first needs to identify the "faulty" component, which in this case is defined as the one that takes the biggest slice of the total time to serve a request. This information is obtained from the *requestPath* metric that can be obtained from the *Monitoring* component. Once the component to be replaced is identified, the *planner* must find a proper replacement. The discovery process is not shown in the example, however we assume that an alternative, more efficient component can be found (if that is not possible, the *planner* can safely fail without producing an action). Finally, the replacement action must be carried on in the appropriate binding. By using the connections between the *Execution* components, the action can be propagated and the binding can be updated. The sequence of the propagation of actions, and the application with all the MAPE components inserted is shown in Figure 22.

This particular implementation is a concrete implementation of an effective autonomic self-optimizing behaviour built through our framework and dynamically inserted in a running application.

4) Providing a self-healing behaviour based on infrastructure: As we have mentioned before, the implementation of sensors and actuators are the only parts of our framework that are heavily dependent on the particular implementation. In the previous example, we have relied in sensors that detect JMX events produced by the GCM/ProActive implementation of the functional code, and actuators that rely on the PAGCMScript scripting language to describe reconfiguration actions.

A different implementation of sensors can be oriented to measure characteristics of the running infrastructure like CPU or memory utilization, by using operating system calls, or communicating with a virtual machine manager. Once these sensors are implemented, their values can be fetched by the *MetricsStore* and they are available for the rest of the components of the framework as any other metric value. These kind of sensors are particularly useful in a Cloud computing environment, where the introduction of autonomic behaviour in the application seems like a promising way to benefit of the elasticity of the running infrastructure.

Infrastructure-based sensors may be used to provide a simple self-healing behaviour in which a metric called *avgLoad* is used to determine the average load of the node where a component is runnning. In case the load surpasses a threshold, a *planner* is activated, which determines the node with the highest load, and migrates one component from that node to another newly acquired node, expecting to achieve a better balance.

Figure 23 shows an example of this behaviour.

5) Integrating adaptation on a cloud infrastructure: We have also integrated the infrastructure monitoring capability of our framework to provide adaptation through the lifecycle of an SOA-based application running on a cloud environment [17].

Figure 24 shows a simplified version of the *TourismService* application where the *Composer* component is duplicated and each component is located in a different node of a cloud infrastructure. The integration of our monitoring capabilities through our framework allows the collection of information both from the infrastructure sensors, and from the runtime levels, and made it available at a higher level view.

From the unified view, it is possible to interact through the *Execution* interfaces and introduce modifications both at the component runtime architecture level as we showed on Figure 22, or by acquiring new nodes from a cloud infrastructure and migrate a component to that node, in order to balance the load of the application. Such example is shown on Figure 25, where component $C2$ is migrated from node C to a newly acquired node D.

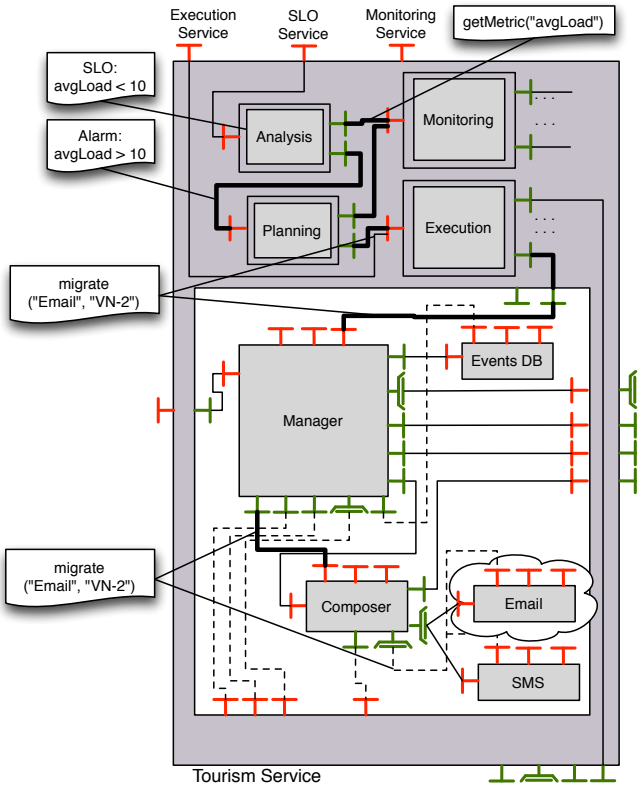


Figure 23. *TourismService* with all MAPE components, providing a self-healing behaviour using sensor over the infrastructure. When a component runs in a node that exhibits a high load. One component is migrated to a newly acquired node, illustrated as a cloud provided node. The complete set of monitoring bindings is not shown for clarity.

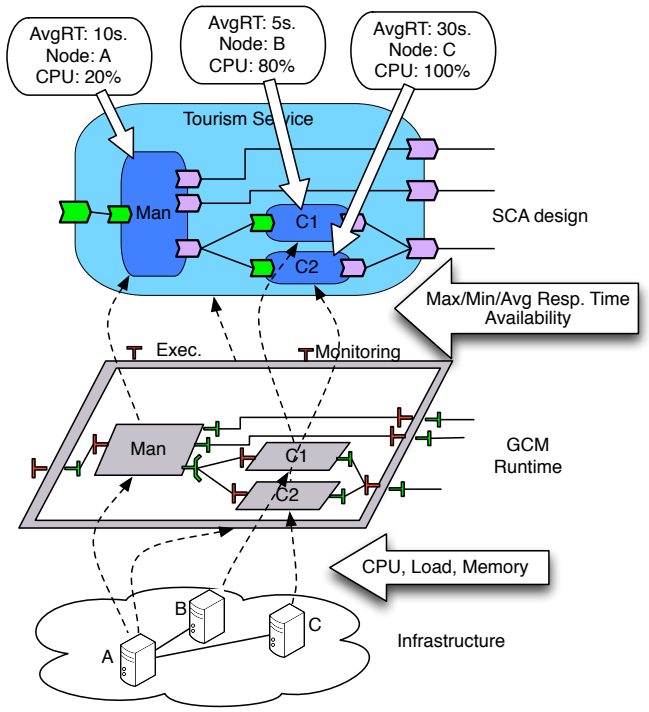


Figure 24. *TourismService* with Monitoring and Execution interfaces. The GCM implementation allows to retrieve information from the infrastructure and the runtime middleware, and associate it to the SCA design.

VI. Related Work

Several works exist regarding monitoring and management of service-oriented applications and about the implementation of autonomic control loops.

A set of works tackle the implementation of each phase mostly in a separate way. We can find infrastructures for monitoring components and services [18], [19], [20], and tools for monitoring grid and cloud infrastructures [21], [22], [23]. The work of Comuzzi et al. [24] proposes a hierarchical monitoring of SLAs with support for event-based communication, pull/push modes and different kinds of metrics. The monitoring requirements are tightly coupled to the services and accessed through a common interface. Their approach differs with ours in that they do not consider the modification of the monitoring requirements, or even SLAs at runtime (nor do they consider components and possible associated hierarchy as we do, in order to ease monitoring information aggregation).

Regarding the Analysis phase, several works integrate SLA monitoring and analysis [25] with SLA fulfillment [8], [26]. For representing the conditions to verify, several languages have been proposed [4] like SLAng [5], WSLA [6] and WS-Policy [7], which are mostly oriented to specify the agreement conditions between providers and consumers. Our claim is that our component based approach allows the integration of one of these languages, specifically in the *SLO Analyzer* shown in Section IV-D to represent the conditions.

On the area of planning strategies for adaptation, several planning algorithms can be found using different techniques. Some of them try to solve the problem of dynamically selecting a set of services that accomplish some determined QoS characteristic [27] using techniques from the genetic algorithms area [28], [29] or using linear and integer programming [30]. Other common way to separate the workflow composition from the selection of services is to rely on *abstract services* with some optionally defined QoS constraints, and bind them to proxies or *brokers* that are in charge of collecting information from a set of *candidate services* and performing the selection to bind *concrete services* to them [31], [32], [33]. Those works intend to compose a service, previous to execution, that complies with the required QoS characteristics. On the other side, other works address the problem of dynamically adjusting a composition at runtime [34], which is closer to the autonomic control loop that we provide, although it makes encapsulation harder as they require a closer integration between the monitoring and analysis phases with the planning phase. The runtime nature of these approaches imposes restrictions on the time spent for

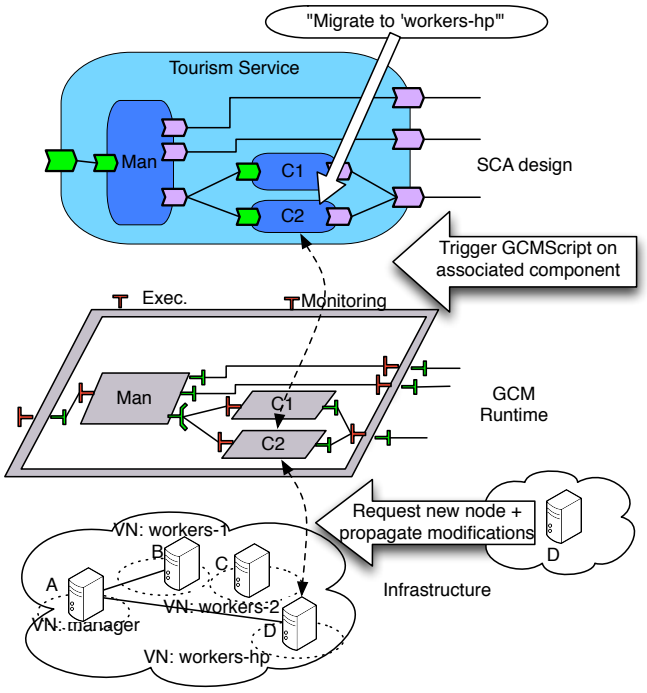


Figure 25. By sending commands through the *Execution* interface, reconfigurations can be enacted both at the runtime level, as well as in the infrastructure level.

computing the necessary rebinding. Some heuristics include K-means clustering of candidate services [35], and filtering of services that combine local and global optimizations [36] and skyline selections [37].

Regarding the *Execution* phase, recent component systems have been designed to take into account support for executing reconfigurations. Among them, works like FraSCAti [15] and SAFRAN [38] include methods for dynamically modifying the composition of an application. FScript [16] is a scripting language closely related to Fractal [12] based applications to describe such reconfigurations, and is the base for our own scripting language PAGCMScript.

Our approach, however, provides support for dynamically building complete autonomic control loops through a meaningful integration of the previous phases. The existing works that provide complete frameworks for the MAPE loop include Rainbow, and architecture-based approach providing a single autonomic control loop [39] that uses a model of the managed architecture to analyze and generate adaptations, which are later mapped to the effective system using a set of sensors and actuators. Another similar work to ours [40] proposes a generic context-aware framework that separates the steps of the MAPE control loop to provide self-adaptation; their work allows the implementation of self-adaptive strategies, though not much is mentioned about runtime reconfigurability, or the possibility to have multiple strategies. Also, we do not necessarily consider that all services require the same level of autonomicity.

CEYLON [41] is a service-oriented framework for integrating autonomic strategies available as services and using them to build complex autonomic applications. They provide the managers that allow the integration and adaptation of the composition of the autonomic strategies according to evolving conditions. In CEYLON, autonomicity is a main *functional* objective in the development of the application, while in our case, we aim to provide autonomic QoS-related capabilities to already existing service based applications. Also, we take benefit of the business-level components intrinsic distribution and hierarchy to split the implementation of monitoring and management requirements across different levels, thus enforcing scalability.

VII. Conclusion and Perspectives

We have presented a generic component-based framework for supporting monitoring and management tasks of component-based SOA applications.

The strengths of our approach include a clear separation of concerns between the functional content and the management tasks, relieving the programmer of the functional application to integrate the managment activities. The framework is generic in the sense that most of its components can be implemented in an independent way from the supporting technology of the application. The necessary implementation-dependent elements, such as sensors and actuators are encapsulated in components, and made available through a common interface to the rest of the framework. Finally we provide two levels of flexibility as we can dynamically insert or remove sensors, conditions, planning strategies and actuators in a previously existent skeleton that provides the autonomic control loop; and we also allow the modification of the composition of the control loop by including phases like analysis and planning only when they are needed and providing different degrees of autonomicity to each component.

We have provided an implementation of our framework as a self-adaptation loop for component-based services, thanks to the composition of appropriate monitoring, SLA management, planning and reconfiguration components. This prototype has been developed in the context of an SCA compliant platform that includes dynamic reconfiguration and distribution capabilities.

This approach provides a high degree of flexibility as the skeleton we have provided for the autonomic control loop can be personalized to support, for example, different planning strategies, and leverage heterogeneous monitoring sources to provide the input data that these strategies may need (for example, performance, price, energy consumption, availability).

One point not targeted by our proposition is the problem of conflict resolution. Indeed we may think about two kinds

of conflicts: one when two or more different planners generate opposite actions, or actions that invalidate each other; and the other situation where the result of an action triggers a chain of autonomic reactions that does not converge to a stable state resulting in a *livelock* situation. Both types of conflicts must be eventually dealt with, and they may arise as a consequence of the fact that conditions are inserted in the system in a way that they may be unaware of each other. For that matter we can consider an additional component that collects the output of each planner involved and that is capable of resolving these kind of conflicts inside the planning component. The specific implementation of a conflict resolution mechanism is not a concern of this work. Nevertheless, its integration is a promising perspective that goes in the direction of improving the autonomic capabilities that can be added to an application.

REFERENCES

[1] C. Ruz, F. Baude, and B. Sauvan, "Flexible Adaptation Loop for Component-based SOA applications," in *IARIA 7th International Conference on Autonomic and Autonomous Systems (ICAS 2011)*, 2011, pp. 29–36.

[2] J. O. Kephart and D. M. Chess, "The Vision of Autonomic Computing," *IEEE Computer*, vol. 36, no. 1, 2003.

[3] (2007, Mar.) Service Component Architecture Specifications. OASIS. Last accessed: April 2011. [Online]. Available: http://oasis-opencsa.org/sca

[4] A. Sahai, V. Machiraju, M. Sayal, A. van Moorsel, and F. Casati, "Automated SLA Monitoring for Web Services," in *Management Technologies for E-Commerce and E-Business Applications*, ser. Lecture Notes in Computer Science, M. Feridun, P. Kropf, and G. Babin, Eds. Springer Berlin / Heidelberg, 2002, vol. 2506, pp. 28–41.

[5] J. Skene, A. Skene, J. Crampton, and W. Emmerich, "The Monitorability of Service-Level Agreements for Application-Service Provision," in *Proceedings of the 6th international workshop on Software and performance*, ser. WOSP '07. New York, NY, USA: ACM, 2007, pp. 3–14.

[6] A. Keller and H. Ludwig, "The WSLA Framework: Specifying and Monitoring Service Level Agreements for Web Services," *Journal of Network and Systems Management*, vol. 11, pp. 57–81, 2003.

[7] (2007, Sep.) Web Services Policy 1.5 - Framework (WS-Policy). W3C. Last accessed: June 2012. [Online]. Available: http://www.w3.org/TR/ws-policy/

[8] P. Leitner, B. Wetzstein, F. Rosenberg, A. Michlmayr, S. Dustdar, and F. Leymann, "Runtime prediction of service level agreement violations for composite services," in *Proceedings of the 2009 international conference on Service-oriented computing*, ser. ICSOC/ServiceWave'09. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 176–186.

[9] ProActive Parallel Suite. Last accessed: June 2012. [Online]. Available: http://proactive.inria.fr/

[10] F. Baude, V. Contes, and V. Lestideau, "Large-Scale Service Deployment–Application to OSGi," in *IARIA 3rd International Conference on Autonomic and Autonomous Services (ICAS 2007)*. IEEE Computer Society Press, Jun. 2007, pp. 19–26.

[11] F. Baude, D. Caromel, C. Dalmasso, M. Danelutto, V. Getov, L. Henrio, and C. Prez, "GCM: a grid extension to Fractal for autonomous distributed components," *Annals of Telecommunications*, vol. 64, pp. 5–24, 2009.

[12] E. Bruneton, T. Coupaye, M. Leclercq, V. Quma, and J.-B. Stefani, "The FRACTAL component model and its support in Java," *Software: Practice and Experience*, vol. 36, no. 11-12, pp. 1257–1284, 2006.

[13] F. Baude, L. Henrio, and P. Naoumenko, "Structural Reconfiguration: An Autonomic Strategy for GCM Components," in *IARIA 5th International Conference on Autonomic and Autonomous Systems (ICAS 2009)*. IEEE Computer Society, 2009, pp. 123–128.

[14] M. Aldinucci, S. Campa, P. Ciullo, M. Coppola, M. Danelutto, P. Pesciullesi, R. Ravazzolo, M. Torquati, M. Vanneschi, and C. Zoccolo, "A framework for experimenting with structured parallel programming environment design," in *Parallel Computing - Software Technology, Algorithms, Architectures and Applications*, ser. Advances in Parallel Computing. North-Holland, 2004, vol. 13, pp. 617 – 624.

[15] L. Seinturier, P. Merle, D. Fournier, N. Dolet, V. Schiavoni, and J.-B. Stefani, "Reconfigurable SCA Applications with the FraSCAti Platform," in *Proceedings of the 2009 IEEE International Conference on Services Computing*, ser. SCC'09. IEEE Computer Society, 2009, pp. 268–275.

[16] P.-C. David, T. Ledoux, M. Léger, and T. Coupaye, "FPath and FScript: Language support for navigation and reliable reconfiguration of Fractal architectures," *Annals of Telecommunications*, vol. 64, pp. 45–63, 2009.

[17] C. Ruz, F. Baude, B. Sauvan, A. Mos, and A. Boulze, "Flexible SOA Lifecycle on the Cloud Using SCA," in *Proceedings of the 2011 IEEE 15th International Enterprise Distributed Object Computing Conference Workshops*, ser. EDOCW'11. IEEE Computer Society, 2011, pp. 275–282.

[18] A. Van Hoorn, M. Rohr, W. Hasselbring, J. Waller, J. Ehlers, S. Frey, and D. Kieselhorst, "Continuous Monitoring of Software Services: Design and Application of the Kieker Framework," 2009, last accessed: June 2012. [Online]. Available: http://www.informatik.uni-kiel.de/uploads/tx_publication/vanhoorn_tr0921.pdf

[19] I. Garcia, G. Pedraza, B. Debbabi, P. Lalanda, and C. Hamon, "Towards a Service Mediation Framework for Dynamic Applications," *2006 IEEE Asia-Pacific Conference on Services Computing*, pp. 3–10, 2010.

[20] P.-C. David and T. Ledoux, "Wildcat: a generic framework for context-aware applications," in *Proceedings of the 3rd international workshop on Middleware for pervasive and ad-hoc computing*, ser. MPAC'05. ACM, 2005, pp. 1–7.

[21] M. L. Massie, B. N. Chun, and D. E. Culler, "The ganglia distributed monitoring system: design, implementation, and experience," *Parallel Computing*, vol. 30, no. 7, pp. 817 – 840, 2004.

[22] Hyperic. CloudStatus Monitoring. Last accessed: June 2012. [Online]. Available: http://www.hyperic.com/products/cloud-status-monitoring

[23] LogicMonitor. Last accessed: June 2012. [Online]. Available: http://www.logicmonitor.com/

[24] M. Comuzzi and G. Spanoudakis, "A Framework for Hierarchical and Recursive Monitoring of Service Based Systems," in *4th International Conference on Internet and Web Applications and Services, 2009. (ICIW'09)*, may 2009, pp. 383 –388.

[25] A. Michlmayr, F. Rosenberg, P. Leitner, and S. Dustdar, "End-to-End Support for QoS-Aware Service Selection, Binding, and Mediation in VRESCo," *IEEE Transactions on Services Computing*, vol. 3, pp. 193–205, 2010.

[26] I. Foster, "Globus toolkit version 4: Software for service-oriented systems," *Journal of Computer Science and Technology*, vol. 21, pp. 513–520, 2006.

[27] C. Ghezzi, A. Motta, V. Panzica La Manna, and G. Tamburrelli, "QoS Driven Dynamic Binding in-the-many," in *Research into Practice – Reality and Gaps*, ser. Lecture Notes in Computer Science, G. Heineman, J. Kofron, and F. Plasil, Eds. Springer Berlin / Heidelberg, 2010, vol. 6093, pp. 68–83.

[28] S. Liu, Y. Liu, N. Jing, G. Tang, and Y. Tang, "A Dynamic Web Service Selection Strategy with QoS Global Optimization Based on Multi-objective Genetic Algorithm," in *Grid and Cooperative Computing – GCC 2005*, ser. Lecture Notes in Computer Science, H. Zhuge and G. Fox, Eds. Springer Berlin / Heidelberg, 2005, vol. 3795, pp. 84–89.

[29] G. Canfora, M. Di Penta, R. Esposito, and M. L. Villani, "A framework for QoS-aware binding and re-binding of composite web services," *J. Syst. Softw.*, vol. 81, pp. 1754–1769, Oct. 2008.

[30] L. Zeng, B. Benatallah, A. H.H. Ngu, M. Dumas, J. Kalagnanam, and H. Chang, "QoS-Aware Middleware for Web Services Composition," *IEEE Trans. Softw. Eng.*, vol. 30, pp. 311–327, May 2004.

[31] T. Yu and K.-J. Lin, "A broker-based framework for QoS-aware Web service composition," in *Proceedings of the 2005 IEEE International Conference on e-Technology, e-Commerce and e-Service*, ser. EEE '05. IEEE Computer Society, 2005, pp. 22–29.

[32] D. A. D'Mello, V. Ananthanarayana, and S. Thilagam, "A QoS Broker Based Architecture for Dynamic Web Service Selection," *2nd Asia International Conference on Modelling & Simulation*, vol. 0, pp. 101–106, 2008.

[33] M. Serhani, R. Dssouli, A. Hafid, and H. Sahraoui, "A QoS broker Based Architecture for Efficient Web Services Selection," in *Proceedings of the 2005 IEEE International Conference on Web Services (ICWS 2005)*, 2005, pp. 113 – 120 vol.1.

[34] G. Canfora, M. Di Penta, R. Esposito, and M. L. Villani, "QoS-aware replanning of composite Web services," in *Proceedings of the 2005 IEEE International Conference on Web Services*, ser. ICWS '05. IEEE Computer Society, 2005, pp. 121–129.

[35] N. B. Mabrouk, S. Beauche, E. Kuznetsova, N. Georgantas, and V. Issarny, "QoS-aware service composition in dynamic service oriented environments," in *Proceedings of the ACM/IFIP/USENIX 10th international conference on Middleware*, ser. Middleware'09. Springer-Verlag, 2009, pp. 123–142.

[36] M. Alrifai and T. Risse, "Combining global optimization with local selection for efficient QoS-aware service composition," in *Proceedings of the 18th International Conference on World Wide Web*, ser. WWW'09. ACM, 2009, pp. 881–890.

[37] M. Alrifai, D. Skoutas, and T. Risse, "Selecting skyline services for QoS-based web service composition," in *Proceedings of the 19th International Conference on World Wide Web*, ser. WWW '10. ACM, 2010, pp. 11–20.

[38] P.-C. David and T. Ledoux, "An Aspect-Oriented Approach for Developing Self-Adaptive Fractal Components," in *Software Composition*, ser. Lecture Notes in Computer Science, W. Löwe and M. Südholt, Eds. Springer Berlin / Heidelberg, 2006, vol. 4089, pp. 82–97.

[39] D. Garlan, S.-W. Cheng, A.-C. Huang, B. Schmerl, and P. Steenkiste, "Rainbow: Architecture-Based Self-Adaptation with Reusable Infrastructure," *IEEE Computer*, vol. 37, pp. 46–54, 2004.

[40] F. André, E. Daubert, and G. Gauvrit, "Towards a Generic Context-Aware Framework for Self-Adaptation of Service-Oriented Architectures," *5th International Conference on Internet and Web Applications and Services (ICIW'10)*, vol. 0, pp. 309–314, 2010.

[41] Y. Maurel, A. Diaconescu, and P. Lalanda, "CEYLON: A Service-Oriented Framework for Building Autonomic Managers," *7th IEEE International Conference and Workshops on Engineering of Autonomic and Autonomous Systems*, pp. 3–11, Mar. 2010.

Metamodel and Formal Logic based Methodology for Modeling, Refining and Verifying Reconfigurable Networked Component Systems

Gabor Batori
Software Engineering Group
Ericsson Hungary
Email: gabor.batori@ericsson.com

Zoltan Theisz
evopro Informatics and Automation Ltd.
Email: zoltan.theisz@evopro.hu

Domonkos Asztalos
Software Engineering Group,
Ericsson Hungary
Email: domonkos.asztalos@ericsson.com

***Abstract*—Reconfigurable networked systems have often been developed via dynamically deployed software components that are executing on top of interconnected heterogenous hardware nodes. The challenges resulting from the complexity of those systems have been traditionally mitigated by individual ad-hoc problem solutions and industrial best practices guidelines tuned to the particular domain specific modeling frameworks and methodologies. Targeting this deficiency, this paper disseminates an alternative, semi-formal methodology that incorporates a first-order logic based structural modeling language, Alloy, in the analysis of component deployment and reconfiguration. This novel approach could help to extend the limits of the generic domain specific metamodeling methodology that has been developed for creating Reconfigurable Ubiquitous Networked Embedded Systems.**

***Keywords*-Alloy; formal model semantics; metamodeling; dynamic component system; platform middleware; RUNES; Erlang; ErlCOM**

I. INTRODUCTION

Reconfigurable networked component systems provide a versatile implementation framework for highly distributed autonomic peer-to-peer applications targeting the domains of sensor networks and autonomous computing environments. The introduction of an effective, high-quality software development methodology, that speeds up the day-to-day tasks of application developers in such an inherently complex environment can be regarded as a rather valuable asset. In fact, the Reconfigurable Ubiquitous Network Embedded Systems (RUNES) IST project successfully completed this endeavor by providing a common distributed component-based platform architecture, on top of heterogeneous networks of computational nodes, and by establishing a corresponding model-based software development methodology and related framework implementation. Nevertheless, the practical building and later validation and verification of such networked component applications turned out to be quite an ambitious technical challenge, which almost always required detailed software engineering know-how that went beyond the usual precise understanding of the problem domain. Hence, we think that practical application development projects may enormously benefit from this beyond state-of-the-art domain specific modeling technique, which also includes some novel, formal logic based practical approaches. Although some of the early results have been reported in [1] the final validation of this methodology via practical application scenarios is still open for further investigation.

One of the major results of the RUNES [2] project was to establish a reflective distributed component-based multi-platform middleware architecture [3] for heterogeneous networks of computational nodes, including metamodel-based software development methodology [4] and graphical development framework. The RUNES metamodel provides all those relevant concepts that software developers must need to know in order to efficiently utilize the computational resources of a reflective distributed component-based environment. The complexity of these distributed reconfigurable component systems is due to the fact that the reflective components can be linked only by compatible provided-required interface pairs and their communication must be served either by these bound links, via pure message sending, or by a temporal storage of (meta-)data located in a distributed database. In the beginning of the RUNES project, only state-of-the art domain specific modeling techniques had been applied, however, later we had to realize that the usage of formal logic based language support, e.g. Alloy, could be taken advantage of in order to go beyond the traditional validation and verification approaches of state-of-the-art model based design methodologies. Therefore, we started experimenting with semi-automated domain specific model analysis techniques in Alloy that can be used to formally handle the evolution of some dynamic component behaviors in certain families of application domains for domain specific verification. As a contribution, this paper extends [1] by putting it into the context of our continuous-life-cycle model based development methodology and in this way also formalizes the relations between the metamodeling, platform and validation and verification part of it. Moreover, we firmly believe that through its practical applicability this methodology can contribute to the better automation of some modeling tasks by eliminating non-trivial dynamic errors or failure situations during the application design of reconfigurable component systems.

The paper is structured as follows: In Section I, the motivation for this research is introduced. In Section II, related works on Alloy usage for system verification and validation are presented. Then, Section III provides a background on the technical domain of reconfigurable networked component systems and the logical formalism of Alloy, which establishes the conceptual frame for the rest of the paper. Next, Section IV describes the methodology and its associated formal and semi-formal methods to designing, refining and verifying the components of reconfigurable systems. A case-study showcasing the usefulness of this methodology, focusing mostly on the application of Alloy in the case of a simplified scenario example, is presented in Section V. Finally, in Section VI, the conclusions and some insights into our future research are provided.

II. Related work

Distributed reconfigurable component systems are inherently complex to analyze, hence, the importance of formal description techniques in system design is well known in the scientific literature. In particular, in this paper, we restrict our work on the usage of Alloy [7] to target only practical scenarios where the model checking capability of a refuter seems to be powerful enough to assist the application developers. So, our methodology, though, reliant on a formal first order logic based description language, which is supported by a fully automated SAT solver based analyzer, it is still some way constrained when it comes to model complexity and scalability. However, we know from related scientific publications that similar formal description techniques of Alloy have been successfully applied to model various complex systems in a wide range of application domains for domain specific model verification purposes. It has been applied in [11] for the analysis of some critical correctness properties that should be satisfied by any secure multicast protocol. The idea of applying Alloy for component based system analysis was suggested also by Warren et al. [12]. That paper describes OpenRec, a framework, which comprises a reflective component model, and then its Alloy model is investigated in some details. This Alloy model served as a conceptual basis for our Alloy component model; however, our model is more detailed, which enables deeper analysis of system behavior. Moreover, [13] demonstrates another Alloy model that identifies various types of dynamic system reconfigurations. It provides a rather good categorization of various problems and corresponding solutions related to dynamic software evolution. Furthermore, Aydal et al. [14] found Alloy Analyzer one of the best analysis tools for state-based modeling languages.

Although individual application scenarios can be easily expressed manually in Alloy we firmly believe that the synergy between metamodel driven design and first order logic based practical model verification could result in a more advantageous unified approach. This approach, in a nutshell, semi-automatically generates all relevant RUNES deployment configuration assets that will also be analyzed within Alloy. In effect, by analyzing a significant subset of frequently reoccurring configurations the boundary between valid and invalid component configurations can be better investigated against proper sets of model-based application and/or middleware feasibility constraints. The analysis results can be later reused for providing useful inputs to the run-time adaptive control logic in order to extend the model-based software development framework [4] with effective autonomicity.

In the rest of this paper, we will describe how this first-order logic based model of the RUNES middleware has been developed in Alloy and how it has been integrated into the RUNES domain specific modeling framework and methodology [4].

III. Background

A. Networked Reconfigurable Dynamic Component System

The aim of any networked reconfigurable component systems is to hide the heterogeneity of the participating nodes from the view of the application. The RUNES architecture consists of a reflective reconfigurable component system and a corresponding Component Run-Time Kernel (CRTK). This means that the reflectivity of the CRTK manifests in the reifiability of all kernel elements via an explicit management interface, and the concepts of a component system lies in the heart of its implementation that complies with well-known component-based software engineering principles. In more details, the reflective components are linked together by their interfaces, they communicate via message sending and store their meta-data in a distributed database. Each computational node incorporates an instance of the CRTK, which provides the basic middleware APIs of component management. These architectural concepts were turned into an effective reference implementation, called ErlCOM [5], which runs on top of the Erlang/OTP distributed infrastructure [6]. ErlCOM being a full-fledged realization of the RUNES CRTK, the RUNES component system will now be described through ErlCOM terms.

A component is the basic unit of the system that corresponds to an active actor-like process, which contains some executable code and has a unique name that is registered in a global registry. The components are spread over caplet hierarchies, caplets being components themselves, in a pool of networked nodes. The root of the caplet hierarchy is called capsule, which is the main process entity of the node. The caplets' main purpose is to provide supervisory facilities for the maintenance of robustness and longevity of the whole component system. The supervisory decisions are taken according to a set of predefined constraints stored within a particular component framework. Examples of robust auto-configuration can be the reactivation of crashed components or the migration of a cluster of running components due

to e.g. load balancing. The main interaction between components is carried out by means of pure message passing through the bindings, which represent the behavioral policy on the communication channels. The bindings themselves are also components with special communication properties. Message passing is synchronous; messages can be intercepted both before entering the interfaces of the recipients and after the replies have been exited those same interfaces. The pre- and post-actions of the bindings constitute a list of additional transformations on individual messages. It is important to emphasize that by the introduction of the binding concept both concurrent code execution inside the components and individual message passing activities through those bindings can be reified and reasoned on via a reflective component configuration graph provided by the middleware. Bindings are created when a receptacle, that is, a required interface, of a particular component is to be bound to a provided interface of another component, provided that they have been found compatible. Finally, both the components and the bindings are facilitated with explicitly attached state information, which may also be associated with some additional metadata stored in a global repository, redundantly distributed over the meshed networked nodes.

The concepts and the specificities of the RUNES component system are specified on various levels of technical details and also from different perspectives. Firstly, the constituting concepts with their corresponding static and dynamic constraints are formalized within domain specific metamodels [4]. Secondly, the dynamics and fine-grained functional and operational details of the ErlCOM reference implementation of the RUNES CRTK have been specified both in Message Sequence Charts (MSC) and related Erlang source code snippets [5]. Finally, a conceptually though simplified, but semantically compatible formal logic based representation of core CRTK elements and operations have been defined in [1].

B. Alloy

For precise validation and verification of application models logic based tools provide exact, though sometimes theoretically complex and practically limited, answers to some of the most important configuration or dimensioning questions. Under validation we mean here the semantic compatibility of the designed system, only from the perspectives of configuration and dimensioning aspects, against the semi-formal and/or verbal specification of given use case scenarios. Verification has also a slightly limited scope in our interpretation since we rather rely on a refuter than a theorem prover in order to gain in practical applicability. Nevertheless, in the particular case of dynamic component systems deployed in the domains of sensor networks, we believe that the theoretical prowess and the practical applicability of some first order logic based techniques can be though efficiently merged for effective applicability. In this paper, our selected choice of formal logic description is based on Alloy [7], which is a textual modeling language relying on structured first-order relational logic with equality. Although other temporal logic based techniques such as Linear Temporal Logic (LTL) or Computational Tree Logic (CTL) constructs could have been applied, instead of Alloy's formalism, to our domain of investigation, we do think that Alloy's syntax lies closer to the spirits of current state-of-the-art programming languages and therefore it is way easier for the practical program developers to use it or understand generated Alloy expressions without having to delve into theoretically precise definitions of its constructs. However, by not directly relying on a temporal logic based model checker such as UPPAAL [8] or SPIN [9] we were, obviously, forced to recreate the temporal aspect of the evolution of component configurations as we reported in [1] and as it will be described more in details in IV-C of this paper. In general, Alloy's syntax is rather simple; a particular model in Alloy contains a set of signature definitions with fields, facts, functions and predicates. Each signature denotes a set of atoms, which represent the smallest building blocks of the language. Atoms are, per definition, immutable and uninterpreted. Each field must belong to a signature and represents a relation with some other signatures. Facts define constraints on other elements of the model. Functions serve as named containments of Alloy definitions and predicates are considered like parameterized constraints that can be invoked within facts, functions or other predicates. Alloy is supported by a fully automated constraint solver, called Alloy Analyzer [10], which can be used to verify model parameters by searching for either valid or invalid instances of the model. Model checking is achieved by automated translation of the model into a Boolean expression, which is analyzed by SAT solver plug-ins, which can be easily incorporated into Alloy Analyzer. Once an instance violating an assertion has been found within the defined scope of a particular analysis task, the result of the verification is declared as not valid. However, if no instance has been found, it is not, in any means, a proof that the assertion is valid, though in practical applications, it could be considered as such, though it still might be invalid within a larger scope. This non-monotonic behavior of the prover may be disturbing in theory, but it works quite well in practical cases since the most relevant errors with practical significance occur in small, though non-trivial sized models in Alloy. Thus, the selection of the proper scope is an important trade-off of Alloy modeling and it should be carried out as precisely as possible within the constraints of practicality.

IV. Methodology

A. Process

All kinds of professional software developments are usually accompanied by some development processes that

safeguard industrial scale applicability of the chosen technology. Although there are many well-established and widely-used model based software development approaches, e.g. Rational Unified Process, that significantly influenced our work, the ambition level of our process design aimed at covering all the stages of component based application development, including generative metamodeling technologies. The overview of the process stages are depicted in Figure 1. Figure 1 is layered into five stages; namely, Scenario, Application Model, Platform Model, Code Repository and Running System. The arrows of the non-iterative part of the process, connecting together the artifacts of the various stages, are labeled by sequence numbers in accordance to their timing. In this paper, we only briefly outline the process by mainly concentrating on the inter-work between the major elements of the stages.

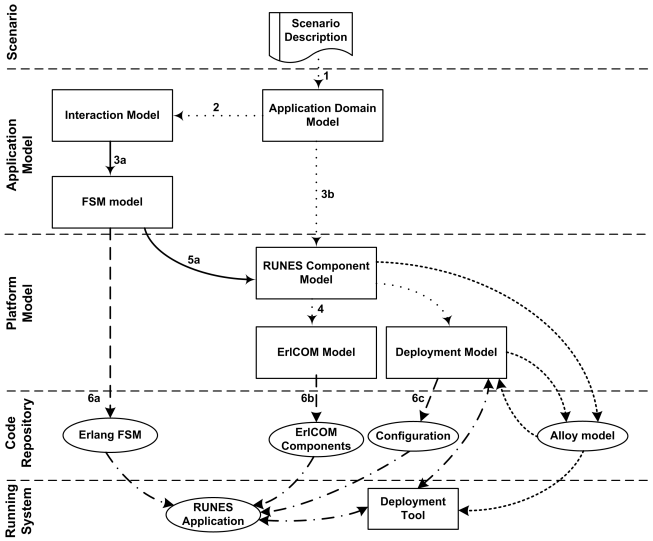


Figure 1. Software Development Process extended with Alloy verification

The Scenario evaluates and finalizes a set of scenario descriptions that establishes the exact scope of the application domain. Our experience gathered during the RUNES project showed clearly that reconfigurable component based applications can only be successfully developed if the application usage scenarios are detailed enough to enable non-trivial application modeling and quality analysis. Because realistic distributed applications involve intense interactions among application components both structural and interaction modeling are equally important. The Application Domain Model is created to cover the scenario in such a way that all use case details must be taken adequately into account and the stakeholders' roles have to be discovered, too.

The roles make up the basic elements of the interaction model, hence the dynamicity of the use cases must be translated into corresponding Message Sequence Charts (MSC). The Application Domain Model and the Interaction Model must be detailed enough so that quality investigations could be carried out in order to check the feasibility of the design. Moreover, this stage involves many creative decisions, so both arrow 1 and 2 in Figure 1 are dotted, this way showing that the activity is mainly carried out manually.

The Interaction Model is transformed into the Finite State Machines (FSM) Model and then a further translation maps it onto the RUNES Component Model. The solid arrow indicates that the translation is executed via graph transformations. The Application Domain Model usually requires creative refinements and only semi-automatically (see dotted line) can get translated onto the RUNES Component Model.

The Platform Model stage has been conceived to support total semantics elaboration, that is, the RUNES Component Model is extended by the semantics of the platform, the components and the FSMs. This step involves some manual coding in Erlang in order to produce a total executable specification of the application.

The final application model takes into consideration the distributed nature of the application; hence, the Deployment Model is populated. It entirely specifies the total component allocation of the application over the available nodes of the network.

The Code Repository is the stage which copes with source code management. The code production is fully automated, which is indicated by dashed lines. The Deployment Model is translated into an initial run-time configuration which is deployed over the available ErlCOM nodes. Any changes of the component configuration at run-time are managed by the Deployment Tool, which continuously updates the Deployment Model.

The Alloy based model verification step extends the standard operation of the Deployment Tool. It contains two additional model transformations; one that originates from the RUNES Component Model and another that takes a compatible RUNES Deployment Model and turns them into a configuration scenario that can be verified within Alloy Analyzer. The model transformations produce configuration scenarios, which include both the structural and the behavioral specifications of the application. However, only those parts of the FSM action semantics are kept from the total dynamic behavior that either directly relate to important control logic elements of the scenario or which belong to the operations provided by the underlying ErlCOM middleware. These steps simplify, though, precisely specify when and with which parameters the application invokes the CRTK of the RUNES middleware. Therefore, the verification of a particular scenario investigates mainly the evolution of the application from the point of view of its component reconfigurations that are allowed by the semantics of the ErlCOM middleware. More precisely, in our work we mostly targeted resource availability investigations over distributed capsules along the lifetime of the application. The results of this verification step provide useful input to the run-time autonomic control mechanisms either embedded inside the

application or defined as explicit rule-sets within autonomic extensions of the Deployment Tool, basically managing pre-calculated adaptive component reconfiguration. The verification step is rather iterative in nature, which is well supported by Alloy Analyzer, and thus the final convergence criteria are mostly decided on a case-by-case basis depending on the particular scenario.

B. RUNES Metamodel

1) Interaction Modeling: Large-scale networked systems can be efficiently comprehended as a large number of interacting services. By combining those services an entity is getting involved in the complete behavior specification for that entity is established. Therefore, the service concept is effectively based on the interaction patterns between the cooperating entities. The notion of a role describes the contribution of an entity within a given interaction pattern. In our work we followed a well-known service oriented approach [15], which maps a particular service specification onto a set of interconnected components, each of them having an internal FSM, and a corresponding pool of abstract communication channels. This methodology also advocates the use of state machine synthesis algorithms so that the scenarios can be quickly simulated and/or validated (see Figure 2).

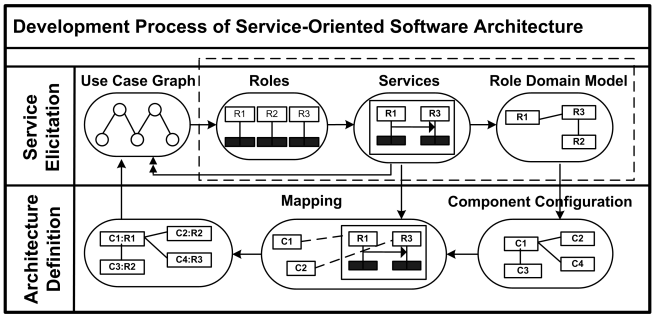


Figure 2. Service-based development

The generated state machines define the intended dynamic behavior of the specified system, thus they can be easily incorporated into our architectural design.

The state machine generation is carried out automatically and relies on two types of MSCs, the basic MSCs and the high level MSCs (HMSC). A basic MSC consists of a set of lines, each labeled by the name of the role and representing a certain unit of the behavior produced by that particular role. An HMSC is a graph whose nodes refer to other (H)MSCs. The semantics of an HMSC is obtained by following these operational paths and by composing the interaction patterns en route through the participating nodes. The output of the transformation is one FSM per role within the domain model; that is, the FSM implementing the respective role's contribution to the services it is associated with.

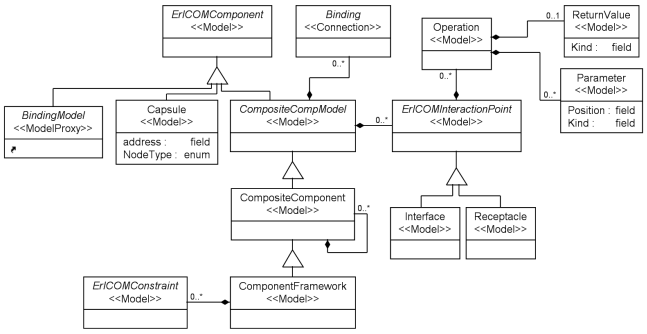


Figure 3. Functional metamodel

2) Functional Modeling: An outline of the component metamodel is illustrated in Figure 3. Components are encapsulated units of functionality and deployment, which interact with each other only via interfaces and receptacles. Interfaces are defined by a list of related operation signatures and associated data types. Components can provide multiple interfaces; embodying a clear separation of concerns (e.g. between base functionality and component management). Capsules and caplets are platform containers providing access to the run-time APIs. Bindings ensure consistent connection setup between a compatible interface and a receptacle. The component model itself is complemented by two other architecture elements: component frameworks and reflective extensions. Component frameworks (CF) are groupings of components with constraint guarantees to allow only "meaningful" component configurations. All entities of the metamodel (Component, Capsule, Interface, Receptacle, Binding, Component Framework) may store arbitrary <key,value> attributes, which contribute to a reflective layer facilitating universal discover at run-time. Component interactions can be intercepted at the bindings by pre- and post-actions to enable additional processing on the level of individual messages.

3) Behavior Modeling: The component behavior description is formalized in an abstract model of action semantics (see Figure 4). This Behavior Model is rather generic, but it though provides an explicit attribute for the specification of the modeled behavior within a particular implementation language. Those entities of the metamodel that may contain behavior descriptions are the Interface and the Component. A component model is translated onto target implementation languages by various model interpreters. In this way, the components can be created in various languages; however, they rely on the same modeling framework. A language specific model interpreter processes only those parts of a component model which contain relevant information for the desired target language environment. Therefore, the metamodel embodies various code snippets; the snippets are later woven together into executable component implementations by the related model interpreter. The most important parts

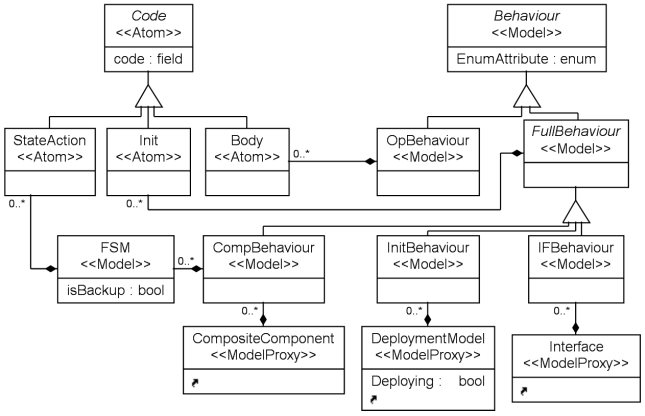


Figure 4. Behavior metamodel

of the code snippets are:

- Init - Initialization code for a component, an interface or the system.
- Body - Executable specification of the operation of an interface. The signature of the operation is defined in the model and automatically generated by the interpreter.
- StateAction - Specifies the semantics inside an FSM state. This action semantics is automatically injected into the corresponding connection point within the generated FSM Model.

4) Deployment Modeling: The complete synthesized platform specific application model contains both the structural configuration and the behavioral semantics of all the constituent components, including their interconnecting bindings and component framework constraints. That model represents the functional view of the application; however, it neither specifies how the application is deployed on the available networked nodes nor how it should start. Therefore, the deployment configuration must be modeled, too (see Figure 5).

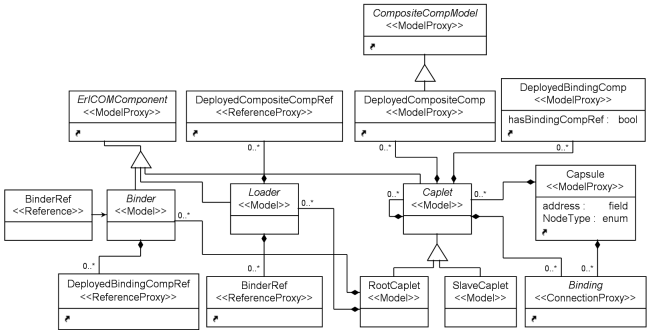


Figure 5. Deployment metamodel

The deployed component configuration, which contains the complete synthesized platform specific application model and the initial configuration of the components, is called the total synthesized platform specific distributed application model.

From the point of view of model based development, the most important element of the deployment infrastructure is the Deployment Tool, which establishes a soft real-time synchronization loop between the model repository and the running application. The schematics of the Deployment Tool based reconfigurability is shown in Figure 6.

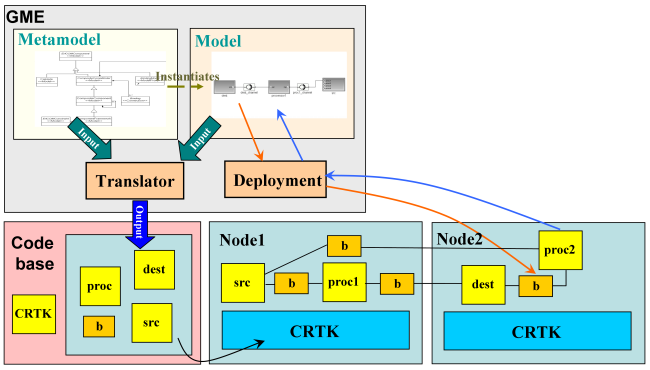


Figure 6. Deployment Tool based reconfigurability of run-time component application

The Deployment Tool analyzes the initial component configuration of the total synthesized platform specific application model and creates the needed ErlCOM elements by relying purely on the ErlCOM API. (Complete API semantics has been reported in [16]) After the initial deployment has been completed the application starts running and the ErlCOM CRTK continuously monitors all component reconfiguration and in case of observable component changes events are sent containing descriptive notifications to the Deployment Tool. The Deployment Tool keeps track of the actual component configuration of the running system by updating the total synthesized platform specific RUNES application model. Deployment Tool plug-ins can also execute policy based rules either re-actively or pro-actively. Any corrective changes on the modeled component configuration of the component application will be reflected by the run-time deployment.

C. RUNES Metamodel Verification with Alloy

1) Introduction: This section revisits the kernel part of metamodel, which defines the basic concepts of Interfaces, Receptacles, Components and Bindings, in order to formally represent those elements in a first order logic based formalism. Figure 7 illustrates that kernel part of the metamodel, including all the relevant relations and cardinalities. The associated OCL expressions are not visualized, though they play a significant role to establish a model-based rapid application development environment in the Generic Modeling Environment (GME) [17]. Mostly these OCL expressions

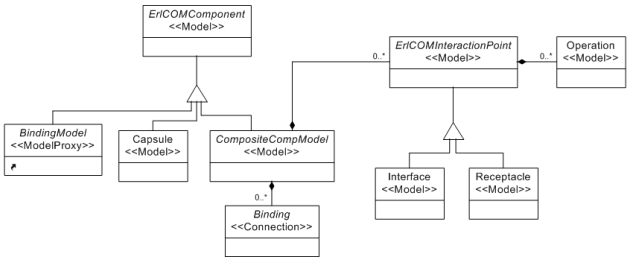


Figure 7. Kernel part of RUNES metamodel

interact with the generic component meta-model by further restricting the compatibility of component interconnection by creating an implicit subsumption hierarchy of bindable required-provided interfaces. Since a generic mapping of these OCL statements onto corresponding logical expressions in Alloy simply too complex and would go far beyond the scope of this paper, our current logic based formalism only relies on some selected elements of those OCL expressions and their mapping to Alloy has been ad-hoc and hand-crafted.

The generic aim of the approach is to to verify particular properties on some configuration sequences of certain modeled application scenarios via semantically anchored precise structural and behavioral formalism expressed in Alloy. The following sections describe the individual mappings between the various metamodeling concepts and their Alloy equivalents in adequate details.

2) Functional Model: In general, the functional specification of any RUNES application must be organized around Components and Bindings. The Components represent the encapsulated units of functionality and deployment. The interactions amongst participating components take place exclusively via explicitly defined Interfaces and Receptacles. The dynamic behavior of the components are automatically generated from MSC and they are represented via concurrent FSM [15]. Intending to rewrite the above specification into Alloy, a generic RUNES Component is, hence, defined as a signature whose fields consist of at most one Finite State Machine and a set of Interfaces and Receptacles, respectively.

```
abstract sig Comp{
   state_machine:set StateMachine,
   provided: set Interface,
   required: set Receptacle,
}{
   lone state_machine
}
```

Both the Interface and the Receptacle inherit the common characteristics of an Interaction Point, which is defined by a set of related operation signatures and associated data types. The Interface represents the "provided", the Receptacle the "required" end-point of a inter-component connection, respectively.

```
abstract sig Signature{}
abstract sig InteractionPoint {
   signatures: set Signature
}
sig Interface extends InteractionPoint{}
sig Receptacle extends InteractionPoint{}
```

A connection between compatible "provided" and "required" communication end-points is set up via Bindings. In fact, a Binding makes sure that connections between Interfaces and Receptacles are created consistently, according to compatible properties defined on the corresponding end-points. Hence, a definition of a Binding is also a signature in Alloy, however, it also contains some more fields; one for the Interface and another one for the Receptacle and finally a third one for a non-identical, component correct mapping connecting together the previous two fields. The connection constraint emanating from the "provided" and "required" characteristics of the end-points is attached to the Binding signature in the form of explicit logical restrictions.

```
abstract sig Binding{
   mapping:Comp -> Comp,
   interface: one Interface,
   receptacle: one Receptacle
}{
   one mapping
   no (mapping & iden)
   receptacle in (Comp.∼(mapping)).required
   interface in (Comp.mapping).provided
}
```

Furthermore, a Receptacle must always represent a required set of operations that is a 'subset' of those operations which are provided by the Interface it intends to be connected to via the Binding. In RUNES application models, this requirement is specified by explicit operation signatures, including parameter lists either via OCL constraints or by the graphical representation of the metamodel. In the case of Alloy, this constraint semantics has been slightly simplified and only an abstract signature matching is enforced.

```
all b:Binding| b.receptacle.signatures in b.interface.signatures
```

3) Deployment Model: Figure 8 revisits the most important deployment concepts of the RUNES Metamodel, which determine the runtime aspects of any component application. The key element is the Capsule, which represents the generic middleware container providing direct access to the functionalities of the runtime API of the CRTK. This set of functionalities incorporates also the robust fault management and recovery and the corresponding redundancy facilities. From the verification perspective, deploying a component into a capsule means that the capsule must be ensured to possess adequate resources made available for loading in components or bindings at any particular instance of time. The deployed components and bindings might be reorganized as time evolves, hence this temporal representation must take into account the explicit definition of time, too. This requirement can be easily satisfied by the introduction of Time into the formal representation of Capsules in Alloy. Hence, a Capsule is defined again as a signature, but this time it has also an explicit field standing for a time instance. The representation of the temporal evolution is not only

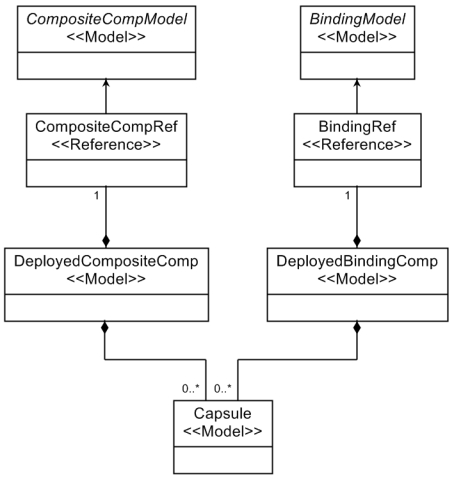


Figure 8. Deployment part of RUNES Metamodel

restricted to the deployed components or bindings, but it also incorporates a middleware related generic resource pool. This resource pool is again a semantic simplification, that abstractly represents the capacity of a capsule to contain either CRTK objects, such as components, or application specific elements that are usually stored in generic run-time repositories within capsules. Another semantic simplification is the time invariant representation of the capsule topology, though it may change in the case of a deployed application in real-time. Nevertheless, by setting the capacity of any particular capsule to zero one can easily simulate all kinds of run-time dynamic reconfigurability of a capsule hierarchy.

```
open util/ordering[Time] as TO
sig Time{}
abstract sig Capsule {
   comps: DeployedComp -> Time,
   bindings: DeployedBinding -> Time,
   comp_capacity: Int -> Time,
   neighbours: some Capsule
}{
   all t:Time|int[comp_capacity.t] >= #(comps.t)
   all t:Time|comp_capacity.t >= Int[0]
}
```

The formal specification of a deployed component must contain all those pieces of information that the active process aspect of the component's functionality requires, including the explicit definition of all state transitions in its FSM during the whole lifetime. In other words, structurally a deployed component has to be formally regarded as a dynamic instance of a component in accordance to its "ModelProxy" declaration in GME [18], as depicted in Figure 8. Considering the temporal aspect of its behavior, the state transitions are defined twofold in the signature of DeployedComp: on one hand, the fire mapping describes the fired transitions, one-by-one at a time; on the other hand, the field current_state tracks all state changes as time flies by.

```
sig DeployedComp{
   deploy: one Comp,
   fire: Transition -> Time,
   current_state: State -> Time
}{
   deploy in FunctionalConf.comps
   all t:Time|lone fire.t
   all t:Time|lone current_state.t
}
```

Some additional constraints are also appended to the definition of DeployedComp. Firstly, it must be safeguarded that the deployed component honors the functional definition of its component description in such a way that its provided logical representation fully satisfies the "ModelProxy" declaration in GME. Secondly, the behavior of the FSMs is restricted to enable only one of them to fire a single transition at a particular instance of time. This is a sequencing constraint on causality of time evolution, which is a restriction globally applicable to the component application. Thirdly, letting a component have an FSM internally it must be deterministic in nature, hence, there is only one current state representing the total dynamic status of the component. The latter two conditions can be relaxed, but the detailed investigation of their consequences is still work in progress.

In contrast to the deployed components, a deployed binding does not have to declare its time evolution explicitly; nevertheless, its formal definition in Alloy contains a mapping field that anchors the DeployedBinding into two deployed components it is connecting together. Hence, the signature of DeployedBinding looks time independent, though it tracks time evolution, but indirectly. Due to the intimate linkage between the definition of a binding and the bound two components, the compatibility of their reliance must be ensured. This extra condition is made explicit through three additional logical constraints appended to the signature of DeployedBinding, stating the uniqueness and functional compatibility of the connection. In other words, if the functional compatibility of the participating components of a binding can be proven, then, also the deployment of such components is assumed to be valid. This formal set of extra Alloy expressions is homologue to their "ModelProxy" declaration in the GME metamodel.

```
sig DeployedBinding{
   mapping: DeployedComp -> DeployedComp,
   deploy: one Binding
}{
   one mapping
   (DeployedComp.~mapping).deploy =
     Comp.~(deploy.mapping)
   (DeployedComp.mapping).deploy = Comp.(deploy.mapping)
}
```

Being our main motivation of applying Alloy for the verification of allowed component configurations, the deployed component applications must be also represented in a compatible Alloy formalism, that is, via a collection of capsules that are continuously tracking the temporal evolution of each of the deployed components and bindings. The Capsule having already been formally defined, the deployed component application is represented via its deployment configuration as a set of Capsules. Therefore, the formal definition of DeploymentConf is quite trivial. Furthermore, Alloy's trace statements help verify this time evolution of the deployed application as will be shown in Section V; thus, successful runs are easily and interactively visualized

for human inspection, too.

```
sig DeploymentConf{
   capsules: some Capsule
}
```

4) Middleware Model: The ErlCOM middleware API supports a complete set of component management operations such as [un]loading, [un]binding and migrating of components. These operations contain complex negotiation protocols among various elements of the ErlCOM CRTK and the deployed components, thus, the execution of a particular API invocation may require some time to complete its functionality. The operations usually modify only the local states of the distributed application and keep the rest of the application's state space unchanged. Obviously, these complex concurrent middleware activities must be considerably simplified so that a compatible logical formalism can be within reach of practical usability. Therefore, in general, all of the potentially concurrent atomic API operations are to be serialized in such a way that one and only one of them is allowed to be executed at one particular instance of time. As a good example showing this simplification approach, the Alloy definition of the 'migrate' operation will be explained in detail in the sequel. In the case of the remaining ErlCOM operations [5] similar techniques have been applied in order to translate them into corresponding Alloy expressions.

A component migration is carried out between two capsules by moving an already deployed component between two consecutive points of time. In effect, the migration itself can be conceptualized as a sequence of invocations of individual ErlCOM API operations: 'Create Component', 'Load Component', 'Update Component', 'Unload Component' and 'Destroy Component'. Obviously, the CRTK provides an optimized single API operation for completing the component migration in one single step; however, for the sake of explaining our approach of simplification the above sequence is considered to be valid. Since our formal Alloy representation is agnostic to the means by which the local state of a component is being migrated from one capsule into another, the operations of 'Load Component', 'Update Component' and 'Unload Component' can be either totally disregarded or taken into account in such a way that only the application relevant state space of the component is copied from time t to time t'. Hence, the only API invocations to be mapped into Alloy are 'Create Component' and 'Destroy Component'. Due to their analog treatment, let us examine only the operation 'Create Component'. The executable specification of the operation in Erlang is the following (see Figure 9 for corresponding MSC):

```
%create in CRTK
create(CapletName,InstanceName)->
   gen_server:call(global,CapletName , create,InstanceName ),
   insert_component(InstanceName,component,CapletName,CapletName).

%create in Caplet
create(InstanceName,Type)->
   CapsuleName = crtk:getOwner(crtk:getSelfName()),
   gen_server:call(global,CapsuleName, create,InstanceName),
   insert_component(InstanceName,Type).
```

```
%create in Capsule
create(InstanceName)->
   gen_server:start_link(global,ComponentName,
      e_EmptyComp, [InstanceName],[]).

%insert_component in Caplet
insert_component(InstanceName,Type)->
   ets:insert(get(componentTable),#componentcomponentName=InstanceName,
      componentData=#componentDatacomponentType=Type,state=created).

%insert_component in CRTK
insert_component(ComponentName,ComponentType,Owner,RegistryOwner) ->
   NodeName=node(),
   Fun = fun() ->
      mnesia:write(#componentcomponentName=ComponentName,
         componentType=ComponentType,owner=Owner,registryOwner=RegistryOwner,
         nodeName=NodeName)
   end,
   mnesia:transaction(Fun).
```

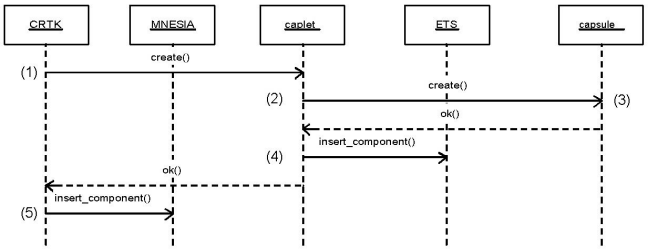


Figure 9. MSC of Create Component in ErlCOM

The message flow of a component creation is the following (see Figure 9): the CRTK first calls the create operatin on the caplet which forwards this request to the corresponding capsule. When the capsule responds OK the component related data will be stored into the caplet's local cache (ets). After the CRTK has received OK from the caplet, it registers the component data into the distributed Mnesia database.

Taking into account that the current Alloy specification of the Deployment Model (see Section IV-C3) only allows flat configurations of Capsules instead of a full hierarchy of Caplets with a leading root Capsule, steps 1 and 2 should be considered as a single combined activity. Moreover, steps 4 and 5 are only relevant to the internals of ErlCOM, therefore the first order logic based abstraction of 'Create Component' consists of one major task only, it being the addition of a new component into the receiving capsule. The operation of 'Destroy Component' can be handled similarly. Hence, in summary, the elements of the formal expression in Alloy of a migration operation are as follows: First the preconditions are checked if it is a real migration between two different capsules. Next, as capsules are abstracted to possess a generic capacity parameter, it is also checked if there are enough resources available in the receiving capsule. Then, the local states of the two respected capsules are updated, which is the homologue of the actual component migration in ErlCOM. Finally, three more constraints are to be satisfied in order to ensure that the rest of the application state remains unchanged. This restriction enforces our serialization concept of causality, which we intend to relax in our future research work.

```
pred migrate(c_src,c_dst:Capsule,d:DeployedComp,t,t':Time){
   c_src != c_dst
   #(c_dst.comps.t) < int[c_dst.comp_capacity.t]
```

```
    c_dst.comps.t' = c_dst.comps.t+d
    c_src.comps.t' = c_src.comps.t-d
    all capsule:Capsule|capsule.bindings.t'=capsule.bindings.t
    all capsule:Capsule-c_src-c_dst| capsule.comps.t'=capsule.comps.t
    all capsule:Capsule| capsule.comp_capacity.t' = capsule.comp_capacity.t
}
```

Going beyond the dynamic aspects of ErlCOM CRTK, there are still some structural restrictions of the middleware left from the RUNES Metamodel. These enforce RUNES specific constraints over potential component configurations in order to safeguard the semantic correctness of component reconfigurations. In the metamodel (see Figure 7 and Figure 8) those rules are expressed either via cardinality constraints or by additional OCL statements. Consequently, their formal Alloy representation must be incorporated into our set of definitions, too. There are many such extra restrictions, though here we only introduce the most relevant elements of that constraint set.

- A Binding or a Component must be contained within at most one single Capsule. Until the binding or the component has not been deployed to or removed from a particular capsule its association with that capsule is non-existent. However, while it is deployed, one and only one capsule can contain it at any point of time.

```
no disj capsule1,capsule2:Capsule|
 some (capsule1.bindings) & (capsule2.bindings)
no disj capsule1,capsule2:Capsule|
 some (capsule1.comps) & (capsule2.comps)
```

- Two Bindings of the same type must not be deployed if they share the same Receptacle. This constraint enforces that connections of a particular type between two components, via a compatible Interface and Receptacle, cannot be shared at any point of time.

```
no disj b1, b2:DeployedBinding| (b1.deploy = b2.deploy)
 and (b1.mapping.DeployedComp = b1.mapping.DeployedComp)
```

- There must not be such a Binding within a Capsule that has a connected Component which is not deployed in any of the Capsules. This constraint is critical since a binding can only connect together already deployed components at its related end-points. Unconnected or half-bound bindings are semantically incorrect since the bind and unbind operations of the ErlCOM API are both atomic in nature.

```
no deployedBinding:DeployedBinding|some t:Time|
 deployedBinding in Capsule.bindings.t and
 (deployedBinding.mapping.DeployedComp not in Capsule.comps.t
 or deployedBinding.mapping[DeployedComp] not in Capsule.comps.t)
```

5) Behavioral Model: The dynamic behavior of the component application is modeled via Finite State Machines (FSM) that are either automatically generated directly from the scenario MSCs or manually elaborated and added to selected components based on application specific requirements. Therefore, in essence, the internal dynamics of the components' functional behavior must be specified in Alloy by an explicit transcription of an FSM that specifies all changes in internal state of the component, including the preconditions of state transitions and the necessary action semantics required by the postconditions of the transition at entering the new state. Due to the complexity of practical applicability, only the vital components of the application are mapped onto their Alloy representation. Nevertheless, our FSM specification in Alloy is generic and mirrors the formal mathematical model following the abstract principle of semantic anchoring [19].

The formal Alloy definition of a Finite State Machine (FSM) relies on the signatures of State and Transition; the latter being specified as a mapping between two States. The initial state of the FSM is designated by the <StartState, StartTransition> pair. Since our verification approach targets the evolution of the FSMs in time a predicate named transition is introduced, which tracks the time instances and records the transitions being executed between every time t and t' inside the deployed component that has currently been chosen for letting its FSM fire.

```
abstract sig State{}
abstract sig Transition{
   trans: State -> State
}{
   one trans
}
abstract sig StartState extends State{}
abstract sig StartTransition extends Transition{}
pred transition[d:DeployedComp,t,t':Time]{
   (d.fire.t).trans.State = d.current_state.t
   (d.fire.t).trans[State] = d.current_state.t'
}
abstract sig StateMachine{
   states: some State,
   startState: one StartState,
   transitions: some Transition,
   startTransition: one StartTransition,
}{
   no (states & startState)
   no (transitions & startTransition)
}
fact Traces{
   ...
   all t:Time-TO/last[],d:DeployedComp|let t'=TO/next[t]|
     some d.fire.t => (transition[d,t,t'])
   all t:Time|some DeployedComp.fire.t
}
```

The definition of the fact Traces puts the FSM into action, basically letting at most one transition fire at a particular point of time. The allowed firings are selected according to the defined transition rules within the FSMs; therefore, the deployed component application is totally FSM driven in our Alloy specification.

6) Example Model: Having all the elements of our Alloy formalism specifying the ErlCOM based, RUNES metamodel compatible models described in details, here the graphical visualization of such an example model is shown in Alloy Analyzer. Figure 10 depicts a model that represents a snapshot of a dynamically evolving component configuration of a sensor network scenario example. The components (black hexagons) have been deployed over a cross shaped capsule (gray pentagons) topology. The connections among the capsules of this topology are indicated by green arrows. The internal resources, here the maximum number of deployed components/bindings, of the capsules are pooled and limited in their capacity. The concrete mapping of the components and bindings (white rhombuses) onto the capsules, at a particular instance of time, is visualized by the brown and red arrows, respectively.

This figure shows only a particular snapshot of a dynamically evolving component application, therefore, for the

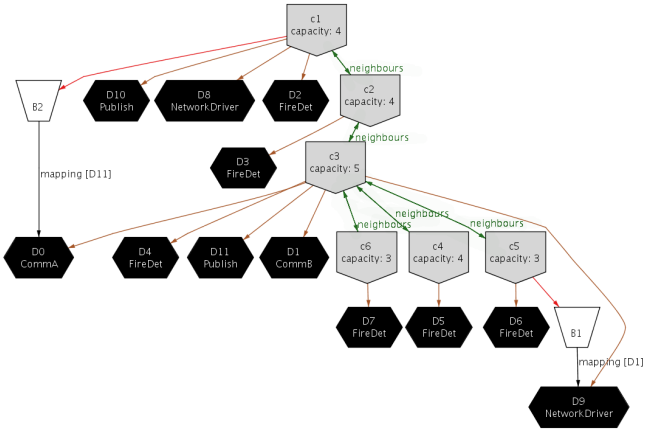


Figure 10. Scenario analysis snapshot

validation of an application scenario or the verification of a certain logical property full sequences of these snapshots must be analyzed. In the sequel, a simplified scenario model will be analyzed to demonstrate the Alloy driven verification step of our methodology in some detail.

V. Simplified Scenario Example

The scenario example that is used to showcase the usability of our proposed approach is based on the Fire in the Road Tunnel scenario [2] of the RUNES IST project. This simple scenario example is part of one of the RUNES demonstrators; hence, its elements have been extracted directly from a bigger component application in order to make it manageable for practical analyses in Alloy. For better understanding, some steps of the RUNES application development process (see Section IV-A) will also be shown in the case of this particular example. The relevant excerpt from the overview of the scenario story [2] is as follows:

"At the beginning of our story traffic is flowing normally in the road tunnel. Tunnel fires can be detected by the wired system that is part of the tunnel infrastructure. The fire sensors do, however, have the capability to operate wirelessly if required. An accident within the road tunnel has resulted in a fire. The fire is detected and is reported back to the TunnelControl Room. ... As a result of the fire the wired infrastructure is damaged and the link is lost between fire detection nodes. Using wireless communication, information from the fire detection nodes is still delivered to the Tunnel Control Room seamlessly. ... As the firemen move towards the fire the sensors reporting periodic data on external temperatures detect a rise in temperature and respond by increasing the frequency of reporting so that the EmergencyControl can assess the danger to the fire fighters. The fire becomes more severe. A node is lost..."

For the sake of being able to show dynamic behaviour modeling in Alloy, we are, first, focusing on the Interaction Modeling (see Section IV-B1). It is obvious to recognise that there is a Fire Detector service lying in the heart of the Fire in the Road Tunnel scenario. It was specified, within the RUNES project, via five MSCs, which are depicted in Figure 11.

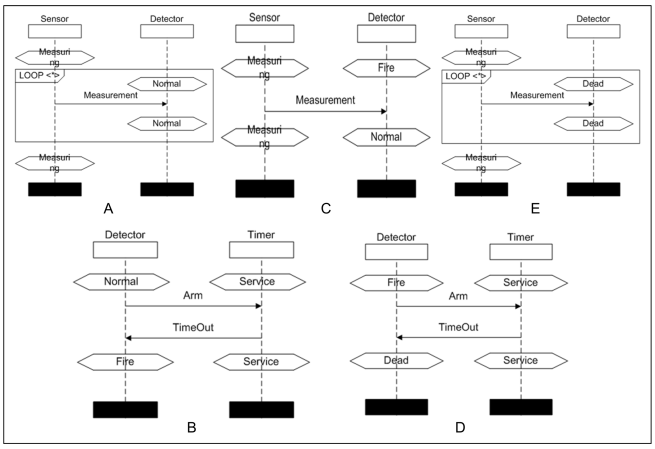


Figure 11. Message Sequence Charts of Fire Detector

Then, the MSCs are translated via a sequence of transformations [4], including a non-trivial graph transformation, into an equivalent FSM representation, which is shown in Figure 12.

Regarding its Functional (See Section IV-B2) and Deployment (See Section IV-B4) Modeling the scenario example has been simplified by having been selected only two capsules and 8 deployed components. In Figure 13, the Alloy representation of the functional configuration of the component system is depicted. The blue hexagons show the components, the beige rectangles represent the bindings and the green diamonds stand for the finite state machines.

This functional view contains, thus, five different components; namely, three network related components (NetworkDriver, CommA and CommB) and two application specific components (Publish and FireDet). The components CommA and CommB implement two different kinds of com-

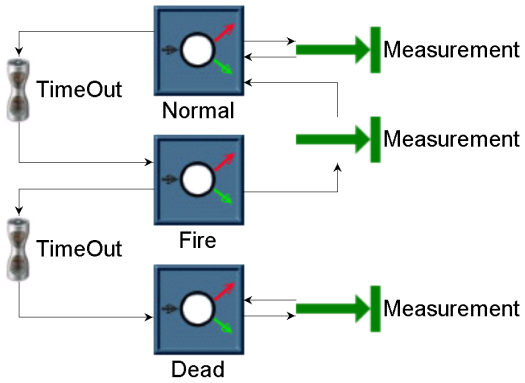


Figure 12. Platform Independent Behavior for Fire Detector

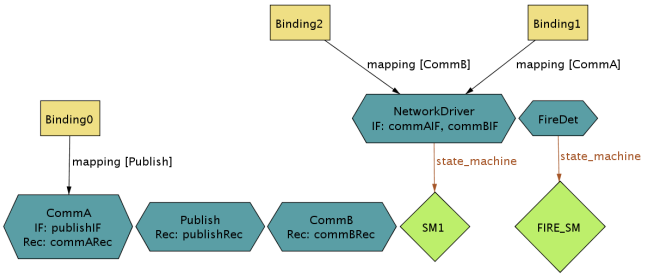


Figure 13. Functional configuration of the example system

munication paradigms, both of them relying on the shared functionalities of the common NetworkDriver component. Their connections are made available through Binding1 and Binding2. The main functionality of the Publish component is to broadcast different sensor measurement data towards the processing end points, such as the Tunnel Control Room. The FireDet component combines and simulates both the effects of spreading fire and the decisions that could have been taken by a real control component being responsible for reconfiguring the other components whenever a fire situation has been detected. Due to the complexity of its real-life homologue, FireDet in Alloy has a rather modified, adapted version of the original FSM that is associated with the Fire Detection service. Its control logic has been reduced to fit the trivial topology of the deployed components in this new functional setup. Nevertheless, the main goal of its functionality, which is to keep the sensor system in operation even in case of extreme fire conditions, has been left unchanged. Therefore, in the scenario example, the reconfiguration is carried out by letting the application components migrate to other capsules located in the neighborhood. In effect, the original states Fire and Dead have been combined and the substitute state causes a gradual loss in capsule capacity. By decreasing this generic capacity parameter of the capsule taken "by fire" the receiving capsule will not be able to immediately reinsert the recently migrated component; hence, ping-pong effects are eliminated.

Both NetworkDriver and FireDet possess proper state machines, which are represented by the green diamonds in Figure 13.

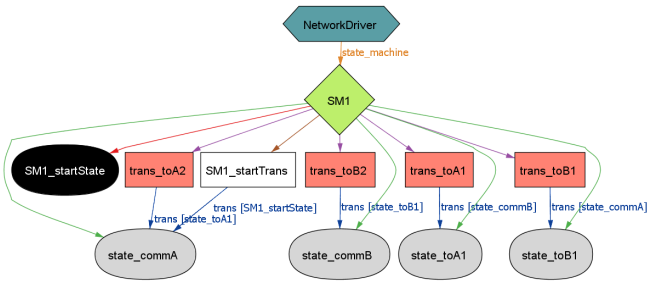


Figure 14. NetworkDriver state machine

Figure 14 shows the state machine of the NetworkDriver component. Its initial status is given by the start state, SM1_startState (black ellipse), and the initial transition, SM1_startTrans (white rectangle), leading from the start state to state_commA. (The states are represented by gray colored ellipses, while the transitions are shown via red rectangles.) Via a consecutive transition from state_commA to state_commB, through a temporal state_toB1, the unbinding of component CommA from NetworkDriver and the binding of component CommB to NetworkDriver will take place. This state transition sequence is a simplified version of the Behavioral model (see Section IV-B3) of the component and simulates the reconfiguration of the communication paradigms within the scenario example. The binding and the unbinding operations represent the invocations of the ErlCOM middleware API. (see Section IV-C4). The Alloy specification of NetworkDriver's FSM is as follows:

```
sig SM1 extends StateMachine{}
{
   startState = SM1_startState
   startTransaction = SM1_startTrans
   one SM1_state_commA
   one SM1_state_commB
   one SM1_state_toB1
   one SM1_state_toA1
   states = SM1_state_commA + SM1_state_commB +
       SM1_state_toB1+SM1_state_toA1
   one SM1_trans_toA1
   one SM1_trans_toA2
   one SM1_trans_toB1
   one SM1_trans_toB2
   transactions = SM1_trans_toA1 + SM1_trans_toA2 +
      SM1_trans_toB1 + SM1_trans_toB2
}

sig SM1_startState extends StartState{}

sig SM1_state_commA extends State{}{
   no t:Time-TO/last[]|
      let t' = TO/next[t]|this in getEndState[DeployedComp.fire.t] and
      not SM1_state_commA_action[t,t']
}

pred SM1_state_commA_action[t,t':Time]{
   some b:DeployedBinding|
      let d = getDeployedComp[SM1_state_commA,t]|
      b.mapping[DeployedComp] = d and
      ((b.mapping.DeployedComp).deploy = CommA) and
      bind[getCapsule[d,t],b,t,t']
}

sig SM1_state_commB extends State{}{
   no t:Time-TO/last[]|
      let t' = TO/next[t]|this in getEndState[DeployedComp.fire.t] and
      not SM1_state_commB_action[t,t']
}

pred SM1_state_commB_action[t,t':Time]{
   some b:DeployedBinding|
      let d = getDeployedComp[SM1_state_commB,t]|
      b.mapping[DeployedComp] = d and
      ((b.mapping.DeployedComp).deploy = CommB) and
      bind[getCapsule[d,t],b,t,t']
}

sig SM1_state_toB1 extends State{}{
   no t:Time-TO/last[]|
      let t' = TO/next[t]|this in getEndState[DeployedComp.fire.t] and
      not SM1_state_toB1_action[t,t']
}

pred SM1_state_toB1_action[t,t':Time]{
   some b:DeployedBinding|
      let d = getDeployedComp[SM1_state_toB1,t]|
      b.mapping[DeployedComp] = d and
      ((b.mapping.DeployedComp).deploy = CommA) and
      unbind[getCapsule[d,t],b,t,t']
}

sig SM1_state_toA1 extends State{}{
   no t:Time-TO/last[]|
      let t' = TO/next[t]|this in getEndState[DeployedComp.fire.t] and
      not SM1_state_toA1_action[t,t']
}

pred SM1_state_toA1_action[t,t':Time]{
   some b:DeployedBinding|
      let d = getDeployedComp[SM1_state_toA1,t]|
      b.mapping[DeployedComp] = d and
```

```
      ((b.mapping.DeployedComp).deploy = CommB) and
      unbind[getCapsule[d,t],b,t,t']
}

sig SM1_startTrans extends StartTransaction{} {
   trans[SM1_startState] = SM1_state_commA
}

sig SM1_trans_toB1 extends Transaction{} {
   trans.State = SM1_state_commA
   trans.State = SM1_state_toB1
   no t:Time|this in DeployedComp.fire.t and
      not SM1_trans_commA_pred[t]
}

pred SM1_trans_commA_pred[t:Time]{}

sig SM1_trans_toB2 extends Transaction{} {
   trans.State = SM1_state_toB1
   trans.State = SM1_state_commB
}

sig SM1_trans_toA1 extends Transaction{} {
   trans.State = SM1_state_commB
   trans.State = SM1_state_toA1
}

sig SM1_trans_toA2 extends Transaction{} {
   trans.State = SM1_state_toA1
   trans.State = SM1_state_commA
}
```

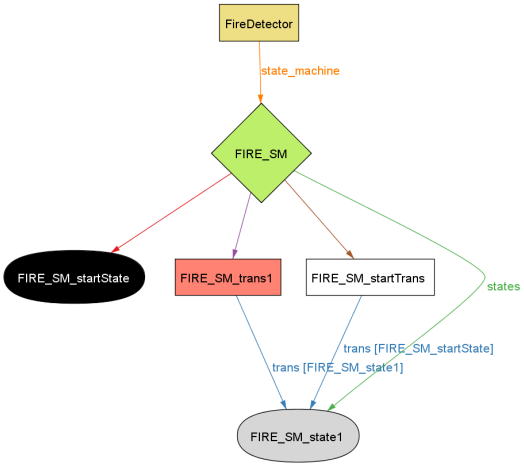


Figure 15. FireDet state machine

The Figure 15 shows the state machine of the FireDet component. As previously explained, this FSM is a reduced version of the original one possessing only two states. Its initial state, FIRE_SM_startState, represents the Normal state of the FSM associated with the Fire Detection service, while the state FIRE_SM_state1 stands for the combination of states Fire and Dead. (see Figure 12) The transition is one-way only and results in the decrease of capsule's capacity. The Alloy specification of FireDet's FSM is as follows:

```
sig FIRE_SM extends StateMachine{} {
   startState = FIRE_SM_startState
   one FIRE_SM_state1
   states = FIRE_SM_state1
   startTransaction = FIRE_SM_startTrans
   one FIRE_SM_trans1
   transactions = FIRE_SM_trans1
}

sig FIRE_SM_startState extends StartState{}

sig FIRE_SM_state1 extends State{}{
   all t:Time-TO/last[]|let t' = TO/next[t]|
      this in getEndState[DeployedComp.fire.t] =>
      FIRE_SM_state1_action[t,t']
}

pred FIRE_SM_state1_action[t,t':Time]{
   let c = getCapsule[getDeployedComp[FIRE_SM_state1,t],t]|
      decrease_capacity[c,t,t']
}

}

sig FIRE_SM_startTrans extends StartTransaction{} {
   trans[FIRE_SM_startState] = FIRE_SM_state1
}

sig FIRE_SM_trans1 extends Transaction{} {
   trans.State = FIRE_SM_state1
   trans.State = FIRE_SM_state1
}
```

In the sequel, a simple validation sequence of the above defined scenario example will be analyzed step-by-step. Figures 16–19 show the snapshots of an Alloy trace sequence. The model evolution is projected over Time in such a way that the relations of a model in different points of Time are represented through a sequence of consecutive models. Expressed it more precisely in Alloy parlance, it means that one Time instance is connected to one and only one particular Model snapshot.

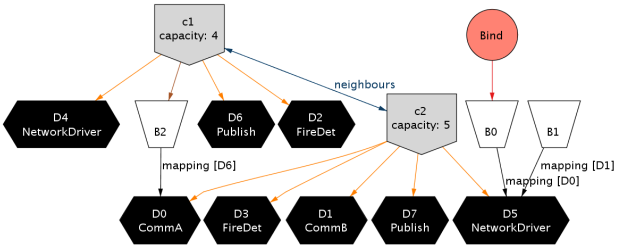


Figure 16. Component binding step

Figure 16 presents the first step of the sequence. When NetworkDriver_startTrans has been activated the red circle labeled by the Bind tag, which represents the invocation of the bind operation of the ErlCOM API, points to the deployed binding B0. The deployed component D5, in capsule c2, is going to be bound to D0 in the next step (see Figure 17).

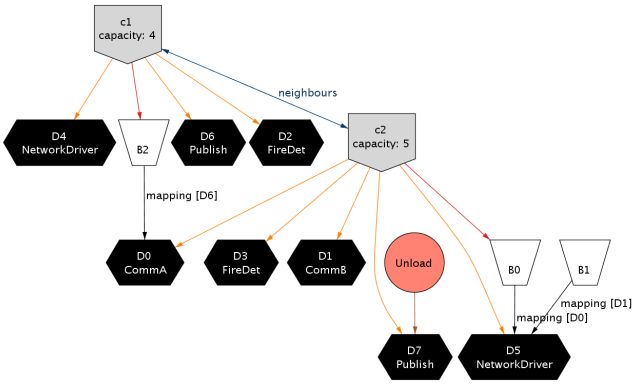


Figure 17. Component reconfiguration (unload) step

In Figure 17 the first reconfiguration of the system is to be seen. The FireDet component's state machine is activated; therefore, the migration of some application functionality has been started. FireDet selected the Publish component, in capsule c2, for migration; however, since another Publish

component had already been deployed to the neighboring capsule c1, the marked Publish component is going to be unloaded from capsule c2 instead of being migrated into capsule c1. Moreover, FireDet will also decrease the capacity of capsule c2.

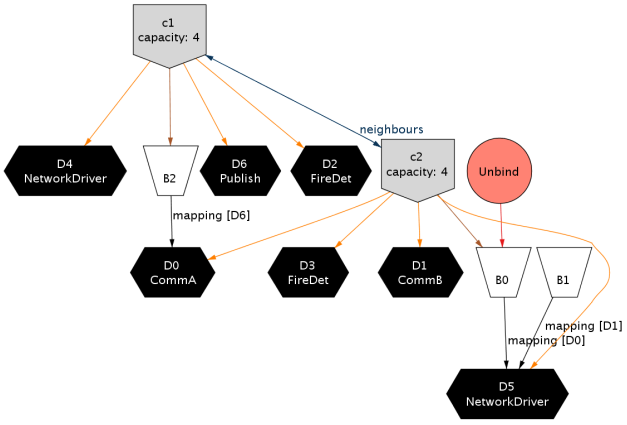


Figure 18. Component unbinding step

In Figure 18, the reconfiguration of the Network-Driver component has started changing from communication paradigm CommA to CommB. In Figure 19, the second migration attempt is demonstrated. In this case, component CommA is migrating to capsule c2 because this required functionality has not been deployed yet to that capsule so far.

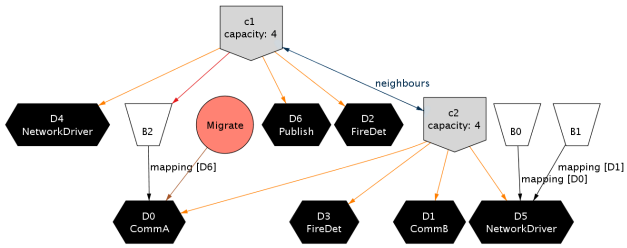


Figure 19. Component reconfiguration (migration) step

Although this example is a rather simplified one in nature, it indicates well the way how a particular validation or verification session may take place using Alloy Analyzer. Validation only generates a set of potential runs of a scenario, while verification also injects logical properties into the Alloy specification of the component application before it looks for counter-examples and shows them visually if found. In general, this approach helps enormously to analyze configuration sequences so that they both comply with some application constraints and avoid non-trivial pitfalls. The result of these analyses is later fed back to the control logic of the Deployment Tool (see Section IV-B4).

VI. Conclusion and Future Work

This paper has investigated a new way of combining domain specific metamodeling techniques with first order logic based model verification so that dynamic component applications could benefit from better quality reconfiguration mechanisms thanks to active scenario validation and verification. We have introduced the semantical foundations of our approach by describing the most relevant items of the RUNES metamodel, its development methodology and, most importantly, the first order logic based definition of those metamodel elements in Alloy representation. We have also illustrated the applicability of the approach in the case of reconfigurable component based sensor networks by a simplified scenario example that has been disseminated in detail. Our current work focuses on further extending the presented methodology by combining the assets of the RUNES and the GANA [20] metamodel in order to fully automate the generation of the adaptive control logic for autonomic component applications. So we are currently investigating the information extraction and feed-back of the results of Alloy based validation and verification of component model configurations so that we could explicitly manage the deployed system via multi-faceted control paradigms. Obviously, we are fully aware of the scalability issues of our current approach, so further studies will be carried out in this regard. Moreover, the results of these studies will be incorporated, as best practices guidelines, into future model translators, which are supposed to produce the major parts of the Alloy specifications and to evaluate the results of the analysis runs. Ultimately, our aim is to create a generic framework which iteratively and interactively modifies and verifies the component model of sensor application scenarios and continuously indicates the most probable, correct runtime configuration sequences thereof.

References

[1] Z. Theisz, G. Batori, and D. Asztalos, “Formal logic based configuration modeling and verification for dynamic component systems,” *Proceedings of MOPAS 2011*, 2011.

[2] K.-E. Arzén, A. Bicchi, G. Dini, S. Hailes, K. H. Johansson, J. Lygeros, and A. Tzes, “A component-based approach to the design of networked control systems,” *European Journal of Control*, 2007.

[3] P. Costa, G. Coulson, C. Mascolo, G. P. Picco, and S. Zachariadis, “The RUNES middleware: A reconfigurable component-based approach to networked embedded systems,” *Proceedings of the 16th Annual IEEE International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC'05), Berlin, Germany*, September 2005.

[4] G. Batori, Z. Theisz, and D. Asztalos, “Domain specific modeling methodology for reconfigurable networked systems,” *ACM/IEEE 10th International Conference on Model Driven Engineering Languages and Systems (MoDELS 2007)*, 2007.

[5] G. Batori, Z. Theisz, and D. Asztalos, "Robust reconfigurable erlang component system," *Erlang User Conference, Stockholm, Sweden*, 2005.

[6] J. Armstrong, "Making reliable distributed systems in the presence of software errors," *SICS Dissertation Series 34*, 2003.

[7] D. Jackson, *Software Abstractions: Logic, Language, and Analysis*. The MIT Press, London, England, 2006.

[8] G. Behrmann, A. David, and K. G. Larsen, "A tutorial on UPPAAL," *Proceedings of the 4th International School on Formal Methods for the Design of Computer, Communication, and Software Systems (SFM-RT'04), LNCS 3185*, 2004.

[9] G. Holzmann and R. Joshi, "Model-driven software verification," *Proceedings of SPIN2004, Springer Verlag, LNCS 2989*, 2004.

[10] D. Jackson, "Alloy analyzer," *http://alloy.mit.edu/*, 2008.

[11] M. Taghdiri and D. Jackson, "A lightweight formal analysis of a multicast key man-agement scheme," *Formal Techniques for Networked and Distributed Systems (FORTE 2003)*, vol. 2767 of LNCS., pp. 240–256, 2003.

[12] I. Warren, J. Sun, S. Krishnamohan, and T. Weerasinghe, "An automated formal approach to managing dynamic reconfiguration," *21st IEEE International Conference on Automated Software Engineering (ASE 2006), Tokyo, Japan*, pp. 37–46, September 2006.

[13] D. Walsh, F. Bordeleau, and B. Selic, "A domain model for dynamic system reconfiguration," *ACM/IEEE 8th International Conference on Model Driven Engineering Languages and Systems (MoDELS 2005)*, vol. 3713/2005, pp. 553–567, October 2005.

[14] E. G. Aydal, M. Utting, and J. Woodcock, "A comparison of state-based modelling tools for model validation," *Tools 2008*, June 2008.

[15] I. H. Krueger and R. Mathew, "Component synthesis from service specifications," *In Proceedings of the Scenarios: Models, Transformations and Tools International Workshop, Dagstuhl Castle, Germany, Lecture Notes in Computer Science, Vol. 3466*, pp. 255–277, September 2003.

[16] G. Batori, Z. Theisz, and D. Asztalos, "Configuration aware distributed system design in erlang," *Erlang User Conference, Stockholm, Sweden*, 2006.

[17] A. Ledeczi, M. Maroti, A. Bakay, G. Karsai, J. Garrett, C. Thomason, G. Nordstrom, J. Sprinkle, and P. Volgyesi, "The generic modeling environment," *In Proceedings of WISP'2001, Budapest, Hungary*, pp. 255–277, May 2001.

[18] "GME documentation," *http://www.isis.vanderbilt.edu/Projects/gme*.

[19] K. Chen, J. Sztipanovits, S. Abdelwahed, and E. Jackson, "Semantic anchoring with model transformations," *European Conference on Model Driven Architecture -Foundations and Applications (ECMDA-FA), Nuremberg, Germany*, November 2005.

[20] A. Prakash, Z. Theisz, and R. Chaparadza, "Formal methods for modeling, refining and verifying autonomic components of computer networks," *Springer Transactions on Computational Science (TCS) - Advances in Autonomic Computing: Formal Engineering Methods for Nature-Inspired Computing Systems in LNCS 7050*, pp. 1 – 48, 2012.

Utility Functions in Autonomic Workload Management for DBMSs

Mingyi Zhang[†], Baoning Niu[§], Patrick Martin[†], Wendy Powley[†], Paul Bird[‡]
[†]School of Computing, Queen's University, Kingston, ON, Canada
{myzhang, martin, wendy}@cs.queensu.ca
[§]Taiyuan University of Technology, Shanxi, China
niubaoning@tytu.edu.cn
[‡]Toronto Software Lab, IBM Canada Ltd., Markham, ON, Canada
pbird@ca.ibm.com

***Abstract*—Utility functions are a popular tool for achieving self-optimization in autonomic computing systems. Utility functions are used to guide a system in optimizing its own behavior in accordance with high-level objectives specified by the system administrators. It is, however, difficult to define a new utility function or evaluate whether an existing utility function is appropriate for a specific system management scenario. In this paper, we discuss the fundamental properties of an effective utility function for autonomic workload management in database management systems (DBMSs). We present two concrete examples of utility functions to illustrate the properties. The utility functions are used for dynamic resource allocation and for query scheduling in DBMSs. The utility functions help the systems translate high-level workload management policies into low-level tuning actions, and therefore ensure the workloads achieve their required performance objectives. A set of experiments are presented to illustrate the effectiveness of the two example utility functions.**

Keywords-Self-Optimization; Utility Function; Autonomic Computing; Workload Management; Database Management Systems

I. INTRODUCTION

A database workload is a set of requests that have some common characteristics such as application, source of request, type of query, priority, and performance objectives (*e.g.*, response time or throughput objectives) [2]. Workload management in database management systems (DBMSs) is a performance management process. The primary objectives of workload management in DBMSs are to achieve the performance goals of all workloads (particularly, the critical ones, such as the workloads for directly generating revenue for business organizations, or those issued by a CEO or VP of the organizations), maintain DBMSs running in an optimal state (*i.e.*, neither under-utilized nor overloaded), and balance resource demands of all requests to maximize performance of the entire system.

For both strategic and financial reasons, many business organizations are consolidating individual data servers onto a single shared data server. As a result, multiple types of requests are present on the data server simultaneously. Request types can include on-line transaction processing (OLTP) and business intelligence (BI). OLTP transactions are typically short and efficient, consume minimal system resources, and complete in sub-seconds while BI queries tend to be more complex and resource-intensive and may require hours to complete. Requests generated by different applications or initiated from different business units may have unique performance objectives that are normally expressed in terms of service level agreements that must be satisfied for business success.

Multiple requests running on a data server inevitably compete for shared system resources, such as system CPU cycles, buffer pools in main memory, disk I/O bandwidth, and various queues in the database system. If some requests, for example, long BI queries, are allowed to consume a large amount of system resources without control, the concurrently running requests may have to wait for the long queries to complete and release their used resources, thereby resulting in the waiting requests missing their performance objectives and the entire data server suffering degradation in performance. Moreover, the mix of arriving requests present on a data server can vary dynamically and rapidly, so it becomes virtually impossible for database administrators to manually adjust the system configurations to dynamically achieve performance objectives of all the requests during runtime. Therefore, a*utonomic workload management* becomes necessary and critical to control the flow of the requests and manage their demands on system resources to achieve their required performance objectives in a complex request mix environment.

Since *autonomic computing* was introduced [3], a great deal of effort has been put forth by researchers and engineers in both academia and industry to build autonomic computing systems. An autonomic computing system is a self-managing system that manages its own behavior in accordance with high-level objectives specified by human administrators [3] [4]. Such systems regulate and maintain themselves without human intervention to reduce the complexity of system management and dynamically achieve system objectives, such as performance, availability and security objectives. In particular, an autonomic workload management system for DBMSs is a self-managing system that dynamically manages workloads present on a data server in accordance with specified high-level business objectives such as workload business importance policies.

Achieving the goal of autonomic workload management may involve using utility functions to facilitate the mapping

of high-level business objectives to low-level DBMS tuning actions in order to guide a database system to optimize its own behavior and achieve required performance objectives. Utility functions are well known as a measure of user preference in economics and artificial intelligence [5]. In this paper, we illustrate the use of utility functions in different aspects of database workload management, namely dynamic resource allocation and query scheduling, to ensure mixed-type requests on a data server achieve their required performance objectives. The contribution of this study is a set of fundamental properties of a utility function used for building autonomic workload management systems, and the use of the properties to evaluate whether an existing utility function is appropriate for autonomic workload management in DBMSs. The methods and properties were first presented in our (ICAS'11) paper [1] and have been elaborated upon and extended with experimental validation here.

The paper is organized as follows. Section II reviews the background and related work, in which a short review of workload management for DBMSs, a brief description of autonomic computing, and utility functions used for building autonomic computing systems are presented. Section III discusses the fundamental properties of a utility function that can be used in realizing autonomic workload management for DBMSs. Section IV provides two examples to illustrate the properties of two different types of utility functions that are used in our studies. Section V presents experiments to evaluate and compare the two utility functions in accordance with some given high-level workload business importance policies. Finally, we conclude our work and propose future research in Section VI.

II. BACKGROUND AND RELATED WORK

In the past several years, considerable progress has been made in workload management for DBMSs. New techniques have been proposed by researchers, and new features of workload management facilities have been implemented in commercial DBMSs. These workload management facilities include IBM® DB2® Workload Manager [6], Teradata® Active System Management [7], Microsoft® SQL Server Resource and Query Governor [8] [9] and Oracle® Database Resource Manager [10]. The workload management facilities manage complex workloads (*e.g.*, a mix of business processing and analysis requests) present on a data server using predefined procedures. The procedures impose proper controls on the requests, based on the request's characteristics such as estimate costs, resource demands, or execution time, to achieve their required performance objectives.

Recent research [11] [12] shows that the process of workload management in DBMSs may involve three typical controls, namely admission, scheduling, and execution control. Admission control determines whether or not an arriving request can be admitted into a database system, thus it can avoid increasing the load while the system is busy. Request scheduling determines the execution order of admitted requests based on some criteria, such as the request's level of business importance and/or performance objectives. Execution control dynamically manages some running requests to limit their impact on other concurrently running queries. In this paper, we demonstrate our techniques used for workload management in DBMSs.

In 2001, IBM presented the concept of autonomic computing [3]. The initiative aims to provide the foundation for computing systems to manage themselves according to high-level objectives, without direct human intervention in order to reduce the burden on the system administrators. An autonomic computing system (*i.e.*, a self-managing system) has four fundamental properties, namely *self-configuring*, *self-optimizing*, *self-protecting* and *self-healing*. Self-configuring means that a system is able to configure itself automatically to allow the addition and removal of system components or resources without system service disruptions. Self-optimizing means that a system automatically monitors and controls its resources to ensure optimal functioning with respect to the specified performance goals. Self-protecting means that a system is able to proactively identify and protect itself from arbitrary attacks. Self-healing means that a system is able to recognize and diagnose deviations from normal conditions and take action to normalize them [3] [4].

In the past decade, autonomic computing has been intensively studied. Many autonomic computing components (with some self-managing capabilities) have been developed and proven to be useful in their own right, although a large-scale fully autonomic computing system has not yet been realized [13] [14]. In particular, Tesauro *et al.* [15] and Walsh *et al.* [16] studied autonomic resource allocation among multiple applications based on optimizing the sum of the utilities for each application. In their work, a data center consisting of multiple and logically separated application environments (AEs) was used. Each AE provided a distinct application service using a dedicated, but dynamically allocated, pool of servers, and each AE had its own service-level utility function specifying the utility to the data center from the environment as a function of some service metrics. The authors compared two methodologies, a queuing-theoretic performance model and model-free reinforcement learning, for estimating the utility of resources.

Bennani *et al.* [17] presented another approach for the same resource allocation problems in the autonomic data center. They observe that the table-driven approach proposed by Walsh *et al.* [16] has scalability limitations with respect to the number of transaction classes in an AE, the number of AEs, and the number of resources and resource types. Moreover, they claim that building a table from experimental data is time consuming and has to be repeated if resources are replaced within the data center. They instead proposed using predictive multi-class queuing network models to implement the service-level utility functions for each AE. In this paper, we show the principles of autonomic computing applied in workload management for DBMSs, and applications of utility functions in building autonomic workload management systems.

III. UTILITY FUNCTIONS IN WORKLOAD MANAGEMENT

Achieving autonomic workload management for DBMSs can involve the use of utility functions. In this section, we consider the following questions:

- Why are utility functions appropriate for autonomic workload management?
- What utility functions are most suitable (*i.e.*, what properties does a utility function need to possess) for autonomic workload management?

The first question can be answered based on the research of Kephart *et al.* [5] and Walsh *et al.* [16], who proposed the use of utility functions to achieve self-managing systems. In their work, the authors presented utility functions as a general, principled and pragmatic way of representing and managing high-level objectives to guide the behavior of an autonomic computing system. Two types of policies were discussed in guiding behavior of a system, namely action policies and goal policies. An action policy is a low-level policy that is represented in the form of *IF (conditions) THEN (actions*). Namely, if some conditions are satisfied, then certain actions must be taken by the system. In contrast with an action policy, a goal policy only expresses high-level objectives of a system, and the system translates the high-level objectives into specific actions for every possible condition. Utility functions are proposed for the translation as they are capable of mapping system states to real numbers with the largest number representing a system's preferred state. In using utility functions, a computing system, via maximizing its utilities under each condition, recognizes what the goal states are, and then decides what actions it needs to take in order to reach those states. Thus by maximizing utilities, a computing system optimizes its own behavior and achieves the specified high-level objectives.

As introduced in Section I, in a mixed request data server environment, the concurrently running requests can have different types, levels of business importance, performance objectives and arrival rates. These properties may dynamically change during runtime rendering it impossible for human administrators to manually make an optimal resource allocation plan for all workloads in order to meet their resource requirements. A utility function, however, is suited for this situation, based on the properties discussed above. It dynamically identifies resource preferences for a workload during runtime, and the utility functions of the workloads can be further used to define an objective function. A solution to optimizing the objective function is an optimal resource allocation plan. Autonomic workload management systems use the resource allocation plan to allocate resources to the workloads and to achieve the required performance objectives. Thus, to manage workloads in DBMSs, using utility functions is naturally a good choice.

To answer the second question, we begin by discussing performance behavior of a workload. The performance of a running workload on a data server depends on the amount of desired system resources that the workload can access. Typically, the performance of a workload increases non-linearly with additional resources assigned to it. As an example, in executing a workload in an OLTP system, by increasing the multi-programming levels, the throughput of the workload initially increases, but at a certain point the throughput starts to level off. That is, at the beginning when the workload starts to run with a certain amount of resource allocated, performance of the workload increases rapidly. However, with additional resources allocated to the workload, the performance increment of the workload becomes very small. This can be caused either by a bottleneck resource among the system resources, such as too small buffer pools, which significantly limits the workload performance increase, or it may be the case that the database system has become saturated (*e.g.*, system CPU resource is fully utilized).

Utility functions in database workload management must capture the performance characteristics of a workload and represent the trend of the changes in performance based on the amount of assigned resources. A utility function defined for database workload management should be a monotonically non-decreasing function, and it should be capable of mapping the performance achieved by a workload with a certain amount of allocated resources into a real number, u.

There is no single way to define a utility function. However, we believe the following properties are necessary for an effective utility function in autonomic workload management for DBMSs:

- The value, u, should follow the performance of a workload. Namely, it should increase or decrease with the performance.
- The amount of change in the utility should be proportional to the change in the performance of a workload.
- The input to a utility function should be the amount of resources allocated to a workload, or a function of the resource allocation, and the output, u, should be a real number without unit.
- The value, u, should not increase (significantly) as more resources are allocated to a workload, once the workload has reached its performance objective.
- In allocating multiple resources to a workload, a utility function should capture the impact of the allocation of a critical resource on performance of the workload.
- For objective function optimization, a utility function should have good mathematical properties, such as an existing second derivative.

The first two properties describe the general performance behavior of a workload that a utility function needs to capture, and the third property presents the domain and codomain of a defined utility function. These three properties are fundamental for a utility function that can be used in building autonomic workload management in DBMSs. The fourth and fifth properties represent the relationships among workload performance, resource provisions, and performance objectives. Namely, if a workload has met its required performance objective, the value produced by the utility function would not increase (significantly) as additional resources are allocated to the workload. So, by checking the marginal utility (the value is very small), the database system can know it should stop allocating additional resources to the workload. If there is a critical resource for a workload, then the utility function

should reflect the impact of changes to the allocation of that resource. The database system then knows to provide the resource to the workload for meeting its performance objective. The last property provides a way of effectively optimizing objective functions.

IV. UTILITY FUNCTION EXAMPLES

Two examples from our work of the use of utility functions in autonomic workload management for DBMSs are presented in this section. The first example demonstrates Dynamic Resource Allocation, which is driven by workload business importance policies [18]. The second example shows a Query Scheduler managing the execution order of multiple classes of queries [19]. The two utility functions are discussed with respect to the properties listed in Section III.

A. Dynamic Resource Allocation

In workload management for DBMSs, dynamic resource allocation can be triggered by workload reprioritization (a workload execution control approach) [6] [18]. That means a workload's priority may be dynamically adjusted as it runs, thereby resulting in immediate resource reallocation to the workload according to the new priority.

Two shared system resources are considered in the study, namely buffer pool memory pages and CPU shares, as they are key factors in DBMS performance management. The DBMS concurrently runs multiple workloads, which are classified in different business importance classes with unique performance objectives. A certain amount of the shared resources is allocated to a workload according to its business importance level. High importance workloads are assigned more resources, while low importance workloads are assigned fewer. The resource allocation is based on an *economic model* [18]. Namely, the DBMS conducts "auctions" to sell the shared system resources, and the workloads submit "bids" to buy the resources via an auctioning and biding based trade mechanism. All the workloads are assigned some virtual "wealth" to reflect their business importance levels. High importance workloads are assigned more wealth than low importance ones.

The dynamic resource allocation approach consists of three main components, namely the *resource model*, the *resource allocation method* and the *performance model*. The *resource model* is used to partition the resources and to determine an available total amount of the resources for allocation. We consider that each competing workload is assigned its own buffer pool, so buffer pool memory pages can be directly assigned to a workload. The CPU resources, on the other hand, cannot be directly assigned to a workload, so we partition CPU resources by controlling the number of database agents that are available to serve requests on a database server. In our study, we use a DB2 DBMS and configure it such that one database agent maintains one client connection request from the workloads. We conducted experiments and verified the relationship between the number of database agents and system CPU utilization of a workload, and observed that the more database agents that are allocated to serve requests for a particular workload, the more CPU resources the workload receives [18]. The available total amounts of resources are parameters in the resource allocation approach, so it can adapt to different system configurations.

The *resource allocation method* determines how to obtain an optimal resource pair of buffer pool memory pages and CPU shares for a workload in order to maximally benefit the workload performance. Namely, a workload needs to capture the resources in an appropriate amount such that none of the resources become a bottleneck resource. In our approach, a greedy algorithm is used for identifying resource preferences of a workload in a resource allocation process. The resource allocation is determined iteratively. In an iteration of the algorithm, by using its virtual *wealth*, a workload bids for a unit of the resource (either buffer pool memory or CPU) that it predicts will yield the greatest benefit to its performance. Figure 1 shows a representation of the search state space for the allocation of buffer pool memory and CPU to a workload in our experiments, as described in Section V. The starting node, $n_{1,1}$, represents the minimum resource allocation to a workload, namely one unit of buffer pool memory and one unit of CPU, at the beginning of a resource allocation process. The workload then traverses the directed weighted graph to search for the optimal <*cpu*, *mem*> pair in order to achieve its performance objective.

The *performance model* predicts the performance of a workload with certain amount of allocated resources in order to determine the benefit of the resources. In our approach, queuing network models (QNM) [20] are used to predict performance of a workload at each step of the algorithm, that is, to assign the weights to the edges of the graph in Figure 1. We consider OLTP workloads and use throughput as the performance metric to represent the performance required and achieved by the workloads. We model the DBMS used in our experiments for each workload with a single-class closed QNM, which consists of a CPU service center and an I/O service center. The CPU service center represents the

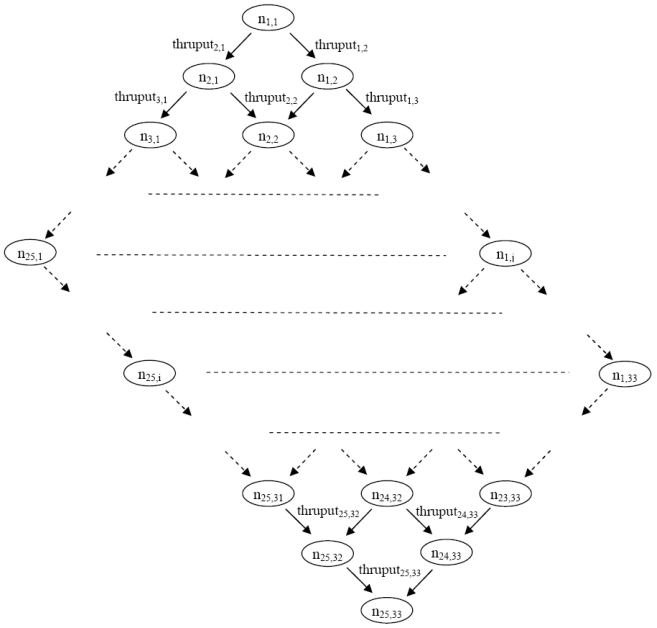


Figure 1. Resource Pair Search State Space

system CPU resources and the I/O service center represents buffer pool and disk I/O resources. The request concurrency level of a workload in the DBMS is the number of database agents (*i.e.*, CPU resources) assigned to the workload. The average CPU service demand of requests in the workload can be expressed as a function of the CPU shares allocated to the workload, using equation (1).

$$S_{CPU} = 1/(a * n + d) \quad (1)$$

We experimentally defined the relationship between the CPU service demand and the number of database agents used in a DBMS. In the equation (1), n is database agents, $n \in \mathbf{N}$, and a and d are constants, $a \in \mathbf{R}^+$, $d \in \mathbf{R}^+$, that can be determined through experimentation.

For an OLTP workload, the average I/O service demand can be expressed as a function of buffer pool memory size, which can be derived from *Belady's* equation [21]. The I/O service demand is:

$$S_{IO} = c * m^b \quad (2)$$

where c and b are constants, $c \in \mathbf{R}^+$, $b \in \mathbf{R}^-$, and m is buffer pool memory pages assigned to the workload, and $m \in \mathbf{N}$. In the equation the constants c and b can be determined through experimentation.

Performance of a workload with some allocated resources, <*cpu, mem*>, can be predicted by solving this analytical performance model (*i.e.*, the QNM) with Mean Value Analysis (MVA) [20]. The predicted throughput of a workload can be expressed as a function of its allocated resources, using equation (3).

$$X = MVA(n, S_{CPU}(cpu), S_{IO}(mem), Z) \quad (3)$$

where, X is the predicted throughput of a workload by using MVA on the QNM for a workload with its allocated resource pair, <*cpu, mem*>; n is the number of requests from the workload concurrently running in the system (*i.e.*, the number of database agents assigned to the workload); $S_{CPU}(cpu)$ is the average CPU service demand determined in equation (1); $S_{IO}(mem)$ is the average I/O service demand determined in equation (2); and Z is think time.

To guide workloads to capture appropriate resource pairs, utility functions are employed in the approach. We define a utility function that normalizes the predicted throughput from the performance model relative to the maximum throughput that the workload could achieve when all the resources are allocated to it. The utility function is given by:

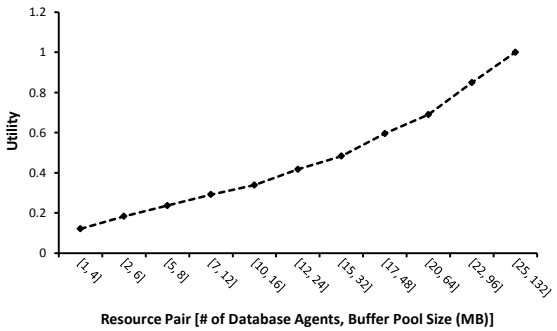


Figure 2. Sample Curve of Utility Function in Resource Allocation

$$u = MVA_{throughput}(n, S_{CPU}(cpu), S_{IO}(mem), Z)/X_{max} \quad (4)$$

where, $MVA_{throughput}(n, S_{CPU}(cpu), S_{IO}(mem), Z)$ is the predicted throughput determined in equation (3), and X_{max} is the maximum throughput achieved by a workload with all the resources allocated, which can be determined through experimentation.

This utility function, as shown in Figure 2, maps performance achieved by a workload given a certain amount of resources into a real number u, $u \in [0, \ldots, 1]$. If the utility of resources allocated to a workload is close to 1, it means the performance of the workload is high, while if the utility of resources allocated to a workload is close to 0, it means the performance of the workload is low. Workloads employ the utility function to calculate marginal utilities, that is, the difference in utilities between two possible consecutive resource allocations in a resource allocation process. As the utility function is non-decreasing, the value of a marginal utility is also in the range [0, …, 1].

The marginal utility reflects potential performance improvement of a workload. For some resources, if the calculated marginal utility of a workload is close to 1, then it means these additional resources can significantly benefit the workload's performance, while if the calculated marginal utility is close to 0, then the additional resources will not greatly improve the workload's performance. By examining the marginal utility value, a workload can determine the preferred resources for bid. The bid of a workload is the marginal utility multiplied by current available wealth of the workload, and indicates that a workload is willing to spend the marginal-utility percentage of its current wealth as a bid to purchase the resources. Wealthy workloads, therefore, can acquire more resources in the resource allocation processes. A workload ceases bidding for additional resources when it has reached its performance objective.

B. Query Scheduling

Our *query scheduler* [19] is built on a DB2 DBMS and employs DB2 Query Patroller (DB2 QP) [6] (a query management facility) to intercept newly arriving queries. Information about the queries is then acquired, and the scheduler determines an execution order for the queries. The *query scheduler* works in two main processes, namely the *workload detection* and the *workload control*. The workload detection process classifies arriving queries based on their service level objectives (SLOs), and the workload control process periodically generates new plans to respond to the changes in the SLOs of arriving requests.

In the query scheduler's architecture shown in Figure 3, DB2 QP is set to inform the *query scheduler*'s *monitor* when an arriving query has been intercepted. The *monitor* collects information about the query from the DB2 QP control tables, which includes query identification, query costs and query execution information, and passes the query's information to the *classifier* and the *scheduling planner*. The *classifier* assigns the query to a service class based on its performance goals and puts the query in a queue, which is associated with the service class and managed by the *dispatcher*. The *dispatcher* receives a

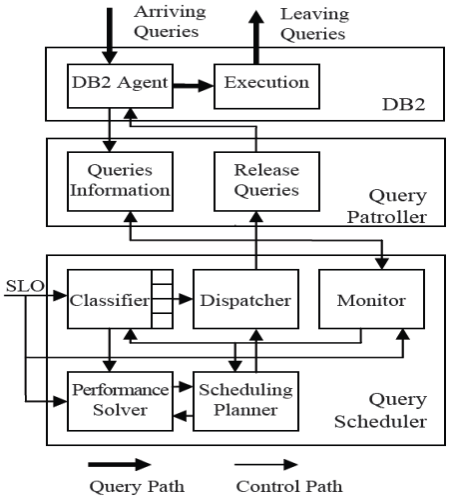


Figure 3. Architecture of Query Scheduler

scheduling plan from the *scheduling planner* and releases the queries in the queues according to the plan's specifications. The *scheduling planner*, given SLOs, receives query information from the *monitor*, and consults the *performance solver* to make a scheduling plan for all the queued queries.

We consider a system with *n* service classes for arriving requests, each with a performance goal and a level of business importance, denoted as $\langle \overline{g}_i, m_i \rangle$, where $\overline{g}_i$ is the performance goal of the *i-th* service class, and m_i is the class business importance level. The pair $\langle \overline{g}_i, m_i \rangle$ is a service level objective. We denote $g_1, g_2, \cdots, g_n$ as the predicted performance of the *n* service classes given a resource allocation plan $r_1, r_2, \cdots, r_n$ (*i.e.*, multi-programming levels in our case). The performance of the *i-th* service class, g_i, can be predicted by using a performance model (queuing network models [20] are used in our study) given r_i, the amount of resources allocated to the service class. The utility of the *i-th* service class, u_i, can be expressed as a function of $\overline{g}_i$, m_i and g_i, namely $u_i = f_i(\overline{g}_i, m_i, g_i)$, and the *n* SLOs can be encapsulated into an objective function $f(u_1, u_2, \cdots, u_n)$. Thus, the scheduling problem can be solved by optimizing the objective function *f*.

We specifically consider business analysis requests, such as those found in decision support systems. In emulating the environment, we use the TPC-H benchmark [22] as the database and workloads in our experiments. Since queries in decision support systems can widely vary in their response times, we employ the performance metric *query execution velocity*, which is the ratio of expected execution time of a query to the actual time the query spent in the system (*i.e.*, the total time of execution and delay), to represent the performance required and achieved by the queries. Query execution velocity captures both the performance goals and the business importance levels of queries.

Through our experiments we found the following general form of utility functions satisfies our requirements:

$$u = 1 - \mathrm{e}^{\mathrm{am}\frac{\overline{g}-g}{g-\hat{g}}} \quad (5)$$

where, $\overline{g}$ is the performance goal of a service class to be achieved, m is the importance level of the service class, $m \in \mathbf{N}$, $\hat{g}$ is the lowest performance allowed for the service class, g is the actual performance, and a is an importance factor that is a constant, $a \in \mathbf{N}$, and can be experimentally determined or adjusted to reflect the distance between two adjacent importance levels. In using a, we control the size and shape of the utility function, as shown in Figure 4.

The objective function, *f*, is then defined as a sum of the service class utility functions, using equation (6):

$$f = \sum_{i=1}^{n} u_i \quad (6)$$

In *query scheduler*, the *performance solver* employs a performance model to predict query execution velocity for a service class. That is, given a new value of service class cost limit, the performance of the service class can be predicted for the next control interval, which is based on its performance and service class cost limit at the current control interval. The performance at the next control interval is predicted by:

$$V_i^k = \begin{cases} V_i^{k-1} C_i^k / C_i^{k-1} & if\ V_i^{k-1} C_i^k / C_i^{k-1} \le 1 \\ 1 & if\ V_i^{k-1} C_i^k / C_i^{k-1} > 1 \end{cases} \quad (7)$$

where, V_i^{k-1} and V_i^k are query execution velocity of service class *i* at *(k-1)-th* and *k-th* control intervals, respectively; C_i^{k-1} and C_i^k are cost limits of service class *i* at the *(k-1)-th* and the *k-th* control intervals, respectively.

Therefore, a scheduling plan can be determined. From equations (5), (6) and (7), we have:

$$f = \sum_i^n u_i^k \quad (8)$$

$$u_i^k = 1 - e^{a_i m_i \frac{\overline{V}_i - V_i^k}{V_i^k - \hat{V}_i}} \quad (9)$$

$$V_i^k = V_i^{k-1} C_i^k / C_i^{k-1} \quad (10)$$

replacing V_i^k in equation (9) with equation (10) and u_i^k in equation (8) with equation (9), the solution for maximizing the objective function, $f(C_1^k, C_2^k, \cdots, C_n^k)$, is the query scheduling plan for *k-th* control interval, where the object function must maintain the constraint, $C_1^k + C_2^k + \cdots + C_n^k \le C$, and C is the system cost limits.

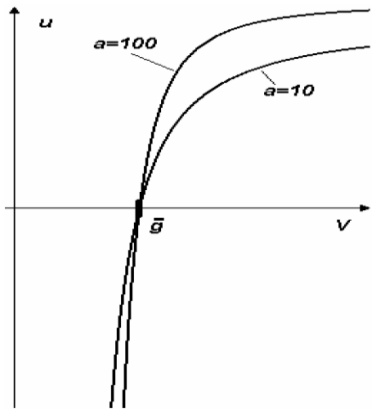


Figure 4. Sample Curves of Utility Function in Query Scheduling

V. EXPERIMENTS

The experimental objective was to validate the utility functions defined in our studies of autonomic workload management for DBMSs. We developed a dynamic resource allocation simulator and implemented a prototype of the Query Scheduler to examine whether the utility functions can effectively guide the dynamic resource allocation and query scheduling actions in accordance with a given high-level workload business importance policy. We present the results of experiments run using the simulator and the prototype in Subsection *A* and *B*, and discuss the two utility functions in Subsection *C*.

A. Experiments for Dynamic Resource Allocation

To allocate the buffer pool memory and CPU resources, we first experimentally determined the appropriate amount of total resources for a given data server as well as set of workloads. Our experiments were conducted with DB2 database software [6] running on an IBM xSeries® 240 PC server with the Windows® XP operating system. The data server was equipped with two Pentium® processors, 2 GB of RAM and an array of 11 disks. The databases and workloads were taken from the TPC-C benchmark [22]. The size of the database was 10GB. The three workloads were similar to TPC-C OLTP batch workloads.

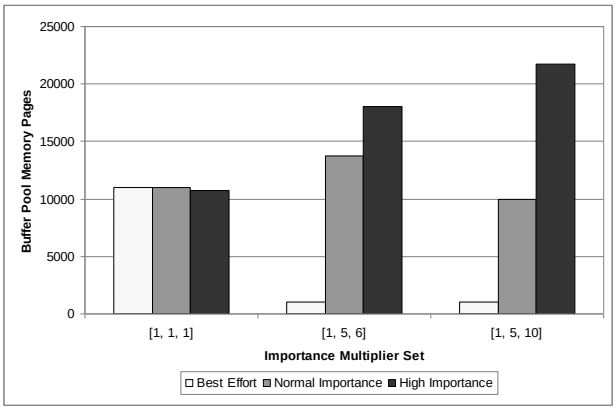


Figure 5. Buffer Pool Memory Allocation for Different Business Importance policies

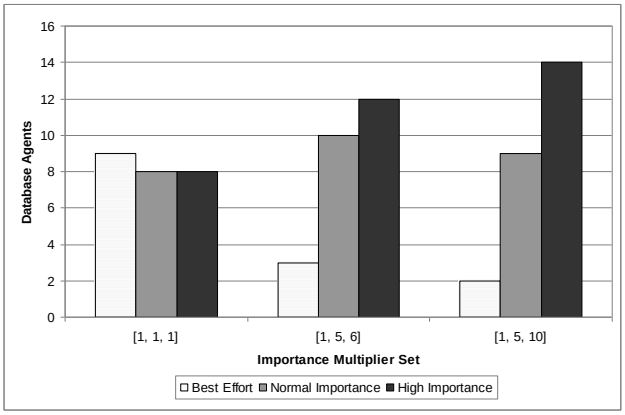


Figure 6. Database Agent Allocation for Different Business Importance policies

We consider the case of a single DB2 instance with three identical databases for three competing workloads from different importance classes. Each database has one workload running on it, thus each workload has its own buffer pool and CPU shares while still having accesses to all the same database objects. Our dynamic resource allocation technique allocates buffer pool memory space and CPU (*i.e.*, database agents) resources across the three identical databases based on a given workload business importance policy.

We selected a minimum amount of each resource (*i.e.*, buffer pool memory and CPU) where maximum system performance was achieved. We experimentally determined 32,768 buffer pool memory pages as the total buffer pool memory and 25 database agents as the total CPU resources [18]. We use 1,000 buffer memory pages as one unit of buffer pool memory and 1 database agent as one unit of CPU resources in our resource allocation experiments (as discussed in the following paragraphs) as these granularities give a reasonable workload performance increment and make the resource allocation process efficient.

We developed a simulator of our dynamic resource allocation approach to generate the resource allocations for competing workloads on a DBMS based on a given workload business importance policy. The simulator was written in Java™ and the three workloads (*i.e.*, the TPCC-like OLTP batch workloads) were used as the simulator input. The output of the simulator was resource allocations, that is, a list of the number of buffer pool memory pages and database agents for each of the workloads.

A set of experiments was conducted to determine whether our approach generates the resource allocations which match a given workload business importance policy. The workloads were assigned one of three different importance classes, namely the *high importance* class, the *normal importance* class, and the *best effort* class. The relative importance of the classes was captured by a set of importance multipliers for the base wealth assigned to the classes. We experimented with three different sets of importance multipliers that were of the form [best effort, normal importance, high importance]: [1, 1, 1], [1, 5, 6], and [1, 5, 10]. The multiplier sets were chosen to demonstrate the effect of business importance policies on the resource allocations.

Figures 5 and 6 respectively show buffer pool memory page and database agent (representing system CPU resources) allocations produced by the simulator using the three workload business importance multiplier sets. The workload importance multiplier set [1, 1, 1] represents the case where the three competing workloads are from three different business importance classes of equal importance. In this case, the three workloads are allocated approximately the same amount of buffer pool memory and CPU resources as shown in Figures 5 and 6. Using the importance multiplier set [1, 5, 6], the high importance and the normal importance classes are much more important than best effort class, and the high importance class is also slightly more important than the normal important class. When the simulator is used to allocate resources in this case, the high importance and normal importance workloads are allocated

significantly more resources than the best effort workload, while the high importance workload is allocated slightly more resources than the normal importance workload. The set [1, 5, 10] represents the case where the high importance class is much more important than the normal importance class, and the normal importance class is much more important than the best effort class. In this case, the high importance workload is allocated more resources than the normal important workload, and the normal importance workload wins significantly more resources than the best effort workload.

By observing the experimental results shown in Figure 5 and Figure 6, we have that the defined utility functions (the key components of the dynamic resource allocation) can effectively guide the resource allocation processes and generates resource allocations for the competing workloads which match the given workload business importance policies (that is, more important workloads assigning more resources than less important ones).

B. Experiments for Query Scheduling

The same data server, as described in Subsection *A*, was used in the experiments. Our experiments were conducted with DB2 database software as well as DB2 Query Patroller as a supporting component [6]. The database and workloads were taken from the TPC-H benchmark [22]. The size of the database was 500MB, and two workloads that consisted of TPC-H queries were submitted by interactive clients with zero *think time* [20]. Each workload was assigned to a service class described in Section *IV-B*, namely either *class 0* or *class 1*, with a different business importance level and a unique performance goal, where we considered *class 0* is more important than *class 1*. The intensity of a workload in the data server was controlled by the number of clients used by the workload. Each experiment was run for 12 hours that consisted of 6 *2-hour periods* (as shown in Figure 7, 8 and 9).

To evaluate whether our Query Scheduler can manage multiple classes of workloads towards their performance goals based on given workload business importance policies, we first need to determine the *total cost limits*, as mentioned in Section *IV-B*, for the DBMS and workloads. Thus, we experimentally determined 300,000 *timerons*, a measure unit for the resources required by the DB2 database manager to execute the plan for a query [6], as the *total cost limits* in our query scheduling experiments [19].

The first experiment was conducted to show performance of the workloads without control and served as the baseline measure to observe how the performance of the workloads changes as they run. The performance goals of *query execution velocity*, as described in Section *IV-B*, for the workload (belonging to *class 0*) and the workload (belonging to *class 1*) were set as 0.65 and 0.45, respectively. The results are shown in Figure 7. It shows that the "class 0" workload missed its performance goal in *periods* 2 and 3, and the "class 1" workload over performed almost all the time in the experiment.

The experiments were then conducted using our Query Scheduler to control the workloads. The performance goals for *class 0* and *class 1* were still 0.65 and 0.45, respectively. The results are shown in Figure 8. The dynamic adjustment of *service class cost limits* to achieve the performance goals is shown in Figure 9. The experimental results show that our Query Scheduler can provide differentiated services for competing workloads. As shown in Figure 8, for the Query Scheduler, the "class 0" workload could better meet its performance goal than the "class 1" workload, which was in accordance with the given importance policy. Although the

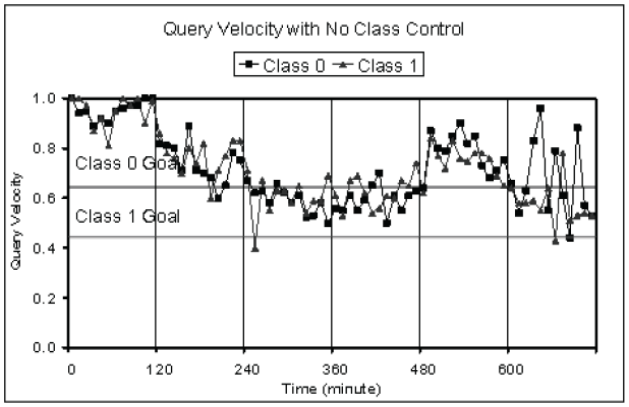


Figure 7. No Service Class Control for Competing Workloads

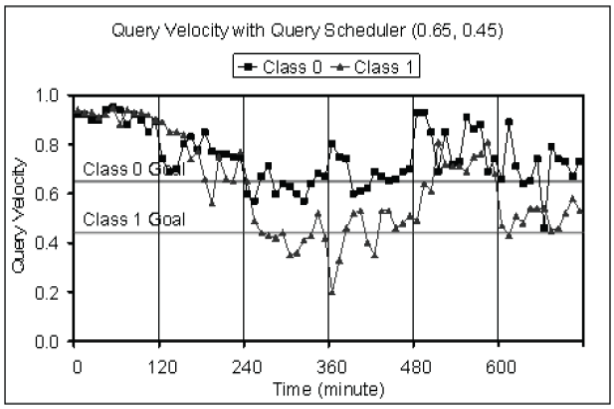


Figure 8. Query Execution Velocity for Multiple Competing Workloads

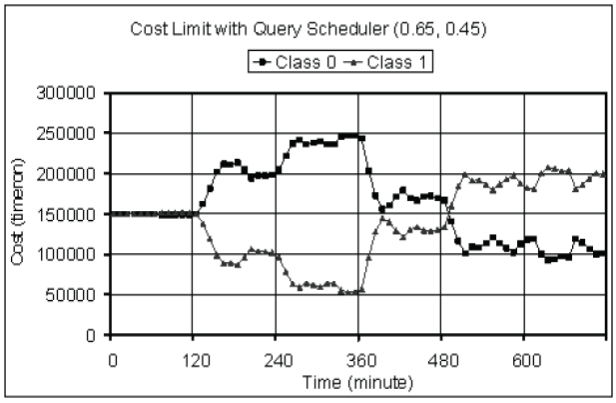


Figure 9. Dynamically Assigned Service Class Cost Limits for Multiple Competing Workloads

Query Scheduler gave preference to the important class, *class 0*, it never allocated too many resources (*i.e.*, multi-programming levels, discussed in Section *IV-B*) *class 0* to prevent *class 1* from meeting its performance goal. When the workloads were too heavy to meet both performance goals as shown at *periods* 3 and 4 in Figure 8, Query Scheduler was still able to help both classes approach their goals. From Figure 9, we can observe that our Query Scheduler dynamically adjusts the *service class cost limits* according to the workload changes. The amount of resources allocated to a class is based on its need in order to meet its performance goal, as shown in Figure 9.

By observing the experimental results shown in Figures 7, 8 and 9, we have that our Query Scheduler is able to respond to query changes and give preference to the queries assigned to an important service class, and to the service class whose performance goals are violated. These results also validate the utility functions as they are key components defined in the Query Scheduler. The results show that the utility functions effectively guide the Query Scheduler to dynamically generate query scheduling plans for competing workloads bases on a given workload business importance policy with more important workloads receiving more shared system resources than less important ones.

C. Discussion

In dynamic resource allocation, the utility function was defined based on a single-class multi-center closed QNM, while in query scheduling, the utility function was chosen based on an exponential function. These two types of utility functions are different in their forms and research requirements, but both strictly maintain the same fundamental properties listed in Section III.

The input to the dynamic resource allocation utility function is an amount of allocated resources (*i.e.*, the resource pair, <*cpu*, *mem*>), the output is a real number in the range [0, …, 1], and the applied QNM properly predicts performance behavior of the workload. A workload ceases bidding for additional resources using assigned virtual *wealth* when it has reached its performance objective.

In query scheduling, the input to the utility functions is the query execution velocity of the service classes predicted by the performance model given a level of allocated resources and the output is a real number in ($-\infty$, $+\infty$). Based on the exponential function properties, as the input of the utility function increases, the output (*i.e.*, the utility) increases and at a certain value, it begins to level off. That means, when the service class approaches its performance goal, the utility increase is less, and it indicates that the database system should not assign more resources to the service class.

If an objective function is continuous, the *Lagrange* method can be applied to solve it [19], otherwise searching techniques may be used. In query scheduling, the second derivative of the utility function exists and this allows mathematical methods to be applied to optimize the objective function. In dynamic resource allocation, instead of defining an objective function based on the utility functions, *economic models* (the use of virtual wealth and auctions and bids) [18] are applied to coordinate the utility functions to allocate the shared system resources to competing workloads.

TABLE I. COMPARISION OF THE TWO UTILITY FUNCTIONS

	Utility functions in Dynamic Resource Allocation	**Utility functions in Query Scheduling**
Utility Increasing Normally	yes	yes
Marginal Utility Increasing Normally	yes	yes
Utility Function Input	allocated resources	a function of the allocated resources
Utility Function Output	a number in [0…1]	a number in **(**$-\infty$**,** $+\infty$**)**
Critical Resource Identifying	yes	no
Having Mathematical Property	no	yes
Utility Increase Stops as Goals Achieved	yes	yes

In evaluating the two types of utility functions (using the set of properties listed in Section III), both utility functions preserve the fundamental properties, that is, *a)* the utility increases as a workload performance increases, and decreases otherwise; *b)* the marginal utility is large as a workload performance increases quickly, and is small otherwise; *c)* the input and output are in the required types and values. In comparing the two utility functions presented in Table 1, we observe that the utility function used in dynamic resource allocation has the property of identifying critical resources for a workload, but it does not have mathematical properties for optimizing objective functions (as there is not an objective function defined in the approach). The utility function used in query scheduling possesses a good mathematical property for optimizing its objective function, but it does not have the property of identifying system critical resources (as it is not necessary to identify critical resources in the problem). In Table 1, *Utility Increasing Normally* means whether the utility increases as a workload performance increases, and decreases otherwise, and *Marginal Utility Increasing Normally* means whether the marginal utility is large as a workload performance increases quickly, and is small otherwise.

Since the utility functions were strictly defined based on their research requirements, the specific research problems shaped the utility function's properties. So, we conclude (based on the properties listed in Section III) that the two types of utility functions are good in terms of their specific research requirements and considered acceptable based on the set of properties listed in Section III.

VI. CONCLUSION AND FUTURE WORK

In this paper, we have presented two concrete examples to illustrate how utility functions can be applied to database workload management, namely dynamic resource allocation and query scheduling. Based on the examples, we generalized a set of function properties that are fundamental for defining utility functions in building autonomic

workload management for DBMSs in future practice and research. Through experiments, we validated the utility functions defined in our studies of autonomic workload management for DBMSs.

As more workload management techniques are proposed and developed, we plan to investigate the use of utility functions to choose during runtime an appropriate workload management technique for a large-scale autonomic workload management system, which can contain multiple techniques. Thus, the system can decide what technique is most effective for a particular workload executing on the DBMS under certain particular circumstance.

ACKNOWLEDGMENT

This research is supported by IBM Centre for Advanced Studies (CAS), IBM Toronto Software Lab, IBM Canada Ltd., and Natural Science and Engineering Research Council (NSERC) of Canada.

TRADEMARKS

DISCLAIMER

The views expressed in this paper are those of the authors and not necessarily of IBM Canada Ltd. or IBM Corporation.

REFERENCES

[1] M. Zhang, B. Niu, P. Martin, W. Powley, P. Bird, and K. McDonald. "Utility Function-based Workload Management for DBMSs". In Proc. of the 7th Intl. Conf. on Autonomic and Autonomous Systems (ICAS'11), Mestre, Italy, May 22-27, 2011, pp. 116-121.

[2] D. P. Brown, A. Richards, R. Zeehandelaar and D. Galeazzi, "Teradata Active System Management: High-Level Architecture Overview", A White Paper of Teradata, 2007.

[3] IBM Corp., "Autonomic Computing: IBM's Perspective on the State of Information Technology". On-line, retreieved in June 2012. http://www.research.ibm.com/autonomic/manifesto/autonomic_computing.pdf.

[4] J. O. Kephart and D. M. Chess, "The Vision of Autonomic Computing", Computer, Volume 36, Issue 1, January 2003, pp. 41-50.

[5] J. O. Kephart and R. Das, "Achieving Self-Management via Utility Functions," IEEE Internet Computing, Vol. 11, Issue 1, January/February, 2007, pp. 40-48.

[6] IBM Corp., "IBM DB2 Database for Linux, UNIX, and Windows Information Center". On-line, retreived in June 2012. https://publib.boulder.ibm.com/infocenter/db2luw/v9r7/index.jsp.

[7] Teradata Corp., "Teradata Dynamic Workload Manager", On-line, retrieved in June 2012. http://www.info.teradata.com/templates/eSrchResults.cfm?prodline=&txtpid=&txtrelno=&txtttlkywrd=tdwm&rdsort=Title&srtord=Asc&nm=Teradata+Dynamic+Workload+Manager.

[8] Microsoft Corp., "Managing SQL Server Workloads with Resource Governor". On-line, retrieved in June 2012. http://msdn.microsoft.com/en-us/library/bb933866.aspx.

[9] Microsoft Corp., "Query Governor Cost Limit Option", On-line, retrieved in June 2012. http://msdn.microsoft.com/en-us/library/ms190419.aspx.

[10] Oracle Corp., "Oracle Database Resource Manager", On-line, retrieved in June 2012. http://download.oracle.com/docs/cd/B28359_01/server.111/b28310/dbrm.htm#i1010776.

[11] S. Krompass, H. Kuno, J. L. Wiener, K. Wilkison, U. Dayal and A. Kemper, "Managing Long-Running Queries", In Proc. of the 12th Intl. Conf. on Extending Database Technology: Advances in Database Technology (EDBT'09), Saint Petersburg, Russia, 2009, pp. 132-143.

[12] A. Mehta, C. Gupta and U. Dayal, "BI Batch Manager: A System for Managing Batch Workloads on Enterprise Data-Warehouses", In Proc. of the 11th Intl. Conf. on Extending Database Technology: Advances in Database Technology (EDBT'08). Nantes, France, March 25-30, 2008, pp. 640-651.

[13] J. O. Kephart, "Research Challenges of Autonomic Computing". In Proc. of the 27th Intl. Conf. on Software Engineering (ICSE'05). St. Louis, MO, USA, 15-21 May, 2005, pp. 15-22.

[14] D. A. Menasce and J. O. Kephart, "Guest Editors' Introduction: Autonomic Computing", In IEEE Internet Computing, Volume 11, Issue 1, January 2007, pp. 18-21.

[15] G. Tesauro, R. Das, W. E. Walsh and J. O. Kephart, "Utility-Function-Driven Resource Allocation in Autonomic Systems", In Proc. of the 2nd Intl. Conf. on Autonomic Computing (ICAC'05), Seattle, Washington, USA, June 13-16, 2005, pp.342-343.

[16] W. E. Walsh, G. Tesauro, J. O. Kephart and R. Das. "Utility Functions in Autonomic Systems", In Proc. of the 1st Intl. Conf. on Autonomic Computing (ICAC'04), New York, USA, 17-18 May, 2004, pp.70-77.

[17] M. N. Bennani and D. A. Menasce, "Resource Allocation for Autonomic Data Centers using Analytic Performance Models", In Proc. of the Intl. Conf. on Autonomic Computing, (ICAC'05), Seattle, Washington, USA, 13-16 June, 2005, pp. 229-240.

[18] M. Zhang, P. Martin, W. Powley and P. Bird. "Using Economic Models to Allocate Resources in Database Management Systems", In Proc. of the 2008 Conf. of the Center for Advanced Studies on Collaborative Research (CASCON'08), Toronto, Canada, Oct. 2008, pp. 248-259.

[19] B. Niu, P. Martin and W. Powley, "Towards Autonomic Workload Management in DBMSs", In Journal of Database Management, Volume 20, Issue 3, 2009, pp. 1-17.

[20] E. Lazowska, J. Zahorjan, G. S. Graham and K. C. Sevcik "Quantitative System Performance: Computer System Analysis Using Queueing Network Models", Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1984.

[21] L. A. Belady. "A Study of Replacement Algorithms for a Virtual-Storage Computer". IBM Systems Journal, Volume 5, Issue 2, June 1966, pp. 78-101.

[22] Transaction Processing Performance Council. On-line, retrieved in Feb. 2012. http://www.tpc.org.

Multiuser Simulation-Based Virtual Environment for Teaching Computer Networking Concepts

Ammar Musheer, Oleg Sotnikov and Shahram Shah Heydari
University of Ontario Institute of Technology
Oshawa, Ontario, Canada
ammar.musheer@mycampus.uoit.ca, oleg.sotnikov@mycampus.uoit.ca, shahram.heydari@uoit.ca

***Abstract*—In this research we focus on primary design principles for creating interactive activities and serious games based on Cisco Packet Tracer tool for the purpose of training in the field of Computer Networks. We present a general architecture for a library of simulation-based interactive activities and games for CCNA-level content. We demonstrate how students' skills in important networking topics such as routing, remote access and security help them succeed in these activities. We discuss various challenges in setting up efficient multi-user environments based on Packet Tracer, and propose solutions for a number of technical problems such as scalable addressing, VLAN handling, individual student monitoring, and offline exporting. Conceptual collaborative activities and their design principles are also discussed in this research. We also provide testing results to evaluate the performance of packet trace in an interactive multiuser environment.**

Keywords-component; IT learning tools; Serious Gaming; Cisco Networking Academy; Blended Learning.

I. INTRODUCTION

A. Background

Advances in personal computer technology, as well as widespread access to the Internet, have allowed development of new and more efficient approaches for teaching environments. Technology has greatly facilitated the transition from a traditional lecture-style teaching environment in which the teacher had the central role in educational activities and student were merely listener, to a knowledge-centered environment in which students are the main actors and share their process of learning while the teacher mostly plays the role of a mentor and facilitator [1] . Technology often becomes the main tool for creating such environment, and numerous researches have indicated that a technology-oriented, knowledge-centered environment increases student attendance, motivation and self-reliance in the learning process [1].

The use of technology in learning environments provides the opportunity to create virtual collaborative learning environments in which educational interactions between students occur in a designed information space [2] in form of shared online activities. Such interactions must have a carefully designed structure that would promote collaborations and provide incentives for specific learning achievements and milestones. Educational games (a.k.a. serious gaming), for instance, provide such opportunities. In particular, the maturing of the video gaming industry has triggered significant interest in developing simulated and interactive serious gaming platforms. The main advantage of serious gaming platforms is that they combine in-depth educational topics with goal-oriented and realistic scenarios. To date there have been many applications of serious gaming, spanned over many industries, from urban planning to business, military, healthcare training etc. These applications have provided the means to an individual, or a group of individuals, to engage in an artificial conflict, assess, and learn the complexities associated with each of their individual area of work. Some real world applications of serious games include INNOV8 Business Process Management simulator by IBM [3], physical conditioning games for the general public, e.g. Fitness Shape Evolved for the new Microsoft Kinect hardware platform [4], Foldit for biologists [5], and flight simulator applications for the military [6]. Serious gaming has allowed users of the application to attain skills and knowledge related to a specific activity. The subject can be as difficult as addressing physical and physiological disorders, or be as simple as promoting physical activity. The main goal of serious gaming is to provide a powerful means of encouraging people to learn and provide them with a more entertaining way to obtain the skills and knowledge addressed by the specific serious gaming activities.

B. Computer Networks Training

Interactive learning is particularly becoming more prominent and shows great potential in teaching IT concepts [7] [8]. The introductory computer networking education in colleges and universities typically focuses on teaching students both the theory and the practical knowledge about networking technology that has rarely been covered at their prior educational levels (high school). Students often have to bring themselves up to speed quickly on a large amount of content that includes the entire OSI model and TCP/IP protocol suite, various routing and switching protocols, and local and wide area network technologies. The sheer volume of technical details in this subject area can quickly result in a dry and non-interactive environment that can impede student learning process. Lab components often improve the dryness of the material by giving students practical examples to reinforce theory. However, more can be gained by increasing the level of interaction during lecture times and in the classroom. This can be achieved by implementing short in-class activities that encourage students to collaborate with their peers throughout the learning process.

In order to create the lab-like interaction in a classroom environment, e-learning tools based on simulation of components and networks have been developed to create a

virtual lab environment. The level of details for modeling each component in such simulators often depends on the desired knowledge and expertise level that the students are expected to achieve. One such tool in the field of computer networking is Cisco's Packet Tracer (PT) simulator [9] that provides a virtual network simulation environment with sufficient details of the network operating system on individual devices that would allow students to test and learn different scenarios in an environment that resembles a real computer network. The PT allows creation of realistic assignments for challenging students' knowledge of networking fundamentals, system configuration and network troubleshooting.

While PT has been used extensively in introductory networking curriculum of the Cisco Networking Academy, in particular for training of networking professionals for industry certifications such as Cisco-Certified Network Associate (CCNA) and Cisco-Certified Network Professional (CCNP) [10], its usage has mostly been limited to individual scenarios in which each student would install and run PT on his/her computer and work independently on the assignment or scenario. This restriction did not allow interactive collaboration among students or between students and the teacher, and thus limited the extent and effectiveness of the virtual learning environment. In 2008, Cisco added a multiuser capability to PT that allowed synchronization and communication between instances of PT that are running on different machines. This new feature has opened the door to develop many interesting new activities such as interactive and dynamic troubleshooting and serious gaming for introductory networking classes.

The objective of this work is to use the new multiuser feature in PT to create in-class interactive learning activities that would enhance students' understanding of complex networking concepts. The multiuser feature in PT allows the students to work in an environment that is affected by their peers and is under control of the instructor. Despite some technological problems, PT multi-user activities can make networking more interesting to learn and lead to greater student engagement. The basic multiuser capability in PT merely allows connection of remote instances of PT on separate machines. However, proper design of the interconnected simulation environment and learning scenarios would require a number of technical and educational considerations that is the subject of this study.

C. Related Work

At the time of writing this paper Cisco has not yet developed any curriculum activities that feature multiuser operation, leaving it up to individual academies. In 2010 the Open University of UK reported on implementation of PT's multiuser functionality into their blended distance learning CCNA courses [11]. Their results offered an extensive guide to the multiuser architecture as well as the implementation of multiuser over the WAN and the inherent problems resulting from network delays and congestion. We build on their work by implementing our multiuser architecture in a traditional classroom through a LAN. This method bypasses the majority of the technical limitations in [11] and gives students more interaction with the class.

Basic technical specifications of the Packet Tracer Messaging Protocol (PTMP) and Inter-Process Communication is available in [11] [12], and while many proprietary details were not available or could not be made public, the available information helped understanding the communication between two hosts running a Packet Tracer multiuser connection.

Cisco has also developed a number of serious games for networking education. Most recently, Cisco ASPIRE Beta 4 was released in January 2012 [13]. This is a strategic game that utilizes the Packet Tracer platform to provide the players a role-playing/SimCity-like serious game focused on the business of networking. Players take on the role of a network engineer who applies his/her entrepreneurial and networking skills to complete several contracts that arise during the game. They must purchase the correct hardware and apply the correct configuration schemes in order to complete the contract. For the correct completion of contracts players receive credits which they can spend toward improving their network. Cisco Systems has also released Cisco myPlanNet 1.0 [14]. Players are put into the shoes of a service provider CEO. They must direct their business through various technological ages ranging from dial-up all the way up to the broadband/mobile connected ages. Along the way, players learn about the various technologies that make the networking world possible. The game provides a SimCity-like overlay where issues arise in the city and must be resolved. There is a credit system that players must use to purchase new technologies to advance their businesses and provide better services to the civilians in their cities. There is also a leaderboard to encourage players to attain higher scores and improve their skills. Additionally, Cisco provides many smaller serious games that allow users to improve their skills, such as: Wireless Explorer, Unified Communications Simulation Challenge, Cisco Mind Share, and Edge Quest. The objective of these activities is to provide players with a means to enjoy learning the complicating subjects related to the networking field, albeit on a smaller scale.

D. Contributions

This paper discusses in detail a project at University of Ontario Institute of technology (UOIT), Oshawa, Canada, to develop interactive learning modules and educational games using the virtual simulation environment of packet trace for use in introductory networking classes at the CCNA certification level. We provide the concepts, a client-server architecture and high level design for such activities. We also summarize technical challenges in efficient design and delivery of such modules, and propose methods to deal with them. We also measure the scalability of multiuser PT-based simulation environments, and provide performance results regarding CPU load, memory usage, offline saving time and network resource requirements for such deployments.

The rest of this paper is organized as following. In Section II a brief description of PT multiuser feature is presented and detailed specifications of the learning modules are discussed. In Section III we will have a close look at several technical challenges faced in building efficient interactive virtual environments based on multiuser PT. Our design for serious educational multiplayer games based in PT environment is discussed in Section IV. In Section V we present some extended features that can provide more efficient and comprehensive learning experience for networking student. An analysis of performance and resource requirements is presented in Section VI, and our conclusions and proposed future works are detailed in Section VII.

II. LEARNING MODULE SPECIFICATIONS

The problem of engaging students effectively in the process of learning is normally solved by using blended learning techniques that involve a hands-on approach, such as the PT simulator. However, current PT activities are highly scripted, having little interactivity and class participation. Multiuser activities would allow PT to be used in a more dynamic way, allowing the instructor to affect student's PT environment in real-time. Having multiuser activities as part of a networking course would fill a gap of short, interactive and extensible activities that can be used to promote student participation in lectures.

The multiuser feature in PT uses a proprietary application-layer protocol, PMTP [12], for communication between PT instances. A description of its operation can be found in [11]. PMTP uses TCP as its transport protocol.

The design of the PT-based activities in this work closely followed the framework described in the following. The activities are currently being integrated into the two introductory networking courses at UOIT that cover the CCNA curriculum.

A. PT Multiuser Environment

There are many operational differences between standard and multiuser CCNA activities, as outlined in [11] and described in the following. These differences create a whole new set of factors that has to be taken into account. The main task is minimizing the disadvantages and limitations of the multiuser feature and maximizing the advantages it offers in interactivity and real-time communication.

PT allows users to use a simple drag and drop cloud icon to connect to peer clouds. Each multiuser cloud supports one-to-one, many-to-one and many-to-many peer-connection configurations. The activities that are created operate on a client-server environment in which most of the complexity and configurations are handled on the instructor side. The central file of the activity scenario is hosted on an instructor PC and the student side file is used by each student to connect to the central PT file hosted on the instructor PC. This instructor/student architecture allows for easy management and control, and provides an overall view for the instructor operating the activity. The multiuser capabilities of PT during testing allowed the connection of up to 60-75 users simultaneously to a single activity over LAN.

UOIT has a laptop-based learning environment in which all students use pre-configured university-issued laptops in the classroom; this greatly facilitated the implementation and use of the instructor-student model. The instructor-side module is responsible for providing the hub part of a hub-and-spoke topology. The students can then start a multi-user PT activity on their respective laptops and establish connections to the instructor side of the activity. Student-specific configurations or modifications must be kept to minimum. Figures 1 and2 show examples of the activity files on the instructor and student sides.

The instructor should also be able to save the student's progress at any point and inspect it on his/her own time. This approach is ideal for completing an activity in the classroom and marking it later. Depending on who the instructor file is released to, the activities are open for modification and can feature any new content that gets added to PT in the future. In general, a laptop, personal computer or other device capable of running PT is required for each student. The requirements for the instructor computer depend on the number of students that are expected to connect and will be discussed later in this paper. The number of connections possible is directly limited by the hardware performance of the instructor PC. We will explore this limitation further in Section VI.

B. Multi-user Activity Specifications

To retain student attention, the multi-user activities and serious games were designed to be completed within 10-20 minutes. The activities would be presented in the middle of the lecture and allow the instructor to demonstrate a particular networking concept interactively with the students. The activities assume that the students have limited hands-on experience and are mainly used to demonstrate the networking concept. In general, the activities follow these requirements:

- Minimal configuration or Pre-configured Student-side files
- Can be used to assess student progression
- Ensure maximum collaboration
- Task variety between activities
- Easy deployment in large user environments
- Scalability to ensure reusability
- Provide an educational experience

Serious gaming activities have to provide a psychological perception of having won or lost. The psychological perception of win or loss is one of the key elements that make up a serious game. The enthusiasm of a student is very crucial in a serious gaming aspect. It is an emotional driver that encourages students to improve his/her networking abilities in order to better compete against fellow classmates. Implementing this perception improves our chances of encouraging students to participate and learn the skills and knowledge needed to complete the activities and course material more efficiently.

A major limitation to utilizing the multiuser architecture is that PT's non-real time simulation mode cannot be

used during a multiuser session. This mode would allow students to analyze the traversal of packets through a network visually step-by-step. By eliminating the use of simulation mode we lose a valuable option but push the focus of the activities to the real-time interaction between the instructor and student networks further encouraging communication and collaboration.

C. Modifying and extending activities

Creating and editing router and switch configurations in text files is considerably easier and more manageable than working within PT. To create or extend the functionality of a certain activity or serious game, it is easiest to first export all configurations of each device into text, and edit the configurations using a text editor. This allows us to either erase the configuration in the old topology later, or load the new configurations. This was found to be the easiest method when altering the game scenarios of the serious games. One must be aware that certain configuration options (e.g. Virtual Trunking Protocol) are not saved in the running configuration and must be configured each time.

The intent of multiuser is not to replace the physical hands-on portions of networking within real lab environments. Instead it is meant to increase student engagement during lectures. As such, in multiuser activities the learning curve to get started and the amount of tedious configuration should be minimized. This can be done by putting commonly accessed devices in the simulation scenario, such as servers and core devices, on the instructor side. To minimize configuration, the student side has devices that are preconfigured, allowing the instructor to limit the focus to a particular topic discussed in class. Minimal configuration from the student is necessary to differentiate the students and allow the networks to communicate. DHCP mechanisms were implemented inside the simulator wherever possible to ensure that the automatic allocation of simulated IP addresses occurred. The IP addresses (i.e. subnets) that were constructed within the serious gaming activities ensured that a wide variety of network topologies could be supported. Therefore, IP address subnets and schemes could be left unchanged and designing new scenarios would only require changing routing protocols or other configuration parameters. This reduces the time it takes to produce new serious gaming activities, which generally take more time to create than the topic based in-class activities.

Simulated student devices can be assigned simulated IP addresses in a unique network depending on how the simulated Virtual LANs (VLAN) are configured. This feature allows students to configure routing protocols between each other. Alternatively, students can be split up into groups on different VLANs to assign them addresses accordingly.

D. Student Evaluation and Monitoring

PT's activity wizard offers an extensible script-based evaluation system that examines the parameters of each device in the student network. For in-class topic-based activities, students were evaluated based on their participation in the specific multiuser activity. The TCP traffic between PT instances is unencrypted and information such as hostnames can be found by capturing that traffic. Additionally, PT allows the ability to save an offline file. The offline file saves can be described as a snapshot of the entire network at an instance of time. The file consists of all of the peer connected multiuser clouds and each device in that peer cloud. The offline file saves also includes all of the devices current state, configuration, and connection status. Typically, during the setup phase of the activity students can be asked to name the devices which they configure with their student ID numbers. This method will easily help us later identify each remote student-side network in an offline save for evaluation purposes. The offline file proves to be a great tool to use when assessing the results of an activity or grading students depending on the configuration status of the devices they were responsible for. This method is used to evaluate a student performance on the two serious gaming activities in Section IV, but can also be used on the topic based activities if needed.

III. Design Challenges

In designing multiuser PT activities, several good practices were developed. The main principle is to maximize the advantages gained by having interactive activities. Configuring multi-user activities can be split into three layers. Layer 1 is the physical and data connection between hosts running the PT instances. Layer 2 is the data connections between switches and relates to simulated MAC addressing and VLAN assignment within the PT environment. Layer 3 is responsible for the rest of the connectivity between PT environments such as simulated IP addressing and routing.

A. Layer 1 Issues

Documentation on how to setup multi-user connections between PT instances is readily available [9]. Connections between instances are done by matching IP address, port, specific cloud name, and password parameters from the instructor file. PT offline saving option accounts for the most overhead activity on computer and network resources. Remote student computers are polled at the same time when offline file saves occur. This results in bursts of traffic that is well handled over the LAN but may result in problems in WAN implementations, as reported in [11]. The serious gaming activities require extra explanation from the instructor on how to set-up the multiuser clouds. Each cloud within a serious gaming activity must be renamed specifically to a student name or ID so that the instructor can distinguish which users have successfully gained access into the instructor side network. PT traffic is unencrypted; so if necessary, useful information can be gathered by collecting PT the traffic data within Wireshark; this however can prove to be a difficult task to commit to for 60-75 students.

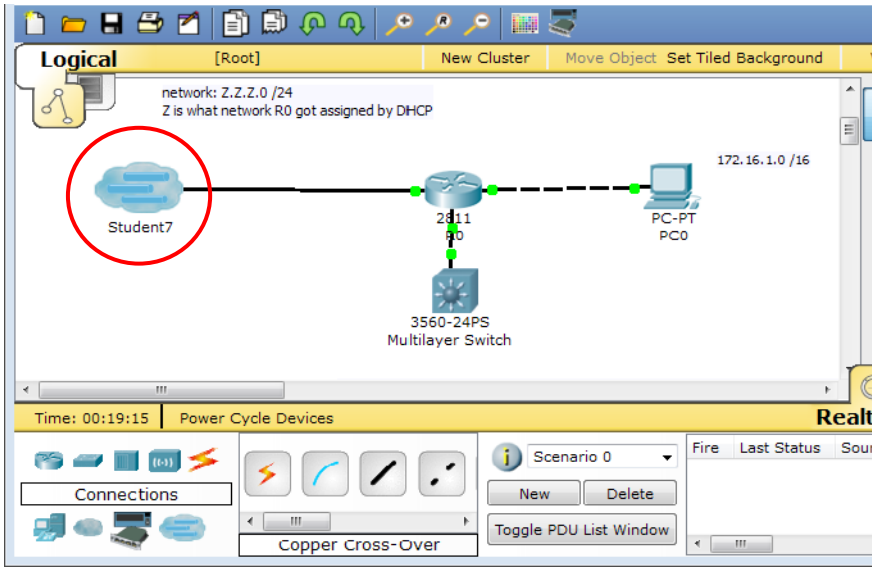


Figure 1: Example of a Student-Side Module

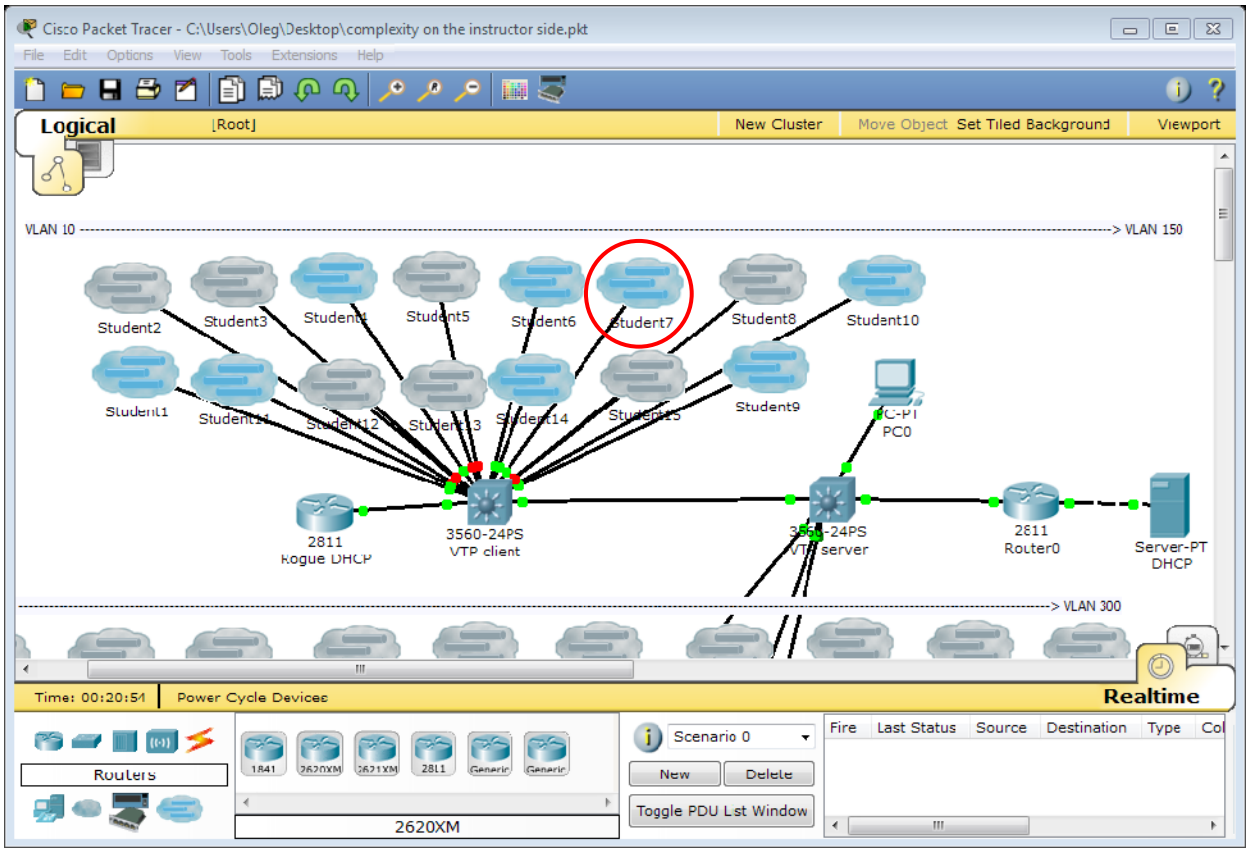


Figure 2: Example of an Instructor-Side Module

B. Layer 2 Issues

As mentioned in the previous section, in most activities students start out with an identical student-side file of the network. This presents a problem because all students now share the same simulated MAC addresses within the PT environment. One way to work around duplicate MAC addresses is by using NAT (Network address translation), but this effectively cuts off inter-student communication. Multiple VLAN's on the other hand maintain inter-student communication when configuring routing protocols. Duplicate MAC addresses can still interfere with various protocols on the instructor-side. STP (Spanning tree protocol) must be properly configured on the switches within PT. The switchport connections can be inadvertently blocked off by STP due to the adjacent port MAC addresses being identical. The switches on the instructor side should be configured with a very low STP priority to guarantee that they become the root and no port gets blocked. There have been cases observed during testing periods when STP has behaved incorrectly and unexpectedly.

When designing activities that teach VLAN concepts, problems can be encountered on the instructor side if past configurations with different VLAN schemes are reintroduced to the instructor network. It is recommended to save the start configuration, erase the vlan.dat file (stores VLAN information) and restart all the routers simultaneously to avoid VLAN/VTP problem when reconfiguring VLANs. Despite the additional complexity on the instructor side, in most activities no other changes have to be made on the student side. Activates designed to explore the routing and WAN topics within the course did not encounter many Layer 2 issues. The instructor side file is almost identical for each routing scenario, every student was separated into their own VLAN and router-on-a-stick is used along with a topic specific routing protocol to allow communication between all student files.

The serious gaming activities focused on testing the ability of the students to troubleshoot layer 3, NAT, DHCP and other protocols. Layer 2 issues were largely avoided because creating layer 3 oriented problematic scenarios for students to solve was simpler and much more manageable.

C. Layer 3 Issues

DHCP is primarily responsible for achieving layer 3 connectivity with minimal configuration from the students. Prior to developing addressing schemes, layer 2 issues should be resolved. While DHCP servers can be set up on routers, for classless networks the DHCP has to be setup on a separate server within the instructor file. Although DHCP is not intended to provide IP addresses to devices with identical MAC address, if layer 2 is problem-free otherwise, the only DHCP problems encountered can be solved by issuing DHCP release/renew commands. Instructor-side files for serious games usually have all of the devices that are needed within them, whereas topic based activities have topic related devices configured on student-side files and devices that allow intercommunication between student side files on the instructor side. With this difference, DHCP services are configured inside each student's cluster within serious games. This ensures easy manageability and provides isolation of student clusters.

D. Scalability, Security and other Limitations

The scalability of the activities is limited. Within a 60 user activity session, the hub/instructor computer will be responsible for any bottleneck encountered. It may be necessary to split the class into a number of student groups that will connect to different hub computers (instructor laptops). The hub computers can also optionally be connected to each other, providing connectivity in a more scalable manner than a single hub computer.

Accountability between authenticated users is very limited and the productive use of PT would require mutual trust between the users. Also, offline saving as well as capturing Wireshark traces provide limited logging capability. While specific task evaluations can be automated using PT's Activity Wizard, it is not possible to find a practical method to account for every detailed student action in the multiuser network unless the PT traffic captured by Wireshark is inspected packet-by-packet.

Denial of service is possible; students can configure a Layer 2 loop between switches in PT which can affect the instructor side, although technological solutions may be written and added onto the PT platform by using the included extensions interface. Due to lack of documentation and added complexity, these resources were not considered in this work when implementing module activities. The risk of such attacks also depends on the weight of the activity completion in the course grade; if these activities constitute a significant portion of the grade, security measures must be implemented in the activity design.

The limited scope of supported Command-Line Interface (CLI) commands in PT is probably the most common limitation that will be encountered in designing multiuser activities. The level of CLI commands is acceptable for a CCNA level at the student side. However to facilitate large interconnected network, it often requires a CCNP level set of commands on the instructor side, which PT often does not provide. The level of complexity on the instructor side is also much higher than the student side, making troubleshooting more difficult.

The lack of supported commands and the increased level of complexity on the instructor side files limit the creativity that can be expressed while constructing serious gaming activities. Serious gaming activities can only be constructed in linear fashion. The team attempted to explore the idea of creating a network traversal game but was restricted by the lack of logging and management capabilities of PT. Time limitations placed inferred that activities could not be too complex. The team also had to ensure that the educational content being delivered to the students was not beyond the scope of the course.

E. Exporting

PT is limited to interact only with devices that are simulated in the PT environment; external syslog servers cannot receive logs. Text, such as router configurations is the most easily exportable information, but not all devices have text interfaces. For example, DHCP servers that address all the VLANs can only be configured through a GUI. As such it can be more practical to build on prior versions of the instructor file rather than create a new one. Unlike GNS/Dynagen [20] and other network simulators, PT cannot generate a network from a text file, nor export a network into any other format. The simplest way to save the work of a student is by using the offline saving feature on the instructor side. PT does however contain a logging services that logs all of the input typed into any device in no specific format that can be exported. This idea was deemed infeasible for our application and consumed too much time for a limited in-class lab scenario.

IV. SERIOUS GAMING ACTIVITIES

Each of the games presented students with artificial conflicts that resembled various real world networking problems that had to be fixed before achieving the game's end goal. Simulated DHCP mechanisms were used to provide each connected player with a set of unique IP address ranges for management and game play structure purposes. Each activity tested the configuration and troubleshooting abilities of the players while providing an educational experience that was novel and entertaining for the students. A detailed explanation of each game follows.

A. Domination Game

The game topology is broken down into four sections. Each section consists of layer 3 switches, multiuser clouds, and clusters. The Section switches all connect to a central Main Domination Switch. Each section consists of 15 cluster clouds and 15 multiuser clouds, all of which are connected to a Section Domination Switch, as shown in Figure 3. Each cluster cloud contains an identical network topology that presents students with a network problem. Each of the clusters have been assigned a /24 virtual subnet, from which the first host IP is assigned to the default gateway. Student side PT instances connect into the multiuser clouds assigned to them by the instructor. Multiuser clouds are distinguished by the Peer Network Name property within each cloud. The game is initiated by the instructor when all players have obtained a connection to the peer multiuser cloud. Once all students clouds are active, students begin by using the telnet protocol in PT to obtain access into their specific cluster.

1) Domination Gameplay

The student's main goal is to fix the network problems presented to them within their clusters. The problems within the cluster can vary in complexity. By solving their cluster problems, the students eventually gain telnet access into the directly connected section switch which enables them to shutdown all other interfaces except the interface directed toward the main domination switch. Once a student has managed to dominate their sections switch, they must quickly telnet into the Main Domination Switch and shutdown the three other ports to block other section switches to telnet into the main switch. The first person to quickly dominate their section's switch and the main switch is the winner. The other three that have managed to dominate only their sections switch are runner ups.

The domination game comes with only one student side file that will work with all of the Multiuser clouds within the instructor file. The student file consists of a single workstation and a multiuser cloud to allow connectivity to the instructor file. By providing a standard student file with basic configurations already applied to the workstation allows for easy deployment in large user environments. Furthermore, virtual IP addresses are dynamically assigned by a DHCP server located in each cluster. Simulating DHCP on each cluster eliminates the need for students to configure an initial IP address to their student side PT Workstation.

The complexity of cluster problems that the students must solve can be increased as needed. Additional devices can be added within a single cluster and be duplicated easily across the clusters because of the easy IP addressing scheme. We simulated EIGRP protocol on the multilayer switches to allow virtual telnet capabilities to the students. IP addressing schemes were designed to be as simple as possible to encourage future development within the activity. The number of VTY lines available within each switch limits the section sizes to 15.

2) Domination Educational Value

The domination game allows students to experience the pressure sometimes put on network engineers in the real world environment. It forces students to apply all of their learnt networking knowledge to troubleshoot a problem. By gaining the ability to combine the hands-on and theoretical knowledge learnt throughout the introductory networking course, students will gain a better understanding of how to apply these skills and tools in the real world. Moreover, the game can also be used as an evaluation tool to see if the students understand the concepts delivered in the course.

B. Relay Race

Adapted from the original Relay Race game presented by Cisco, this Relay Race game incorporates new features. The topology is broken down into four sections, each consisting of five Main Line Routers, four network clusters and five multiuser clouds. The whole topology is brought together at a central Finish Line Router. Each cluster contains problematic network scenarios that students must correct in order to allow a Runner, designated in each team, to move closer to the Finish Line Router. The network scenarios within the clusters progressively become harder the closer the cluster is to the Finish Line Router. The five Main Line Routers act as doorways, locked, preventing the runner from moving forward. Figure 4 shows an overview of Relay Race topology.

1) Relay Race Gameplay

Although slightly more complicated, the concept of the game remains somewhat similar to the Domination game. Relay Race consists of 4 Teams each consisting of 5 team members. Each team consists of 1 Runner and 4 members responsible for solving the problematic network clusters. Once the problem within the cluster is solved they must move forward to their Main Line Router and no shutdown their routers Serial 0/0/0 interface. This will allow the runner of the team to telnet into their routers in order to move forward towards the Finish Line Router. Several access lists have been put in place so that only the appropriate hosts can telnet into the appropriate devices. For example, none of the team members responsible for solving cluster problems can telnet into the Finish Line Router; only the runner will be able to telnet into that router. The access lists also ensure that telnets to other Main Line Routers will only work from the Runner's Workstation on the student PT file. The Goal of the game is to have the fastest team to enable all of their S0/0/0 interfaces within the Main Line Routers. The Runner must then run (telnet) into the Finish Line Router before any of the other teams and shutdown all of the other interfaces to block the opposing teams' access to the Finish Line Router.

A common student file is used in this game and DHCP has been simulated in PT in a manner allowing the same file to be used among all multiuser peer-connection within the game. The instructor file contains the Relay Race game that students must telnet into. The students connect to the multiuser clouds according to the name of the cloud in correspondence to the role of the student within each team. By providing this instructor/student architecture we ensure easy deployment of the game during play time. The clusters problem can be easily changed if need be.

2) Relay Race Educational Value

Relay Race pits students together in a team environment where they must communicate and apply their skills to a problem. The activity will help hone communication skills and also improve the ability of the students to solve network problems. The game will force students to work rapidly and correctly to solve their problems the fastest in order to win. This environment emulates the fast paced environments of the networking world today, where problems arise quickly and must be solved rapidly. The game can also be used as an evaluation tool to see how far student's configuration and troubleshooting skills have progressed.

V. EXPLORING UNIQUE NETWORK ACTIVITIES

This section discusses some ideas for further development of PT-based activities.

A. Network Traversal Activities

During this project a small scale network traversal activity was created for one of the in-class modules. The unique PT activity required students to attempt to discover the network by using various telnet and show commands. The goal of the activity was to learn how to fix the router problems they encounter within a topology before moving forward. Most of the devices in the activity are hidden and cannot be accessed by double clicking on them nor can be moved. The activity forces the students to use various CLI commands that are necessary to know at the CCNA level. Connectivity will be achieved when the lab is successfully completed.

A large scale version of the activity can be created to encompass multiple students. The activity can be presented as a race to the last node within the topology. Each node within the topology would include the privileged password to a router that would allow them to move to a different segment within the topology. The password can easily be stored within the description of a specific link or the router itself. Each segment within the traversal topology would present the racers with a problem that they must solve and traverse to each node within the segment using telnet/show commands. The concept will require an instructor file that contains multiple identical traversal networks. Each identical traversal topology would be assigned to a student and accessed via a multiuser cloud. The start of the race can be controlled by releasing the passwords to the first node to every student simultaneously at the beginning of the race. The main issue with this concept is that the instructors PCs will have to be computationally powerful machines to handle multiple, possibly 60, identical networks containing a minimum of 10 devices each.

B. PT Plus Web App Activity

A PT plus Web App activity is a conceptual activity that will be utilized for mid-term or final lab examination evaluation purposes. The concept also relies on a Web based application that has accounts for each student and can process multiple choice questions on it. Most universities utilize online student accounts managing system that can be utilized for this purpose. The concept is similar to network traversal in that at the end of each segment within a network traversal topology a password is given to the student that can be used to unlock a multiple choice question on the Web application as well as allow them to progress through the traversal topology. This way a student can be evaluated for their knowledge in practical configuration expertise as well as theoretical knowledge questions via the multiple choice answers.

This concept can also be used to deliver interactive labs that present students with a situation. The lab can progress online on the Web App as the students complete tasks within the lab. This type of activity is not a collaborative activity however it does deliver content in a different manner that can enable the possible creation of collaborative labs as future experimentation is done with the concept.

VI. RESOURCE REQUIREMENTS AND SCALABILITY

In order to test the scalability of the learning environment, the activities and games were deployed in an instructor set up with multiple multiuser PT clouds connecting to the instructor server and executing the activities from their side. Packet Tracer 5.3 was chosen as the platform for these activities.

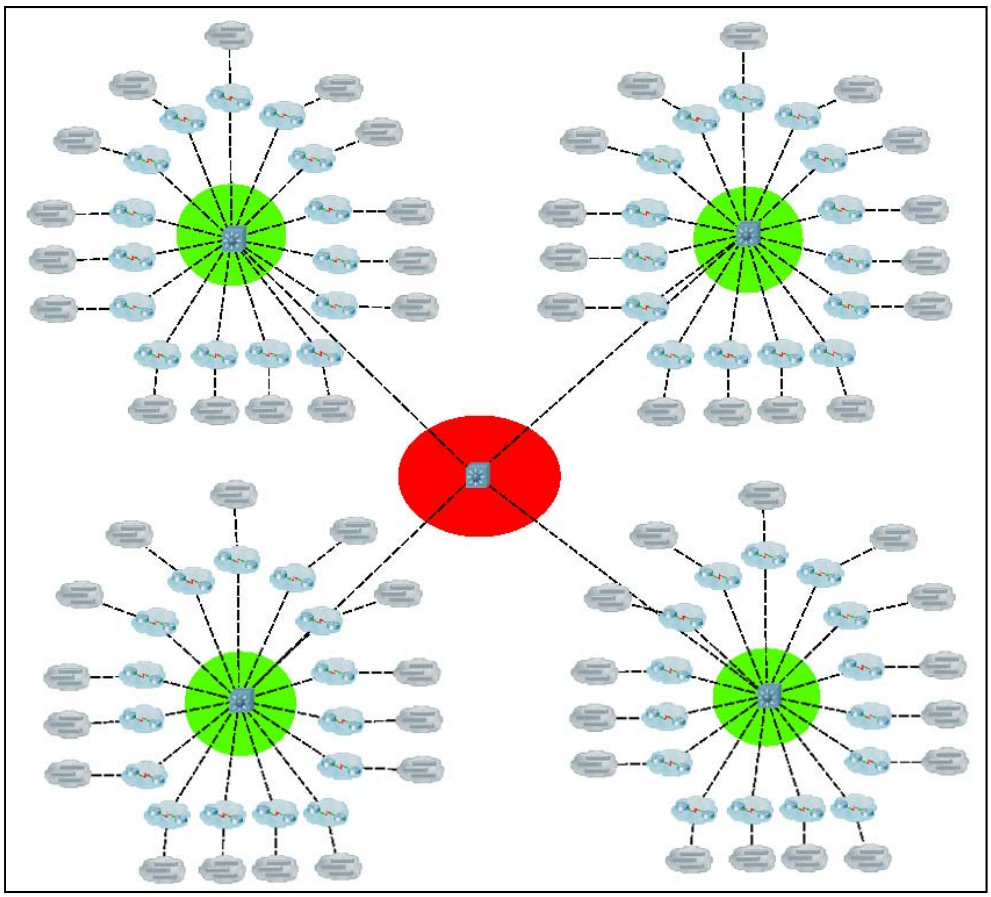


Figure 3: Domination Game Screen Shot

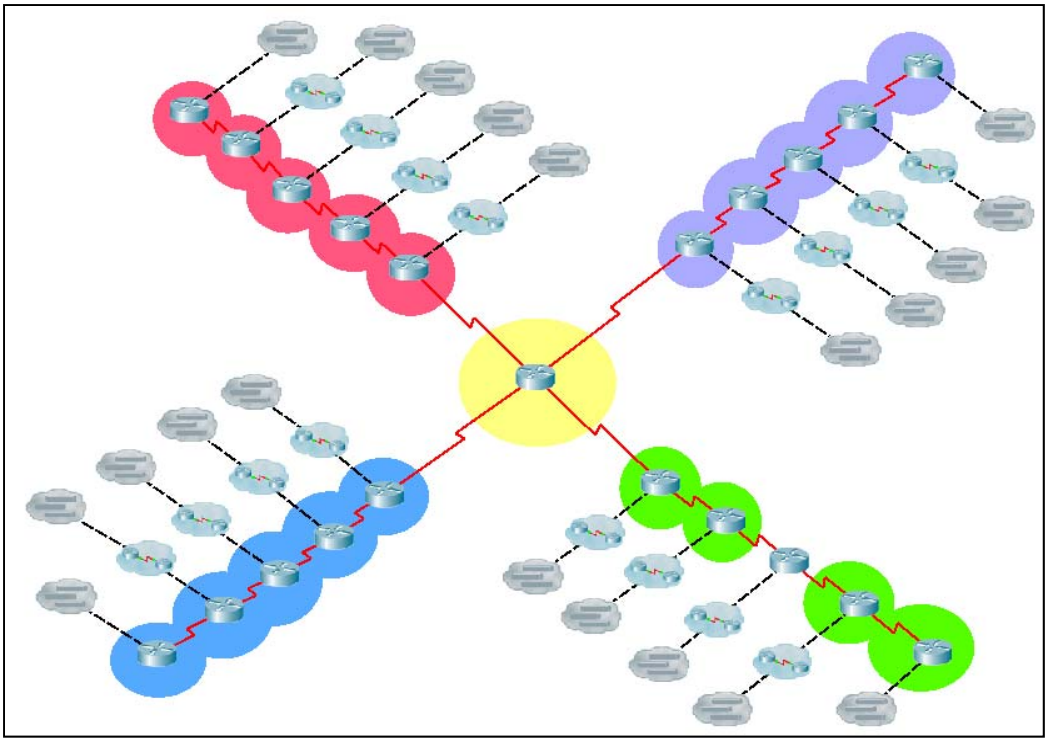


Figure 4: Relay Race Game Screen Shot

A. CPU load and offline save time

In this test case a single prototype activity file was used for all the student connections, and a single instructor file was used that would accept all the connections. Statistics about resource consumption were recorded by observing CPU, memory, and network usage as more and more students were connected to the instructor's simulated network. The hub was based on a school Lenovo T61p laptop (Core 2 Duo T8300 2.4GHz, 2GB RAM). The method of gathering the system information has its limitations in precision but portrays an accurate picture of the increase in system resources as the number of student rises.

Figure5 rage offline save shows the CPU load and ave stime. During this test, frequently the instructor file became unresponsive and the testing had to be restarted. The growth for memory resource was very scalable to the amount of students. However, CPU usage and loading times grew at a greater rate. As such, multiuser activities remain most practical in classes that have less than 60 students per instructor server. Likewise, if classes are small, the requirements for reliability can be lower, as technological problems can be easily managed by the instructor.

B. Scalability

Two sets of tests were conducted on each of the two activities instructor files. The first was done using the Domination instructor file and its companion student file. The second was done using the Relay Race instructor file and its companion student file. During the testing procedure for the Domination game a total of 60 multiuser clouds were connected in increments of 5 users to the instructor file. Each time five users connected to the game, new data was collected. The Relay Race game has a maximum capacity of 20 users playing at the same time therefore, 20 multiuser clouds were connected to the Relay Race instructor file in increments of five users.

The results for this test were very satisfying for both of the activities. Figure Figure 6 shows the CPU utilization for Domination and Relay Race. It is evident that the as more hosts connect to the instructor files, the CPU utilization increases in a logarithmic form. This indicates that the instructor side was capable of handling higher levels of stress if need be.

Test results for both activities indicated linear growth rates in memory utilization. Figure 7 shows that growth rate for both the Domination activity and the Relay Race activity is linear as the number of users increases. Although memory is not an issue with 50 users connected to an activity, if we were to connect up to 100 or more users' memory could potentially become an issue. For our university use cases memory does not pose a threat to the functionality of our activities.

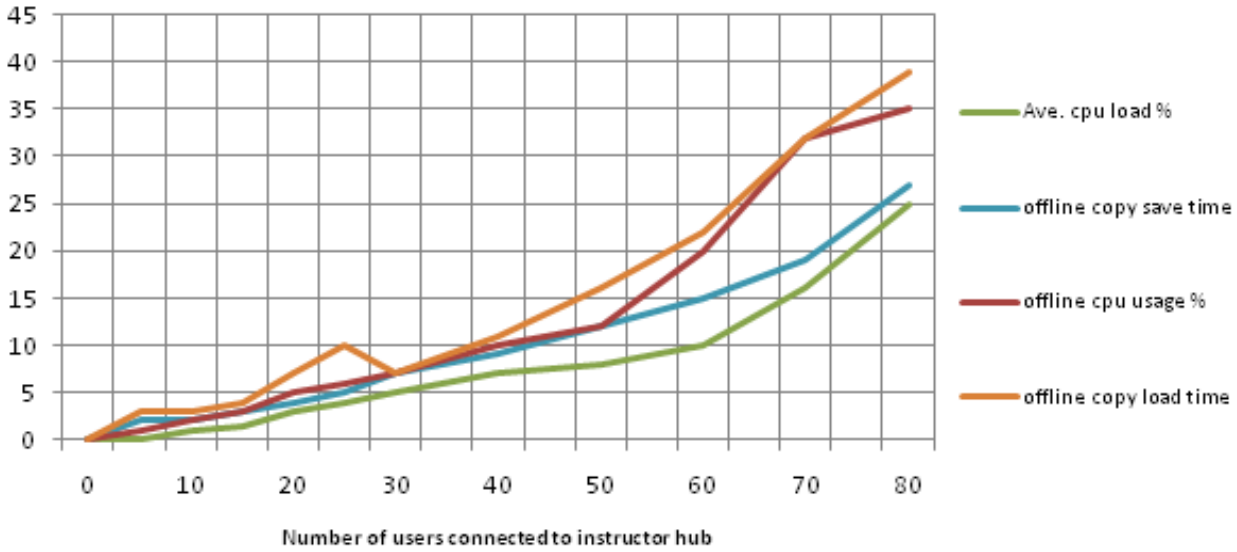


Figure 5: Packet Tracer scalability with 1 hub

We also measured the required network resource and bandwidth for these scenarios, and the test results indicated that the multiuser environment places minimal burden on network resources. It indicated that we could have a large number of users connected to a single instructor file without worrying about the network capacity and bandwidth. The network utilization test increased in a linear fashion and these values are bound to increase as the number of users increase. An important issue to note is that these values are bound to change depending on how actively the students interact with the instructor side network.

The offline file-save for recording students' works could potentially become a bottleneck issue. The offline file consists of all of the peer connected multiuser clouds and each device in that peer cloud. It also saves all of the devices current state, configuration, and connection status. Offline saving times of greater than five minutes for a 30 user Multiuser environment could render useless a 20-minute activity and a class room limited to an 80 minute class. Therefore it is important to test how long it actually takes to create the offline file. The results indicated that for the Domination activity the time to create the offline file grew exponentially. The Relay Race activity indicated a linear growth rate. The exponential growth rate for the Domination file could prove to be troublesome if more than 60 users are connected to the Domination activity. Currently it takes about 25 seconds for a 50 user multiuser environment offline file to save. Although it serves our purpose to provide 60 students a reliable platform to use the activity, connecting more than 100 users to a single instructor file could take much longer and may require multiple interconnected instructor servers. These values are bound to change if the complexities of the problems in each cluster are increased.

VII. Conclusion and Future Work

Networking courses that cover introductory, CCNA-level course material usually have to cover a large amount of theory for students that may have little prior networking experience. In-class lectures can be improved by using interactive activities that foster student collaboration. Multiuser achieves greater class interaction by allowing students to form what is essentially one large network supervised by the instructor. Using the multiuser architecture we can also develop serious gaming activities that can provide a simple and entertaining solution to present complex networking scenarios to networking students enrolled in our courses. We can train students to not only improve their configuration and troubleshooting skills but also improve communications skills in an IT environment. We can also increase the speed at which they tackle these problems and provide them with the expertise and knowledge to excel in the fast paced networking environments of today.

In this project, we developed a set of modules containing learning activities and educational games based on Cisco Packet Tracer simulation environment for use in introductory networking classes. Along the way we faced and solved several challenges for efficient design of such activities, which we presented here as a guideline for creating efficient multiuser interactive modules in a virtual environment. Our results indicated that the proposed design can be used in medium size classes without the need for significant investment in servers and network resources.

Future work on the multiuser feature in packet tracer could improve the quality and user experience offered by multiuser activities. Other academies may seek to modify the instructions in order to change the difficulty of some activities. Student feedback is critical to effectively improving activities and bringing them up to the standard of activities offered by Cisco. In the meantime it is up to the instructor judgment to adjust the difficulty or instructions on the spot. Various multiuser architectures may be attempted. In contrast to client-server topology in use, ad-hoc PT connections are low resource and can make use of the simulation mode of PT. The problem with ad-hoc connection is that the instructor is not present to assist students and it is difficult to account for the activities completed. This may offer a different learning experience to the student. However user input would be needed for a successful implementation. The activity wizard provided by PT can be used to create more locked down activities for the student and may address certain security limitations. The challenge with using the activity wizard lies with the increased development time and the requirement to account and restrict/facilitate all the student actions that may occur in the activity.

In the meantime implementing some of the new ideas presented in section V may prove to be a challenge worthwhile. Compared to regular PT activities, by being connected to the same network, multiuser allows students to collaborate and work towards a common goal. This allows the creation of activities that have before been impractical to implement, such as group troubleshooting, capture the flag, and relay race games. This research has shown the great potential of the topic and the educational values PT holds.

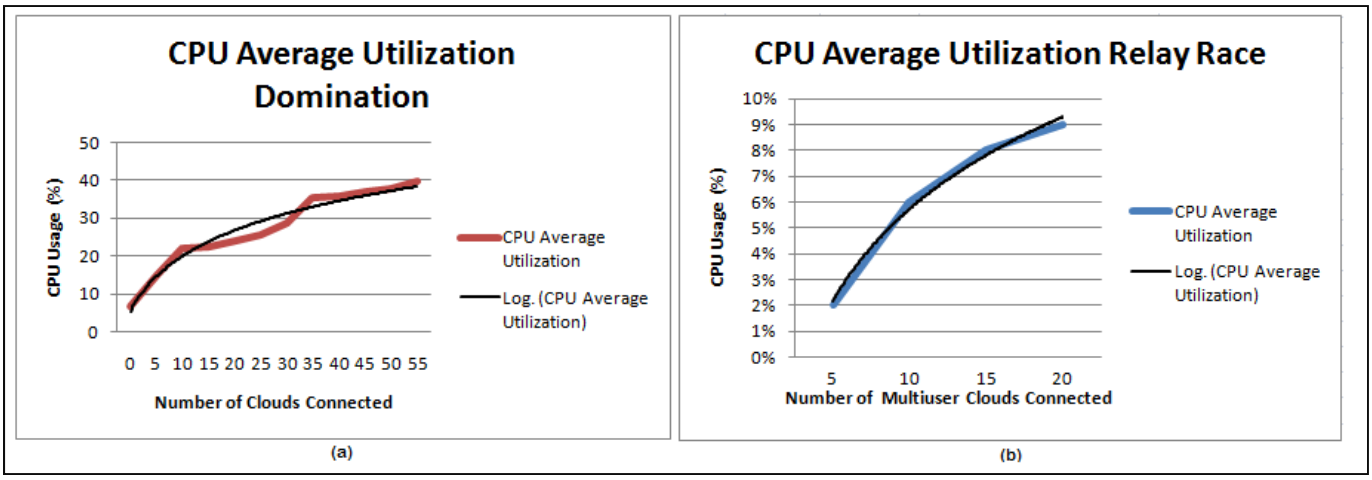


Figure 6: CPU Utilization: (a) Domination CPU Utilization (b) Relay Race CPU Utilization

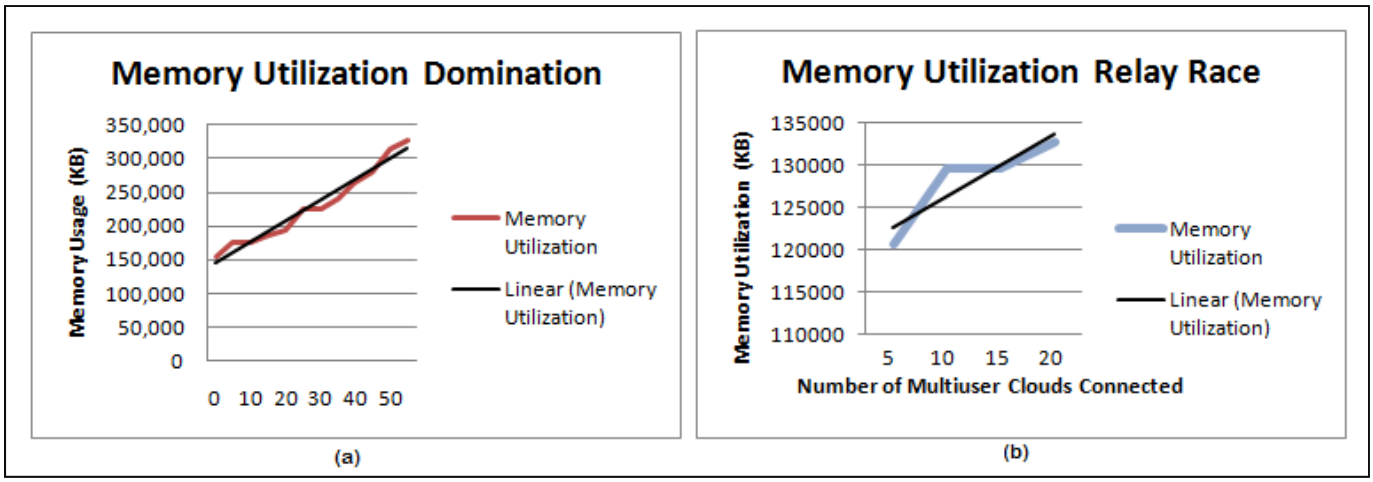


Figure 7: Memory Utilization (a) Domination Memory Utilization (b) Relay Race Memory Utilization

ACKNOWLEDGMENT

This research was supported through a Teaching Innovation Funding (TIF) grant from the office of the Associate Provost, Academic, of the University Of Ontario Institute Of Technology. Packet Tracer is a product of Cisco Networks and is provided free of charge to Cisco Networking Academy students and instructors.

REFERENCES

[1] R.G. Muir-Herzig, "Technology and its impact in the Classroom," *Computers & Education*, vol. 42, no. 2, pp. 111-131, February 2004.

[2] P. Dillenbourg, D. Schneider, and P. Synteta, "Virtual Learning Environments," in *Proceedings of the 3rd Hellenic Conference on Information & Communication Technologies in Education*, 2002, pp. 3-18.

[3] A. All. (Accessed 2011, March 20) "Serious Games Entertain, Educate Employees", IT Business Edge, August 5, 2009. [Online]. http://www.itbusinessedge.com/cm/community/features/articles/blog/serious-games-entertain-educate-employees/?cs=34730

[4] C. Kohler. (Accessed 2011, March 20) "Xbox Kinect Games Give You a Serious Workout", Wired Magazine, June 15, 2010. [Online]. http://www.wired.com/gamelife/2010/06/kinect-hands-on/

[5] J. Bohannon. (Accessed 2011, March 20) "Unravel the Secret Life of Protein", Wired Magazine, issue 17.05, 20 April 2009. [Online]. http://www.wired.com/medtech/genetics/magazine/17-05/ff_protein?currentPage=all

[6] M. Macedonia. (2001) "Games, simulation, and the military education dilemma", Internet and the University. [Online]. http://www.educause.edu/ir/library/pdf/ffpiu018.pdf

[7] D.D. Burdescu, M.C. Mihaescu, C.M. Ionascu, and B. Logofatu, "Support system for e-Learning environment based on learning activities and processes," in *Fourth International Conference on Research Challenges in Information Science* , 2010, pp. 37-42.

[8] M. Chang and Kinshuk, "Web-Based Multiplayer Online Role Playing Game (MORPG) for Assessing Students' Java Programming Knowledge and Skills," in *Third IEEE International Conference on Digital Game and Intelligent Toy Enhanced Learning*, 2010, pp. 103-107.

[9] "Packet Tracer Reference Guide and Tutorials," Cisco Networking Academy, 2010.

[10] F. Jakab, M. Bucko, I. Sivy, L. Madarasz, and P. Cicak, "The system of career promotion of networking professionals based on industrial certificates," in *International Conference on Intelligent Engineering Systems (INES 2009)*, 2009, pp. 221 - 226.

[11] A. Smith and C. Bluck, "Multiuser Collaborative Practical Learning Using Packet Tracer," in *Proceedings of the Sixth International Conference on Networking and Services (ICNS)*, 2010, pp. 356-362.

[12] "Packet Tracer Messaging Protocol (PTMP) Specification Document," Cisco Networks, 2008.

[13] Cisco Aspire CCNA Edition. (Accessed 2012, February 14) [Online]. https://learningnetworkstore.cisco.com/market/prod/aspireFAQ.se.work

[14] M. Torrieri. (Accessed 2011, March 20) "Cisco's MyPlanNet Simulation Game Touches on Broadband Growth and Other Hot Communications Topics", TMCnet, Nov. 4, 2009. [Online]. http://4g-wirelessevolution.tmcnet.com/broadband-stimulus/topics/broadband-stimulus/articles/68180-ciscos-myplannet-simulation-game-touches-broadband-growth-other.htm

[15] M. Virvou, G. Katsionis, and K. Manos, "Combining Software Games with Education: Evaluation of its Educational Effectiveness," *Educational Technology & Society*, vol. 8, no. 2, pp. 54-65, 2005.

[16] K.D. Squire, "Video games in education," *International Journal of Intelligent Games & Simulation*, vol. 2, no. 1, pp. 49-62, 2003.

[17] J.T. Behrens, D.C. Frezzo, R. Mislevy, J. Kroopnik, and D. Wise, "Structural, Functional and Semiotic Symmetries in simulation based games, and assessments," in *Assessment of Problem Solving Using Simulations*., 2007, pp. 59-80.

[18] A. Musheer, O. Sotnikov, and S. Shah-Heydari, "Packet Tracer as an Educational Serious Gaming Platform," in *Proceedings of the 7th International Conference on Networking and Services*, 2011, pp. 299-305.

[19] O. Sotnikov, A. Musheer, and S. Shah-Heydari, "Building Interactive multi-user in-class learning modules for computer networking," in *Proceedings of the 7th International Conference on Networking and Services*, 2011, pp. 326-331.

[20] GN3/Dynagen (Accessed 2012, June 24) [Online] http://www.gns3.net/dynagen/

An MDA-based Approach to Crisis and Emergency Management Modeling

Antonio De Nicola, Alberto Tofani, Giordano Vicoli, Maria Luisa Villani
Computing and Technological Infrastructure Lab.
ENEA: Italian National Agency for New Technologies, Energy and Sustainable Economic Development
Rome, Italy
{antonio.denicola, alberto.tofani, giordano.vicoli, marialuisa.villani}@enea.it

Abstract— **Managing crisis and emergency requires a deep knowledge of the related scenario. Simulation and analysis tools are considered as a promising mean to reach such understanding. Precondition to these types of tools is the availability of a graphical modeling language allowing domain experts to build formally grounded models. To reach this goal, in this paper, we propose the CEML language and the related meta-model to describe structural aspects of crisis and emergency scenarios. The meta-model consists of a set of modeling constructs, a set of domain relationships, and a set of modeling rules. Then we introduce a set of methodological guidelines to reach an executable code, consisting of a system architecture and the mapping rules to transform the CEML modeling constructs into others typical of discrete event simulation. Finally, we propose a preliminary set of collaboration design patterns to model interaction and communication exchange arising among emergency services providers and citizens to solve the crisis. An emergency scenario example demonstrates the applicability of the presented approach.**

Keywords - Conceptual Modeling; Collaborative Networks; Critical Infrastructures; Model Driven Architecture.

I. INTRODUCTION

Recent natural disasters (e.g., earthquakes, floods, fires) and technical faults (e.g., power outages) and their impact on critical infrastructures (CI) and population have caused a growing attention on how to manage crisis and emergency. In this context, CI services (e.g., telecommunications network, water pipelines) may not work or could not guarantee an acceptable level of service. Since dependencies among CI services are often unpredictable, they could generate further unexpected faults in the CI network. Communications channels could be unavailable to teams needing to collaborate to solve the crisis. Furthermore, beneficiaries of CI, not provided with the needed resources, can act in uncontrolled mode, hindering the work of operators who are trying to restore CI services.

To cope with such complexity and mitigate such effects, a promising approach is to simulate these scenarios. Simulation allows creating a portfolio of virtual crisis and emergency management experiences to be used, for instance, for training institutional operators with the responsibility of solving the crisis.

A precondition to build effective simulation tools is the availability of a modeling language and a modeling methodology allowing domain experts to build formally grounded models that can be converted into simulation models. The MDA (Model-Driven-Architecture) [1] approach can help us to this aim as it provides methods and tools that can be used by domain experts, i.e., institutional operators with a deep knowledge of crisis and emergency scenarios but with limited high-level IT skills. The first required feature of such language is the *domain adequacy*, i.e., how the language is suitable to represent the addressed domain [2]. This is achieved by providing experts with modeling constructs and relationships better reflecting their knowledge about the domain. In the CI domain, it is required to allow modeling of collaboration and interaction among CI services, population, institutional operators and stakeholders operating in crisis and emergency scenarios. Then the language has to permit modeling of both structural and behavioral aspects. It has to be formally grounded to allow models to be processed as source code of appropriate simulation programs. It has to be based on widely accepted existing standards to support model interoperability between different simulation tools. Finally, it has to be supported by a graphical notation to allow intuitive and user-friendly modeling.

In this paper we propose CEML (Crisis and Emergency Modeling Language), an abstract level language to model crisis and emergency management scenarios. In particular, we describe the related CEML meta-model, consisting of a set of modeling constructs, a set of relationships, a set of modeling rules, and its formalization using SysML [3] and OCL [4]. Here, we focus mainly on presenting how CEML supports structural modeling of a crisis and emergency scenario. Modeling of behavioral aspects will be treated in another paper.

CEML's objective is to support domain experts in building a model of a CI scenario. However, a CEML model has not been conceived to be directly simulated, since it needs to be transformed into a format closer to the computer programs. For this reason, with respect to [5], here we propose some methodological guidelines to reach an executable code. They consist of a system architecture and a set of mapping rules to transform the CEML modeling constructs to the constructs typically used in the discrete event simulation tools.

Then we propose a modeling methodology tailored to model collaboration needed in crisis and emergency scenarios. This methodology is based on Collaboration

Design Patterns (CDP)s. A design pattern is a reusable solution to a recurrent modeling problem [6]. In particular, collaboration design patterns model interaction and communication exchange arising during the crisis. As example, here we propose five CDPs: clustered service, basic communication, heterogeneous networking, single service provider, and infrastructure.

The rest of the paper is organized as follows. Section 2 presents related work in the area. Section 3 describes the meta-model for crisis and emergency scenarios and its formalization. Section 4 presents some guidelines to implement the CEML language into a simulation platform. Section 5 proposes a preliminary set of collaboration design patterns for crisis scenarios. Section 6 describes an example concerning emergency management after earthquake events and shows an application of the proposed modeling framework. Section 7 contains a discussion on the evaluation of our approach and, finally, Section 8 presents conclusions and future work.

II. Related Work

Nowadays there is an increasing interest on crisis and emergency management modeling and simulation. The aim is to propose effective modeling and simulation approaches to analyze crisis scenarios, and to test crisis and/or disaster management procedures.

The main concepts and definitions related to critical infrastructures (CI) are presented in [7]. An interesting approach to describe various aspects of CI is the ontological approach. In [8], for instance, five meta-models are proposed to characterize various aspects of an infrastructure network, such as managerial, structural and organizational aspects. These meta-models are defined as a UML profile with the aim to completely describe the critical infrastructures domain and their interdependencies. Instead, here we concentrate on the problem of graphically building structural models of crisis management scenarios, also involving humans, for simulation purposes.

Ontologies to describe either emergency plans or disasters affecting critical infrastructures are presented in [9], [10], [11], and [12].

All these works, which we have considered as a starting point for our research, are complementary to our result, as they provide means to semantically enrich simulation models realized with our language.

Several papers propose an MDA approach to simulation. Among them we cite [13], [14], and [15]. In particular, we share with them a layered approach proposing a different type of model representation for the PIM (i.e., Platform Independent Model) level and the PSM (i.e., Platform Specific Model) level [16]. The main difference with these approaches concerns the scope. Whereas they are general purpose, we focus on the crisis and emergency management domain. Consequently, CEML has been conceived to model such domain whereas it is not the best solution for a different one.

In [17] and [18] SysML is proposed as "standard" meta-model for high level discrete event simulation models to be mapped to Arena and DEVS programs. Indeed, this is proposed to ease the access to simulation technology to non ICT experts and to allow exchange of simulation models between tools.

Instead, in [19] UML is proposed as modeling language for agent-based simulators of interdependent critical infrastructures. To this aim, the authors define a methodology for the development of the simulator that suggests the UML diagrams to be used and how these may map to an agent-based model. As the UML meta-model is used as it is and the methodology is given in the form of design suggestions, this work is addressed to software engineers and does not aim at the formalization required by MDA.

With all of these works we share the choice of the UML meta-model (and/or of its profiles) as a root for a modeling language in this domain. Indeed, generally, UML is the most used language for the specification and development of software applications, and, specifically for our work, many tools are available, especially in the MDA world, to implement and validate our approach.

In addition to what is presented by others in the same field, we propose a set of CDPs to support crisis management experts in modeling crisis scenarios. At the best of our knowledge there is no similar proposal in the crisis and emergency management sector.

III. A Meta-Model for Crisis and emergency scenarios

In this section we present the CEML meta-model aimed at guiding a modeler in representing the structural aspects of a crisis and emergency scenario. A meta-model is a design framework describing the basic model elements, the relationships between them, and their semantics. Furthermore it defines the rules for their use [20]. As stated in the introduction, CEML is defined at a high level of detail since it has been conceived mainly for domain experts. In fact, according to the MDA approach, CEML is located at the PIM level.

For this reason, the modeling constructs and relationships have been defined starting from an analysis of crisis and emergency scenarios and from interviews with domain experts. In particular, we have given importance to two requirements. One is simplicity: domain modelers prefer a limited number of constructs and more focused rather than many abstract constructs most of which not needed for their purposes. The other requirement is that models should be service-based: services are the abstractions used by the domain experts for the entities of their scenarios and are at the right level of granularity compared, for example, to individual functions.

A. CEML Modeling Constructs

The CEML modeling constructs define the "terms" used when describing crisis and emergency scenarios. They can be classified as active and passive constructs. The active constructs allow modeling entities able to perform activities (e.g., processing a resource, issuing a message) and their behavior. They are: the abstract service, specialized as service, human service, and communication service; the

behavior; the external event; and the user. The passive constructs allow modeling entities managed or processed by active entities. They are: the message, the resource, and the connectivity. A natural language description of the CEML modeling constructs now follows.

Abstract Service. It represents the active entity processing either a *resource* entity or a *message* entity or a *connectivity* entity. It can be either a *service* or a *human service* or a *communication service*.

Service. It represents the active entity either producing (e.g., power house) or providing (e.g., information service) or transporting (e.g., electrical power grid) a given *resource* entity.

Human service. It represents the active entity providing a given *resource* in the form of human activities (e.g., fire brigades).

Communication service. It represents, from a physical perspective, the active entity allowing communication and information exchange between two of the following entities: *service*, *human service*, and *user* (e.g., between two *services*, between a *service* and a *user*).

Behavior. It represents an operational feature of either a *service* or a *human service* or a *communication service* or a *user* entity. This allows completing the structural model with behavioral specifications.

External event. It represents the active entity (e.g., failure, earthquake) affecting the operational status of either a *service* entity or a *human service* entity or a *communication service* entity or affecting the wellness of a *user* entity.

User. It represents the entity using or consuming a *resource* entity (e.g., hospital). It is characterized by a wellness level.

Message. It represents information content exchanged in a communication.

Resource. It represents the passive entity processed (i.e., produced, provided, transported) by either a *service* entity or a *human service* entity. It can be either material (e.g., water) or immaterial (e.g., fire brigades activity). It can be input to either another *service* entity or a *communication service* entity or a *human service* entity or a *user* entity. It can contribute significantly to *user*'s wellness level.

Connectivity. It represents, from a physical perspective, the output of a *communication service* entity.

B. CEML Relationships

The CEML relationships allow modeling flowing of passive entities, through the flow and the port relationships, and how an external event affects another entity through the impact relationship. Flow relationships are the resource flow, the connectivity flow and the message flow. Port relationships are: the abstract port, specialized as communication port, message port, and resource port; and the connection port group. A natural language description of the CEML relationships now follows.

Resource Flow. It represents resource passing through ports from a *service* or *human service* entity to either a *user* or a *service* or a *human service* or a *communication service* entity.

Connectivity Flow. It represents, from a physical perspective, the communication channel provision (through ports) from a *communication service* entity to either a *service* or a *human service* or a *user* or another *communication service* entity.

Message Flow. It represents, from a logical perspective, the exchange of information content through ports between two of the following entities: *service*, *human service*, and *user* (e.g., between two *services*, between a *service* and a *user*).

Abstract Port. It represents the abstract entity linking either an *abstract service* entity or an *user* entity to either one or more *connectivity flow* entities, or one or more *message flow* entities, or one or more *resource flow* entities. It can be either a *message port* or a *communication port* or a *resource port*.

Communication Port. It represents the abstract entity linking either a *communication service* or *a human service* or *a service* or *a user* entity to one or more *connectivity flow* entities.

Message Port. It represents the abstract entity linking either a *service* or a *human service* or a *user* entity to one or more *message flow* entities.

Resource Port. It represents the abstract entity linking either a *service* or a *human service* or a *communication service* or a *user* entity to one or more *resource flow* entities.

Connection Port Group. It represents the abstract entity grouping one *communication port* entity and one or more *message port* entities and belonging to either a *service* or a *human service* or a *user* entity.

Impact. It represents how an *external event* entity affects one or more of the following entities: *service*, *communication service*, *human service*, and *user*.

C. CEML Modeling Rules

The CEML modeling rules are the syntactic rules to generate well-formed CEML models. The 13 modeling rules now follow.

C1. An element can be categorized only as a modeling construct or as a relationship.

C2. A **service** element has *0..n* incoming **resource port** elements, *1..n* outgoing **resource port** elements, and *0..n* **connection port group** elements.

C3. A **human service** element has *0..n* incoming **resource port** elements, *1..n* outgoing **resource port** elements, and *0..n* **connection port group** elements.

C4. A **communication service** element has *0..n* incoming **resource port** elements and *1..n* outgoing **communication port** elements.

C5. The **service** element, the **human service** element, and the **communication service** element are specializations of the **abstract service** element.

C6. The **message port** element, the **communication port** element, and the **resource port** element are specializations of the **abstract port** element.

C7. Every **abstract service** element is characterized by *0..n* **behavior** elements.

C8. Every **abstract service** element is affected by *0..n* **external event** elements by means of the **impact** element.

C9. A **user** element has *0..n* incoming **resource port** elements and *0..n* **connection port** group elements**.**

C10. A **user** element is affected by *0..n* **external event** elements by means of the **impact** element.

C11. A **message flow** element is linked to *1..n* **message** elements and holds between two **message port** elements belonging to two **connection port group** elements.

C12. A **resource flow** element is linked to *1..n* **resource** elements and holds between 2 **resource port** elements. The **resource flow** element is directed from a **resource port** element belonging either to a **service** or **human service** element and to a **resource port** belonging either to an **abstract service** element or to an **user** element.

C13. A **connectivity** element is directed from a **communication port** element, belonging to a **communication service** element, to a **message port** element, belonging to a **connection port group**.

D. CEML Meta-model formalization

In order to equip the language with a sort of formal grounding, so that smart editors could be defined with validation facilities, we have identified SysML [3], a standard language sponsored by OMG (Object Management Group), as a good candidate. SysML comes as a *profile* of UML 2.0, that is, extends the UML meta-model with constructs to enable "system" other than "software" modeling and provides some new diagram types. Therefore, SysML inherits all the advantages of UML: the multi-views representation of a system model; the simplicity of the notation, which is addressed to stakeholders with different levels of technical knowledge; the XML schema for tools interoperability (XMI); and, finally, the "semi-formal" specification, which has been better clarified starting from version 2.0, that allows model-driven development to take place. Our meta-model is an application of SysML profile tailored to critical infrastructures modeling and, as such, it is a domain-specialization of a subset of SysML. We do this by creating a new profile following the stereotype extension mechanism specified by UML.

Specifically, we consider the components of the *Internal Block Diagram* of SysML, which is based on the *Block* entity. According to the OMG specification, blocks "are modular units of a system description, which define a collection of features to describe a system or other elements of interest. These may include both structural and behavioral features, such as properties and operations, to represent the state of the system and behavior that the system may exhibit".

Figure 1 shows the relationship of the *User* and *AbstractService* constructs of our meta-model with the *Block* entity of SysML. They can have a behavior specified and can be connected with other blocks through ports. However, differently from services, a *User* does not provide functions/resources to other model elements. Note that the *User* construct in our meta-model cannot be mapped to the UML (or SysML) Actor meta-class as we intend the *User* be inside the model (and not part of the environment).

Flow ports are introduced in SysML as a specialization of UML ports "to specify the input and output items that may flow between a block and its environment". *Flow ports* are generally typed with respect to the item that can flow (in, out, or inout). In our meta-model we have decided to introduce three port types as shown in Figure 2.

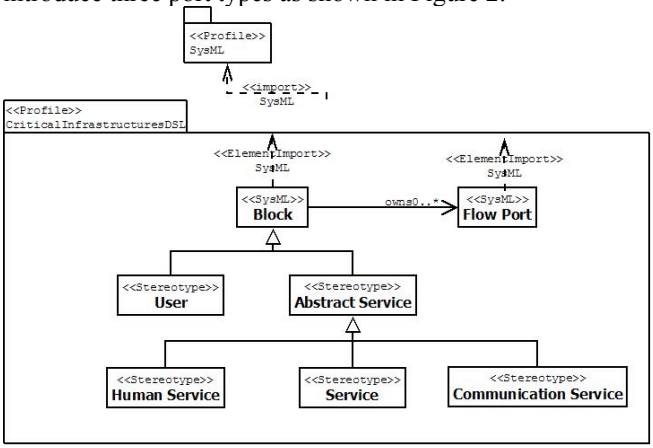


Figure 1: Relationship of the *Abstract Service* and *User* constructs with the *Block* entity of SysML

In order to relate the message flow generating from a service/user with the transport mean that allows it (e.g., internet connection), we have identified a particular type of (non-atomic) *Flow Port*, namely the *Connection Port Group*, with the aim of grouping together one or more message ports with one (in) communication port.

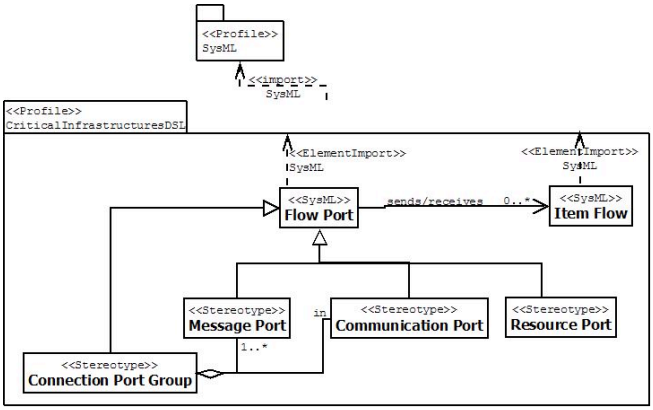


Figure 2: Relationship of the *Message Port*, *Communication Port*, *Resource Port*, and *Connection Port Group* with the *Flow Port* of SysML

The specialization of *Flow Ports* in three types obviously requires that also *Item Flow* be specialized accordingly. The type of the item that can flow through an atomic port (e.g., water, power) in SysML is specified by the *FlowProperty* stereotype, which can be simply a label. In our case, we want to distinguish between: *message*, *connectivity*, and *resource*, which we define as a specialization of *FlowProperty*. Instead, non-atomic *Flow Ports* in SysML are defined through a FlowSpecification object, which is a collection of FlowProperty objects, each referring to a single item. In SysML, items flow through *Connectors*, used to link blocks. For graphical convenience only, we have defined a SysML

connector specialization for *message flow* to represent it as a dashed arrow line (see Table II below).

As we want to design analysis scenarios for crisis management, we need to represent the events that may happen and what services/users they may affect. Here we want to represent just the type of the *external event*, such as earthquake, flood, and so on, and its "affecting" relationship to one or more scenario entities. Therefore, we intend the event being an abstract element outside the model (part of the environment) but influencing it, and so this definition specializes that of the *Actor* in UML.

Finally, each kind of service or user element, being a UML Class, might be modeled internally through a *Behavior* object, which is the link to one or more behavioral descriptions of the scenario that we will treat as future work.

The following tables include the list of all the constructs (Table I) and relationships (Table II) of the proposed CEML meta-model, with the corresponding formal notation describing the extension from the SysML profile and UML references, and the graphical symbol we have associated to them to be used in our diagrams.

TABLE I. CEML MODELING CONSTRUCTS, BASE SYSML META-CLASS, AND CORRESPONDING GRAPHICAL NOTATION

CEML Modeling Constructs	*Base SysML Metaclass*	*Graphical Notation*
Abstract Service	SysML::Blocks::Block	NA
Service	SysML::Blocks::Block	Service Name
Human Service	SysML::Blocks::Block	Human Service Name
Communication Service	SysML::Blocks::Block	Com. Service Name
Behavior	UML::CommonBehaviors:: BasicBehaviors:: Behavior	NA
External Event	SysML::Actor	External Event name
User	SysML::Blocks::Block	User name
Message	SysML::Property::Flow Property	Message name
Resource	SysML::Property:: FlowProperty	Resource name
Connectivity	SysML::Property:: FlowProperty	NA

TABLE II. CEML RELATIONSHIPS, BASE SYSML META-CLASS, AND CORRESPONDING GRAPHICAL NOTATION

CEML Relationships	*Base SysML Metaclass*	*Graphical Notation*
Resource Flow	SysML::Ports&Flows:: ItemFlow	Resource name →
Connectivity Flow	SysML::Ports&Flows:: ItemFlow	Connectivity name →
Message Flow	SysML::Ports&Flows:: ItemFlow	Message name ⇢
Abstract Port	SysML::Ports&Flows:: FlowPort	NA
Connection Port Group	SysML::Blocks::Block	Connection Port Group Name
Message Port	SysML::Ports&Flows:: FlowPort	Message Port Name
Communication Port	SysML::Ports&Flows:: FlowPort	Communication Port Name
Resource Port	SysML::Ports&Flows:: FlowPort	Resource Port Name
Impact	UML4SysML::Association	Affecting →

In a UML profile, "well-formedness" rules, such as the constraints listed in sub-section C, can be encoded in OCL, which is a declarative formal language to express properties of UML models. An OCL rule is defined within a context, that is, the element to which some Boolean expression, specified by the rule, should apply. We give here some representative examples of OCL implementation of the constraints of our meta-model. Specifically, through rule C4, we show how to link one or more subtypes of *Port* to the corresponding subtype of the *AbstractService* construct. Instead, through the first part of rule C12, we show how to link an Item specialization to the corresponding *ItemFlow* and *Port* subtypes. Finally, both rules C8 and C10 are based on a invariant on the use of connector subtypes. Namely, in this example, the *Impact* relationship is *always* originated by an *ExternalEvent* construct towards a *User* or an *AbstractService* construct.

C4. A **communication service** element has *0..n* incoming **resource port** elements and *1..n* outgoing **communication port** elements.

```
Context SysML::Blocks::Block
self. oclIsTypeOf(CommunicationService) implies
(self.attributes->select(oclIsTypeOf(MessagePort))->size()=0) and
(self.attributes>
select(oclIsTypeOf(CommunicationPort).direction='out')->size()>0)
and (self.attributes->
select(oclIsTypeOf(CommunicationPort).direction='in')->size()=0)
and (self.attributes->
select(oclIsTypeOf(ResourcePort).direction='out')->size()=0)
and (self.attributes->
select(oclIsTypeOf(ResourcePort).direction='inout')->size()=0)
and (self.attributes->
select(oclIsTypeOf(CommunicationPort).direction='inout')->size()=0)
```

C8. Every **abstract service** element is affected by *0..n* **external event** elements by means of the **impact** element.

C10. A **user** element is affected by *0..n* **external event** elements by means of the **impact** element

```
Context UML4SysML::Association
self.oclIsTypeOf(Impact) implies
(self.memberEnd->size()=2 and self. memberEnd -> exists(p |
p.oclIsTypeOf(ExternalEvent)) and
(self.navigableOwnedEnd->size()>0 and
self.navigableOwnedEnd -> forAll(p |
p.oclIsTypeOf(AbstractService) or p.oclIsTypeOf(User)))
```

(First part of) **C12**. A resource flow element is linked to *1..n* resource elements and holds between 2 resource port elements.

```
Context SysML::Ports&Flows:: FlowPort
self.oclIsTypeOf(ResourcePort) implies
(self.type.oclIsTypeOf(FlowSpecification) and self.type.attributes-
>size()>0 and self.type.attributes ->forAll(r |
r.type.oclIsTypeOf(Resource)))
```

IV. GUIDELINES FOR CEML MODELS SIMULATION

CEML is a language to support experts of the various critical infrastructures in creating global representations of these systems, and their interactions, in order for them to analyze problems and take strategic decisions. As simulation plays a key role in this activity, it is desirable that the constructed CEML models are then used to generate input code for simulators. This means that a CEML model needs to be implemented in a simulation language of some kind, and integrated with the simulation-specific constructs of the language required to run the experiments.

This objective is achieved by following a model-based design methodology. Indeed, referring to the model driven architecture paradigm [16], CEML is located at the Platform Independent Model (PIM) layer, being a domain specific language formally defined as a profile of SysML. In order to be actually used in the context of simulation, mapping rules of the CEML meta-model to some lower level simulation language need to be provided to convert CEML models into Platform Specific Models (PSM), executable by specific simulators.

In this section we present a software system architecture allowing building and verifying a CEML model and, finally, transforming it into a platform-specific code. Then, we present how the transformation is performed by means of a set of mapping rules from the CEML modeling constructs to a set of generic modeling constructs typical of discrete event simulation. This approach has been applied on a real case study within the EU project MOTIA [21] [22] [23].

A. The Architecture for CEML Modeling and Simulation

The CEML approach is enabled by the architecture depicted in Figure 3, that shows how CEML can be used together with existing simulation environments.

The architecture highlights a complete decoupling between the modeling and simulation functions. Indeed, although every simulator provides its own design interface, often these interfaces require some programming skills and are not graphical. Moreover, in a simulator program the model of the real world to test is mixed with simulation programming functions, thus making models reuse and evolution a more complex task. In the proposed architecture, instead, the modeling and simulation design activities may have a different focus and so can even be performed by different users.

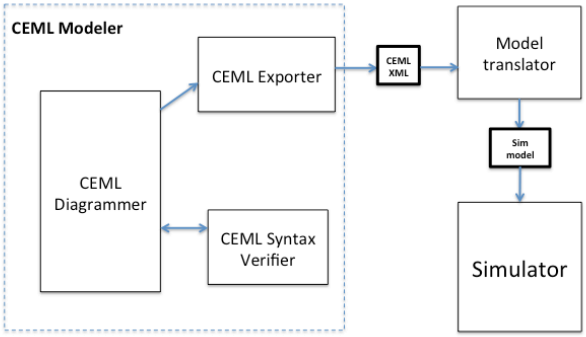


Figure 3: Software system architecture to support CEML models definition and simulation

The ***CEML Modeler*** component allows for editing correct CEML models which may be exported to an XML format. Specifically, the modeling environment consists of: the ***CEML Diagrammer***, that implements the CEML's meta-model and provides a graphical interface for editing the models; the ***CEML Syntax Verifier***, that, during the editing, allows to verify that the model is well-formed with respect to the CEML modeling rules; and a ***CEML Model Exporter*** for an XML serialization of the CEML model. As CEML is formally defined as a SysML profile, with the modeling rules expressed by OCL constraints, the *CEML Modeler* component can be implemented by any UML tool that supports SysML and profiles, such as Topcased [24] or UModel [25]. All of these tools provide an XML export function of the models. However, an XML schema has been created specifically for CEML to simplify the XMI serialization of CEML models leaving just the relevant data for the simulation stored in the XML document.

The ***Model Translator*** component implements the mapping rules that allow transforming a CEML model into code for a specific simulator. The output of the *Model Translator* essentially contains the data and specifications needed to instantiate the structural model or code of a simulation scenario. Clearly, the user of the simulator must configure the simulation scenario and add the required simulator specific code to obtain an executable program.

B. The CISP Simulation Platform

Although CEML has not been initially conceived for any particular technology, an existing discrete event simulator for critical infrastructures developed at ENEA, called CISP [22] [26], has been used to experiment CEML's integration with a simulation environment. Other than critical infrastructures, CISP provides a set of predefined components that can be easily extended in order to simulate other domain specific

scenarios. Indeed, these components implement the building blocks of a general discrete simulation language, such as:

- **entity**: a simulation element that flows into the system (e.g., materials or workpieces, documents or data packets in a computer systems);
- **source**: a simulation element that places entities in the system;
- **sink**: a simulation element that receives entities;
- **queue**: a simulation element that collects entities;
- **decider**: a simulation element that takes decisions, for instance about entities flowing;
- **event**: a simulation element, modeling the change of state in the simulation environment. It is instantaneous and does not require time;
- **activity**: a simulation element, modeling a set of operations that transform the state of an entity and require time to be executed.

CISP is based on Discrete Event Simulation (DES) paradigm [27]. DES is an operational research technique where the simulation is advanced from event time to event time rather than using a continuously advancing time clock as in continuous simulation.

C. *PIM-PSM Mapping Rules: from the CEML Modeling Constructs to the CISP Components*

In the following we show how the transition from the CEML modeling constructs (at PIM level) to the CISP components (at PSM level) can be performed. In particular, we specify a set of mapping rules, covering a subset of the CEML modeling constructs that allow transforming a CEML model into a specification of the structure of a CISP simulation scenario. This scenario has to be completed with behavioral aspects by CISP users.

***AbstractService* mapping rule**

The *AbstractService* CEML modeling construct is implemented at the PSM level by using the *Activity*, the *Queue*, the *Sink*, and *0..n Decider* components. The same applies to the *Service*, the *HumanService*, and the *CommunicationService* CEML modeling constructs. At the PSM level, the *Activity* component allows modeling of operations related to the *AbstractService*; the *Queue* and the *Sink* components allow to store, respectively, input and output resources either to be processed or processed by the *AbstractService*. Finally, the *Decider* component allows modeling decisions taken by the *AbstractService* about its operations and depending on the internal status and external events affecting its behavior. Figure 4 presents the *AbstractService* mapping rule.

***User* mapping rule**

As the *AbstractService* mapping rule, the *User* CEML modeling construct is implemented at the PSM level by using the *Activity*, the *Queue*, the *Sink*, and the *Decider* component. At the PSM level, the *Activity* component allows modeling of user operations to interact with the external world (e.g., exchange of messages, receipt of resources). The *Queue* and the *Sink* components allow modeling, respectively, of issuing and receiving messages and resources. Please note that, with respect to the *AbstractService* mapping rule, here the *Decider* component allows modeling decisions taken by the *User* about message sending and depending on the wellness level and external events affecting it. Figure 5 presents the *User* mapping rule.

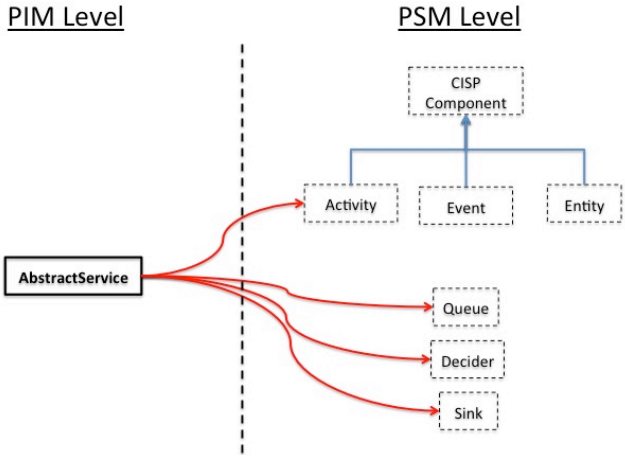


Figure 4: *AbstractService* Mapping to the PSM Level

***ExternalEvent* mapping rule**

The *ExternalEvent* CEML modeling construct is implemented at the PSM level by using the *Event* CISP component. Figure 6 presents the *ExternalEvent* mapping rule.

***Resource* mapping rule**

The *Resource* CEML modeling construct is implemented at the PSM level by using the *Entity* CISP component. The same applies to both the *Message* and the *Connectivity* CEML modeling construct. Figure 7 presents the *Resource*/*Message*/*Connectivity* mapping rule.

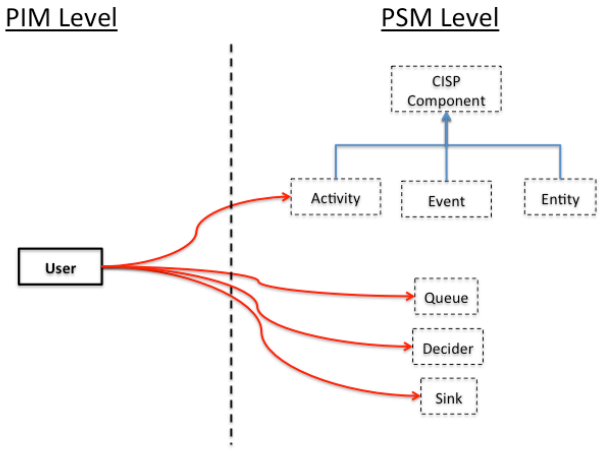


Figure 5: *User* Mapping to the PSM Level

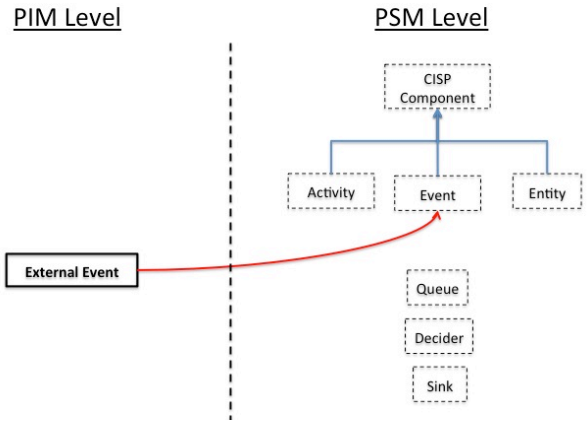


Figure 6: *ExternalEvent* Mapping to the PSM Level

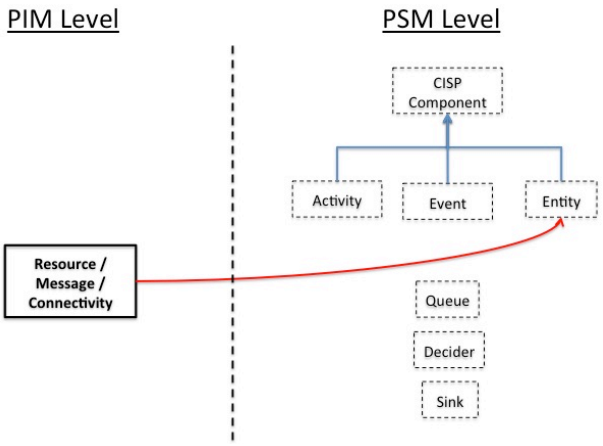


Figure 7: Mapping of either *Resource* or *Message* or *Connectivity* to the PSM Level

V. CRITICAL INFRASTRUCTURES COLLABORATION DESIGN PATTERNS

Design patterns are proving to be one of the most promising methodological tools to support building of models and, more in general, ICT artifacts like software programs. Currently, there are several proposals of design patterns in different fields, e.g., UML design patterns for software engineering [28], workflow patterns for business process management [29], and ontology design patterns for ontology building [30]. Here we propose some examples of domain-specific design patterns, devoted to facilitate modeling of interaction and communication exchange arising among emergency services providers and citizens to solve the crisis. In particular, a CDP allows representing a chunk of the reality where collaboration is performed. By using this approach, modelers can create a repository of CDPs to be reused to describe similar scenarios. Furthermore, these patterns may result useful when analyzing the modeled system. Especially in the case of analysis by simulation of big size models, structured diagrams may help users in the identification of criticalities and in planning ways to probe them. In the following, five CDP examples we have identified are presented: clustered service, basic communication, heterogeneous networking, single service provider, and infrastructure.

CDP1. Clustered Service

Figure 8 shows the clustered service CDP devoted to model collaboration arising among different services working together to either provide or produce or transport a resource (e.g., energy, water). In particular, the objective of this CDP is to model exchange of resources and information. Furthermore, this CDP models the physical connection provided by a communication service and allowing information exchange.

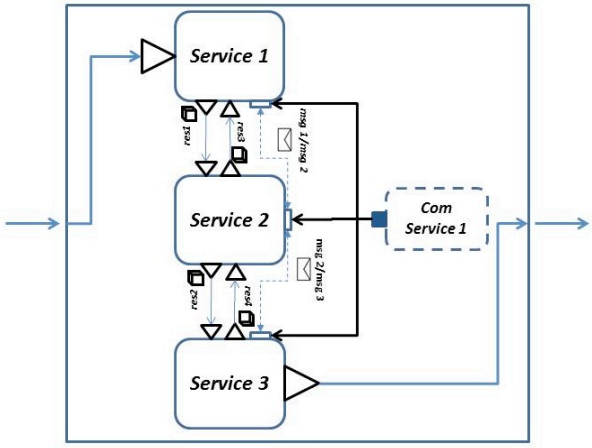


Figure 8 Clustered Service CDP

CDP2. Basic Communication

Figure 9 shows different cases concerning the basic communication CDP representing a simple exchange of information where the physical connection is not deemed relevant for the modeling purposes (e.g., oral communication).

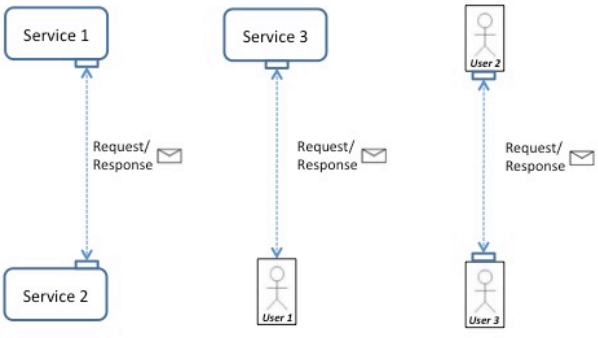


Figure 9 Different cases concerning the Basic Communication CDP

CDP3. Heterogeneous Networking

Figure 10 presents the heterogeneous networking CDP modeling a network of different communication services, guaranteeing the physical connection between two services.

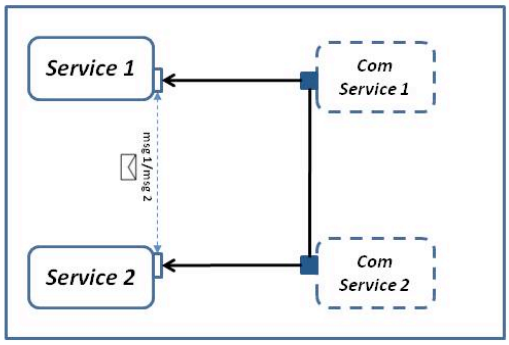


Figure 10 Heterogeneous Networking CDP

CDP4. Single Service Provider

Figure 11 shows the single service provider CDP representing a potentially risky situation where a user (or a service) receives a resource just from a service and this service has vulnerability in case of an external event. This type of CDP is inherently different from the above mentioned CDPs. Whereas CDP1, CDP2, and CDP3 have been mainly conceived to support the domain expert in the modeling phase, CDP4 can be used also in the analysis phase. In fact, CDP4 can be used as a basis for a "situation awareness service", by detecting its presence in a previously built model of scenario.

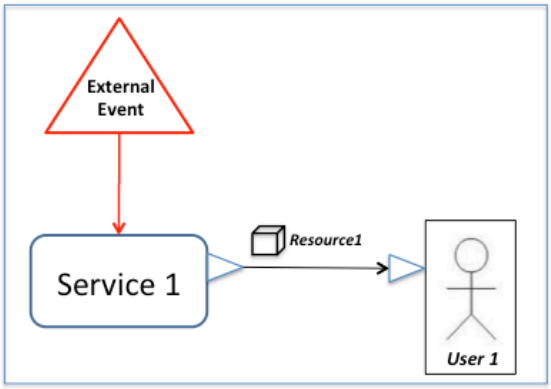


Figure 11 Single Service Provider CDP

CDP5. Infrastructure

Figure 12 shows an example of CDP that can be useful in the analysis phase of the modeled system: the infrastructure CDP. This represents a means to highlight in the model that some services belong to the same infrastructure. The aim is to enable the design and analysis of the infrastructure as a whole, through its relationship with other infrastructures of the model. With respect to the clustered service CDP, the component services of the infrastructure CDP are not necessarily physically connected.

VI. EMERGENCY SCENARIO EXAMPLE

The objective of this section is to demonstrate the usability and flexibility of the proposed modeling framework by describing a real emergency scenario occurred after an earthquake [31]. In particular, here we focus on the main services, resources and users related to the Italian Civil Protection (ICP) emergency management protocol. Figure 13, Figure 14 and Figure 15 present different excerpts from the addressed scenario model. Please note that when some elements of a model appear in more than one figure we omit to repeat some parts of the model itself due to presentation purposes. For the sake of clarity, we also omit some details, as our aim is to demonstrate the usability and flexibility of the proposed modeling framework. A detailed description of the scenario is available in [31]. After an earthquake event, the ICP is able to have a global picture of the impact of this event by using sensor networks, simulation tools, and specific expert team reports. The *Mixed Operative Center* (*COM* in Figure 13) is established near the areas mostly damaged by the earthquake. In this example, the *COM* plays the role of final user. Then there are the *Emergency Services*, the *Emergency Call Service*, and the *Lifeline networks*. The *Emergency Services* represent all actors involved in the emergency management protocol. We describe the details about this service using the clustered service CDP (Figure 14). The *Emergency Call Service* represents the network of emergency call centers devoted to receive feedbacks from user in order to assess how well ICP is facing the emergency. The *Lifeline Networks* element models the infrastructure networks (e.g., electrical distribution and telecommunications network, gas and water pipelines, water treatment systems) of the damaged area. Evaluation of the lifeline performances is one of the most important tasks during an emergency to allow rescue teams to properly and safely operate during an emergency. The networks and their dependencies can be further specified using an appropriate clustered service CDP.

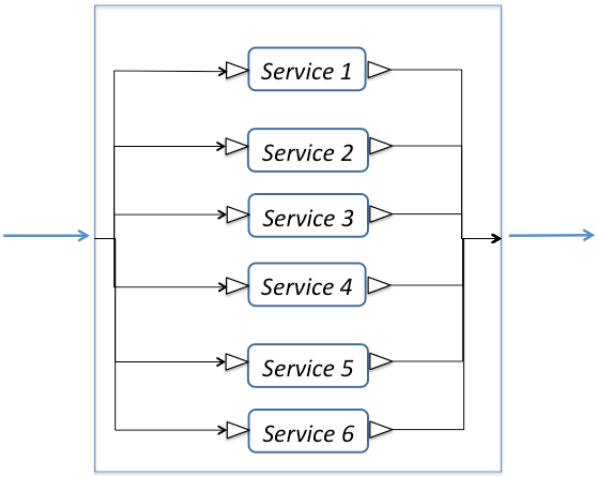


Figure 12 Infrastructure CDP

The *Telco Network* communication service models the connectivity services and resources operating in the area.

By using the clustered service CDP, it is possible to refine the definition of the *Emergency Services* to model the

coordination messages that are exchanged among the major actors during emergency management (Figure 14). The decisional board is represented by the *National Civil Protection Service* (*SNPC*). The coordination messages aim to gather information about available resources at a national, regional, provincial, and local level. The *Direction and Command on site* (*DiComaC*) service is in charge of resources distribution and operations management. All decisions rely on the information about the lifeline performance provided by the *Lifeline Owners* service. Figure 14 shows also the output resources of the *Emergency Services* to the *COM*.

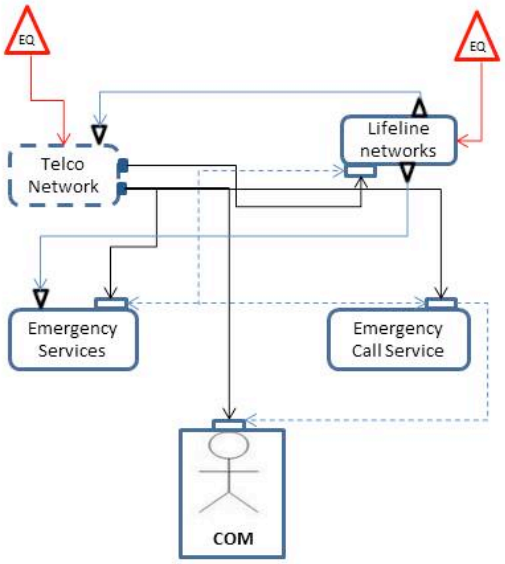


Figure 13 Emergency scenario example

Figure 15 shows how heterogeneous networking CDP can be used to represent different physical connections among services. The *SNPC* uses the public telecommunication network and the internet to exchange messages with the *DiComaC* (Figure 15 a.). On the other hand, for the communication between the *DiComaC* and the ICP rescue teams service it is possible to have several ICT emergency communication channels: telecommunication network, ad hoc network, radio network, and the internet (Figure 15 b.).

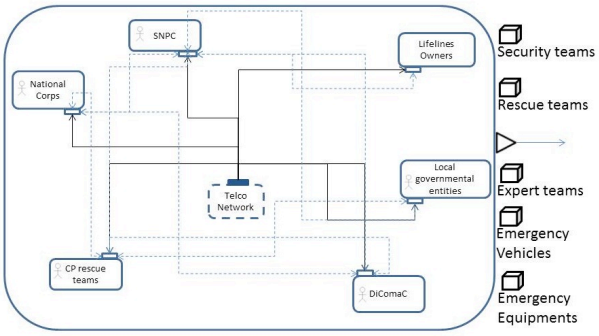


Figure 14 Clustered Service CDP example

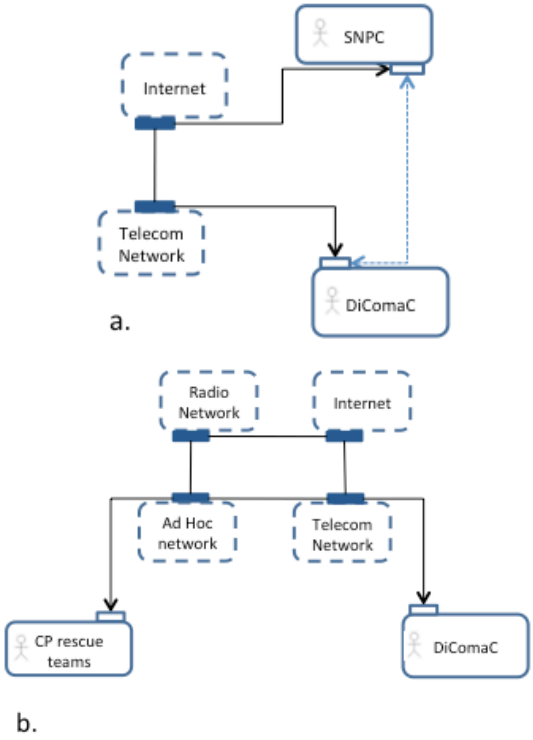


Figure 15 Heterogeneous networking CDP examples

VII. EVALUATION

A further step towards a proper validation of the language has been made by using CEML to model the Italian System for Public Connectivity (SPC), within the activities of the MOTIA project. SPC is a system of federated technological ICT infrastructures of Public Administrations (PAs) to provide eGovernment services to citizens via a shared interface. The various PAs may communicate through a Qualified Internet Service Provider (Q-ISP).

The interest of the MOTIA project for this case study has been to use a CEML model as a means to a simulation-based quantitative analysis of dependencies of the PAs logical network from the underlying telecommunication network. In particular, the functioning of the modeled system has been simulated under normal and perturbed conditions caused by external events. For deeper insights into this system and the experiments we have conducted we remind the reader to [22]. Instead, here we report in greater details some lessons learnt about usability of our approach.

From the modeling side, we have had the opportunity to apply CEML to another real case of crisis management and so to verify the effectiveness of the constructs of the language and of the CDPs. The modeling approach has been evaluated under two perspectives: that of the *domain expert* who knows the system and can judge the adequacy of CEML in representing the system through one or more models; and that of the *system analyst* (a network analyst for this case study) who needs to configure the simulator and hence can judge how the CEML models are useful for his/her understanding of the system.

The structure of the modeled system is simpler than that for the emergency scenario presented in Section VI. Indeed, we have defined only two types of services, namely the *PA service*, as human service, and the *Q-ISP service*, a type of user representing the *Citizen*, and an external event to inject faults in the Q-ISP service. Instead, the complexity of the system is in the size of the PAs logical network and hence on the communication flow between this network and the underlying telecommunications network. This complexity needs to be handled by the network analyst when configuring the simulator with appropriate dependencies metrics, based on the analysis of the structural model.

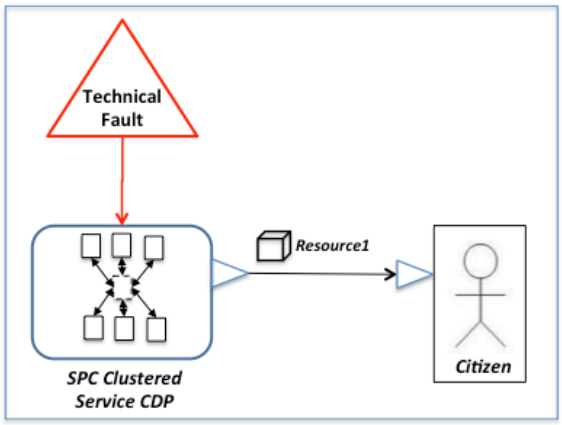


Figure 16 Single service provider CDP example in the SPC case study

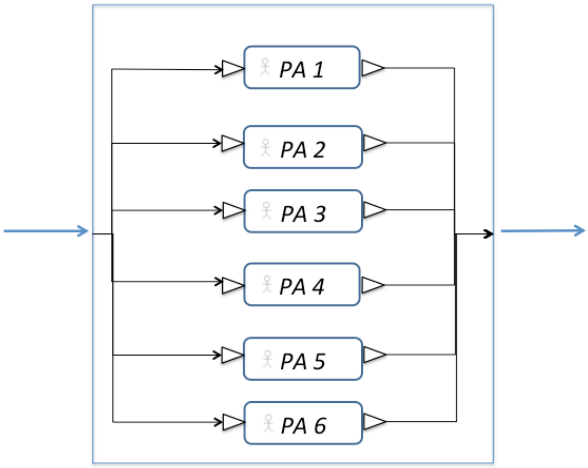


Figure 17 Simplified infrastructure CDP example in the SPC case study

For this activity, which has been carried out with the support of network analysis experts, the use of three CDPs in the model, namely the single service provider CDP together with a clustered service CDP, as shown in Figure 16, and the infrastructure CDP shown in Figure 17, has been convenient. The first two patterns have been used to highlight to the network analyst that the system to study consists of two collaborating ICT infrastructures that produce resources (i.e., PA documents like certifications) to people and that no backup system exists to provide the same resource in case of failure. The other pattern has been used to highlight that one of the two infrastructures (the PAs network) is composed of various services that do not communicate directly and that this infrastructure needs be evaluated as a whole through simulation.

From this experience we have obtained the following important feedbacks:

- in some fields, like critical infrastructures protection, a domain specific modeling language and implementing tools can be used to support the activity of simulation experts (system analysts);
- the domain experts have appreciated the fact that CEML is more concise than a general purpose modeling language (e.g., SysML);
- a domain specific language is a first step to help system analysts in building simulation models but a smarter editor could be implemented to elicit more knowledge from the domain experts and make it explicit;
- CEML and the CDPs can be effective means to develop methodologies and/or processes for the analysis of dependencies in complex systems, like that for the quantitative analysis of interdependencies of ICT systems defined in the MOTIA project [23].

As a future work we intend to formally evaluate these results.

VIII. CONCLUSIONS

In this paper we presented an approach to build models concerning crisis and emergency scenarios. Our approach is based, first of all, on the CEML language and the related meta-model consisting of a set of modeling constructs, a set of relationships, and a set of modeling rules. Then, it proposes some methodological guidelines, consisting of a system architecture and a set of PIM-PSM mapping rules, to allow CEML models to be part of the input data required by simulation environments. Finally, it proposes a modeling methodology based on collaborative design patterns, i.e., reusable solutions to recurrent modeling problems, tailored to model interaction and communication exchange arising during the crisis.

Currently, CEML supports modeling structural aspects of a scenario. We are working on extending the language and the related meta-model to behavioral aspects. For example, in [32], we present a method based on ECA (Event Condition Action) rules [33]. Finally, we are implementing the proposed architecture to permit CEML models to be simulated by other existing simulation tools.

ACKNOWLEDGEMENTS

This work is partially supported by the European Project MOTIA (JLS/2009/CIPS/AG/C1-016). This publication reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

REFERENCES

[1] OMG-MDA, "MDA Guide, version 1.0.1," Available at: http://www.omg.org/mda/presentations.htm, 2003. Retrieved on 4th April, 2011.

[2] F. D'Antonio, M. Missikoff, and F. Taglino, "Formalizing the OPAL eBusiness ontology design patterns with OWL," Proc. of the IESA 2007 Conf., pp. 345-356, 2007.

[3] OMG-SysML, "OMG Systems Modeling Language" version 1.2. Available at: http://www.omgsysml.org/. 2010.

[4] J. Warmer and A. Kleppe, "The Object Constraint Language: Getting Your Models Ready for MDA" (2 ed.). Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2003.

[5] A. De Nicola, A. Tofani, G. Vicoli, M. L. Villani, Modeling Collaboration for Crisis and Emergency Management. The First International Conference on Advanced Collaborative Networks, Systems and Applications - COLLA 2011 Luxembourg June 19-24, 2011 isbn:978-1-61208-008-6, 2011.

[6] http://en.wikipedia.org/wiki/Design_pattern_(computer_science). Retrieved on 26th February 2011.

[7] S.M. Rinaldi, J.P. Peerenboom, and T.K. Kelly, "Identifying, understanding, and analyzing critical infrastructure interdependencies," Control Systems Magazine, IEEE , vol.21, no.6, pp.11-25, Dec 2001.

[8] G. A. Bagheri Ebrahim, "UML-CI: A reference model for profiling critical infrastructure systems," Inf. Syst. Frontiers, 12(2), 2010.

[9] W. Wang, X. Zhang, C. Dong, S. Gao, L. Du, and X. Lai, "Emergency Response Organization Ontology Model and its Application," Proc. of ISISE 2009 Symp., pp. 50-54, 2009.

[10] C.X. Dong, W.J. Wang, and P. Yang, "DL-Based the Knowledge Description of Emergency Plan Systems and Its Application," Proc. of MUE'09, pp. 364-368, 2009.

[11] P. C. Kruchten, "A human-centered conceptual model of disasters affecting critical infrastructures," Proc. of ISCRAM 2007 Conf., pp.. 327-344, 2007.

[12] M. A. Sicilia and L. Santos, "Main Elements of a Basic Ontology of Infrastructure Interdependency for the Assessment of Incidents," LNCS, Springer-Verlag, Vol.5736, pp. 533-542, 2009.

[13] S. Parr, A Visual Tool to Simplify the Building of Distributed Simulations Using HLA. Information Security 12, 151-163, 2003.

[14] A. Tolk, Avoiding Another Green Elephant – A Proposal for the Next Generation HLA based on the Model Driven Architecture", 02F-SIW-004, Fall Simulation Interoperability Workshop, Orlando, Florida, September 2002.

[15] H. Zhang, H. Wang, D. Chen, and G. Zacharewicz, A model-driven approach to multidisciplinary collaborative simulation for virtual product development, Advanced Engineering Informatics, Volume 24, Issue 2, Pages 167-179, ISSN 1474-0346, 10.1016/j.aei.2009.07.005, April 2010.

[16] A. G. Kleppe, J. Warmer, and W. Bast, MDA Explained: The Model Driven Architecture: Practice and Promise, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2003.

[17] L. McGinnis and V. Ustun, "A simple example of SysML-driven simulation," in Proc. of WSC'09, pp. 1703-1710, 2009.

[18] M. Nikolaidou, V. Dalakas, L. Mitsi, G.-D. Kapos, and D. Anagnostopoulos, "A SysML Profile for Classical DEVS Simulators", in Proc. of the 3rd Int. Conf. on Software Engineering Advances, IEEE, pp. 445-450, 2008.

[19] V. Cardellini, E. Casalicchio, and E. Galli. Agent-based Modeling of Interdependencies in Critical Infrastructures through UML, In Proc. of the 2007 spring simulation multiconference (SpringSim '07) – Vol 2, pp. 119—126, 2007.

[20] M. Rosemann and P. Green, "Developing a meta-model for the Bunge-Wand-Weber ontological constructs," Inf. Syst. 27(2): 75-91, 2002.

[21] G. D'Agostino, G. Cicognani, A. De Nicola, and M. L. Villani, Case Study. Deliverable of the Activity 4 of the European Project MOTIA (JLS/2009/CIPS/AG/C1-016), 2012.

[22] G. D'Agostino, A. De Nicola, A. Di Pietro, G. Vicoli, M.L. Villani, and V. Rosato, A Domain Specific Language for the Description and the Simulation of Systems of Interacting Systems, in Advances in Complex Systems, Vol. 15, Suppl. No. 1, 2012.

[23] The MOTIA European Project (JLS/2009/CIPS/AG/C1-016). http://www.motia.eu. Last accessed on 20th June 2012.

[24] TOPCASED The Open-Source Toolkit for Critical Systems, http://www.topcased.org.

[25] UModel Enterprise Edition, Altova, http://www.altova.com/umodel/sysml.html.

[26] G. Vicoli, Discrete Event Simulation. Oral presentation at Workshop UTMEA, Soluzioni per l'energia e l'ambiente, Rome, 2010-09-07.

[27] S. Robinson, Simulation: The Practice of Model Development and Use, Wiley, Chichester, UK, 2004.

[28] A. Spiteri Staines, Modeling UML software design patterns using fundamental modeling concepts (FMC), In Proc. of ECC'08 Conf., pp. 192-197, 2008.

[29] W.M.P. van der Aalst, A.H.M. ter Hofstede, B. Kiepuszewski, and A.P. Barros, "Workflow Patterns," Distributed and Parallel Databases, 14(3), pp. 5-51, 2003.

[30] A. Gangemi and V. Presutti, "Ontology design patterns," In: Handbook on Ontologies, 2nd edn. International Handbooks on Information Systems. Springer, Heidelberg, 2009.

[31] M. Dolce, S. Giovinazzi, I. Iervolino, E. Nigro, and A. Tang, "Emergency management for lifelines and rapid response after L'Aquila earthquake," Progettazione sismica, n. 3, 2009.

[32] A. De Nicola, G. Vicoli, and M. L. Villani, A Rule-based Approach for Modeling Behaviour in Crisis and Emergency Scenarios, Enterprise Interoperability V, Ed. R. Poler, G. Doumeingts, B. Katzy, and R. Chalmeta, Springer, 2012.

[33] K. R. Dittrich, S. Gatziu, and A. Geppert. The Active Database Management System Manifesto: A Rulebase of ADBMS Features. Lecture Notes in Computer Science 985, Springer, pp. 3-20, 1995.

Integration of Up-to-Date Technologies for Emergency Response

Fire Response Community in Smart Space

Alexander Smirnov, Tatiana Levashova, Nikolay Shilov, Alexey Kashevnik
Laboratory of Computer Aided Integrated Systems
St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences
SPIIRAS, 39, 14th line, St. Petersburg, 199178, Russia
{smir, tatiana.levashova, nick, alexey}@iias.spb.su

***Abstract*—The paper addresses the problem of organizing a resource community in a smart space. The resources making up the community aim at joint emergency response actions. A smart framework for integrating emerging technologies of smart space, Web-services and Web-based communities was developed. In this framework, Web-services represent smart space's resources and Web-based community members. A service-oriented architecture was designed to coordinate Web-service interactions. The smart framework applicability was tested via a scenario-based organization of an emergency response community aiming at fire response actions. The main research challenge is to show how facilities provided by the emerging technologies of Web-based communities and smart spaces can be used for emergency management.**

Keywords—smart space; service-oriented architecture; Web-services; Web-based community; emergency response

I. INTRODUCTION

Emerging technologies of Web-based communities, smart spaces, and Web-services, have the potential to impact emergency management dramatically. The research on the investigation of the possibilities of these technologies for emergency response was initially introduced in [1].

Web-based communities offer advantages of instant information exchange that is not possible in real-life communities. Availability of operational information [2][3] as well as potentialities to instant information exchange [4][5][6] are of great importance to success in emergency response operations. Usually, in such operations joint efforts of independent parties are required. To involve the parties in the emergency response actions and to coordinate them, operational information about the parties' facilities, availabilities, locations, etc. is needed. In this connection, organization of a community of emergency response actors as a Web-based community, whose members can share and exchange operational information, seems to be a promising idea.

Unfortunately, in real life it is occurred quite often that people would not like sharing information – "At the agency level, and even within agencies, there has been the culture that you don't share information for a variety of reasons, whether it's because of classification or "need-to-know," you just don't share information. There are also sometimes some bureaucratic or personal reasons [7]". Smart spaces provide good facilities to overcome this problem since a smart space is a sharable system by definition. Smart space is any virtual or real location equipped with passive and active artifacts. These artifacts have the processing and communication capabilities to interact with each other in a (mutually) beneficial way [8]. The smart spaces gather information from the environment and provide embedded services according to this information. This means first, that people do not have to provide any information if they do not have intentions of doing that; instead, the smart space will do this, and second, that smart spaces act in a context-aware manner.

The information sharing facilities provided by Web-based communities and smart spaces have suggested an idea to combine these facilities for organization of emergency response communities. These facilities are supplemented with smart spaces' capabilities to context aware service provision.

Any smart space is comprised of a large number of informational, computational, and acting resources. Web-services offer advantages of seamless information exchange between autonomous resources of smart spaces [9]. This fact was a reason to use Web-services as mediators between resources of the smart space and members of the emergency response community.

Research presented in this paper addresses the organization of a resource community in a smart space. The purpose of the community organization is participation of its members in emergency response actions. The main research challenge is to show how facilities provided by the emerging technologies of Web-based communities and smart spaces can be used for emergency management.

A smart framework that serves to integrate concepts of smart space, Web-services and Web-based communities is proposed to achieve the research purpose. This framework is based on the earlier developed hybrid technology supporting context aware operational decision support in pervasive environments [10]. Although some research has been done since the hybrid technology was published, this paper presents first extension of this technology with Web-based communities.

In the framework, resources of the smart space are represented by sets of Web-services. As a result of this representation, the emergency response community comprises Web-services representing units taking these actions. Service-oriented architecture is used to coordinate

Web-service interactions. The Web-services constituting this architecture implement resources' functionalities, produce model of the emergency situation, provide emergency response services, and represent participants of the emergency response actions and other people somehow involved in the emergency situation. An applicability of the proposed framework is demonstrated via a scenario-based organization of a Web-based community aiming at fire response.

The rest of the paper is structured as follows. Section II provides a comparative analysis of the present research with related ones. In Section III the scenario of fire response actions is described. The smart framework is discussed in Section IV. Results of scenario execution are given in Section V. Main research findings and approach limitations are discussed in the Conclusion.

II. Related Research

This Section focuses on approaches dealing with integration of different emerging technologies to multi-parties cooperation, particularly to emergency response. The main focus of the discussion is Web-based communities and smart spaces since the combination of facilities provided by these technologies is the main research challenge. Other problems that the present research concerns as e.g., ontology management, service composition, constraint satisfaction problem solving, etc. are out of consideration in this Section.

The role of social media and online communities in emergency situations is being thoroughly investigated within the research area of crisis informatics [11]. Online forums [12], Web portals [13], Tweeter [14][15], micro-blogging [16], social networks [17][18], and other forms of social media are believed to be powerful tools enabling collaboration of different parties to respond more effectively to emergencies.

There is no extensive literature on the subject of emergency management in smart spaces. One of the possible examples is DrillSim environment [19]. The purpose of this environment is to play out a crisis response activity where agents might be either computer agents or real people playing diverse roles. An activity in DrillSim occurs in a hybrid world that is composed of (a) the simulated world generated by a multi-agent simulator and (b) a real world captured by a smart space. In order to capture real actors in the virtual space, DrillSim utilizes a sensing infrastructure that monitors and extracts information from real actors that is needed by simulator (such as agent location, agent state, etc.).

To some extent potentialities of smart spaces in emergency have been exploited in an architecture that intends to improve the collaboration of rescue operators in emergency management via their assistance by a Process Management System [20]. This system is installed on the smart phones and PDAs of the rescue operators. It manages the execution of emergency-management processes by orchestrating the human operators with their software applications and some automatic services to access the external data sources and sensors.

In part of integration of emerging technology-driven paradigms for different purposes, ideas of an integration of paradigms of Web services, Web 2.0, pervasive, grids, cloud computing, situated computing, and crowd sourcing that are considered to be the candidates that can support collective resource utilization and multi-parties cooperation with mutual interests [21] can be pointed out. Integration of paradigms of virtual organizations and Semantic Web is offered to be used for organization of resources and services into a collaborative association to handle different kinds of emergency events [22].

The above approaches address different aspects of emergency management. All they integrate various emerging technologies to achieve their goals. The novelty of the present research lies in combination of information from the smart space and from Web-based communities for coordination of emergency response activities.

Like the approaches considering the problem of searching for efficient transportation routes within the emergency response problem (e.g., [23][24]), this research searches for such routes and uses them as the basis for joining independent units from diverse locations in a collaborative community. The community members are coordinated via Web-based interface. They are provided with the ability to exchange operational information and interact on-line using different Internet accessible devices.

III. Scenario

Suddenly, in some area inside a smart space the emergency event of a fire has started. Resources of the smart space as, e.g., fire sensors recognize it and send the appropriate signal to a smart space's service taking the role of the dispatcher. In the surroundings of this area available mobile fire brigades and emergency teams as well as hospitals with free capacities are found. Based on some criteria (see Section V) several of the brigades, teams, and hospitals are selected for the joint fire response actions. A plan for these actions is proposed to the selected emergency responders. The plan is a set of emergency responders with transportation routes for the mobile responders, required helping services, and schedules for the responders' activities. The plan is displayed on Internet accessible devices of the hospital administrators and the leaders of the fire brigades and emergency teams. These persons are organized in a Web-based community to exchange information about their abilities, availabilities, surrounding conditions, etc. with the purpose of the joint actions coordination.

Potential victims are evacuated from the fire place using the ridesharing technology. Potential victims here are people who have been out of danger so far or have got themselves out of the dangerous area. In the scenario it is proposed that persons who need to be evacuated set the location where they would like to be conveyed into an application installed in their mobile devices. The application finds drivers able to transport these persons. The found drivers receive an

appropriate signal. In the mobile devices of the drivers and persons the ridesharing routes are displayed.

Generally speaking, the destinations for the evacuated people do not matter. In actual usage evacuee can just run the appropriate application and it will search for bypassing cars.

It is supposed that the scenario takes place in a smart space. The main requirement to fulfill the scenario is Internet accessibility for the persons involved in it. A smart framework has been developed for this scenario.

IV. SMART FRAMEWORK

Smart Framework is defined here as a framework that is intended to coordinate operations of various autonomous resources of a smart space in context aware way to assist people in attaining their objectives. Sensors, databases, applications and other kinds of components of the smart space including humans and organizations are regarded as resources. The framework is planned to conceptually show how smart spaces and Web-based communities can facilitate the coordination and effectiveness of emergency response operations. Technical problems like failed Internet connections, discharged devices, power off, etc. are not addressed in the framework. As well, reliability and security problems (unregistered services, information incompleteness, unauthorized access, etc.) are out of the research scope.

The framework (Figure 1) is supported by an application ontology that represents non-instantiated domain & problem solving knowledge of the emergency management domain [25]. This ontology is formalized by means of the formalism of object-oriented constraint networks. Problems represented in such a way can be processed as constraint satisfaction problem.

The application ontology specifies knowledge that can be needed in various emergency situations. Generally, in different situations different problems can arise independently on the situation type. In particular situation only a piece of knowledge relevant to this situation is needed. In this connection, whenever an emergency event occurs, knowledge and information relevant to the current emergency situation are extracted from the application ontology and integrated into an *abstract context*. This context reduces the volume of knowledge and, correspondingly, the complexity of the problem to be solved.

The task of relevant knowledge determination is treated as ontology slicing operation. The abstract context is an ontology-based model of the current emergency situation. As the two components make up the application ontology, the context specifies domain knowledge describing the current emergency situation and problems to be solved in this particular situation.

The domain constituent of the abstract context is instantiated by resources of the smart space. An *operational context* is then produced. The operational context embeds the specifications of the problems to be solved. The input parameters of these problems, which correspond to properties of the classes of the domain constituent, are instantiated. The operational context reflects any changes in information, so it is a near real-time picture of the current emergency situation. The operational context is the base for organization of a community that unites members whose aim is taking joint actions on emergency response.

In the framework, the resources of the smart space as well as people involved in the emergency in any way are represented by Web-services. Service profiles capture capabilities of the resources, organizations, and people and delivery constraints, i.e., the profiles describe the functional

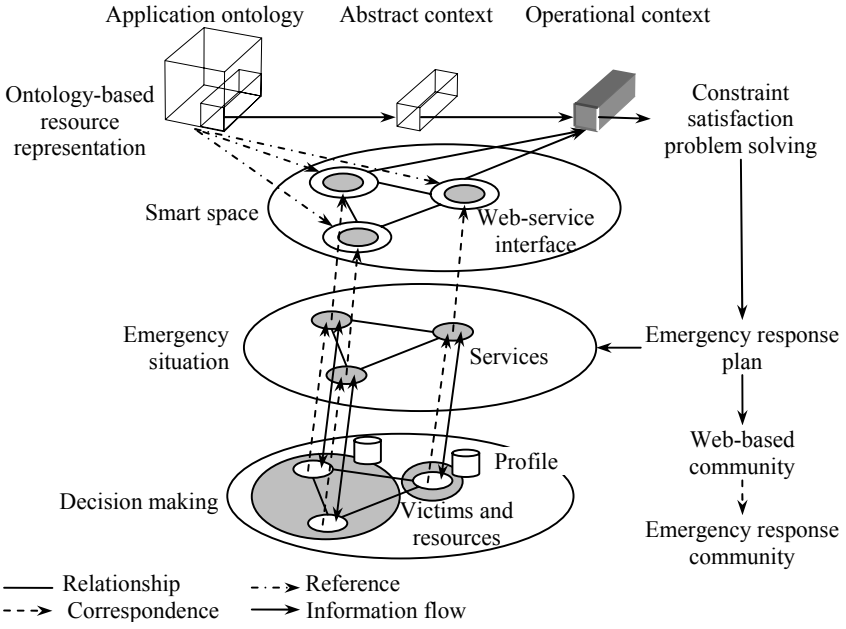


Figure 1. Generic scheme of smart framework

and non-functional service semantics [26]. The functional service semantics is described in terms of the input and output parameters of the service by means of WSDL. The WSDL service descriptions are complemented with SA-WSDL [27] annotations. The annotations enable to describe the non-functional service semantics, which is expressed with respect to service's cost model, availability, competence, and weight. The problem of compliance of service data models with the internal data model are resolved by wrappers. Due to the representation used the community purposed to emergency response actions comprises Web-services representing entities taking these actions.

The community is organized by specially developed emergency response services embedded in the smart space. Input data for the community organization are information characterizing the current emergency situation, particularly the situation type, and types of services relevant to the response actions. The types of services are represented in the abstract context. The current situation is represented by the operational context.

The emergency response services select possible community members and generate a set of feasible plans for actions. The set of plans is generated using the constraint satisfaction technology. A heuristic-based algorithm implements the plan generation [28]. Then, an efficient plan is selected from the set and submitted to the possible community members to their approval. This is the case of Web-based communications on the plan implementation. The members participating in such communications organize a Web-based community. If the plan is approved by all the members the emergency response community is considered to have been organized. Otherwise, another plan is taken up. The option of rejection is provided for due to the rapidly changing emergency situations – something may happen between the moment when a plan is selected and time when the possible community members receive this plan. The process of re-planning is an iterative process repeated till a plan suited all the members is found. The approved plan is thought to be the guide to joint actions for the members of the emergency response community.

As practice has shown, emergency response actions, besides actions on emergency control and first aid, have to foresee opportunities to evacuate potential victims from the dangerous areas. In the smart framework this purpose is achieved by applying functions that the ridesharing technology provides.

A. Service-Oriented Architecture

To coordinate interactions of the Web-services within the smart framework a service-oriented architecture is proposed. It comprises three groups of services (Figure 2).

The first group is made up of core services responsible for the registration of the Web-services in the service register and producing the real-world model of the emergency situation, i.e., creation of the abstract and operational contexts. Services belonging to this group are as follows:

- *registration service* registers the Web-services in the service register;

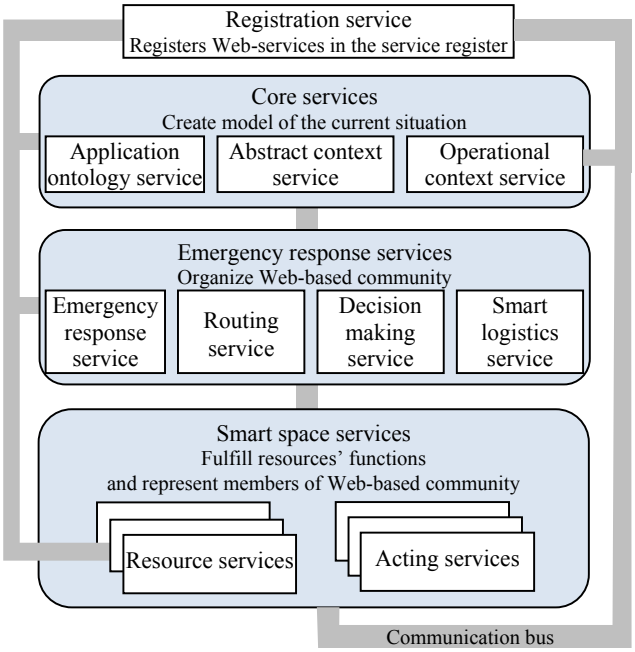


Figure 2. Service-oriented architecture

- *application ontology service* provides access to the application ontology;
- *abstract context service* creates, stores, maintains, and reuses the abstract contexts;
- *operational context service* produces operational contexts.

Web-services comprising the second group are responsible for the generation of alternative plans for actions and the selection of an efficient plan. This group contains:

- *emergency response service* integrates information provided by different resources about the number of injured people, and the location, intensity and severity of an emergency event;
- *routing service* generates a set of feasible plans for emergency response actions;
- *smart logistics service* implements the ridesharing technology;
- *decision making service* selects an efficient plan for actions and coordinates the (re)planning procedure.

The third group comprises sets of services responsible for the representation of the resources, organizations, and people and implementation of their functions. This group includes:

- *resource services* provide data stored in the resources' profiles and implement functions of the resources (smart ones as well);
- *acting services* provide data stored in the profiles of the emergency responders and victims; represent roles played by people or organizations; communicate on the plan implementation.

B. Organization of Web-based Community

We describe a Web-based community aimed at fire response actions.

The starting point for community organization is receiving by *emergency response service* of the signal that a fire event is taking place. Fire-prevention smart sensors had recognized some fire and sent this signal. Other kinds of smart information resources inform *emergency response service* of the number of injured people, and the location, intensity and severity of the fire.

Based on the information about the fire location, *emergency response service* requests the GeoInformation System (GIS) for a map of the fire area and the adjacent territory. The map contains some predetermined information as locations of the airports, buildings, roads, railway lines, water bodies, etc.

Using knowledge represented in the application ontology *abstract context service* determines what kinds of mobile teams and organizations providing response services are needed for the fire response actions and kinds of roles of the individuals involved in the fire situation. This service extracts knowledge related to the listed kinds of concepts from the application ontology and integrates it into an abstract context. In the case of fire, such kinds of teams are fire brigades and emergency teams; kinds of organizations are hospitals; kinds of roles are leader of a team, car driver, victim, etc. The referred kinds of concepts represent objects to be instantiated in the operational context.

Operational context service instantiates the abstract context and produces in that way an operational context. For the instantiation *operational context service* uses information provided by the following resources of the smart space:

- GPS-based devices installed on the vehicles of mobile emergency teams and fire brigades to fix the positions of these teams and brigades and to determine what types of vehicles they use;
- databases to find addresses and contact information of the fire departments, emergency services organizations, and hospitals;
- smart sensors to receive information which routes are available (e.g., where traffic jumps are, or some roads can be closed for traffic for some reasons);
- hospital administration systems to find out free capacities of the hospitals.

Operational context service passes the operational context to *routing service*. *Routing service* analyses types of routes (roads, airlines) that the emergency teams and fire brigades can follow depending on the vehicles they use. Based on the information about the number of injured people, the intensity and severity of the fire *routing service* calculates number of emergency teams and fire brigades needed to succeed in the response actions. The information about the number of injured people, the intensity and severity of the fire is received from *emergency response service*.

Then, *routing service* selects possible fire brigades, emergency teams, and hospitals that can be involved in the response operation and generates a set of feasible plans for actions. The actions are scheduled taking into account the availabilities of fire brigades, emergency teams, and hospitals; the types of vehicles that teams and brigades use; the routes available for these types; and the hospitals' free capacities. The problem of transportation routes planning incorporates the shortest-path problem.

Decision making service using a set of criteria selects an efficient plan from the set of feasible plans. The selected plan and the operational context are submitted to the leaders of the emergency teams, fire brigades that have been included in the plan, and to the hospitals' administrators. They have access to the operational context through any Internet-accessible devices (notebooks, PDAs, mobile phones, etc.). These persons organize a Web-based community to communicate on the plan implementation.

Persons who need to be evacuated invoke *smart logistics service* that is responsible for the evacuation. Clients of this service are supposed to be installed on the Internet-accessible devices of car drivers and other people involved in the fire situation. The persons enter the locations they would like to be conveyed. *Smart logistics service* determines the persons' locations and searches for cars going to or by the same or close destinations that the persons would like to be. It searches the cars among the vehicles passing the persons' locations. This service reads information about the destinations that the car drivers are going to from the navigators that the drivers use or from the drivers' profiles. The profiles store periodic routes of the drivers.

Based on the information about locations and destinations of the person and the found cars, *routing service* generates a set of feasible routes for person transportations. *Decision making service* determines efficient ridesharing routes. The criteria of the efficiency are minimum evacuation time and maximum evacuation capacity.

Smart logistics service sends appropriate signals to the drivers included in the ridesharing routes and displays on the drivers' devices the routes each driver is selected for. The points where the driver is expected to pick up the passenger(s) is indicated in the routes. The ways the passengers have to walk to these points are routed for them as well. Besides the routes, the passengers are informed of the model, color, and license plate number of the car intended for their transportation.

The view of the routes displayed on the devices of the individuals involved in the fire situation depends on the roles of these individuals.

C. Communications on Plan Implementation

The used model of decision making oversteps the limits of the three-phase model [29] towards communications of emergency responders on the implementation of the decision proposed by *decision making service*. The emergency responders communicate online using Internet-accessible devices and Web-based interface. Procedures of making decisions on plan implementation by professional emergency responders (emergency teams, fire brigades, hospitals) and by car drivers and evacuees are different.

The procedure of making decisions by the professional emergency responders is as follows (Figure 3). If the plan is approved by all the responders, this plan is supposed to be the plan for actions. Otherwise, either this plan is adjusted (so that the potential participant who refused to act

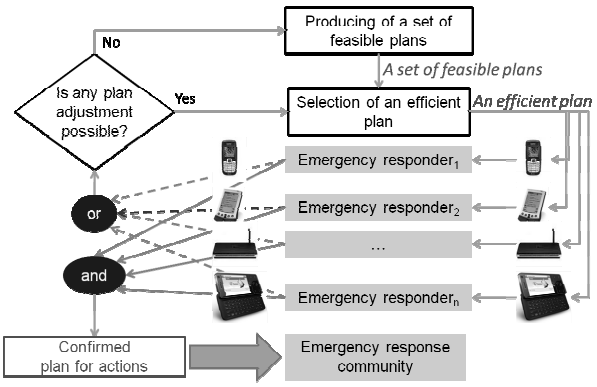


Figure 3. Decision making by professional emergency responders

according to the plan does not appear in the adjusted plan) or another set of plans is produced.

The plan adjustment is a redistribution of the actions among emergency responders that are contained in the set of feasible plans. If such a distribution does not lead to a considerable loss of time (particularly, the estimated time of the transportation of the injured people to hospitals does not exceed "The Golden Hour" [30]) then the adjusted plan is submitted to the renewed set of emergency responders for approval. If a distribution is not possible or leads to loss of response time a new set of plans is produced, from which a new efficient plan is selected and submitted to approval.

As soon as representatives of all the emergency teams, fire brigades, and hospitals have approved the plan they are in, *decision making service* sends them an appropriate signal that the joint actions can be started.

Figure 4 shows service interactions when all the emergency responders agree to participate in the joint actions according to the plan selected by *decision making service* (in the figure the emergency responders are

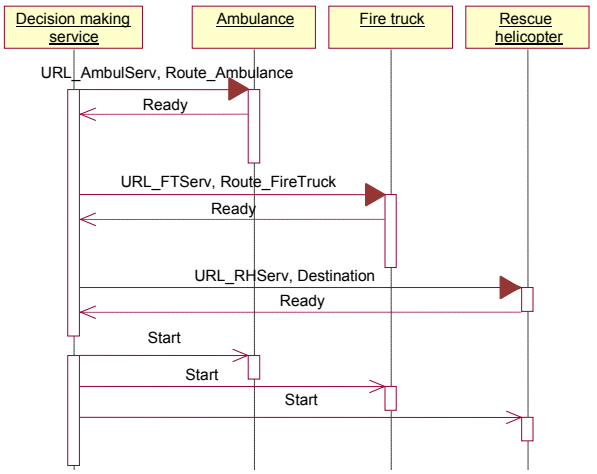


Figure 4. Emergency responders accept emergency response plan

represented by vehicles that they use – ambulance, fire truck, and rescue helicopter). We could see that *decision making service* sends simultaneous messages to all the emergency responders with the plan for each responder, waits their replays on plan acceptance (Ready), and sends them simultaneous messages to take the response actions (Start).

Figure 5 demonstrates service interactions in case when all the ambulances selected for the response actions are not ready to participate in them and *routing service* does not manage to adjust the selected plan. Two ambulances (Ambulance 1 and Ambulance 2) replay "Not ready" to the messages of *decision making service*. This replay is accompanied with the messages to *decision masking service* and *operational service* with the reasons of their refusals. Examples of such reasons are the road has been destroyed, the ambulance has blocked, etc.

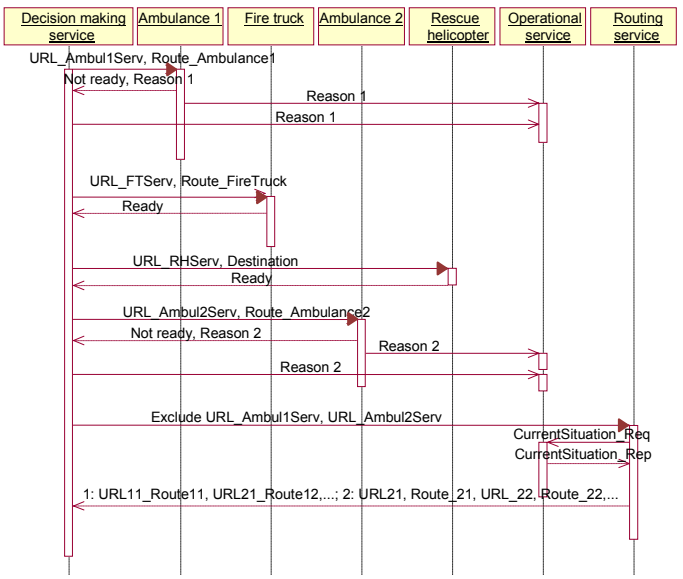


Figure 5. Plan regeneration

Decision making service duplicates the messages with the reasons for *operational service*. The duplication is a guarantee that *operational service* will receive information that it was unaware of up to this moment. As well *decision making service* sends the message on excluding the two ambulances from the list of available emergency responders to *routing service*.

Operational service corrects the operational context according to the information contained in the reasons. *Routing service* requests *operation service* of the operational context that represents the up-to-date information of the emergency situation, generates a new set of plans, and sends it to *decision making service*.

Decision making on an evacuation plan is in making agreement between the driver and the evacuee to go according to the scheduled ridesharing route (Figure 6). In case, when there is no agreement between a driver and an evacuee, another car for evacuation of this passenger is sought for. At that, the confirmed routes are not revised.

The emergency responders that are in the approved plan intended for professional emergency responders and the drivers participating in the evacuation organise the emergency response community.

V. SCENARIO USE CASE

The scenario (Section III) execution is demonstrated via organizing an emergency response community aimed at joint actions to response on a fire event happened in an urban area. The fire event was simulated using an internal platform that supports a GIS-based simulation. The platform is able to generate random failures and locations of professional emergency responders, random route availabilities, random flows of cars; it allows ones to input contextual information on types of emergency events, number of victims, etc.

In the scenario it is simulated that the fire has happened in a building, its level of severity is low, 9 injured people have to be transported to hospitals.

The application ontology used to create model of the fire situation had been created by experts via integration of parts of existing ontologies accessible through the Internet. To support the integration and necessary ontology modifications an ontology management tool – WebDESO [31] – was used. The application ontology has 7 taxonomy levels, contains more than 600 classes, 160 class attributes, and 120 relationships.

Figure 7 presents the abstract context created to model the fire situation. This context has 4 taxonomy levels, contains 17 bottom-level classes to be instantiated, 38 class attributes, and around 30 relationships of different types. Problem solving knowledge is hidden in the class "emergency response". This class specifies the following problems:

- select feasible hospitals, emergency teams, fire brigades, and car drivers;
- determine feasible transportation routes for ambulances, and fire engines depending on the transportation network and traffic situation;
- calculate the shortest routes for transportation of the emergency teams by ambulances, fire brigades by fire engines, and evacuees by cars;
- produce a set of feasible response plans for emergency teams, fire brigades, and hospitals;
- produce a set of feasible ridesharing routes.

In the simulated scenario 7 available fire brigades, 8 emergency teams, 5 hospitals having free capacities for 4, 4, 2, 3, and 3 patients are found in the territory adjacent to the fire place; 6 fire trucks and 1 fire helicopter are allocated to the fire brigades, 7 ambulances and 1 rescue helicopter are allocated to the emergency teams; 1 fire brigade is calculated to be required to extinguish the fire. The plan for actions designed for the emergency teams supposes that one vehicle can house one injured person.

A set of feasible plans for actions was generated for the criteria of minimal time and cost of transportation of all the victims to hospitals, and minimal number of mobile teams involved in the response actions. The set of feasible plans comprised 4 plans.

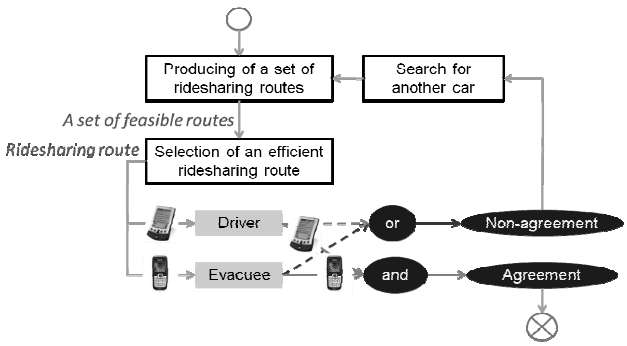


Figure 6. Decision making by car drivers and evacuees

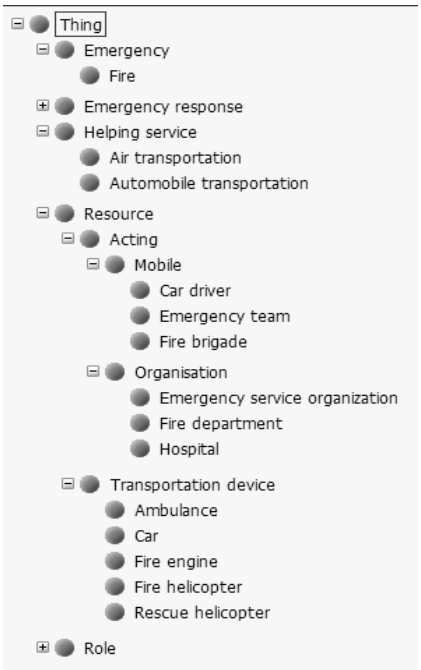


Figure 7. Fire situation: abstract context (a piece)

An efficient plan (Figure 8) was selected based on the key indicator of minimal time of victim transportations. In Figure 8 the big dot denotes the fire location; dotted lines depict routes to be used for transportations of the emergency teams and fire brigades selected for the response actions. The plan is approved by all the action participants. As it is seen from the figure, Web-based community comprises 1 fire brigade going by 1 fire helicopter, 7 emergency teams allocated to 1 rescue helicopter and 6 ambulances, and 3 hospitals having free capacities for 4, 2, and 3 patients. 1 ambulance (encircled in the figure) and the rescue helicopter go from the fire location to hospitals twice. The estimated time of the operation of transportations of all the victims to hospitals is 1 h. 25 min. Figure 9 shows part of the plan displayed on the smart phone of a member of an emergency team going by ambulance.

Results of evacuation of safe people using the ridesharing technology are as follows: 26 persons desire to be evacuated from the scene of fire; 22 persons have been driven directly to the destinations by 16 cars whereas for 4 persons no cars have been found. Examples of ways routed for a driver and a passenger are given in Figure 10 and Figure 11. The encircled car in the figures shows the location where the driver is offered to pick up the passenger. The persons that cannot be evacuated by passing cars are informed that they can be evacuated by taxi. If they agree, *smart logistics service* makes orders for taxi.

The Web-based community organised comprises 1) the professional emergency responders scheduled in the fire response plan (Figure 8) in the persons of the leaders of the emergency teams and fire brigades as well as the administrators of the hospitals, 2) the cars' drivers participated in the confirmation of the ridesharing routes, and 3) the evacuees. The emergency teams, fire brigades, hospitals, and car drivers constitute the emergency response community.

The Smart-M3 platform [32] has been used for the

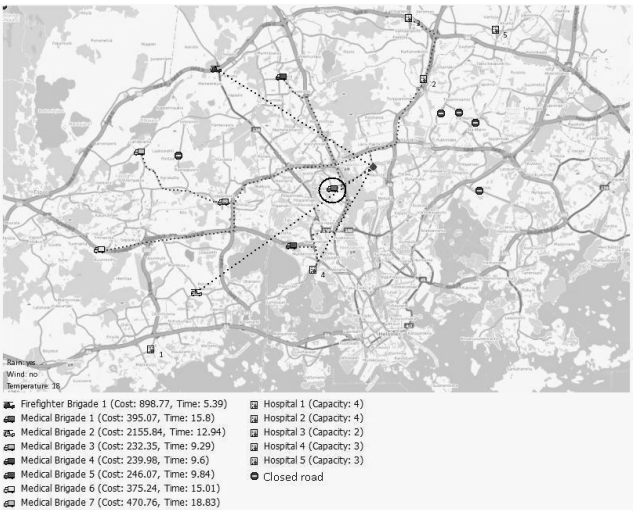


Figure 8. Plan for actions for fire brigades, emergency teams, and hospitals

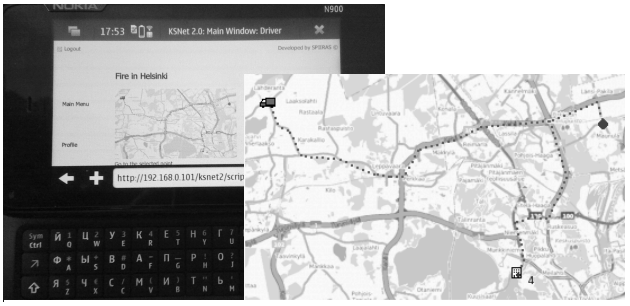


Figure 9. Plan for actions for an emergency team

scenario implementation. Tablet PC Nokia N810 (Maemo4 OS) and smart phone N900 (Maemo5 OS) play role of user devices. Personal PCs based on Pentium IV processors and running under Ubuntu 10.04 and Windows XP are used for hosting other services.

In the experiments with different datasets the execution time from the moment the emergency event was registered to the moment of producing the operational context took around 0.0007 s. The time taken to generate the sets of action plans for different datasets is shown in Table 1 and Figure 12. The approximating equation is quadratic for the total amount of objects involved in the response actions. The experimentation showed that the system already takes a reasonable time for result generation. Presented results are based on the usage of a research prototype running on a desktop PC. In a production environment the system is aimed to be run on dedicated servers and it is expected to be responsive enough to handle a large amount of objects. The future development of Smart-M3 up to the production level with a higher capacity could also contribute to the system performance.

VI. CONCLUSION

The problem of integration of the emerging technology-driven paradigms of smart spaces, Web-services, and Web-based communities for the fire response purposes was investigated. Most probably, judging from the literature, this is the first investigation on the integration of the mentioned technologies for emergency management aims.

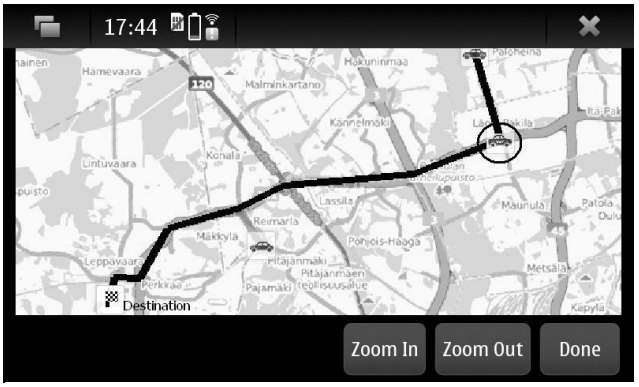


Figure 10. Ridesharing route: driver's view

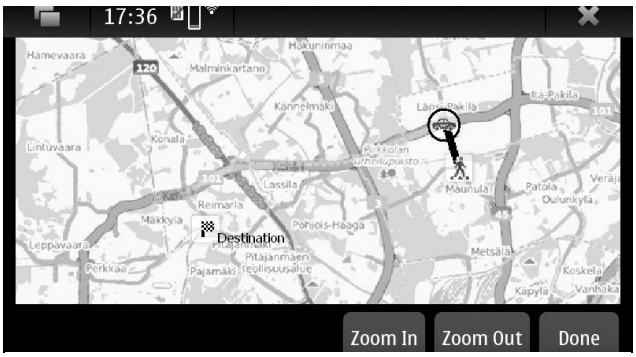


Figure 11. Ridesharing route: passenger's view

A smart framework that serves to integrate concepts of smart space, Web-services and Web-based communities has been proposed. This framework is developed to operate with Web-services representing the physical resources of a smart space and parties and individuals involved in a fire situation. The parties and individuals that are fire responders form a Web-based community. It is shown that they can communicate online independently on the devices they use, to exchange the operational information or make decisions on their readiness to participate in the joint response actions. In this direction, the present research exceeds the bounds of the three-phase Simon's model [29] towards actor communications on the decision implementation.

Due to the smart framework is built around the application ontology of the emergency management domain, this framework can be applied to organization of emergency management communities for response to different types of emergencies.

An original feature of the way the fire response actions are planned is in the involvement of ridesharing technology. Previously, the authors of this paper considered professional emergency responders to act on emergency response. In this paper, the community of professionals is extended with volunteers. Ridesharing serves as an example of the technology based on which volunteers can be involved in the emergency response actions.

To coordinate Web-service interactions within the smart framework the service-oriented architecture has been designed. The architecture contains a set of Web-services that is supposed to be sufficient to organize any fire response communities independently on types of operational units to be involved in response actions.

So far, the applicability of the smart framework was tested for response to traffic accidents and different kinds of

TABLE I. EXPERIMENT RESULTS

Number of emergency responders	Number of victims	Total number of objects	Time of plan generations, s.
10	10	20	4.85
10	20	30	9.12
20	20	40	17.51
30	30	60	37.93
40	40	80	66.13
50	50	100	101.29

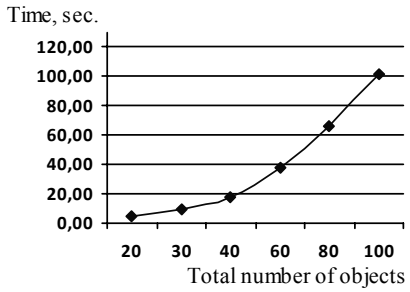


Figure 12. Dependence between number of objects involved in emergency and times of response plan generations

fire events (fire in a building, a port, a city area). This paper presents the scenario of planning fire response actions in an urban area. The scenario execution has shown that the paradigm of smart space provides efficient facilities to successful emergency response. Moreover, it can be concluded that ridesharing technology can be used for evacuation of potential victims from dangerous areas.

Some limitations of the developed framework are worth mentioning. The framework does not take into account cases when it is not found enough available acting resources or when some resources become disabled at time of the response actions. As well, the framework does not address the problem of lack of passing cars for evacuation of people from the fire area and the problem of searching for a route with changes if there are not any cars nearby the fire area going directly to the person destination. The listed limitations will be subjects for future research. Some more future research will address the problem of dynamic adaptability to following emergency events or to events concurrently happening in near-by locations.

ACKNOWLEDGMENTS

The present research was supported partly by projects funded by grants 10-07-00368, 11-07-00045, 11-07-00058, 12-07-00298 of the Russian Foundation for Basic Research, the project 213 of the research program "Information, control, and intelligent technologies & systems" of the Russian Academy of Sciences (RAS), the project 2.2 of the Nano- & Information Technologies Branch of RAS, and the contracts 14.740.11.0357 and 11.519.11.4025 of the Ministry of Education and Science of Russian Federation.

REFERENCES

[1] A. Smirnov, A. Kashevnik, T. Levashova, and N. Shilov, "Web-Based Community for Fire Response Actions: Scenario and Smart Framework," Proc. first Intl. Conf. on Advanced Collaborative Networks, Systems and Applications (COLLA 2011), IARIA, 2011, USB-flash drive.

[2] K. Luyten, F. Winters, K. Coninx, D. Naudts, and I. Moerman, "A situation-aware mobile system to support fire brigades in emergency situations," in CAMS 2006, the 2nd Intl. Workshop on context-aware mobile systems, 2006, pp. 1966–1975.

[3] P. Murphy, A. McGinness, and D. Guinan, Major incident review of toodyay fire December 2009, Tech. Rep., Australia, Manuka: Noetic Solutions Pty, 2010.

[4] L. Hauenstein, T. Gao, T. W. Sze, D. Crawford, A. Alm, and D. White "A cross-functional service-oriented architecture to support

real-time information exchange in emergency medical response," in IEEE Eng. Med. Biol. Soc., 2006, pp. 6478–6481 [EMBS '06, IEEE 28th Annual Intl. Conf.], doi: 10.1109/IEMBS.2006.260878.

[5] N. Owens, A. Armstrong, P. Sullivan, C. Mitchell, D. Newton, R. Brewster,and T. Trego, Traffic incident management handbook, U.S. Department of Transportation, Federal Highway Administration, Office of Transportation Operations, 2010.

[6] M. Turoff, M. Chumer, B. Van de Walle, and X. Yao, "The design of a dynamic emergency response management information system," in J. Inf. Tech. Theor. Appl., vol. 5, no. 4, 2004, pp. 1–36.

[7] D. Wyllie, "Technology isn't the (biggest) problem for information sharing in law enforcement," in PoliceOne, April 30, 2009 [online] available at: <http://www.policeone.com/police-products/-communications/articles/1816539-Technology-isn-t-the-biggest-problem-for-information-sharing-in-law-enforcement/> [accessed 25.06.2012].

[8] B. Moltchanov, C. Mannweiler, and J. Simoes, "Context awareness enabling new business models in smart spaces," in ruSMART/NEW2AN 2010, LNCS, vol. 6294, Springer, 2010, pp. 13–25.

[9] C. Schroth, "The Internet of services: global industrialization of information intensive services," in ICDIM'07 [2nd Intl. Conf. on Digital Information Management, Lyon, France, 2007], vol. 2, pp. 635–642.

[10] A. Smirnov, T. Levashova, N. Shilov, and A. Kashevnik, "Hybrid technology for self-organization of resources of pervasive environment for operational decision support," in Int. J. Artif. Intel. T., vol. 19, no. 2, World Sci. Publ. Co., 2010, pp. 211–229, doi: http://dx.doi.org/10.1142/S0218213010000121.

[11] C. Hagar, "Introduction to special section on crisis informatics," in Bulletin of the American Society for Information Science and Technology, vol. 36, no. 5, 2010, pp. 10–12.

[12] L. Palen, R. H. Starr, and S. Liu, "Online forums supporting grassroots participation in emergency preparedness and response," in Commun. ACM, vol. 50, no. 3, 2007, pp. 54–58.

[13] L. H. Mandel, C. R. McClure, J. Brobst, and E. C. Lanz, "Helping libraries prepare for the storm with Web portal technology," in Bulletin of the American Society for Information Science and Technology, 2010, vol. 36, no. 5, pp. 22–26.

[14] K. Starbird and L. Palen, "Voluntweeters:" self-organizing by digital volunteers in times of crisis," Proc. ACM CHI 2011 Conf. on Human Factors in Computing Systems, 2011, pp. 1071-1080.

[15] K. Starbird and L. Palen, "(How) will the revolution be retweeted?: Information propagation in the 2011 Egyptian uprising," Proc. 2012 ACM Conf. on Computer Supported Cooperative Work, 2012, to appear.

[16] S. Vieweg, A. Hughes, K. Starbird, and L. Palen, "Microblogging during two natural hazards events: What twitter may contribute to situational awareness," Proc. ACM 2010 Conf. on Computer Human Interaction, 2010, pp. 1079–1088.

[17] G. Armour, "Communities communicating with formal and informal systems: being more resilient in times of need," in Bulletin of the American Society for Information Science and Technology, 2010, vol. 36, no. 5, 2010, pp. 34–38.

[18] A. Krakovsky, "The role of social networks in crisis situations: public participation and information exchange," Proc. 7th Intl. ISCRAM Conf., 2010, pp. 52–57.

[19] V. Balasubramanian, D. Massaguer, S. Mehrotra, and N. Venkatasubramanian, "DrillSim: a simulation framework for emergency response drills," in AAMAS'06 [5th Intl. Conf. Autonomous Agents and Multiagent Systems, Japan, 2006], LNCS, vol. 3975, Springer, pp. 237–248.

[20] T. Catarci, M. Leoni, A. Marrella, and M. Mecella, "The WORKPAD project experience: improving the disaster response through process management and Geo collaboration," Proc. 7th Intl. ISCRAM Conf., [online] available at: <http://www.iscram.org/ISCRAM2010/-Papers/136-Catarci_etal.pdf> [accessed 11.02.2012].

[21] N. Bessis, E. Asimakopoulou, T. French, P. Norrington, and F. Xhafa. "The big picture, from grids and clouds to crowds: a data collective computational intelligence case proposal for managing disasters," in 3PGCIC 2010, [Intl. Conf. P2P, Parallel, Grid, Cloud and Internet Computing, Japan, 2010], IEEE Comput. Soc., 2010, pp. 351–356, doi: 10.1109/3PGCIC.2010.58.

[22] Z. Kang-kang, Y. Feng, Z. Wen-yu, and L. Pei-guang, "EDVO: a "one-station" emergency response service model based on ontology and virtual organization," Proc. 2008 IEEE Intl. Conf. Computer Science and Software Engineering, IEEE Comput. Soc., 2008, pp. 223–226, doi: 10.1109/CSSE.2008.1519.

[23] C.W.W. Ng and D.K.W. Chiu, "e-government integration with Web services and alerts: a case study on an emergency route advisory system in Hong Kong," Proc. 39th Hawaii Intl. Conf. System Sciences (HICSS'06), vol. 4, IEEE Comput. Soc., 2006, pp. 70.2–70.2, doi: 10.1109/HICSS.2006.135.

[24] A. Ling, X. Li, W. Fan, N. An, J. Zhan, L. Li, and Y. Sha, "Blue arrow: a Web-based spatially-enabled decision support system for emergency evacuation planning," Proc. 2009 IEEE Intl. Conf. Business Intelligence and Financial Engineering, IEEE Comput. Soc., 2009, pp. 575–578, doi: 10.1109/BIFE.2009.135.

[25] A. Smirnov, T. Levashova, A. Krizhanovsky, N. Shilov, and A. Kashevnik, "Self-organizing resource network for traffic accident response," in ISCRAM 2009, [6th Intl. Conf. Information Systems for Crisis Response and Management, Gothenburg, Sweden, 2009], J. Landgren and S Jul, Eds., 2009, URL: http://www.iscram.org/-ISCRAM2009/papers/Contributions/ 177_Self-Organizing%20-Resource%20Network%20for%20Traffic_Smirnov2009.pdf (access date: 26.06.2012).

[26] M. Klusch, "Semantic Web Service Description," in Intelligent Service Coordination in the Semantic Web, H. Helin and H. Schuldt, Eds., Birkhaeuser Verlag, Springer, 2008, pp. 41–67.

[27] "Semantic annotations for WSDL and XML schema", in W3C Recommendation, 2007, URL: http://www.w3.org/TR/sawsdl/ (access date: 26.06.2012).

[28] A. Smirnov and N. Shilov, "AI-based approaches to solving a dynamic logistics problem," in Künstliche Intelligenz, vol. 24, no. 2, Springer, 2010, pp. 143–147.

[29] H.A. Simon, "Making Management Decisions: The Role of Intuition and Emotion," in Academy of Management Executive, 1987, no. 1, pp. 57–64.

[30] E. B. Lerner and R. M. Moscati, "The Golden Hour: scientific fact or medical "Urban Legend?", in Acad. Emerg. Med., vol. 8, no. 7, 2001, pp. 758–760.

[31] A. Smirnov, M. Pashkin, N. Chilov, and T. Levashova, "KSNet-approach to knowledge fusion from distributed sources," in Comput. Inform., 2003, vol. 22, pp. 105–142.

[32] J. Honkola, H. Laine, R. Brown, and O. Tyrkko, "Smart-M3 information sharing platform," Proc. IEEE Symp. Computers and Communications, IEEE Comput. Soc., 2010, pp. 1041–1046, doi.ieeecomputersociety.org/10.1109/ISCC.2010.5546642.

Agent-based Versus Macroscopic Modeling of Competition and Business Processes in Economics and Finance

Aleksejus Kononovicius, Vygintas Gontis
Institute of Theoretical Physics and Astronomy,
Vilnius University
Vilnius, Lithuania
aleksejus.kononovicius@gmail.com,vygintas@gontis.eu

Valentas Daniunas
Institute of Lithuanian Scientific Society
Vilnius, Lithuania
mokslasplius@itpa.lt

***Abstract*—We present examples of agent-based and stochastic models of competition and business processes in economics and finance. We start from as simple as possible models, which have microscopic, agent-based, versions and macroscopic treatment in behavior. Microscopic and macroscopic versions of herding model proposed by Kirman and Bass new product diffusion are considered in this contribution as two basic ideas. Further we demonstrate that general herding behavior can be considered as a background of nonlinear stochastic model of financial fluctuations.**

***Keywords*-agent-based modeling; stochastic modeling; business models; financial market models.**

I. Introduction

Statistically reasonable models of social and economic systems, first of all stochastic and agent-based, are of great interest for a wide scientific community of interdisciplinary researchers dealing with diversity of complex systems [1], [2], [3], [4]. Computer modeling is one of the key aspects of modern science, be it physical or social or economic science, [5], [6]. In case of complex system modeling it serves as a technique in the quest for the understanding of the interrelation between microscopic interactions of individual agents and macroscopic, colective, dynamics of the whole complex system. Nevertheless, some general theories or methods that are well developed in the natural and physical sciences can be helpful in the development of consistent micro and macro modeling of complex systems [3], [4], [7], [8], [9].

As computer modeling is very prominent and important in modern science, we start this paper by discussing our online publishing and collaboration platform, see Section III. The open-source applets made available online on the website "Physics of Risk", see [10], allow to reproduce most of the results presented in this paper. This is very important as reproducibility of the results is one of the key demands in scientific society [5], [6].

From the Section IV we start discussing various models applicable in economics and finance, which highlight the important correspondence between microscopic, agent-based, and macroscopic, stochastic, modeling. In the opening Section IV, we present Kirman's agent-based model (see [11] for original paper) and derive its stochastic alternative, which was also done by Alfarano et al. in [12] using a more complex manner. In the Section V we show that modified, unidirectional, Kirman's agent based model can be seen as microscopic alternative to the widely known Bass diffusion model [13]. Further, in Section VI, we apply the stochastic treatment of the Kirman's model for financial markets and obtain stochastic model of absolute return similar to the CEV process [14] and earlier proposed model of $1/f$ noise [15], [16], [17]. In the Section VIII we show that Kirman's model possesses multifractal features, which are seen as an important feature of many natural phenomena [18]. Section IX closes presented discussion with some definitions and results regarding burst duration statistics generated by the class of nonlinear SDE and observable in the financial markets.

In the last section, Section X, we sum up everything discussed in this paper and share some ideas on future developments of the discussed research.

II. Review of the related works

Current on-going financial economic crisis provoked many papers calling for a revolution of economical thought and emphasizing a need for a wider applications of statistical physics in the research of social complexity [3], [8], [9], [19], [20], [21], [22], [23], [24], [25]. Most of them pointing out that agent based models are very important if one wants to effectively understand what is going on in the complex social and economic systems and the physical intuition might provide the important bridging between the macroscopic and microscopic modeling. These ideas somewhat traceback to the thoughts put down by Waldrop and Axelrod in the 1990s (see [4], [7]).

In the recent decades there were many attempts to create an agent-based model for the financial markets, yet no model so far is realistic enough and tractable to be considered as an ideal model [26]. One of the best examples of realistic models is so-called Lux and Marchesi model [27], which is heavily based on the behavioral economics ideas

mathematically put down as utility functions for the agents in the market, thus it is considered to be very reasonable and realistic [26]. Yet this model is too complex, namely it has many parameters and complex agent interaction mechanics, to be analytically tractable. Another example of a very complex agent-based model would be Bornholdt's spin model [28], [29], which is based on a certain interpretation of the well-known Ising model (for the details on the original model see any handbook on statistical physics (ex., [30])).

Some might argue that agent-based models need not to be analytically tractable and in fact that agent-based models are best suited to model phenomena, which is too complex to be analytically described [31]. But the recent developments show that many groups attempt to build a bridge between microscopic and macroscopic models. Possibly one of the earliest attempts to do so started from not so realistic, nor tractable "El Farol bar problem" [32]. This simple model quickly became known as the Minority Game [33] and over few years received analytic treatment [34]. Another prominent simple agent-based model was created by Kirman [11], which gained broader attention only very recently [12], [35], [36], [37]. In [38] we have given this model and extended analytical treatment and have shown that this model coincides with some prominent macroscopic, namely stochastic, models of the financial markets (see Section VII of this work for more details). Another interesting development was made by following the aforementioned Bornholdt's spin model, which has recently received an analytical treatment via mean-field formalism [39].

Our work in the modeling of complex social and economic systems has begun from the applications of nonlinear stochastic differential equations (abbr. SDE) seeking reproduce statistics of financial market data. The proposed class of equations has power law statistics evidently very similar to the ones observed in the empirical data. As all of this work (for broad review see [16]) was done by relying on the macroscopic phenomenological reasoning, we are now motivated to find the microscopic reasoning for the proposed equations. The development of the macroscopic treatments for the well established agent-based models appears to be the most consistent approach, as the movement in the opposite direction seems to be very complex and ambiguitious task. Thus we decided that we should select the simple agent-based models, which would have an expected macroscopic description. In this contribution we present a few examples of the agent-based modeling, based on the Kirman's model, in the business and finance while showing that the examples have useful and informative macroscopic treatments.

Kirman's ant colony model [11] is an agent-based model used to explain the importance of herding inside the ant colonies and economic systems (see the later works by Kirman (ex. [40]) and other authors, which develop on this idea, [12], [35]). The analogy can be drawn as human crowd behavior is ideologically and statistically similar in many senses. On our website, [10], we have presented interactive realizations of the original Kirman's agent-based model (see [41]), of its stochastic treatment by Alfarano et al. [12] (see [42]) and of its treatment in the financial market scenario done by our group [38] (see [43], [44]).

The diffusion of new products is one of the key problems in marketing research, and also one of the fields where we see that Kirman's model might be applied. The Bass diffusion model is a very prominent model related to this problem. This model is formulated as an ordinary differential equation, which might be used to forecast the number of adopters of the new successful product or service [13]. There were suggestions that such basic macroscopic description in marketing research can be studied using the agent-based modeling as well [45]. Thus it is a great opportunity to explore the correspondence between the micro and macro descriptions looking for the conditions under which both approaches converge. The Bass Diffusion model is of great interest for us as representing very practical and widely accepted area of business modeling. Web based interactive models, presented on the site [46] serve as an additional research instrument available for very wide community. On our website we also provide an interactive applet for the Bass diffusion treatment in terms of the modified Kirman's model [47] (for details on modification see Section V of this work).

Another interesting problem tackled in this work is related to the dynamics of the intermittent behavior. This kind of behavior is observed in many different complex systems ranging from the geology (ex., earthquakes [48]) and astronomy (ex., sunspots [49]) to the biology (ex., neuron activity [50]) and finance [51]. Great review of the universality of the bursty behavior is given by Karsai et al. [52] and Kleinberg [53]. In [52] the bursting behavior is considered as a point process with threshold mechanism. In this contribution we analyze the class of nonlinear SDE exhibiting power law statistics and bursting behavior, which was derived from the multiplicative point process [54], [55], [56] with applications for the modeling of trading activity in financial markets [57], [58]. This provides a very general, via hitting time formalism [14], [59], [60], approach to the modeling of bursty behavior of trading activity and absolute return in the financial markets [61].

III. Web platform

Our web site [10] was setup using WordPress webloging software [62]. The setup pays to be user-friendly, powerful and easily extensible web publishing platform, which with some effort can be adapted to the scientist's needs. There is a wide choice of plugins, which enable convenient usage of equations (mostly using LaTeX). While during the setup we found that bibliography management plugins were lacking at the time.

To accommodate our needs for equations we have worked on improving WP-Latex plugin (available from [63]). Namely we have introduced a possibility to write equations in both inline and ordinary math modes. Implemented equation labeling, numbering and referencing. And finally fixed some noticeable problems with vertical placement of the inline mode equations.

Another important task was to implement bibliography management and citations. For this cause we have used the bibtexParse PHP code (available from [64]) to setup BiBTeX backend. From this point on we have written our own original PHP code to link between bibtexParse, our database and WordPress. By using this plugin we can now easily manage and present our own papers (ex. generate our own bibliographies), papers we have read (tag them with keywords, write our own comments and etc.) and also communicate with the visitors using numerous citations.

Interactive models themselves are independent from the publishing framework. Most of them were implemented using the Java applet technology. Some of the applets were created using multi-paradigm simulation software AnyLogic [65], while the others were programmed from scratch using Java programing language [66]. AnyLogic was used in the most of agent-based scenarios as it is a very convenient tool for agent-based modeling, while programing from scratch gave us more control over the applets behavior needed while doing stochastic modeling.

Either way by compiling appropriate files one obtains Java applets, which can be included in to the articles written using WordPress. This way articles become interactive - visitor can both theoretically familiarize himself with the model and test if the claims made in the post describing model were true. This happens in the same browser window, thus the transition between theory and modeling appears to be seamless. Due to the fact that models are implemented as Java applets all of the numerical evaluation occurs on client machine, while the visitor must have Java Runtime Environment installed, and server load stays minimal. The requirement for JRE might appear to be cumbersome, but the technology is somewhat popular and freely available from Oracle Corp.

One of the goals of developing these models on the web site was to provide theoretical background of Bass Diffusion model and discuss practical steps on how such computer simulations can be created even with limited IT knowledge and further applied for varying purposes (see [46]). Thus, we have targeted small and medium enterprises to encourage them to use modern computer simulation tools for business planning, sale forecasting and other purposes.

Consequently computer models and their corresponding descriptions published at the [46] provide a relatively easy starting point to get acquainted with computer simulation in business. The published content enables site visitors to familiarize themselves with these models interactively, running the applets directly in a browser window, changing the parameter values and observing results. This significantly increases accessibility and dissemination of these simulations.

Our web site also offers another level of reproducibility by including source code files inside the Java applet files. In this way any willing user may use modern archiver software (ex., 7Zip) to obtain the source code. After doing so one can analyze source code and more deeply understand the presented models and their implementations. This is a very important level of reproducibility in the modern scientific context [5], [6].

IV. Extended macroscopic treatment of Kirman's model

There is an interesting phenomenon concerning behavior of ant colony. It appears that if there are two identical food sources nearby, or two identical paths to the same food source (the experiment done by Pasteels and Deneubourg [67], [68]), ants exploit only one of them at a given time. Evidently the food source which will be used at a given time is not certain. It is so as switches between food sources occur, though the food sources, or paths, remain the same.

One could assume that those different food sources are different trading strategies or, if putting it simply, the actions available to traders. Thus, one could argue that speculative bubbles and crashes in the financial markets are of similar nature as the exploitation of the food sources in ant colonies - as quality of stock and quality of food in the ideal case can be assumed to be constant. Thus, model [11] was created using ideas obtained from the ecological experiments [67], [68] can be applied towards the financial market modeling.

Kirman, as an economist, actually developed this model as a general framework in context of economic modeling (see [11], [40] and his other works). Recently his framework was also used by other authors who are concerned with the financial market modeling (see [12], [35]). Thus basing ourselves on the main ideas of these authors and our previous results in stochastic modeling (see [16]) we introduce specific modifications of Kirman's model providing a class of nonlinear stochastic differential equations [17] applicable for the financial variables.

Kirman's one step transition probabilities might be expressed in the following form [11],

$$p(X \to X+1) = (N-X)\,(\sigma_1 + hX)\,\Delta t, \qquad (1)$$

$$p(X \to X-1) = X\,(\sigma_2 + h[N-X])\,\Delta t, \qquad (2)$$

where X is a number of agents exploiting the chosen trading strategy (the one used to describe system state), while N is a total number of agents in the system (thus the other trading strategy is used by the $N-X$ agents). In the above the original Kirman's approach was extended by introducing fixed event time scale Δt by replacing the original models individual decision $\varepsilon_i \to \sigma_i \Delta t$ and herding $(1-\delta) \to h\Delta t$

parameters. Later we will need a more general assumption that parameters σ and h may depend on X and N, but for now we omit it.

Note that the transition probabilities (1) and (2) describe a scenario where the interaction among agent groups depends on the overall number of agents in alternative state. Such a choice makes the transition rates non-extensive, the connectivity between agent groups increases with the number of agents N. The herding interactions have a global character. Opposite scenario - extensive one will be also used further in this paper.

The lack of memory of the agents is the crucial assumption to formalize the population dynamics as a Markov process. Furthermore to describe the aforementioned dynamics in a continuous time we will need to obtain the transition rates, transition probabilities per unit time, which for continuous $x = X/N$ may be expressed as

$$\pi^+(x)=(1-x)\left(\frac{\sigma_1}{N}+hx\right), \quad (3)$$

$$\pi^-(x)=x\left(\frac{\sigma_2}{N}+h[1-x]\right). \quad (4)$$

Here the large number of agents N is assumed to ensure the continuity of variable x, which expresses the fraction of agents using the selected trading strategy, X. Relation between the discrete transition probabilities, (1) and (2), and continuous transition rates, (3) and (4), should be evident:

$$p(X \to X \pm 1) = N^2\pi^{\pm}(x)\Delta t. \quad (5)$$

One can compactly express the Master equation for the system state probability density function, $\omega(x,t)$, by using one step operators $\mathbf{E}$ and $\mathbf{E^{-1}}$ (see [69] for a details on this formalism) as

$$\begin{aligned}\partial_t\omega(x,t) =& N^2\left\{(\mathbf{E}-1)[\pi^-(x)\omega(x,t)]+\right.\\ &\left.+(\mathbf{E^{-1}}-1)[\pi^+(x)\omega(x,t)]\right\}. \quad (6)\end{aligned}$$

By expanding $\mathbf{E}$ and $\mathbf{E^{-1}}$ using the Taylor expansion (up to the second term) we arrive at the approximation of the Master equation

$$\begin{aligned}\partial_t\omega(x,t) =& -N\partial_x[\{\pi^+(x)-\pi^-(x)\}\omega(x,t)]+\\ &+\frac{1}{2}\partial_x^2[\{\pi^+(x)+\pi^-(x)\}\omega(x,t)]. \quad (7)\end{aligned}$$

By introducing custom functions

$$A(x)=N\{\pi^+(x)-\pi^-(x)\}=\sigma_1(1-x)-\sigma_2 x, \quad (8)$$

$$\begin{aligned}D(x)=&\pi^+(x)+\pi^-(x)=2hx(1-x)+\\ &+\frac{\sigma_1}{N}(1-x)+\frac{\sigma_2}{N}x, \quad (9)\end{aligned}$$

one can make sure that the (6) is actually a Fokker-Planck equation:

$$\partial_t\omega(x,t) = -\partial_x[A(x)\omega(x,t)]+\frac{1}{2}\partial_x^2[D(x)\omega(x,t)]. \quad (10)$$

Note that in the limit of large N one can neglect individual behavior terms in the $D(x)$. The above Fokker-Planck equation was first derived in a slightly different manner in the [12].

It is known (for details see [59]) that the Fokker-Planck equation can be rewritten as Langevin equation, or in other words stochastic differential equation,

$$\begin{aligned}\mathrm{d}x=&A(x)\mathrm{d}t+\sqrt{D(x)}\mathrm{d}W=\\ =&[\sigma_1(1-x)-\sigma_2 x]\mathrm{d}t+\sqrt{2hx(1-x)}\mathrm{d}W, \quad (11)\end{aligned}$$

here W stands for Wiener process. This step was also present in the [12].

In Fig. 1 we show that the statistical properties obtained from the agent-based model, defined by transition probabilities (1) and (2), match statistical properties of the solutions of (11). Thus the approximations done while deriving the Langevin equation for population fraction are valid. Interestingly enough we have obtained agreement with not so high number of agents - $N = 100$.

Note that the method used to derive Eq. (11) gives us an opportunity to consider parameters σ_1, σ_2, h dependent on the variable x and N. We will need this generalization in the further elaboration on various applications. From our point of view, the general form of SDE (11) derived from the very basic agent-based herding model provides a wide choice of opportunities in consistent micro and macro modeling of complex social systems.

V. Agent based model for the Bass Diffusion

The Bass Diffusion model is a tool to forecast the diffusion rate of new products or technologies [13]. Mathematically it is formulated as an ordinary differential equation

$$\partial_t X(t)=[N-X(t)]\left[\sigma+\frac{h}{N}X(t)\right], \quad (12)$$

$$X(0)=0. \quad (13)$$

where $X(t)$ denotes the number of consumers at time t, N can be seen as the market potential, being a starting number of the potential consumers (agents), σ is the coefficient of innovation, the likelihood of an individual to adopt the product due to influence by the commercials or similar external sources, h is the coefficient of imitation, a measure of likelihood that an individual will adopt the product due to influence by other people who already adopted the product. This nonlinear differential equation serves as a macroscopic description of new product adoption by customers widely used in business planning [45].

The agent-based approach to the same problem is related with modeling of product adoption by individual users, or agents. One can simulate diffusion process using computers, where individual decisions of adoption occur with specific adoption probability affected by the other individuals in the neighborhood. It is easy to show that Bass diffusion process

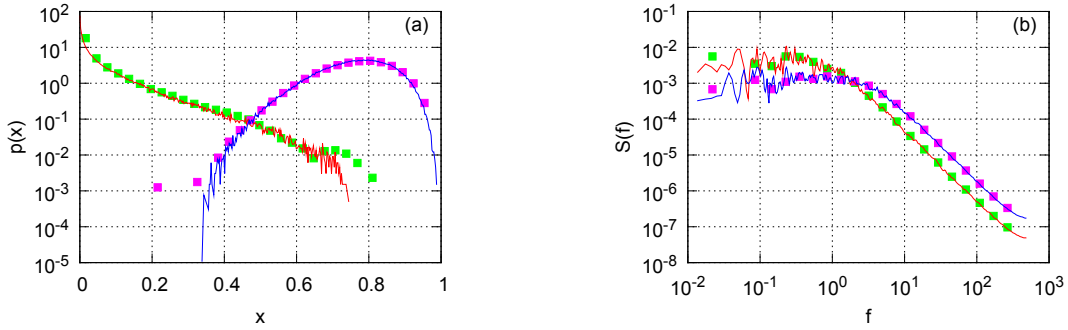


Figure 1. Agreement between statistical properties of population fraction, x, (a) probability density function and (b) power spectral density, obtained from stochastic (red and blue curves) and agent-based (green and magenta squares) models. Two qualitatively different model phases are shown: red curve and green squares correspond to herding dominant model phase ($\sigma_1 = \sigma_2 = 0.2$, $h = 5$), while blue curve and magenta squares correspond to individual behavior dominant model phase ($\sigma_1 = \sigma_2 = 16$, $h = 5$). Agent based model results obtained with $N = 100$.

is a unidirectional case of the Kirman's herding model [11]. Indeed, let us define $x(t)$ in the same way as in previous section $x(t) = X(t)/N$, then the potential users will adopt the product at the same rate as in Kirman's model agents switch from one state to another

$$\pi^+(x) = (1 - x)\left(\frac{\sigma}{N} + \frac{h}{N}x\right). \quad (14)$$

$$\pi^-(x) = 0. \quad (15)$$

The form of (15) should be self explanatory - in case of the product diffusion agent should not be allow to withdraw from the consumer state, thus this transition probability should be forced to equal zero.

The mathematical form of (14) is not as evident, note that we have substituted h with $\frac{h}{N}$ (compare with the original model transition probability (3)), and needs further discussion. Mathematically this substitution can be backed by the need for the stochastic term to become negligible in the limit of large N. In the modeled market terms this substitution means an introduction of the interaction locality - namely it is an assumption that each individual communicates only with his local partners (epidemic case).

One can compare the expression of the transition probability, (14), with the adoption probabilities of the Linear and GLM models of Bass Diffusion discussed in [70]. The match in expressions is clear in the small time step limit, $\Delta t \to 0$.

In case of the transition rates (14) and (15) the macroscopic description functions, namely drift, $A(x)$, and diffusion, $D(x)$, become

$$A(x) = N\pi^+(x) = (1 - x)(\sigma + hx), \quad (16)$$

$$D(x) = \pi^+(x) = \frac{(1 - x)}{N}(\sigma + hx). \quad (17)$$

In the large market potential limit, $N \gg 1$, $D(x)$ becomes negligible and thus one can consider the obtained equation to be equivalent to the Bass Diffusion ordinary differential equation (12) instead of the stochastic differential equation. This serves as a proof that Bass Diffusion is an unidirectional epidemic case of Kirman's herding model. Though this simple relation looks straightforward, we derive it and confirm by numerical simulations in fairly original way.

In Figure 2 we demonstrate the correspondence between the Bass Diffusion model (macroscopic description) and unidirectional Kirman's herding model (microscopic description). Both, agent-based and continuous, descriptions of the product adoption, ΔX, converge while the market potential, N, or the selected observation time interval, τ, become larger.

VI. Nonlinear Stochastic Differential Equation as a Model of the Financial Markets

Earlier we have introduced a class of non-linear SDEs providing time series with power-law statistics, and most notably reproducing $1/f^\beta$ spectral density, [54], [55], [56]. The general form of the proposed class of Ito SDEs is

$$\mathrm{d}y = \left(\eta - \frac{\lambda}{2}\right) y^{2\eta-1}\mathrm{d}t_s + y^\eta \mathrm{d}W_s, \quad (18)$$

here y is the stochastic process exhibiting power-law statistics, η is the power-law exponent of the multiplicative noise, while λ defines the exponent of power-law probability density function (PDF), and W is a Wiener process (the Brownian motion). Note that SDE (18) is defined in the scaled time, $t_s = \sigma_t^2 t$, where σ_t^2 is the scaling parameter. Empirically we have determined that $\sigma_t^2 = 1/6 \cdot 10^{-5}\mathrm{s}^{-1}$ is appropriate in terms of the return model proposed in [71].

From the SDE (18) follows that the stationary probability density function (PDF) of this stochastic process is power-law, $p_0(y) \sim y^{-\lambda}$, with the exponent λ [59]. While in Refs. [72] and later more precisely in [17] it was shown that the time series obtained while solving SDE (18) have power-law

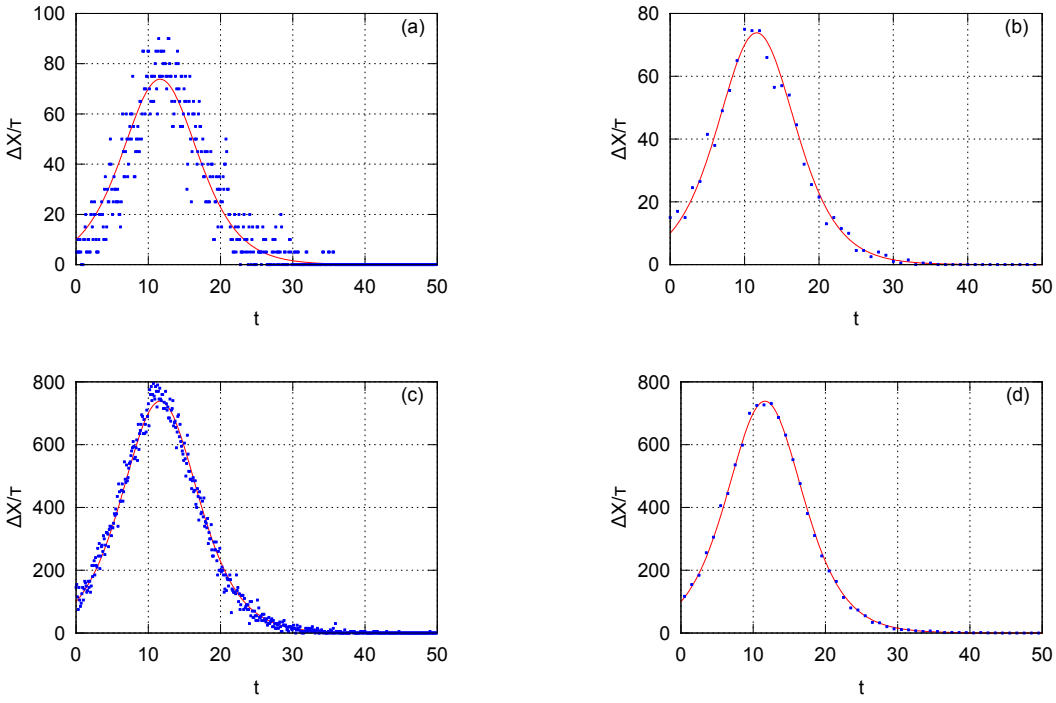


Figure 2. Comparison of the product adoption per observation interval, $\Delta X/\tau$ versus t, obtained from the macroscopic description by the Bass Diffusion model, (12), (red line) and the microscopic description using the unidirectional Kirman's model, (14), (blue points). The models tend to converge when time window, τ, or market potential, N, become larger: (a) $N = 1000$, $\tau = 0.1$; (b) $N = 1000$, $\tau = 1$; (c) $N = 10000$, $\tau = 0.1$; (d) $N = 10000$, $\tau = 1$. Other model parameters were the same for all subfigures and were as follows $\sigma = 0.01$, $h = 0.275$.

spectral density

$$S(f) \sim \frac{1}{f^{\beta}}, \qquad \beta = 1 + \frac{\lambda - 3}{2(\eta - 1)}. \tag{19}$$

Note that exponent of spectral density, β, is defined only for $\eta \neq 1$. In case of $\eta = 1$ the SDE (18) becomes identical to the geometric Brownian motion.

Power law statistics of the signal y obtained by solving SDE (18) and exponents λ, β are defined for large y values. Thus one has to introduce the diffusion restriction terms in the limit of small y values when attempting to solve SDE (18) or applying it in a stochastic modeling. There is a wide choice of restriction mechanisms adjustable to the needs of real systems with negligible influence on the power law exponents. We have introduced a term of additive noise while attempting to model the absolute return [71]

$$\mathrm{d}y = \left(\eta - \frac{\lambda}{2}\right)(1 + y^2)^{\eta - 1} y \mathrm{d}t_s + \\ + (1 + y^2)^{\frac{\eta}{2}} \mathrm{d}W_s. \tag{20}$$

In such case the stationary probability density function of the SDE (18) is a q-Gaussian (see [16], [71])

$$P_{\lambda}(y) = \frac{\Gamma(\lambda/2)}{\sqrt{\pi}\Gamma(\lambda/2 - 1/2)} \left(\frac{1}{1 + y^2}\right)^{\frac{\lambda}{2}}. \tag{21}$$

While modeling the trading activity [58] we have used the exponential diffusion restriction for small values of variable $y \simeq y_{\min}$

$$\mathrm{d}y = \left[\eta - \frac{1}{2}\lambda + \frac{m}{2}\left(\frac{y_{\min}^m}{y^m}\right)\right] y^{2\eta - 1}\mathrm{d}t + \\ + y^{\eta}\mathrm{d}W. \tag{22}$$

Equation (22) has a very general form, which includes the well known models applicable to financial markets such as the *Cox-Ingersoll-Ross* (CIR) process or the *Constant Elasticity of Variance* (CEV) process [14]

$$\mathrm{d}y = \mu y \mathrm{d}t + y^{\eta}\mathrm{d}W, \tag{23}$$

where $\mu = (\eta - 1) y_{\min}^{2(\eta - 1)}$, as a less general cases of the SDE (22).

The class of equations based on SDE (18) gives only a general idea how to model power-law statistics of trading activity and return in the financial markets. The problem is to determine the parameter set λ and η in a way giving the empirical values for the λ and β. The task becomes even more complicated if one considers the more sophisticated behavior of the spectral density - power spectral densities have not one, but two power-law regions with different values of β. In the series of papers [71], [72], [58], [57] we have shown that trading activity and return can be modeled

by a more sophisticated version of the SDE than (18) now including the two powers of the noise multiplicativity. In the case of return instead of Eq. (20) one should use

$$\mathrm{d}y=\left(\eta-\frac{\lambda}{2}\right)\frac{(1+y^2)^{\eta-1}}{(\epsilon\sqrt{1+y^2}+1)^2}y\mathrm{d}t_s+ \\ +\frac{(1+y^2)^{\frac{\eta}{2}}}{\epsilon\sqrt{1+y^2}+1}\mathrm{d}W_s, \quad (24)$$

here ϵ divides the area of diffusion into the two different noise multiplicativity regions to ensure the spectral density of $|y|$ with two power law exponents.

The proposed form of the SDE enables reproduction of the main statistical properties of the return observed in the financial markets. Similarly one can deal with a more sophisticated model for the trading activity [58]. This provides an approach to the financial markets with behavior dependent on the level of activity and exhibiting two stages: calm and excited. Equation (24) models the stochastic return y with two power-law statistics, namely the probability density function and the power spectral density, reproducing the empirical power law exponents of the return in the financial markets.

VII. Kirman's model as a microscopic approach to the financial markets

The drawback of the stochastic models is a lack of direct insights into the microscopic nature of replicated dynamics. Bridging between microscopic and macroscopic approaches is needed for better grounding of stochastic modeling.

Top-down approach, namely starting from the stochastic modeling and moving towards the agent-based models, seems to be a very formidable task, as the macro-behavior of complex system can not be understood as a simple superposition of varying micro-behaviors. While in the case of sophisticated agent-based models [26] bottom-up approach provides too many opportunities. But there is selection of rather simple agent-based models (ex. [11]), whose stochastic treatment can be directly obtained from the microscopic description [12].

Here we consider an opportunity to generalize Kirman's ant colony model [11] with the intention to modify its microscopic approach to the financial market modeling [12] reproducing the main stylized facts of this complex system. In the Section IV we have already introduced Kirman's ant colony model, proposed its generalization and derived stochastic model for the two state population dynamics.

As Kirman's model considers the two available agent states one must define two types of agents acting inside the market in order to relate Kirman's model to financial markets. Currently, the most common choice is assuming that agents can be either fundamentalists or noise traders [26].

Fundamentalists are assumed to have the fundamental knowledge about the market, which is assumed to be quantified by the so-called fundamental price, $P_f(t)$, of the traded stock. By having this knowledge they can make long term forecasts on a notion that infinitely long under-evaluation or over-evaluation of the stock is impossible - the market in some point in the future will have to set a fair price on the stock. Thus their excess demand, which is shaped by their long term expectations, is given by [12]

$$D_f(t)=N_f(t)\ln\frac{P_f(t)}{P(t)}, \quad (25)$$

here $N_f(t)$ is a number of fundamentalists inside the market and $P(t)$ is a current market price. As long term investors fundamentalists assume that $P(t)$ will converge towards $P_f(t)$ at least in a long run. Therefore if $P_f(t) > P(t)$, fundamentalists will expect that $P(t)$ will grow in future and consequently they will buy the stock ($D_f(t) > 0$). In the opposite case, $P_f(t) < P(t)$, they will expect decrease of $P(t)$ and for this reason they will sell the stock ($D_f(t) < 0$).

The other group, the noise traders are investors who attempt estimate the stocks future price based on its recent movements. As there is a wide selection of technical trading strategies, which are used to analyze stocks price movements, one can simply assume that the average noise traders demand is based on their mood, $\xi(t)$, [12]

$$D_c(t)=r_0N_c(t)\xi(t), \quad (26)$$

here $D_c(t)$ is a total excess demand of noise trader group, $N_c(t)$ is a number of noise traders inside the market and r_0 can be seen as a relative noise trader impact factor.

Price and, later after a brief derivation, return can be introduced into the model by applying the Walrassian scenario. One can assume that trading in the market is cleared instantaneously to set a price, which would stabilize the market demand for a given moment. Thus the sum of all groups' excess demands should equal zero:

$$D_f(t)+D_c(t)=N_f(t)\ln\frac{P_f(t)}{P(t)}+r_0N_c(t)\xi(t)= \\ =0, \quad (27)$$

$$P(t)=P_f(t)\exp\left[r_0\frac{N_c(t)}{N_f(t)}\xi(t)\right], \quad (28)$$

where without loosing generality one can assume that fundamental price remains constant, $P_f(t)=P_f$.

Consequently the return, which is defined as logarithmic change of price, in the selected time window T is given by:

$$r(t)=\ln P(t)-\ln P(t-T)= \\ =r_0\left[\frac{x(t)}{1-x(t)}\xi(t)-\frac{x(t-T)}{1-x(t-T)}\xi(t-T)\right], \quad (29)$$

where we have set that $\frac{N_c(t)}{N}=x$ and $\frac{N_f(t)}{N}=1-x$ according to the notation introduced in Section IV. Alfarano

et al. [12] simplified the above by assuming that $x(t)$ is significantly slower process than $\xi(t)$, obtaining adiabatic approximation of the return

$$r(t) = r_0 \frac{x(t)}{1-x(t)} \zeta(t), \tag{30}$$

where $\zeta(t) = \xi(t) - \xi(t-T)$. If $\zeta(t)$ is modeled using spin-noise model, as in [12], then the middle term, $\frac{x(t)}{1-x(t)}$, can be seen as an absolute return.

Using Ito formula for variable substitution [59] in SDE (11) we obtain nonlinear SDE for the $y(t) = \frac{x(t)}{1-x(t)}$

$$\begin{aligned} \mathrm{d}y = & (\sigma_1 - y[\sigma_2 - 2h])(1+y)\mathrm{d}t + \\ & + \sqrt{2hy}(1+y)\mathrm{d}W. \end{aligned} \tag{31}$$

Agreement between the agent-based Kirman's model applied towards financial markets using the ideas discussed above and the new stochastic model for y, (31), is demonstrated in Fig. 3.

Note once again that the actual derivation, and thus, the final outcome, does not change even if σ_1, σ_2 or h are the functions of either x or y. Therefore, one can further study the possibilities of the obtained stochastic model, (31), by checking different scenarios of σ_1, σ_2 or h being functions of either x or y. Nevertheless, the most natural way is to introduce a custom function $\tau(y)$ to adjust the inter-event time according to the system state. From the financial market point of view this can be seen as introduction of variability of trading activity based on the return.

We have chosen the case when h and σ_2 are functions of y, namely we make the substitutions, $\sigma_2 \to \frac{\sigma_2}{\tau(y)}$ and $h \to \frac{h}{\tau(y)}$, in the Kirman's model transition probabilities, (1) and (2), and stochastic model for y, (31). To further simplify the model we can introduce scaled time, $t_s = ht$, and make related model parameter transformations, $\varepsilon_i = \frac{\sigma_i}{h}$. By making these substitutions we arrive at

$$\begin{aligned} \mathrm{d}y = & \left[\varepsilon_1 + y\frac{2-\varepsilon_2}{\tau(y)}\right](1+y)\mathrm{d}t_s + \\ & + \sqrt{\frac{2y}{\tau(y)}}(1+y)\mathrm{d}W_s, \end{aligned} \tag{32}$$

where W_s is appropriately scaled Wiener process. Note that we left σ_1, and consequently ε_1, independent of y on purpose as one could argue that individual behavior of fundamentalist trader should not depend on the observed returns as he is a long term investor uninterested in the momentary fluctuations of the market mood.

Note that absolute return, y, defined in Eqs. (31) and (32), serve as a measure of volatility in the financial markets. It is known that volatility has long-range memory and correlates with trading activity and has probability density function with power law tail [51]. We are particularly interested in the case of $\tau(y) = y^{-\alpha}$. This selection is defined by the fact that trading activity has positive correlation with volatility and the class of SDE (18) is invariant regarding power-law variable transformation, see [56]. In such case the obtained stochastic differential equation, Eq. (32), in the limit of $y \gg 1$ is very similar to the stochastic models discussed in the Section VI.

In the aforementioned limit of y, $y \gg 1$, we can consider only the highest powers in Eq. (32). In such case Eq. (32) is reduce to the

$$\mathrm{d}y = (2-\varepsilon_2) y^{2+\alpha}\mathrm{d}t_s + \sqrt{2y^{3+\alpha}}\mathrm{d}W_s. \tag{33}$$

The direct comparison of Eqs. (18) and (33) yields:

$$\eta = \frac{3+\alpha}{2}, \quad \lambda = \varepsilon_2 + \alpha + 1. \tag{34}$$

Consequently we expect that the stochastic process y defined by Eq. (33) will have the power law stationary probability density function,

$$p(y) \sim y^{-\varepsilon_2 - \alpha - 1}, \tag{35}$$

and also a power law spectral density,

$$S(f) \sim \frac{1}{f^{\beta}}, \quad \beta = 1 + \frac{\varepsilon_2 + \alpha - 2}{1+\alpha}, \tag{36}$$

where we have used the relation between model parameters, Eq. (34).

While if we linearize drift function of Eq. (31) with the respect to the absolute return, y, namely set $\varepsilon_2 = 2$, we would obtain a stochastic differential equation (once again in the limit $y \gg 1$)

$$\mathrm{d}y = \varepsilon_1 y \mathrm{d}t_s + \sqrt{2y^{3+\alpha}}\mathrm{d}W_s. \tag{37}$$

similar to the generalized CEV process [14], [73], which was considered in [73],

$$\mathrm{d}y = ay\mathrm{d}t + by^{\eta}\mathrm{d}W. \tag{38}$$

In [56] the latter was noted to be a special case of Eq. (22) with exponential restriction of diffusion applied. The comparison with this special case is important on its own as this equation generalizes some stochastic models used in risk management. Theoretical prediction of PDF and spectral density for y defined by Eq. (37), is given by [73]

$$p(y) \sim y^{-3-\alpha}, \tag{39}$$

$$S(f) \sim \tfrac{1}{f^{\beta}}, \quad \beta = 1 + \tfrac{\alpha}{1+\alpha}, \tag{40}$$

where we have used the previously obtained relation between model parameters, Eq. (34).

In the Figure 4 we show that the theoretical predictions discussed in this section are valid and that they enable the reproduction of different spectral densities and probability density functions.

Note that while the stochastic model based on herding behavior of agents appears to be too crude to reproduce statistical properties of financial markets in such details as the stochastic model driven by the Eq. (24), which is

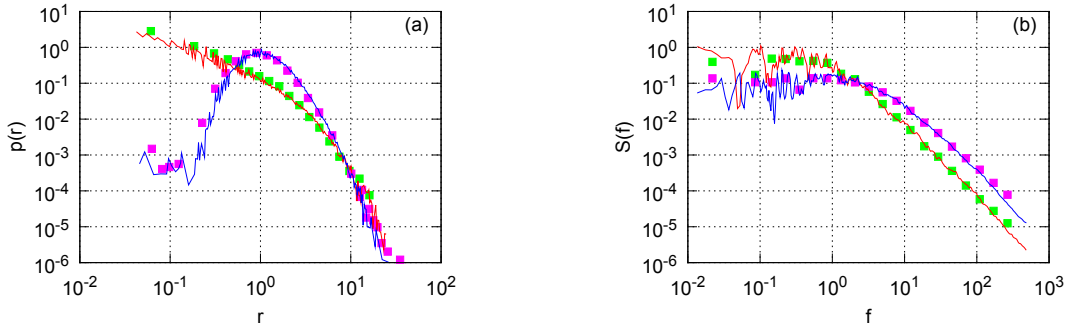


Figure 3. Agreement between statistical properties of y, (a) probability density function and (b) power spectral density, obtained from the stochastic (red and blue curves) and agent-based (green and magenta squares) models. Two qualitatively different model phases are shown: red curve and green squares correspond to herding dominant model phase ($\sigma_1 = \sigma_2 = 0.2$, $h = 5$), while blue curve and magenta squares correspond to individual behavior dominant model phase ($\sigma_1 = \sigma_2 = 16$, $h = 5$). Agent based model results obtained with $N = 100$.

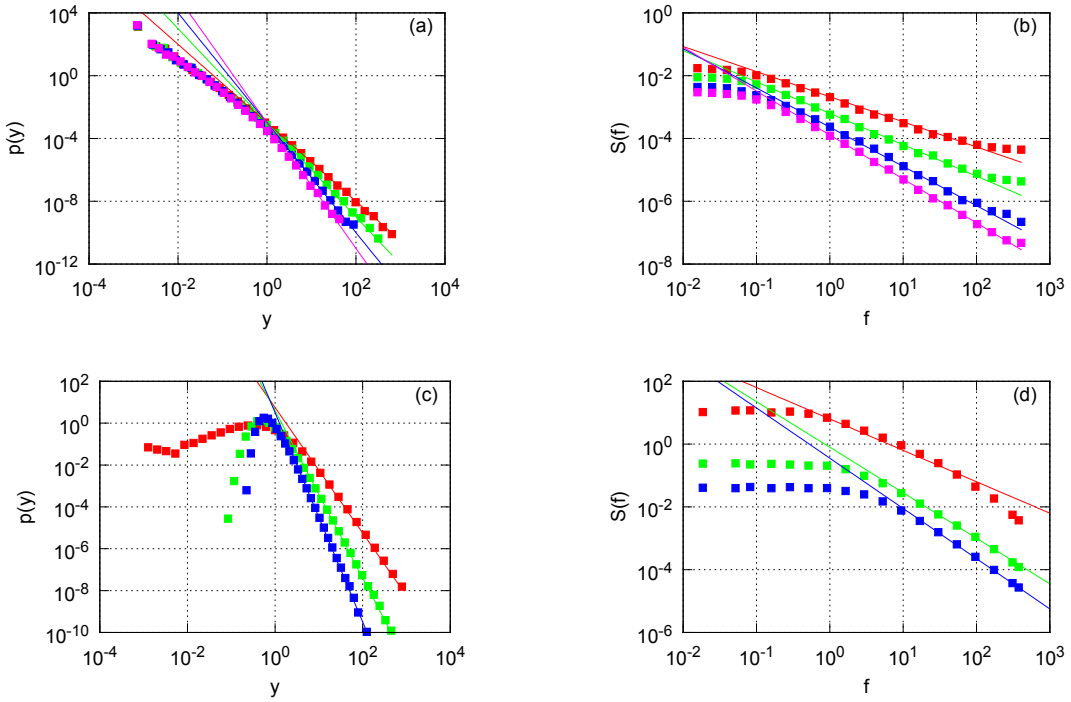


Figure 4. Statistical properties, (a) and (c) - probability density function, (b) and (d) - spectral density, of the time series obtained by solving Eq. (32) (colored squares). Fits are provided by the theoretical predictions made in this section, (a) and (b) are fitted by using (35) and (36), (c) and (d) are fitted by using (39) and (40), (curves of corresponding colors). Model parameters for the (a) and (b) were set as follows: $\alpha = 1$, $\varepsilon_1 = 0$, $\varepsilon_2 = 0.5$ (red squares), 1 (green squares), 1.5 (blue squares) and 2 (magenta squares). Model parameters for the (c) and (d) were set as follows: $\varepsilon_1 = \varepsilon_2 = 2$, $\alpha = 0$ (red squares), 1 (green squares) and 2 (blue squares).

heavily based on the empirical research, it contains very important long range power law statistics of the absolute return. Obtained equations are very similar to some general stochastic models of the financial markets [17], [73] and thus, in future development might be able to serve as a microscopic justification for them and maybe for the more sophisticated model driven by the Eq. (24).

It is possible to extend agent-based model by introducing additional agent groups or splitting old ones. Let us assume that chartist agents may disagree in their expectations and thus divide into pessimists and optimists. Therefore it is natural to introduce three agent groups (see Fig. 5) interacting among themselves. Our first attempts in this direction proves that in case of the three agent groups (as shown in Fig. 5), when the herding parameter $h_{cc} \gg h_{cf}$, might confirm the expectation of a more complex behavior exhibiting fractured

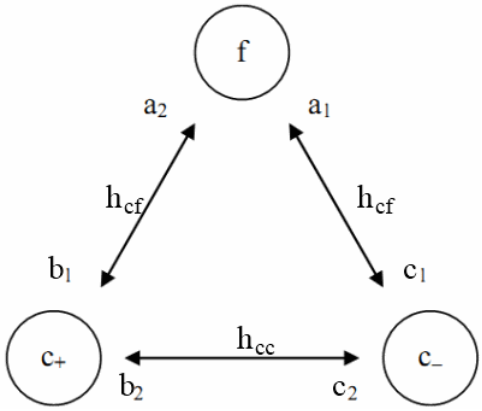


Figure 5. The general case of the three groups of interacting agents: f - fundamentalists, c_+ - chartists optimists, c_- - chartists pessimists. h_{ij} are herding terms, while a_i, b_i and c_i stand for individual transitions in the direction of the arrow.

power spectral density of absolute return. More detailed study of such approach in comparison with macroscopic modeling by SDE (24) is ongoing.

VIII. Multifractal behavior of return series

In the last few decades it was noted that many natural phenomena have very complex intrinsic structure, which has a very specific scaling properties. This notion was generalized as fractal framework [74]. Later it was also noted that the scaling properties of some processes exhibit even more complex scaling behavior - namely they appeared to have features of the multiple fractals. Few examples of such phenomena include geoelectrical processes [75], human heartbeat [76] and gait [77]. The financial market time series apparently are also of the multifractal nature [78], [79], [80].

There are few established methods to detect multifractal time series and two very prominent methods. One of them is generalized height-height correlation function method (GHHCF) and multifractal detrended analysis method (MF-DFA). In our previous approaches [81], [38] we have used the GHHCF method, so let us in this contribution to rely on the MF-DFA method.

To start with the multifractal analysis of the time series, y_k, we have to obtain the profile of the time series, Y_k:

$$Y_k = \sum_{j=1}^{k} (y_k - \langle y \rangle). \tag{41}$$

Next we have divide the Y_k series into equally sized and non overlapping segments. Thus if our segments are of the size s, then we will have $N_s = \mathrm{int}(N/s)$ segments (here N is the length the series, while $\mathrm{int}(\dots)$ is a function which takes an integer part of the argument). For the most of the segment sizes some of the data will be lost, in order to account for it one might want to take another set of segments, but now splitting from the end of the series.

Further, one has to determine the trends in the obtained segments. Generally this can be done using varying polynomial fits, but linear fits in the most cases are more than enough. After the trends, $\bar{Y}$, are known one has to evaluate how well the trend fits the actual series:

$$F_\nu^2(s) = \frac{1}{s} \sum_{i=1}^{s} \left[Y_{(\nu-1)s+i} - \bar{Y}_\nu(i) \right]^2, \tag{42}$$

$$F_\nu^2(s) = \frac{1}{s} \sum_{i=1}^{s} \left[y_{N-(\nu-N_s)s+i} - \bar{y}_\nu(i) \right]^2. \tag{43}$$

The Eq. (42) holds for segments $\nu = 1, \dots, N_s$, while the Eq. (43) should applied towards segments $\nu = N_s + 1, \dots, 2N_s$. Finally one has to average over all segments using

$$F_q(s) = \left\{ \frac{1}{2N_s} \sum_{\nu=1}^{2N_s} \left[F_\nu^2(s) \right]^{\frac{q}{2}} \right\}^{\frac{1}{q}}, \tag{44}$$

here q stands for generalized coefficient, which is the one enabling us to recover multifractal features it is also the only difference from the original detrended fluctuation analysis (DFA) method [18]. Note that in case of $q = 2$ the $F_q(s)$ is the same as the one in the original DFA method.

All that is left is to determine is the power law trend, $h(q)$, of the $F_q(s)$. These trends, $h(q)$, are also frequently named the generalized Hurst exponents. If the Hurst exponents are different for different q, which can be any real number, then the signal can be seen as multifractal. In the opposite case or if the variation is negligible, time series can be assumed as monofractal. For more details on the MF-DFA method see [18].

In Figure 6 we show that the stochastic differential equations obtained for the modeling of financial markets and derived from the Kirman's agent-based model have broad multifractal spectra. The curves capture a region of the Brownian motion, $h(q) \approx 1.5$, and a region of long range memory, $h(q) \approx 1$. Note that in case of $\alpha = 1$ and

$\alpha = 2$ (green and blue curves) $h(2) = 1$, which can be seen as a proof that the obtained time series posses the long term correlated (have so-called long range memory) behavior, while for $\alpha = 0$ interim behavior between the Brownian motion and long range memory is observed, $1 < h(2) < 1.5$.

IX. Statistics of bursts generated by nonlinear SDE

In the Section VII we have shown that the herding model of return in the financial markets leads to the class of stochastic differential equations, whose general form is given by SDE (18). This class of stochastic differential equations reproduces power law statistics, namely the probability density function and the spectral density, of return and trading activity in the financial markets. The burst statistics of the financial markets are also very important for the risk management and would serve as an additional criteria to determine the model consistency. In this section we provide some initial results of burst statistics generated by the SDE (18).

We define a burst as a part of the time series lying above the certain threshold, h_I. In Figure 7 we present an example burst of the simple bursty time series, $I(t)$. Evidently a burst as itself can be described by its duration, $T = t_2 - t_1$, maximum value, I_{max}, and burst size, which we define as an area above the selected threshold yet bellow time series curve (highlighted by x pattern in the Fig. 7), S.

There is a well established passage, or alternatively hitting, time framework, which is frequently used to tackle practical problems in both mathematical finance [14] and physics [59], [60]. One can also apply this framework to understand the burst durations, T. Interestingly enough we can consider the first hitting time of the stochastic process starting infinitesimally near the hitting threshold as the burst duration itself, T.

Brownian motion, geometric Brownian motion and Bessel process are highly applicable models (for examples of the application in the mathematical finance, see [14]) for which hitting times statistics are known. The Bessel process,

$$\mathrm{d}R = \frac{N-1}{2}\frac{\mathrm{d}t_s}{R} + \mathrm{d}W_s, \tag{45}$$

is one of the most interesting as some prominent mathematical finance models can be transformed to a similar form. In order to simplify further handling of the Bessel process it is convenient to introduce $\nu = \frac{N}{2} - 1$, which is known as the index of the Bessel process. While N is also frequently retained and mentioned as it bears an actual physical meaning - the Bessel process is an Euclidean norm, length of the vector, of N-dimensional Brownian motion, which starts at the origin. Note that for $N > 1$, or alternatively $\nu > -0.5$, R tends to diverge towards infinity.

In our case the Bessel process is of high interest as by using the Lamberti transform defined as

$$\ell : y \mapsto z(y) = \frac{1}{(\eta-1)y^{\eta-1}}, \tag{46}$$

we can reduce a general class of SDE (18) to the Bessel process,

$$\mathrm{d}z = \left(\nu + \frac{1}{2}\right)\frac{\mathrm{d}t_s}{z} + \mathrm{d}W_s, \tag{47}$$

with index $\nu = \frac{\lambda-2\eta+1}{2(\eta-1)}$. The corresponding dimension of the Brownian motion is given by $N = 2(\nu+1) = \frac{\lambda-1}{\eta-1}$.

Let us assume that a burst starts at time t_0, with $y_0 = y(t_0)$ slightly exceeding the selected threshold, h_y. By definition the burst lasts until $y(t)$ crosses h_y once again, but now from the above. Equivalently, in the terms of Bessel process the burst lasts until at a certain time, t, the z process crosses the boundary $h_z = \ell(h_y)$ from the below, while the starting position, $z_0 = z(t_0)$, which in the terms of Bessel process is below the threshold, $z_0 = \ell(y_0) < h_z$.

Consequently by choosing z_0 arbitrarily close yet below h_z, we can obtain an estimate for the burst duration, T, in terms of the hitting times of the Bessel process, $\tau^{(\nu)}_{z_0,h_z}$,

$$\begin{aligned} T = \tau^{(\nu)}_{z_0,h_z} &= \inf_{t>t_0}\Big\{t,\ z(t) \geq h_z\Big\}, \\ &0 < h_z - z_0 < \epsilon, \end{aligned} \tag{48}$$

where ϵ is an arbitrary small positive constant. As given in [82], the following holds for $0 < z_0 < h_z$

$$\begin{aligned} \rho^{(\nu)}_{z_0,h_z}(t) = \frac{h_z^{\nu-2}}{z_0^{\nu}} \sum_{k=1}^{\infty} \Bigg[& \frac{j_{\nu,k} J_\nu\left(\frac{z_0}{h_z} j_{\nu,k}\right)}{J_{\nu+1}(j_{\nu,k})} \cdot \\ & \cdot \exp\left(-\frac{j^2_{\nu,k}}{2h_z^2}t\right)\Bigg], \end{aligned} \tag{49}$$

where $\rho^{(\nu)}_{z_0,h_z}(t)$ is a probability density function of the hitting times at level h_z of Bessel process with index ν starting from z_0, J_ν is a Bessel function of the first kind of the order ν, while $j_{\nu,k}$ is a k-th zero of J_ν.

We have to replace $\rho^{(\nu)}_{z_0,h_z}(t)$ by density function regarding h_z to avoid the self-evident convergence of $\rho^{(\nu)}_{z_0,h_z}(t)$ (for $t > 0$) to zero, when $z_0 \to h_z$. This is achieved introducing the probability density function $p^{(\nu)}_{h_z}(t)$ as a probability density function of the burst duration

$$p^{(\nu)}_{h_z}(t) = \lim_{z_0 \to h_z} \frac{\rho^{(\nu)}_{z_0,h_z}(t)}{h_z - z_0}, \tag{50}$$

where we have selected the threshold at level h_z and ν is the original model parameter. To evaluate this limit we have

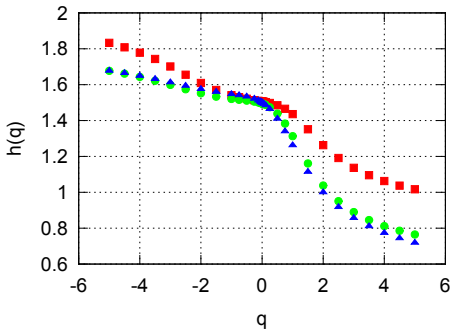


Figure 6. The broad spectra of Hurst exponents, $h(q)$, obtained from time series obtained by solving (32). The model parameters were set as follows: $\varepsilon_1 = 1$, $\varepsilon_2 = 2 - \alpha$, $\alpha = 0$ (red squares), 1 (green circles) and 2 (blue triangles).

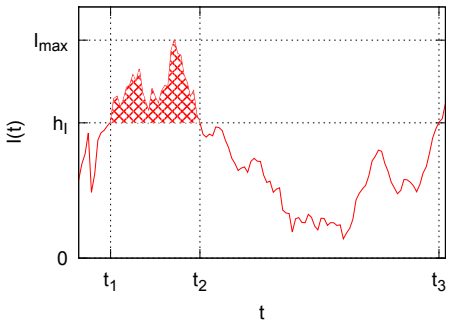


Figure 7. Time series exhibiting bursty behavior, $I(t)$. Here h_I is the threshold value, above which bursts are detected, t_i is the three visible threshold passage events, I_{max} is the highlighted burst's peak value. Burst duration we define as: $T = t_2 - t_1$.

to expand $J_\nu\left(\frac{z_0}{h_z} j_{\nu,k}\right)$ near $\frac{z_0}{h_z} = 1$:

$$J_\nu\left(\frac{z_0}{h_z} j_{\nu,k}\right) \approx J_\nu(j_{\nu,k}) - \left(1 - \frac{z_0}{h_z}\right) \cdot \cdot [\nu J_\nu(j_{\nu,k}) - j_{\nu,k} J_{1+\nu}(j_{\nu,k})] = (1 - \frac{z_0}{h_z}) j_{\nu,k} J_{1+\nu}(j_{\nu,k}). \quad (51)$$

By using this expansion we can rewrite (50) as:

$$p_{h_z}^{(\nu)}(t) \approx C_1 \sum_{k=1}^{\infty} j_{\nu,k}^2 \exp\left(-\frac{j_{\nu,k}^2}{2h_z^2} t\right), \quad (52)$$

here C_1 is a normalization constant. By taking a note that $j_{\nu,k}$ are almost equally spaced, we can replace the sum by integration

$$p_{h_z}^{(\nu)}(t) \approx C_2 \int_{j_{\nu,1}}^{\infty} x^2 \exp\left(-\frac{x^2 t}{2h_z^2}\right) \mathrm{d}x = = C_2 \left[\frac{h_z^2 j_{\nu,1} \exp\left(-\frac{j_{\nu,1}^2 t}{2h_z^2}\right)}{t} + \sqrt{\frac{\pi}{2}} \frac{h_z^3 \mathrm{erfc}\left(\frac{j_{\nu,1}\sqrt{t}}{\sqrt{2}h_z}\right)}{t^{3/2}}\right]. \quad (53)$$

From the expression above follows that the probability density of the burst durations in the time series obtained by solving SDE (18) can be approximated by a power law with exponential cut-off. Or mathematically

$$p_{h_z}^{(\nu)}(t) \sim t^{-3/2}, \quad \text{for} \quad t \ll \frac{2h_z^2}{j_{\nu,1}^2}, \quad (54)$$

$$p_{h_z}^{(\nu)}(t) \sim \frac{\exp\left(-\frac{j_{\nu,1}^2 t}{2h_z^2}\right)}{t}, \quad \text{for} \quad t \gg \frac{2h_z^2}{j_{\nu,1}^2} \quad (55)$$

This result is in agreement with a general property of one dimensional diffusion processes presented in [60], namely that the asymptotic behavior of first hitting times is a power law $t^{-3/2}$ irrespectively of the nature of stochastic one dimensional process or the actual mathematical expressions of the Langevin or the Fokker-Plank equations. The exponential cutoff for longer burst durations can be explained by the direction preference of the Bessel processes (note the positive drift term in case of $N > 1$, or alternatively $\nu > -0.5$). The actual empirical data, as shown in Fig. 8 (b), also has the predicted asymptotic behavior, though the inconsistence in fitting is clearly higher than for the model's probability density Fig. 8 (a).

Our empirical data set includes all trades made on NYSE, which were made from January, 2005 to March, 2007 and involved 24 different stocks, ABT, ADM, BMY, C, CVX, DOW, FNM, GE, GM, HD, IBM, JNJ, JPM, KO, LLY, MMM, MO, MOT, MRK, SLE, PFE, T, WMT, XOM. We have used one hour window moving average filter on

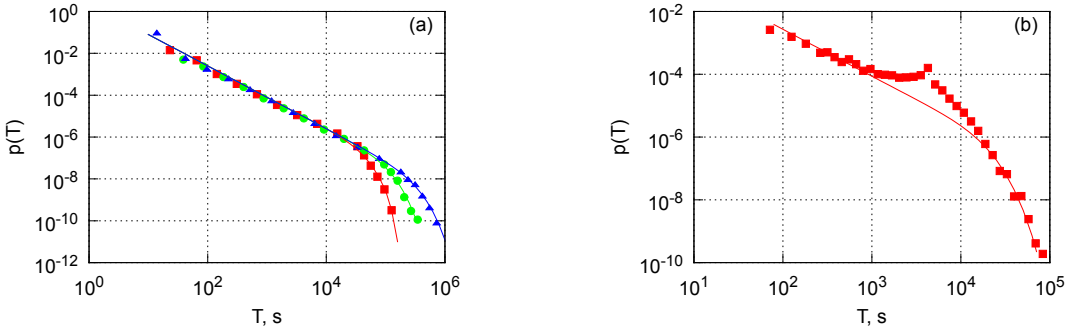


Figure 8. Numerical (a) and empirical (b) PDF of burst durations, $h_y = 2$. In both subfigures numerical and empirical data is represented by filled shapes, while fits, (53), are represented by gray curves. Model, (18), parameters were set as follows: $\sigma_t^2 = 1/6 \cdot 10^{-5}\mathrm{s}^{-1}$ (in all three cases), $\lambda = 4$ (in all three cases), $\eta = 2.5$ (red squares, $\nu = 0$), $\eta = 2$ (green circles, $\nu = 0.5$) and $\eta = 1.5$ (blue triangles, $\nu = 2$). Empirical data fitted by assuming that $\nu = -0.2$.

empirical one minute return series. As we consider the model to be universal, i.e., applicable towards the modeling of varying financial markets and stocks, we can consider each stocks' time series as a separate realization of the same stochastic process. Time series are first normalized and later averaged over the whole set. We back this approach by recalling that in [16] we have shown that the more sophisticated versions of (24) may be well used to model absolute return of different stocks from NYSE and Vilnius Stock Exchange.

There are numerous reasons for the observed inconsistence in fitting of empirical data Fig. 8 (b). Firstly, we were unable to remove intra-day pattern from the time series. But the main reason is that the simple stochastic model, driven by (18), is unable to reproduce the full complexity of empirically observed spectral density. In order to reproduce the correct, fractured, shape of the spectral density one must use double stochastic model, driven by a more sophisticated version SDE (24), [16]. Nevertheless, derived equations for the burst duration distribution (52) and (53) of the general process (18) are in agreement with empirical time series of return. This provides one more argument for the further development of stochastic models based on herding behavior of agents and nonlinear SDE (18).

X. Conclusions and Future Work

Reasoning of stochastic models of complex systems by the microscopic interactions of agents is still a challenge for researchers. Only very general models such as Kirman's herding model in ant colony or Bass diffusion model for new product adoption have well established agent-based versions and can be described by stochastic or ordinary differential equations. There are many different attempts of microscopic modeling in more sophisticated systems, such as financial markets or other social systems, intended to reproduce the same empirically defined properties. The ambiguity of microscopic description in complex systems is an objective obstacle for quantitative modeling. Simple enough agent-based models with established or expected corresponding macroscopic description are indispensable in modeling of more sophisticated systems. In this contribution we discussed various extensions and applications of Kirman's herding model.

First of all, we modify Kirman's model introducing interevent time $\tau(y)$ or trading activity $1/\tau(y)$ as functions of driving return y. This produces the feedback from macroscopic variables on the rate of microscopic processes and strong nonlinearity in stochastic differential equations responsible for the long range power-law statics of financial variables. We do expect further development of this approach introducing the mood of chartists as independent agent-based process.

Nonlinear SDEs derived from the agent herding model generate multifractal time series. This gives more confidence in the modeling of multifractal series observed in financial markets. We derive PDF of burst duration for the basic form of nonlinear SDE (18). This is in agreement with empirical time series of return. Further investigation of burst statistics in financial markets in comparison with analytical results from nonlinear SDE is ongoing. This would serve as an independent method to adjust model parameters to the empirical data.

One more outcome of Kirman's herding behavior of agents is one direction process - Bass diffusion. This simple example of correspondence between very well established microscopic and macroscopic modeling becomes valuable for further description of diffusion in social systems. Models presented on the interactive web site [10] have to facilitate further extensive use of computer modeling in economics, business and education.

Acknowledgment

Work presented in this paper is supported by EU SF Project "Science for Business and Society", project number:

VP2-1.4-ŪM-03-K-01-019.

We also express deep gratitude to Lithuanian Business Support Agency.

REFERENCES

[1] V. Daniunas, V. Gontis, and A. Kononovicius, "Agent-based versus macroscopic modeling of competition and business processes in economics," in *ICCGI 2011, The Sixth International Multi-Conference on Computing in the Global Information Technology*, Luxembourg, 2011, pp. 84–88.

[2] D. Helbing, *Managing Complexity: Insights, Concepts, Applications*. Springer, 2008.

[3] L. Pietronero, "Complexity ideas from condensed matter and statistical physics," *Europhysics news*, vol. 39, pp. 26–29, 2008.

[4] M. Waldrop, *Complexity: The emerging order at the edge of order and chaos*. New York: Simon and Schuster, 1992.

[5] D. C. Ince, L. Hatton, and J. Graham-Cumming, "The case for open computer programs," *Nature*, vol. 482, pp. 485–488, 2012.

[6] K. Niemeyer, "If you want reproducible science, the software needs to be open source," Nature Editorial, 2012. [Online]. Available: http://arstechnica.com/science/2012/02/science-code-should-be-open-source-according-to-editorial/ [Accessed: 2012-06-14]

[7] R. Axelrod, "Advancing the art of simulation in the social sciences," *Complexity*, vol. 3, no. 2, pp. 16–32, 1998.

[8] D. Helbing, "Pluralistic modeling of complex systems," *Science and Culture*, vol. 76, no. 2, p. 315, 2010.

[9] J. H. Johnson, "The future of the social sciences and humanities in the science of complex systems," *The European Journal of Social Science Research*, vol. 23, no. 2, pp. 115–134, 2010.

[10] V. Gontis, A. Kononovicius, and V. Daniunas, "Physics of risk." [Online]. Available: http://mokslasplius.lt/rizikos-fizika/en [Accessed: 2012-06-11]

[11] A. P. Kirman, "Ants, rationality and recruitment," *Quarterly Journal of Economics*, vol. 108, pp. 137–156, 1993.

[12] S. Alfarano, T. Lux, and F. Wagner, "Estimation of agent-based models: The case of an asymmetric herding model," *Computational Economics*, vol. 26, no. 1, pp. 19–49, 2005.

[13] F. M. Bass, "A new product growth model for consumer durables," *Management Science*, vol. 15, pp. 215–227, 1969.

[14] M. Jeanblanc, M. Yor, and M. Chesney, *Mathematical Methods for Financial Markets*. Berlin: Springer, 2009.

[15] B. Kaulakys and M. Alaburda, "Modeling scaled processes and $1/f^{\beta}$ noise using non-linear stochastic differential equations," *Journal of Statistical Mechanics*, p. P02051, 2009.

[16] V. Gontis, J. Ruseckas, and A. Kononovicius, "A non-linear stochastic model of return in financial markets," in *Stochastic Control*, C. Myers, Ed. InTech, 2010.

[17] J. Ruseckas and B. Kaulakys, "1/f noise from nonlinear stochastic differential equations," *Physical Review E*, vol. 81, p. 031105, 2010.

[18] J. W. Kantelhardt, S. A. Zschiegner, E. Koscielny-Bunde, S. Havlin, A. Bunde, and H. E. Stanley, "Multifractal detrended fluctuation analysis of nonstationary time series," *Physica A*, vol. 316, pp. 87–114, 2002.

[19] J. P. Bouchaud, "Economics need a scientific revolution," *Nature*, vol. 455, p. 1181, 2008.

[20] ——, "The (unfortunate) complexity of the economy," *Physics World*, pp. 28–32, April 2009.

[21] ——, "The Bachelier legacy: Why and how do asset prices move?" Siena, Italy, 2010, talk given at International School on Multidisciplinary Approaches to Economic and Social Complex Systems.

[22] J. D. Farmer and D. Foley, "The economy needs agent-based modelling," *Nature*, vol. 460, pp. 685–685, 2009.

[23] T. Lux and F. Westerhoff, "Economic crysis," *Nature Physics*, vol. 5, pp. 2–3, 2009.

[24] M. E. J. Newman, "Complex systems: A survey," *American Journal of Physics*, vol. 79, pp. 800–810, 2011.

[25] C. Schinckus, "Econophysics and economics: Sister disciplines?" *American Journal of Physics*, vol. 78, no. 4, pp. 325–327, 2010.

[26] M. Cristelli, L. Pietronero, and A. Zaccaria, "Critical overview of agent-based models for economics," in *Proceedings of the School of Physics "E. Fermi", course CLXXVI*, Varenna, 2010.

[27] T. Lux and M. Marchesi, "Scaling and criticality in a stochastic multi-agent model of a financial market," *Nature*, vol. 397, pp. 498–500, 1999.

[28] S. Bornholdt, "Expectation bubbles in a spin model of markets: Intermittency from frustration across scales," *International Journal of Modern Physics C*, vol. 12, no. 5, pp. 667–674, 2001.

[29] T. Kaizoji, S. Bornholdt, and Y. Fujiwara, "Dynamics of price and trading volume in a spin model of stock markets with heterogeneous agents," *Physica A*, vol. 316, pp. 441–452, 2002.

[30] J. P. Sethna, *Statistical Mechanics: Entropy, Order Parameters and Complexity*. Oxford: Clarendon Press, 2009.

[31] E. Bonabeau, "Agent-based modeling: Methods and techniques for simulating human systems," *Proceedings of National Academy of Science USA*, vol. 99, no. Suppl 3, pp. 7280–7287, 2002.

[32] W. B. Arthur, "Inductive reasoning and bounded rationality," *American Economic Review*, vol. 84, pp. 406–411, 1994.

[33] D. Challet and Y.-C. Zhang, "Emergence of cooperation and organization in an evolutionary game," *Physica A*, vol. 246, pp. 407–418, 1997.

[34] D. Challet, M. Marsili, and R. Zecchina, "Statistical mechanics of systems with heterogeneous agents: Minority games," *Physical Review Letters*, vol. 84, pp. 1824–1827, 2000.

[35] S. Alfarano, T. Lux, and F. Wagner, "Time variation of higher moments in a financial market with heterogeneous agents: An analytical approach," *Journal of Economic Dynamics and Control*, vol. 32, pp. 101–136, 2008.

[36] V. Alfi, M. Cristelli, L. Pietronero, and A. Zaccaria, "Minimal agent based model for financial markets i: Origin and self-organization of stylized facts," *European Physics Journal B*, vol. 67, no. 3, pp. 385–397, 2009.

[37] ——, "Minimal agent based model for financial markets ii: Statistical properties of the linear and multiplicative dynamics," *European Physics Journal B*, vol. 67, no. 3, pp. 399–417, 2009.

[38] A. Kononovicius and V. Gontis, "Agent based reasoning for the non-linear stochastic models of long-range memory," *Physica A*, vol. 391, no. 4, pp. 1309–1314, 2012.

[39] S. M. Krause, P. Bottcher, and S. Bornholdt, "Mean-field-like behavior of the generalized voter-model-class kinetic ising model," *Physical Review E*, vol. 85, p. 031126, 2012.

[40] A. Kirman and G. Teyssiere, "Microeconomic models for long memory in the volatility of financial time series," *Studies in Nonlinear Dynamics and Econometrics*, vol. 5, no. 4, pp. 281–302, 2002.

[41] A. Kononovicius and V. Gontis, "Kirmans ant colony model." [Online]. Available: http://mokslasplius.lt/rizikos-fizika/en/kirman-ants [Accessed: 2012-06-11]

[42] ——, "Stochastic ant colony model." [Online]. Available: http://mokslasplius.lt/rizikos-fizika/en/stochastic-ant-colony-model [Accessed: 2012-06-11]

[43] ——, "Agent based herding model of financial markets." [Online]. Available: http://mokslasplius.lt/rizikos-fizika/en/agent-based-herding-model-financial-markets [Accessed: 2012-06-15]

[44] ——, "Multifractality of time series." [Online]. Available: http://mokslasplius.lt/rizikos-fizika/en/multifractality-time-series [Accessed: 2012-06-15]

[45] V. Mahajan, E. Muller, and F. M. Bass, "New-product diffusion models," in *Handbooks in Operations Research and Management Science*, J. Eliashberg and G. L. Lilien, Eds. Amsterdam: North Holland, 1993, vol. 5, pp. 349–408.

[46] V. Daniunas, "Verslo modeliai." [Online]. Available: http://mokslasplius.lt/rizikos-fizika/category/business [Accessed: 2012-06-11]

[47] A. Kononovicius and V. Gontis, "Unidirectional kirmans model." [Online]. Available: http://mokslasplius.lt/rizikos-fizika/en/unidirectional-kirman-model [Accessed: 2012-06-11]

[48] A. Corral, "Long-term clustering, scaling, and universality in the temporal occurrence of earthquakes," *Physical Review Letters*, vol. 92, p. 108501, 2004.

[49] M. S. Wheatland and P. A. Sturrock, "The waiting-time distribution of solar flare hard x-ray bursts," *Astrophysics Journal*, vol. 509, p. 448, 1998.

[50] T. Kemuriyama, H. Ohta, Y. Sato, S. Maruyama, and M. Tandai-Hiruma, "A power-law distribution of inter-spike intervals in renal sympathetic nerve activity in salt-sensitive hypertension-induced chronic heart failure," *BioSystems*, vol. 144147, p. 101, 2010.

[51] R. Cont, "Empirical properties of asset returns: Stylized facts and statistical issues," *Quantitative Finance*, vol. 1, pp. 1–14, 2001.

[52] M. Karsai, K. Kaski, A. L. Barabasi, and J. Kertesz, "Universal features of correlated bursty behaviour," *NIH Scientific Reports*, vol. 2, p. 397, 2012.

[53] J. Kleinberg, "Bursty and hierarchical structure in streams," *Data Mining and Knowledge Discovery*, vol. 7, pp. 373–397, 2003.

[54] V. Gontis and B. Kaulakys, "Multiplicative point process as a model of trading activity," *Physica A*, vol. 343, pp. 505–514, 2004.

[55] B. Kaulakys, V. Gontis, and M. Alaburda, "Point process model of 1/f noise vs a sum of lorentzians," *Physical Review E*, vol. 71, no. 051105, pp. 1–11, 2005.

[56] J. Ruseckas, B. Kaulakys, and V. Gontis, "Herding model and 1/f noise," *EPL*, vol. 96, p. 60007, 2011.

[57] V. Gontis and B. Kaulakys, "Long-range memory model of trading activity and volatility," *Journal of Statistical Mechanics*, vol. P10016, pp. 1–11, 2006.

[58] V. Gontis, B. Kaulakys, and J. Ruseckas, "Trading activity as driven poisson process: comparison with empirical data," *Physica A*, vol. 387, pp. 3891–3896, 2008.

[59] C. W. Gardiner, *Handbook of stochastic methods*. Berlin: Springer, 1997.

[60] S. Redner, *A guide to first-passage processes*. Cambridge University Press, 2001.

[61] V. Gontis, A. Kononovicius, and S. Reimann, "The class of nonlinear stochastic models as a background for the bursty behavior in financial markets," 2012.

[62] [Online]. Available: http://wordpress.org [Accessed: 2012-06-15]

[63] [Online]. Available: http://wordpress.org/extend/plugins/wp-latex/ [Accessed: 2012-06-11]

[64] [Online]. Available: http://sourceforge.net/projects/bibliophile/files/bibtexParse/ [Accessed: 2012-06-11]

[65] [Online]. Available: http://www.xjtek.com/anylogic [Accessed: 2012-06-11]

[66] [Online]. Available: http://www.java.com/en/ [Accessed: 2012-06-11]

[67] J. M. Pasteels, J. L. Deneubourg, and S. Goss, "Self-organization mechanisms in ant societies (i): Trail recruitment to newly discovered food sources," in *From Individual to Collective Behaviour in Social Insects*, J. M. Pasteels and J. L. Deneubourg, Eds. Basel: Birkhauser, 1987, pp. 155–175.

[68] ——, "Self-organization mechanisms in ant societies (ii): Learning in foraging and division of labor," in *From Individual to Collective Behaviour in Social Insects*, J. M. Pasteels and J. L. Deneubourg, Eds. Basel: Birkhauser, 1987, pp. 177–196.

[69] N. G. van Kampen, *Stochastic process in Physics and Chemistry*. Amsterdam: North Holland, 1992.

[70] G. Fibich, R. Gibori, and E. Muller, "A comparison of stochastic cellular automata diffusion with the bass diffusion model," NYU Stern School of Business, Tech. Rep., 2010.

[71] V. Gontis, J. Ruseckas, and A. Kononovicius, "A long-range memory stochastic model of the return in financial markets," *Physica A*, vol. 389, pp. 100–106, 2010.

[72] B. Kaulakys, J. Ruseckas, V. Gontis, and M. Alaburda, "Nonlinear stochastic models of 1/f noise and power-law distributions," *Physica A*, vol. 365, pp. 217–221, 2006.

[73] S. Reimann, V. Gontis, and M. Alaburda, "Interplay between positive feedback in the generalized cev process," *Physica A*, vol. 390, no. 8, pp. 1393–1401, 2011.

[74] J. Feder, *Fractals*. New York: Plenum Press, 1988.

[75] L. Telesca, V. Lapenna, and M. Macchiato, "Multifractal fluctuations in earthquake-related geoelectrical signals," *New Journal of Physics*, vol. 7, p. 214, 2005.

[76] P. C. Ivanov, L. A. N. Amaral, A. L. Goldberger, S. Havlin, M. B. Rosenblum, Z. Struzik, and H. E. Stanley, "Multifractality in healthy heartbeat dynamics," *Nature*, vol. 399, pp. 461–465, 1999.

[77] B. J. West and N. Scafetta, "Nonlinear dynamical model of human gait," *Physical Review E*, vol. 67, p. 051917, 2003.

[78] M. Ausloos and K. Ivanova, "Multifractal nature of stock exchange prices," *Computer Physics Communications*, vol. 147, pp. 582–585, 2002.

[79] E. E. Peters, *Fractal market analysis: applying chaos theory to investment and economics*. John Wiley and Sons, 1994.

[80] J. Kwapien, P. Oswiecimka, and S. Drozdz, "Components of multifractality in high-frequency stock returns," *Physica A*, vol. 350, pp. 466–474, 2005.

[81] B. Kaulakys, M. Alaburda, V. Gontis, and T. Meskauskas, *Multifractality of the Multiplicative Autogressive Point Processes*. World Scientific, 2006, pp. 277–286.

[82] A. N. Borodin and P. Salminen, *Handbook of Brownian Motion*, 2nd ed. Basel, Switzerland: Birkhauser, 2002.

Lumen Detection in Endoscopic Images: A Boosting Classification Approach

Giovanni Gallo and Alessandro Torrisi
Department of Mathematics and Computer Science
***I**mage **P**rocessing **Lab**oratory*
University of Catania, Italy
{gallo,atorrisi}@dmi.unict.it

***Abstract*—Intestinal lumen detection in endoscopic images is clinically relevant to assist the medical expert in studying intestinal motility. Wireless Capsule Endoscopy (WCE) produces a high number of frames. Automatic classification, indexation and annotation of WCE videos is crucial to a more widespread use of this diagnostic tool. In this paper we propose a novel intestinal lumen detection method based on boosting. In particular, we propose a customized set of Haar-like features combined with a variant of AdaBoost to select discriminative features and to combine them into a cascade of strong classifiers. Experimental results show the efficacy of boosted classifiers to quickly recognize the presence of intestinal lumen frames in a video. To better assess the accuracy of the proposed boosted classifier, we present an experimental comparison with the results obtained with a Support Vector Machine using a linear kernel.**

***Keywords*-Classification; Pattern Recognition; Boosting; Wireless Capsule Endoscopy; Video Automatic Annotation; Support Vector Machine.**

I. INTRODUCTION

Wireless Capsule Endoscopy [2], [3] is a technique to explore small intestine regions that traditional endoscopy does not reach. A video-capsule, that integrates wireless transmission with image technology, is swallowed by the patient and it is propelled through the gut by intestinal peristalsis. Once activated, the capsule captures two frames per second and transmits images to an external receiver. The exam is concluded after about eight hours, that corresponds to the lifetime of the battery of the capsule. Images taken during the entire route of the capsule through the intestine are successively analyzed by an expert. He/She may spend up to one or more hours to gather the relevant information for a proper diagnosis. This greatly limits the use of the capsule as a diagnostic routine tool.

Such shortcoming may be overcome if the WCE video is automatically segmented into shorter videos, each one relative to a different trait of the bowels, and if reliable automatic annotation tools are available to the clinicians. Unfortunately, the goal of automatically producing a summary of the whole WCE video remains yet unaccomplished. Tools to extract semantic information from such videos are relevant research products for applied Pattern Recognition investigators.

In this paper we present a novel method to automatically discriminate a relevant subclass of frames. In particular, our classifier sorts the frames in two categories: "lumen frames" (images depicting the stages of an intestinal contraction where the shrinkage of lumen intestine is well visible) and "not lumen frames" (Figure 1). "Lumen frames" detection is clinically relevant because it announces the presence of a contraction and helps the physician to study the intestinal motility. Alteration of the physiological intestinal motility is an indicator of disorders in which the gut has lost its ability because of endogenous or exogenous causes. In particular, anomalies in contraction are a common symptom of irritable bowel syndrome, delayed gastric emptying, cyclic vomiting syndrome, and so on.

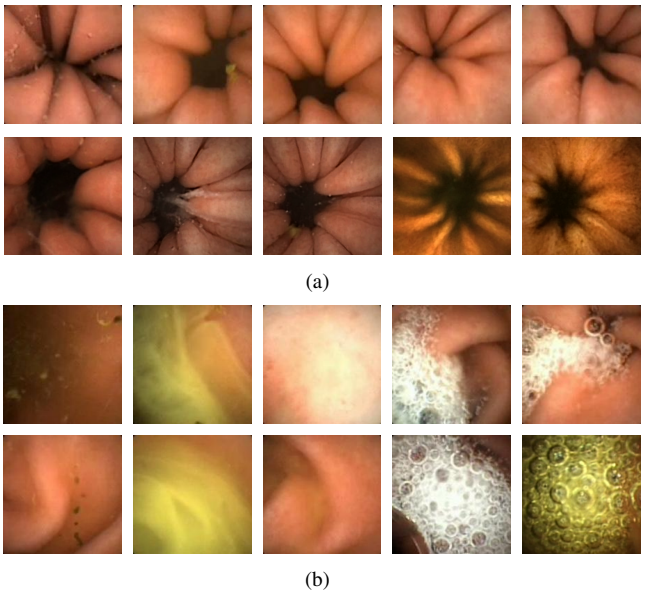


Figure 1. Examples of *lumen* (a) and *not lumen* (b) frames extracted from a WCE video.

Our summarization tool may be deployed in a diagnostic station providing real-time useful shortcuts to the middle phases of an intestinal contraction resulting in reduced time of analysis by the expert.

In our approach "lumen frames" detection is obtained as a special case of object detection. To this aim we choose the Viola and Jones paradigm introduced in 2001 [4]. Although

other techniques, like neural networks, fuzzy rules systems, etc., could be deployed, the main motivation for our choice has been the following. Haar features based classification is readily customizable to recognize different kinds of objects; moreover, boosting allows fast learning even in presence of high dimensionality data. Indeed in the case of boosting as for all ensemble learning method, different classifiers are built using a tiny part of the available features. The classification obtained by combining the responses of different classifiers improves the performance achieved by a standard classification algorithm in a straightforward, efficient, principled way when adaptive boosting is adopted.

This paper is organized as follows: Section II reviews related works and reports examples of object detection based on approaches similar to the proposed one. Section III describes in detail how Viola-Jones technique is customized to address the present problem. Section IV reports the experiments conducted on real WCE videos. It also describes an interesting comparison between the results obtained using Boosting and Support Vector Machine. Finally, Section V draws conclusions and discusses some future works.

This paper is a revised and expanded version of the contribution presented by the same authors to "The Third International Conferences on Pervasive Patterns and Applications" [1].

II. Related Works

Most of the systems reported in literature to recognize intestinal lumen images refer to traditional probe-based endoscopy. The motivation behind these methods is to aid the physician to individuate lumen region to avoid or minimize the collision of the endoscope tip with the intestinal mucosa. In this context, Asari [5] proposes a Region Growing Segmentation to extract lumen from gray level endoscopic images.

Recently, the original WCE has been modified/updated to a novel configuration allowing the movements to be remotely monitored. In this context, the recognition of lumen could help the capsule to go through the intestine minimizing collisions and avoiding to record meaningless frames. To this aim, Zabulis et al. [6] propose a system based on a Mean Shift Segmentation algorithm variant to locate lumen regions in WCE frames.

The problem of the detection of frames with a clear narrowing of lumen in WCE videos to assist the diagnostic and clinical use of this imaging technique is not much investigated. Some works study the general problem of contraction finding to examine the intestinal motility [7]–[9]. If we associate a label to each "lumen frame" extending the selection to a certain number of adjacent images in the video, our task is roughly equivalent to the search of intestinal contractions.

The main idea exploited in this work is to customize the Viola-Jones method for object detection [4], [10]. Initially proposed for face detection, this technique is based on the use of simple features calculated in a new representation of the image. Based on the concept of integral image [11], a huge set of features is tested and the boosting algorithm AdaBoost is used to reduce this set [12]–[14]. The introduction of a tree of boosted classifiers provides a robust and fast detection and minimizes the false positive rate. This strategy has been proven effective to recognize various kinds of objects. Several systems have been proposed for different recognition problems, like face, hands and pedestrian [15]–[18]. The possibility to define a specific set of features and the more recent release of an open source implementation [19] have permitted to use extensively this method in many Computer Vision applications.

III. Proposed Method

In this section we describe an automatically trainable system to detect frames where the front shrinkage of intestinal lumen is well visible. The learning stage for the proposed system can be summarized in the following three steps:

- Evaluation of a customized set of Haar features to the integral images of the training samples.
- Selection of the best discriminative features through AdaBoost algorithm.
- Construction of a final boosted classifier based on a cascade of classifiers whose complexity is gradually increasing.

To obtain, through a reliable learning procedure, a good classifier we must guarantee two requirements: a comprehensive set of examples where the objects of interest may occur; a suitable selection of descriptors to describe each possible occurring pattern. In order to detect an object in an image we should in principle take into account the information provided by each single pixel. This search space may be reduced if we exploit the semantic information enclosed by "lumen frames". These images, indeed, show a strong geometrical coherence that may help in discriminating them from other kinds of frames. To this aim, Haar-like features, a set derived from Haar wavelets [20], recognize objects using intensity contrast between adjacent regions in an image.

Basic Haar features proposed by Viola-Jones and specialized for face detection do not have proper discriminative power for lumen investigation: it is necessary to define customized variations for the present case. In particular, the features needed in this work should provide a strong positive response on a rectangular region with low intensity called generically "lumen" and a brighter surrounding area corresponding to the gut wall. By combining a learned evaluation threshold to each feature, it is possible to assign an image to the appropriate category. Figure 2a shows an example of the first kind of our proposed features that we call "center-surround" feature.

The typical appearance of a frame that shows an intestinal contraction consists in a dark area surrounded by the typical

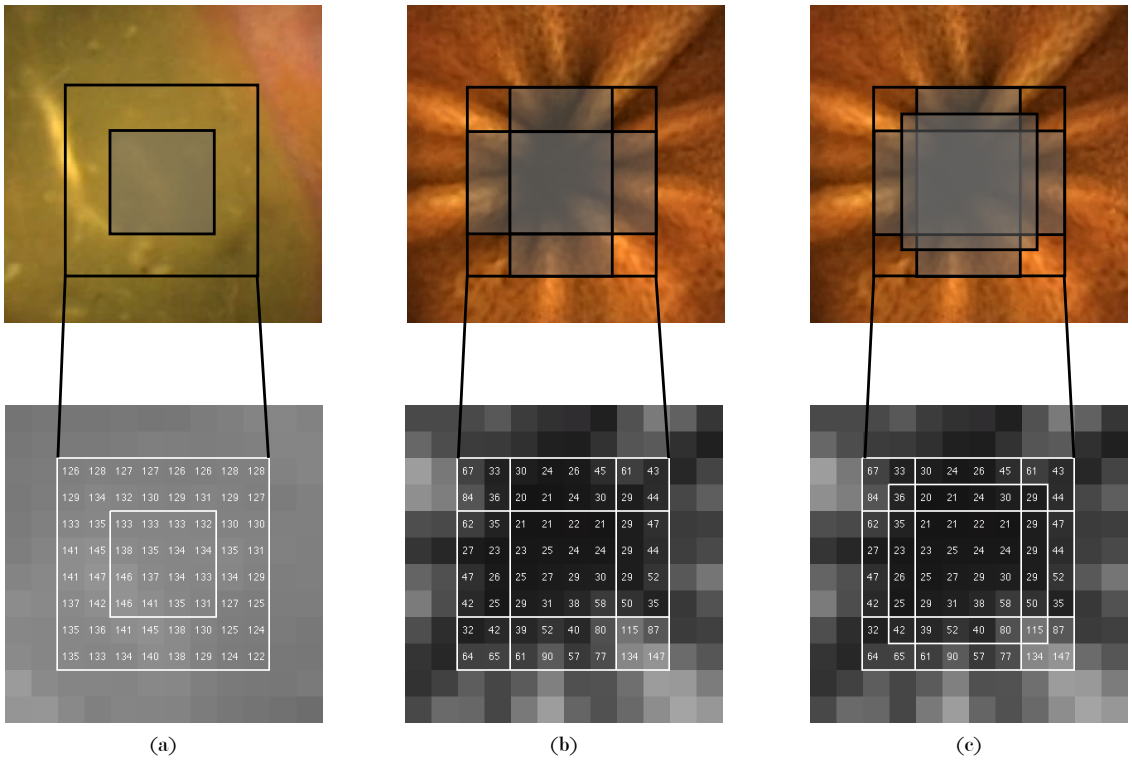


Figure 2. The three proposed kinds of features. For each feature we get a score S calculated as the difference of intensity between light and dark regions of the rectangle. In the first row are shown the images at the original resolution while in the second the images are rescaled to the base resolution 24×24 pixels zooming on the region of interest. (a) Evaluation of a "center-surround" feature in a "not lumen frame" ($S_a = 6348 - 2175 = 4173$). (b) Evaluation of the first cross feature in a "lumen frame" ($S_b = 1083 - 1766 = -683$). (c) Evaluation of the second cross feature in a "lumen frame" ($S_c = 861 - 1988 = -1127$).

rays that muscular tone produces due to the folding of the intestinal wall. We hence introduce two additional "cross-like" features that enhance the discriminative power produced by the simpler "center-surround" feature (Figure 2b - 2c). The computation of this second kind of features may be efficiently obtained as for the simpler "center-surround" feature from the integral image representation.

Using integral image representation, feature evaluation is accomplished by few memory accesses. It is straightforward to verify that to compute "center-surround" features, at any position or scale, only eight look-ups are needed. The remaining two kinds of features require more accesses due to greater number of rectangular areas. "Cross-features" require respectively 16 and 24 references from the integral image. The reader may easily convince himself that indeed this is the minimum number of look-ups needed from a direct analysis of this feature geometry.

Once a feature shape has been assigned, it is necessary to specify its position and scale within the region of interest. Actually, the features are scanned across the image top left to bottom right using a sliding offset of two pixels both in the horizontal and in the vertical directions. The process is iteratively repeated with different feature scales at each round. To keep the computation of the proposed features within the same number of look-ups into the integral image, we choose not to change the scale of the image but to variate instead the size of the features.

The exact representation for the three proposed types of features is as follows:

$$f = [x_w, y_w, s_{wx}, s_{wy}, x_b, y_b, s_{bx}, s_{by}, type, \theta, \rho] \qquad (1)$$

The first four elements x_w, y_w, s_{wx}, s_{wy}, refer to the larger square of the feature. Similarly, the following four elements x_b, y_b, s_{bx}, s_{by}, relate to the inner square. The $type$ parameter is an integer that indicates which type of feature is considered (1 for the "center-surround" feature, 2 and 3 for the two kinds of cross features respectively). The last two parameters are the optimal learned threshold and the polarity to register the category of images discriminated by that feature.

The "center-surround" features are evaluated considering difference between the sum of the pixels within two rectangular regions (Figure 3a). The second type of features considers a cross-shaped region to enhance lumen area. Location and size of this region are constrained by the size of correlated "center-surround" feature (Figure 3b). The third

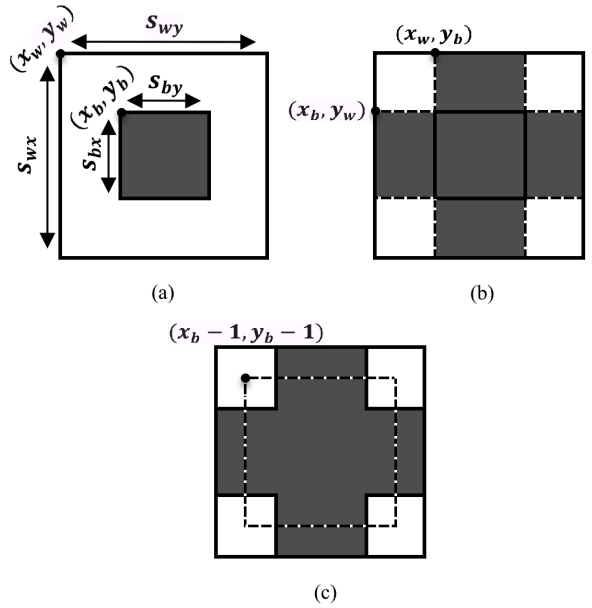


Figure 3. Schematic features representation. (a) Center-surround feature. (b) First cross feature obtained by center-surround feature considering the cross with width s_{by} and height s_{bx}. (c) Second cross feature obtained by the first taking into account a inner square of width and height greater than one pixel respect to the previous version.

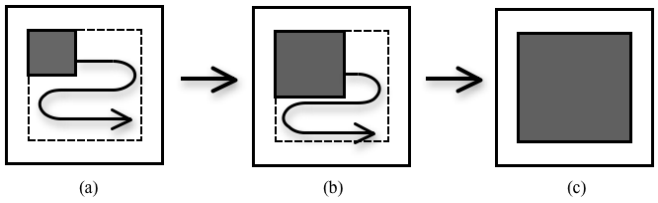


Figure 4. Given feature size, all regions of a fixed scale are considered in each location (a). This cycle is reiterated by increasing the size of the inner square (b) until maximum amplitude is achieved (c).

type of features is processed in a similar way. The central region of the cross is enlarged of one pixel both in the horizontal and in the vertical directions (Figure 3c). We consider the same total number of features for each type. Lumen area presents always a square aspect ratio, i.e., the bounding region of these areas is approximatively a square. This leads to a simplification of the feature definition (1) as follows:

$$f = [x_w, y_w, s_w, x_b, y_b, s_b, type, \theta, \rho] \quad (2)$$

We consider only squared features, i.e., those with equal horizontal and vertical even scale s_w. The internal region relative to lumen varies from a minimum size 2×2 up to $(s_w - 2) \times (s_w - 2)$ pixels. Once we have fixed the size of the external section, the descriptor associated with the lumen is shifted across the external descriptor with a resizing of two pixels at each step (Figure 4).

In this phase of processing the resolution of a WCE frame is reduced to 24×24 pixels. The total number of features per scale is hence equal to the total amount of different features in the image multiplied by the allowed variations of scale. For example, a 8×8 feature contains nine regions of size 2×2, four of size 4×4 and one of size 6×6 pixels. The total number of features of size 8×8 is 1134, equal to the number of windows in the image (assuming a horizontal and vertical offset of two pixels) for the total number of variations. Table I summarizes the feature counting for the chosen scales.

A. Training a cascade of strong classifiers

As it is stated above, during the training phase the dataset is rescaled to the base resolution 24×24 pixels. The integral image representation of gray tone training samples is used to compute feature scores. Application of AdaBoost provides a list of best discriminative features. In particular, we build a binary classifier for each feature (these are traditionally referred in the boosting community as weak classifiers). Initially all the examples have the same weight. For each boosting step, the determination of a new weak classifier involves the evaluation of the relevance of each feature on training data. The "best" feature is selected according to the weighted error that each feature shows on the training data. In the successive round, the samples are reweighted to emphasize the misclassified ones. Since this step has to be iterated several times, this is the most expensive section of the training module.

The result of the training module is a classifier (called "strong classifier" in the boosting jargon) computed as a weighted linear combination of the weak classifiers built during each round of boosting. The whole boosting process is, in turn, iterated, varying at each step the number of weak classifiers. The result is the realization of a cascade of strong classifiers with a gradually increasing number of features.

An appropriate learning process requires that each strong classifier shows a prescribed detection rate, while main-

Table I
FEATURES NUMBER PER SCALE. THE FIRST COLUMN REFERS TO THE SIZE OF THE FEATURE WHILE THE SECOND IS RELATED TO MAXIMUM SCALE ALLOWED FOR THE LUMEN AREA.

Feature size	*Max Internal scale*	*#Features*	*#Variations*	*Total*
4×4	2×2	121	1	121
6×6	4×4	100	5	500
8×8	6×6	81	14	1134
10×10	8×8	64	30	1920
12×12	10×10	49	55	2695
14×14	12×12	36	91	3276
16×16	14×14	25	140	3500
18×18	16×16	16	204	3264
20×20	18×18	9	285	2565
22×22	20×20	4	385	1540
24×24	22×22	1	506	506
				21021

taining a definite rate of false positives. In particular, a minimum detection rate and a maximum false positive rate is required at every level of the cascade. For each strong classifier, a weak classifier is added until it reaches the required parameters for the current level of the cascade. Similarly, a new strong classifier is associated to the cascade until total false positive rate crosses a certain threshold.

One of the advantages of the proposed system is that the user only needs to define the feature set to be used and the false positives and detection rates for each level of the cascade. All the internal parameters are automatically selected during the training phase.

B. Testing a cascade of strong classifiers

In the proposed system, each test image is scaled to 24×24 pixels and it is labelled as "lumen frame" or "not lumen frame". This single scale procedure combined with selection of best features during training allows real time application of our system (up to 600 frames per second). Please notice that, differently than in the case where the object to recognize may appear at different scales, in the present case a "single-scale" choice has been shown adequate. Notice that in this simplifying choice of a single scale we differ from the original Viola and Jones approach. In the case of face detection the issue is to find faces that may appear at different scales within an image. These stringent requirements force Viola and Jones to include different scales in their detection procedure. In our case the problem is simpler: lumens are roughly all at the same scale and we do not require localization of them inside the frame but only to label the frame as a "lumen frame". This justifies our choice of a single scale.

IV. Experimental Results

A. Boosting based classification

In this section, we report the experiments carried out to verify the efficacy of the proposed method. To this aim, we have considered 10033 images extracted from real WCE videos of 12 patients of which 6 were healthy and 6 had suspected bowel disorders. Rather than considering only one training set as was done in an earlier version of this paper [1], we have extracted ten different training sets and control sets from the whole set at our disposal. This more extensive experiment has been aimed to verify if the behavior of the algorithm significantly changes according to the used learning set. To train each one of the cascades of strong classifiers, we take into account the integral images of 3000 images, 1000 positive and 2000 negative, rescaled to 24×24 pixels. The positive images have been previously manually selected from WCE videos labelled by an expert. The selected images represent a comprehensive set of scenes where the intestinal lumen can be present, including location and scale changes within the image. Differently, the negative examples have been randomly selected from videos that not contain any lumen. Both typical smooth images and images containing other judged negative events, like the presence of bubbles, bleedings, residuals, share this set.

During the learning module, we need to establish a maximum false positive rate and a minimum detection rate to satisfy for each layer of cascade. In particular, we require that 98% of positive images must be recognized at each level while maintaining a maximum amount of false positives equivalent to 80%. These values have been experimentally optimized. Notice, however, that higher positive images recognition rate are first of all rarely attainable and even when possible, they may introduce strong overfitting. At the next levels of the cascade these two values are computed relatively to the new dataset whose positives set is composed by every lumen recognized as such by the previous classifier; the negatives set includes the remaining false positives. A strong classifier will be added to the cascade until the total false positive rate drops to zero.

By iterating this process for each training set, we get ten different cascades of strong classifiers whose details are listed in Table II. It is straightforward to understand that the trained cascades are slightly different only in the total number of features, but the proportion of features is often the same: the cross-shaped features ($Cross\ 1$, $Cross\ 2$) are the most discriminative. The number of nodes in the cascade can not be deterministically calculated, but this also depends on the type of images used during learning. We do not impose any constraints on the number of features in each node. It is assured only that the node $i+1$ must have a greater or equal number of features than node i. To clarify this procedure, in Figure 5 is illustrated the cascade of strong classifiers relative to the 8-th dataset. The total detection rate of this cascade, D, and the final false positive rate F, are obtained as a combination of intermediate outcomes on the cascade:

$$D = \prod_{i=1}^{N} d_i = 97,98\% \qquad F = \prod_{i=1}^{N} f_i = 0\% \qquad (3)$$

where N is the total number of layers of the cascade. To test the effectiveness of trained cascades, we have considered

Table II
Details on trained cascades using ten different training sets.

$Train\ Data$	$Nodes$	$features$	$Center\ surround$	$Cross\ 1$	$Cross\ 2$
1	6	217	51	78	88
2	5	291	82	109	100
3	6	397	89	154	154
4	6	342	77	131	134
5	6	256	57	71	128
6	5	185	47	67	71
7	6	257	72	98	87
8	5	205	60	66	79
9	6	184	47	80	57
10	5	272	67	100	105

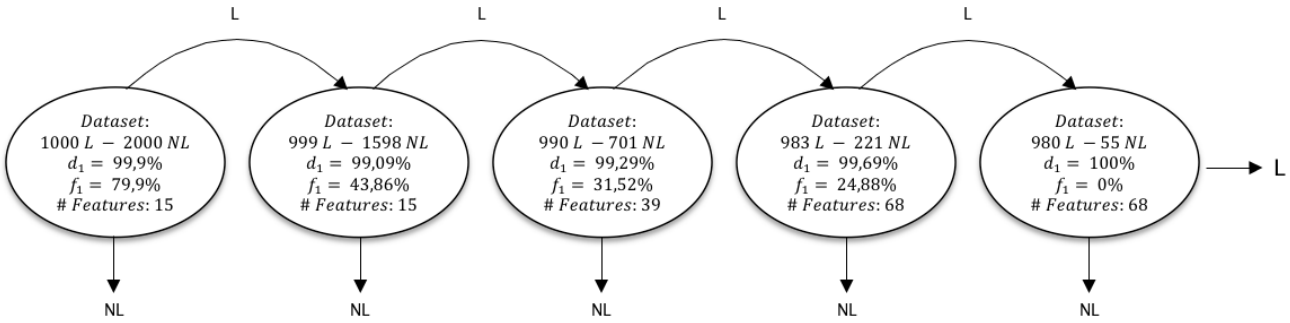


Figure 5. Cascade of strong classifiers. d_i and f_i represent detection and false positive rate at the i-th level of cascade. L and NL indicate *lumen* and *not lumen* frames, respectively.

ten different collections of 7033 images randomly extracted from a set of frames disjoined from each training set. During testing phase, we consider the integral images of test set rescaled to 24×24 pixels with the respective labels, the cascade of boosted classifiers as it has been obtained during training and, finally, a threshold that determines the rigorousness of the classifier. Each test sample gets through each single node of the cascade; a positive outcome is sent by the classifier i to the more complex classifier $i + 1$. An image is labeled as lumen if positively overcomes each node of the cascade. If at any point the test image is judged negative, it is rejected immediately without further test (Figure 5). The classification performance has been evaluated in terms of precision and recall by comparing our results with the annotations provided by the specialist. Table III shows the results. The labeling of images was previously made by a human expert. However, for certain images it is often difficult to understand, even to a skilled human observer, if what we hold as "lumen frame" is actually a particular fold of the intestinal tissue or vice versa.

Each strong classifier in the cascade is constrained by a rigidity threshold. Higher threshold values minimizes both detection and false positive rates. Similarly, a low threshold will lead to acceptance of a greater number of lumens images while increasing the probability of detecting false positives. The optimal value of threshold depends on the preferences of the physician. We expect that a higher amount of false positives than of false negatives is typically preferred. The presence of a high number of false positive results in more time spent by the expert to do a diagnosis. Losing a rightful lumen is a worse event because it means to miss a relevant event with the resulting inaccuracy in the final report. By varying the rigidity threshold from a minimum to a maximum value, we can construct a ROC curve comparing the detection rate versus the number of false positives. Figure 6 reveals that is possible to reach a detection rate above the 90%, keeping the amount of false positives at about 600 instances, i.e., 8% of the test dataset. All experiments have been conducted on a consumer level PC with Intel®Core™2 Duo processor and 4 GB of RAM. Calculations have been performed in MATLAB environment.

Table III
CLASSIFICATION RESULTS USING BOOSTING

Test Data	*Recall*	*Precision*	*Accuracy*
1	88,60%	72,06%	91,32%(6423/7033)
2	89,05%	71,64%	91,24%(6417/7033)
3	91,82%	69,11%	90,67%(6377/7033)
4	91,37%	67,86%	90,16%(6341/7033)
5	87,92%	70,73%	90,81%(6387/7033)
6	88,07%	71,76%	91,17%(6412/7033)
7	88,90%	69,06%	90,34%(6354/7033)
8	90,85%	70,78%	91,15%(6411/7033)
9	86,95%	73,40%	91,55%(6439/7033)
10	91,45%	68,33%	90,34%(6354/7033)

Figure 7 shows some examples of false positives obtained with the proposed method. In many circumstances, the intensity contrast between adjacent regions does not correspond to the presence of a lumen. This is maybe a consequence that Haar features are sensitive to illumination changes. Variations on the lighting conditions may cause the cascade to detect lumen that was not predicted during the training stage. Likewise, in some images, folds of the intestinal wall may produce contrasted regions that confuse the Haar features. If new kind of images are presented to the classifier, detection is difficult and the amount of false positives increases. To deal with this problem, training data must include as many examples as possible to predict only true lumen.

B. Features analysis

One may reasonably ask if the proposed kind of features is optimal: may we obtain good classification results without one of these three kind of features? May we get away with only one kind? Adding some more elaborate Haar-like

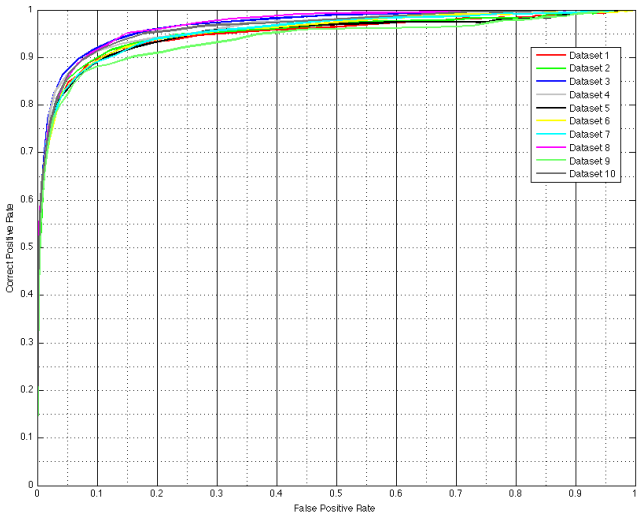


Figure 6. ROC curves for each dataset obtained by varying the stiffness threshold of each classifier from 0.1 to 1.

Figure 7. Example of some false positives detected by the system.

features is worth the gain in accuracy? The authors have tried to perform boosted classification using only one kind of feature among those proposed in this paper at each time. The results were only slightly different than those obtained using the whole set of features. This suggests that we might use only one kind of feature and achieve similar results. It is relevant to point out that the cross-shaped features have been introduced by the authors to improve not the results but the stability of the classifier. The availability of the whole set of features helps to keep down the number of classifiers in each node of the cascade. This happens because AdaBoost achieves more quickly the requirements fixed for the current classifier by the user. Also the number of nodes in the cascade is minimized. We can confirm that the use of additional features can only take effect on the structure of the classifier. The results would not be further significatively improved.

C. Comparing the boosted classifier with Support Vector Machine

The mean recall value we obtained using boosting is 89,5%. This result is efficiently attainable allowing a real-time performance. An interesting question is to compare the results provided by the boosting-based implementation with another "classic" classification method. The main problem in our data is the excessive dimensionality (63,063 features for each image to be classified). The high number of features suggests that comparison with other classification technique is fair only if these other techniques are adequate to handle these cases. For this reason, Support Vector Machine (SVM) is the ideal candidate for comparison. It is well know that SVM may easily deal with very high feature dimension; moreover, standard SVM implementation are available and this makes comparison easier and repeatable. SVM is a supervised learning algorithm used both for classification and regression. It indicates a binary classifier which projects the training samples in a multidimensional space looking for a separating hyperplane in this space. The hyperplane should maximize the margin, i.e., the distance from the closest training examples. SVM is well adapted to handle the curse of dimensionality and its performance has been tested in different application domains. We have considered the same data used in the previous experiments to train different SVMs using a linear kernel. We rely on a particular class of SVM called Least Squares SVM (LS-SVM). In this version it is possible to maximize the margin between support vectors by solving a linear equation with a least squares method. Classification results using this method are shown in Table IV. The superiority of the proposed boosting based technique is evident.

Table IV
CLASSIFICATION RESULTS USING SUPPORT VECTOR MACHINE.

Test Data	*Recall*	*Precision*	*Accuracy*
1	69, 92%	63, 84%	86, 79%(6104/7033)
2	71, 57%	63, 77%	86, 90%(6112/7033)
3	70, 82%	66, 39%	87, 67%(6166/7033)
4	69, 62%	64, 27%	86, 90%(6112/7033)
5	68, 79%	67, 58%	87, 83%(6177/7033)
6	70, 59%	65, 39%	87, 35%(6143/7033)
7	69, 17%	66, 14%	87, 44%(6150/7033)
8	70, 37%	65, 37%	87, 32%(6141/7033)
9	70, 37%	64, 11%	86, 92%(6113/7033)
10	72, 77%	63, 86%	87, 03%(6121/7033)

V. CONCLUSION

In this paper we introduced an automatic lumen detection algorithm for endoscopic images. Inspired by Viola-Jones object detection system, we show that using AdaBoost learning-based algorithm combined with a cascade of strong classifiers leads to a good rate of detection minimizing running time. Experimental results show that the proposed system detects positive images using exclusively Haar-like proposed features. Our detector is flexible and easily extensible to other semantic objects in endoscopic applications.

REFERENCES

[1] G. Gallo and A. Torrisi, "Boosted wireless capsule endoscopy frames classification," in *Proc. of Third International Conferences on Pervasive Patterns and Applications, PATTERNS'11*, September 25-30, 2011, Rome, pp. 25–30.

[2] G. Imaging, "Expanding the scope of gi," Last accessed: May 2012. [Online]. Available: http://www.givenimaging.com

[3] G. Iddan, A. Glukhovsky, and P. Swain, "Wireless capsule endoscopy," *Nature*, vol. 405, pp. 725–729, 2000.

[4] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Computer Vision and Pattern Recognition, CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1, 2001, pp. 511–518.

[5] K. Asari, "A fast and accurate segmentation technique for the extraction of gastrointestinal lumen from endoscopic images," in *Medical Engineering and Physics*, vol. 22, 2000, pp. 89–96.

[6] X. Zabulis, A. Argyros, and D. Tsakiris, "Lumen detection for capsule endoscopy," in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, September 22-26, Nice, 2008, pp. 3921–3926.

[7] P. Spyridonos, F. Vilarino, J. Vitria, F. Azpiroz, and P. Radeva, "Anisotropic feature extraction from endoluminal images for detection of intestinal contractions," in *Medical Image Computing and Computer-Assisted Intervention*, 2006, pp. 161–168.

[8] G. Gallo and E. Granata, "Lbp based detection of intestinal motility in wce images," vol. 7961, no. 1. SPIE, 2011, p. 79614T. [Online]. Available: http://link.aip.org/link/?PSI/7961/79614T/1

[9] J. Lee, J. Oh, S. K. Shah, X. Yuan, and S. J. Tang, "Automatic classification of digestive organs in wireless capsule endoscopy videos," in *Proceedings of the 2007 ACM symposium on Applied computing*, ser. SAC '07. New York, NY, USA: ACM, 2007, pp. 1041–1045. [Online]. Available: http://doi.acm.org/10.1145/1244002.1244230

[10] P. Viola and M. Jones, "Robust real-time object detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2002.

[11] F. C. Crow, "Summed-area tables for texture mapping," in *Proceedings of the 11th annual conference on Computer graphics and interactive techniques*, ser. SIGGRAPH '84. New York, NY, USA: ACM, 1984, pp. 207–212. [Online]. Available: http://doi.acm.org/10.1145/800031.808600

[12] R. E. Schapire, Y. Freund, P. Bartlett, and W. S. Lee, "Boosting the Margin: A New Explanation for the Effectiveness of Voting Methods," *The Annals of Statistics*, vol. 26, no. 5, pp. 1651–1686, 1998. [Online]. Available: http://dx.doi.org/10.2307/120016

[13] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," in *Proceedings of the Second European Conference on Computational Learning Theory*. London, UK: Springer-Verlag, 1995, pp. 23–37. [Online]. Available: http://portal.acm.org/citation.cfm?id=646943.712093

[14] Y. Freund and R. Schapire, "A short introduction to boosting," *J. Japan. Soc. for Artif. Intel.*, vol. 14, no. 5, pp. 771–780, 1999. [Online]. Available: citeseer.ist.psu.edu/freund99short.html

[15] P. Viola, M. Jones, and D. Snow, "Detecting pedestrians using patterns of motion and appearance," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, vol. 2, October 2003, pp. 734–741.

[16] L. Yun and Z. Peng, "An automatic hand gesture recognition system based on viola-jones method and svms," in *Computer Science and Engineering, 2009. WCSE '09. Second International Workshop on*, vol. 2, October 2009, pp. 72–76.

[17] M. Kolsch and M. Turk, "Analysis of rotational robustness of hand detection with a viola-jones detector," in *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, vol. 3, August 2004, pp. 107 – 110.

[18] M. C. Santana, O. Déniz-Suárez, L. Antón-Canalís, and J. Lorenzo-Navarro, "Face and facial feature detection evaluation - performance evaluation of public domain haar detectors for face and facial feature detection," in *VISAPP (2)*, 2008, pp. 167–172.

[19] OpenCV, "Open computer vision library," Last accessed: May 2012. [Online]. Available: http://opencv.willowgarage.com

[20] C. Papageorgiou, M. Oren, and T. Poggio, "A general framework for object detection," in *Computer Vision, 1998. Sixth International Conference on*, January 1998, pp. 555 – 562.

Believing Software: A Method of Practical Proof for Software Engineering

Jerry Overton
Computer Sciences Corporation (CSC)
St. Louis, Missouri, USA
joverton@csc.com

Abstract – **For years, software engineers have tried to achieve the same collective confidence in their software specifications that mathematicians, by way of proof, have in their theorems. Most attempts have been rooted in deduction and have produced methods that are too difficult to use in practice. By borrowing from mathematics its methods of recording, communicating, and scrutinizing arguments instead of its methods of deduction, we introduce a method practical proof in software engineering. The result of this work is a cost-effective method for getting consensus among practicing software engineers about the adequacy of a real-world software design.**

Keywords – Consensus, Proof, Software Engineering, Software Design Pattern, Practical Formal Method, POAD Theory.

I. INTRODUCTION

This paper is an elaboration of the ideas originally published in [1]. We expand on the method of practical mathematical reasoning in software engineering; provide a more detailed account of how the method can be used to argue for the adequacy of a real-world software system; and provide an extended analysis of the significance of this research.

One of the most distinguishing features of mathematics is the level of consensus among mathematicians about the truth or falsehood of their theorems [2]. Mathematicians, by way of proof, enjoy an unusually high collective confidence in their theorems. For years, software engineers have tried to achieve the same collective confidence in their software specifications [3]. So far, most attempts have been limited to verifying software using some form of deduction [4] – an approach rooted in the assumption that proof happens as a result of deductive calculation [2]. Deductive methods all have the same drawback: the cost (in time and effort) of using them to verify a software design is usually an order of magnitude greater than the cost of creating the design itself [5]. Deductive methods of verification are so expensive that, in practice, they are used only to reduce the risk of the most serious design flaws – flaws that may compromise human safety, for example [6].

In this work, we borrow proof from mathematics; use it to argue for the fitness-for-purpose of a software design; and do so in an amount of time that is within same order of magnitude that it took to create the design itself. But rather than assuming that proof is achieved through a series of deductive calculations, we adopt, instead, the view that proof is achieved by a gradual process of collective scrutiny and refinement [3]:

> *First of all, the proof of a theorem is a message. A proof is not a beautiful abstract object with an independent existence. No mathematician grasps a proof, sits back, and sighs happily at the knowledge that he can now be certain of the truth of his theorem. He runs out into the hall and looks for someone to listen to it. He bursts into a colleague's office and commandeers the blackboard. He throws aside his scheduled topic and regales a seminar with his new idea. … If the various proofs feel right and the results are examined from enough angles, then the truth of the theorem is eventually considered to be established.*

We borrow from math its methods of recording, communicating, and scrutinizing arguments – not its methods of deduction. First, we use Pattern-Oriented Analysis and Design (POAD) Theory [7], [8] to structure an adequacy argument based on software design patterns (the details of POAD Theory are given in Section IV, subsection B). Then, we use fuzzy inference to argue that the particular pattern instantiations in the design makes it fit for purpose. The result is what we will refer to as practical proof in software engineering: a cost-effective method for getting consensus among practicing software engineers about the adequacy of a real-world software design.

The rest of this paper is laid out as follows. We start by placing this work within the wider context of existing research on software design verification. Next, we specify the design for a collaborative wireless sensor network – the real-world problem of interest. We use POAD Theory to structure a proof-of-correctness argument for the design and calculation (based on fuzzy inference) to complete the argument. Finally, we close with an analysis of this work and conclusions about its significance.

II. STATE OF THE ART

The prior art for this research is the body of existing proof-of-correctness methods for computer programs. In software engineering, requirements are specifications proposed in the requirements phase and design is the specification proposed in the design phase. Proof-of-correctness happens when it is demonstrated that a design meets its requirements. Proof-of-correctness techniques reduce to a step-by-step reasoning for determining whether or not the design is fit for purpose [9]. Requirements dictate acceptable systems behavior by defining a mapping between

a set of pre-states and a set of post- states [10]. To satisfy a set of requirements, a design must take as input each pre-state and produce as output the prescribed post-state. Regardless of the specific technique, proof-of-correctness happens by process of refinement; where the original specifications of the requirements are replaced by the equivalent or stronger specifications of the design [10]. The body of existing proof-of-correctness methods is vast; however, they all work according to one of the three fundamental laws of refinement: refinement by steps, refinement by parts, and refinement by cases [10].

In refinement by steps, proof-of-correctness proceeds by sequential actions where, in each step, a part of the requirement specification is replaced by a suitable design. The refinement continues until all requirements have been interpreted as sequences of computational steps. In practice, proof-of-correctness techniques based on refinement by steps work by using semantic rules for interpreting requirements, specifying designs, and making comparisons between the two. For example, [11], [12], and [13] all develop competing formal semantics that makes it possible to prove (by steps) the correctness of designs documented in UML state chart diagrams.

In refinement by parts, an analyst normalizes the requirements into orthogonal parts, and then independently replaces each part with a suitable design element. In practice, proof-of-correctness using refinement by parts proceeds by normalizing requirement specifications into domains [14]. The requirements of each domain are replaced by designs represented by mathematical constructs – for example, partial functions as in [15]; actor-based models as in [16]; or by games between the environment and the system as in [17].

In refinement by cases, requirements are specified in terms of a correspondence between pre and post conditions. In Hoare Logic [18], for example, the central construct is the Hoare Triple that relates a pre-condition to a post-condition by way of a command. Refinement occurs by replacing a requirement with a design that achieves the same correspondence.

Existing proof-of-correctness methods (whether they use refinement by steps, parts, or case) require that requirements be replaced by suitable designs and that those replacements be justified by deductive implication [10]. As mentioned in the Introduction, deduction is expensive to use in the proof-of-correctness of real software systems – about an order of magnitude more expense that the cost of creating the design itself. Lightweight formal methods [5] are a way of compensating for the high cost; but instead of reducing the cost of deduction, lightweight formal methods simply limit its use. The central problem in the current state of the art remains – current methods of proof-of-correctness are too expensive for general use in real-world systems.

This research breaks from the state of the art by rejecting the restriction that deduction must be used to justify refinement. Instead, we will propose a proof-of-correctness technique based on Problem Oriented Software Engineering (POSE) [19] and software design patterns. The details of POSE are given in Section IV, subsection A. Deduction is one of many methods for justifying the substitution of requirements with engineering designs. We do not evaluate our method of justification by comparing it to deduction. Instead, in Section V, we evaluate our method of justification by determining whether or not it is logically sound (a standard more general than deduction).

POSE provides a framework for accepting engineering expertise as justification for replacing a requirement with a design. We complement POSE by using software design patterns as ready-made units of justification and engineering expertise. There are prior works that combine both formal methods and software design patterns. Most of these works (for example [20], [21], and [22]) offer proposals for formally representing software design patterns, but they do not offer methods for proof-of-correctness. The works that do offer proof-of-correctness methods (such as [23], [24], and [25]) do so based on deductive calculation; and, therefore, have the same drawbacks as the rest of the works surveyed.

POSE gives us the freedom to choose a more efficient method of reasoning. Software design patterns allow us to easily connect our arguments to the processes of collective scrutiny and feedback already in existence in the pattern community [26]. In the course of this research, we contribute to the state of the art a proof-of-correctness technique that is closer to real-world use of proof in mathematics [3]: rigorous arguments (but not deductive arguments) whose truth is established by a social process of scrutiny and feedback; arguments whose truth could be demonstrated by formal deduction if it were worth the time and effort.

III. A COLLABORATIVE SYSTEM DESIGN

In this section we introduce a real-world design for a software system. This design will be the target of analysis and proof-of-correctness in subsequent sections.

In collaborative systems, otherwise autonomous computing nodes cooperate to achieve a common task that would not be possible with any individual node acting alone [27]. Although the exact definition of a collaborative system can vary depending on context, in this paper, we focus on three defining characteristics:

- Nodes in collaborative systems are autonomous and spatially distributed.
- Task-execution responsibilities are distributed across multiple nodes.
- The communication links between nodes are decentralized and dynamic.

Figure 1 is an example of a collaborative system – a network of environmental sensor stations. The system is designed to report the environmental condition of a given

geographic region. Each sensor is capable of recording and reporting its local conditions, but to record and report the condition of the entire region requires all sensor stations to cooperate.

The nodes in the network are autonomous and spatially distributed across the region shown. Each sensor is capable of recording and reporting its local environmental conditions without the help of any of the other sensor stations. Task-execution is distributed across multiple nodes since reporting conditions for the entire region requires the cooperation of multiple sensor stations. The communication links between the sensor stations are decentralized and dynamic. Sensors can enter and leave the network at anytime. Every station is wirelessly connected to every other station, so no single sensor failure can disrupt the overall network connectivity.

Figure 1: Example Collaborative System [28].

In our system, we expect that node failures will be common and that the wireless communication links will be prone to frequent interruptions. For example, the sensor stations are exposed to adverse weather, they are knocked over and broken easily, and they can be expected to run out of power. People, cars, and animals passing between two sensor stations can cause a temporary loss of communication between them. If any of these things happen at the right time, a controller in a region may miss a sensor update and become out of touch with the current conditions in the region.

A robust design will allow the sensor stations (referred to from now on as nodes) to both detect and mitigate these kinds of failures. Each node must be designed to detect when other nodes become unresponsive; each node must be designed to perform in degraded mode when disconnected from the network; and the network must be capable of using node redundancy to compensate for the loss of any particular node. A satisfactory design must satisfy the following requirements.

Req. 1. *Group Communication*. Each node must be able to communicate with all other nodes and detect when a node becomes unresponsive.

Req. 2. *Fault Tolerance*. The network must be capable of using node redundancy to compensate for the loss of any particular node.

Req. 3. *Degraded Mode Operation.* Each node must be capable of performing limited functions while disconnected from the network, and be capable of resuming full function when network communication is restored.

Figure 2 shows the class diagram of our design for a robust collaborative system. We consider **Figure 2** to be the class diagram of a real-world design since it was taken from the design of an actual software system built to provide fault tolerance in collaborative systems [29]. Each *GroupNode* operates in its own thread of execution. Each node gets its ability to collaborate through an association with a *CommStrategy* object. The *CommStrategy* has an association back to its *GroupNode* in case the *GroupNode* needs to be notified of events from the *CommStrategy*. The *PushPullNode* (which, represents a sensor or controller) is a specific type of *GroupNode*. The *PushPullStrategy* is a specific type of *CommStrategy*. Using the JGroup communication API [30] the *PushPullStrategy* gives each *PushPullNode* the ability to communicate with other *PushPullNodes*.

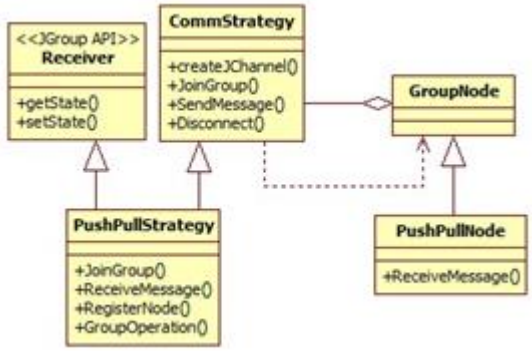


Figure 2: Class diagram of a design for a robust collaborative system.

Figure 3 is a sequence diagram of how nodes participate in group operations. Sensors A and B are controlled by the Controller. Sensors A and B join the same group representing a single physical zone. The Controller relies on both sensor A and sensor B to report temperature for a given region. The controller doesn't care which sensor it uses as long as at least one of them is always available. When the Controller wants a temperature reading from the zone, it joins the zone's group and executes *CommStrategy.groupOperation()*. JGroups elects a leader within the group and calls *getState()* on that node (let's assume that sensor A was chosen). The *getState()* operation of sensor A takes a temperature reading and sets the reading as the operation's return value. JGroups then calls *setState()* on the Controller, passing it the temperature reading from sensor A. In subsequent requests for the zone temperature, if sensor A becomes unresponsive, JGroups will failover to sensor B.

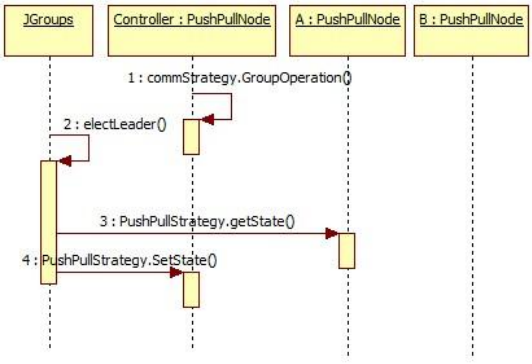


Figure 3: Nodes participating in a group operation

We have a design, but is it a good one? Does it solve our problem and satisfy our requirements? In the remainder of this paper, we will construct a proof-of-correctness-argument for the design.

IV. OUR METHOD FOR PROOF OF CORRECTNESS

A. Our Approach: The Basis

The method that we use in the next section to structure our proof-of-correctness argument (POAD Theory) is based on a system of reasoning known as Problem-Oriented Software Engineering (POSE) [19]. In POSE a software engineering problem has context (a real-world environment), W; a requirement, R; and a solution (which, may or may not be known), S. We write $W, S \vdash R$ to indicate that we intend to find a solution S that, given a context of W, satisfies R. Details about an element of the problem can be captured in a description for that element; and a description can be written in any language (UML in our case) considered appropriate. The problem, P_0 , of designing a collaborative system can be expressed in POSE as:

$$CSystem: W, S \vdash R \qquad (1)$$

where W is the real-world environment for the system (shown in **Figure 1**); S is the system itself and R are requirements Req. 1, Req. 2, and Req. 3. Equation (1) says that we can expect to satisfy R when the system S is applied in context W.

In POSE, engineering design is represented using a series of problem transformations. Transformation steps can be arbitrary in size; large steps can be composed of smaller ones. A problem transformation is a rule where a conclusion problem $P: W, S \vdash R$ is transformed into premise problems $P_i: W_i, S_i \vdash R_i, i = 1, \dots, n\ (n > 0)$ using justification J and a rule named N, resulting in the transformation step $\frac{P_1 \dots P_n}{P}\ \frac{[N]}{\ll J \gg}$. This means that S is a solution of $W, S \vdash R$ whenever $S_i, \dots, S_n$ are solutions of $W_i, S_i \vdash R_i, \dots, W_n, S_n \vdash R_n$. The justification J collects the evidence of adequacy of the transformation step and is validated by all relevant stake-holders. Through the application of rule N_i, problems are transformed into other problems that may be easier to solve or that may lead to other problems that are easier to solve. These transformations occur until we are left only with problems that we know have a solution fit for the intended purpose. POSE allows us to use one big-step transformation to represent several smaller ones. We can apply big-step transformations without having completed justification, with the understanding that we will complete the justification later and solve our problem. The progression of a software engineering solution described by a series of transformations can be shown using a development tree.

$$\frac{\dfrac{\overline{P_3: W_3, S_3 \vdash R_3} \quad P_4: W_4, S_4 \vdash R_4 \quad [N_2]}{P_2: W_2, S_2 \vdash R_2 \ll J_2 \gg} \quad [N_1]}{P_1: W_1, S_1 \vdash R_1 \ll J_1 \gg} \qquad (2)$$

In the tree, the initial problem forms the root and problem transformations extend the tree upward toward the leaves. There are four problem nodes in the tree: P_1, P_2, P_3, and P_4. The problem transformation from P_1 to P_2 is justified by J_1; the transformation from P_2 to P_3 and P_4 is justified by J_2. The bar over P_3 indicates that P_3 is solved. Because P_4remains unsolved, the adequacy argument for the tree (the conjunction of all justifications) is not complete, and the problem P_1 remains unsolved. A complete and fully-justified problem tree means that all leaf problems (in this case P_3 and P_4) have been solved.

For the sake of clarity, we will show the context, solution, and requirement of a problem only when necessary to understanding a given transformation. In many of the subsequent equations, these details are omitted and only the problem's name is shown.

B. Our Approach: Practical Mathematics

In this section, we introduce POAD theory and use it to structure the argument that the design from Section III is fit-for-purpose.

A software design pattern is a tool that a software engineer can use to take a complex, unfamiliar problem and transform it into simpler, more familiar ones [31]. The basis of POAD Theory is that software engineering design can be represented as a series of transformations from complex engineering problems to simpler ones, and software design patterns can be used to justify those transformations:

$$\frac{SimplerProblem}{ComplexProblem}\ \frac{[SolInt]}{\ll Pattern_1, \dots, Pattern_n \gg} \qquad (3)$$

In (3) the patterns $Pattern_1, \dots, Pattern_n$ are used to justify the transformation from the $ComplexProblem$ to the

$SimplerProblem$. The engineering expertise documented in the Object Group pattern describes how to achieve reliable multicast communication among objects in a group [32]. The pattern gives us justification for substituting $CSystem$ with the easier problems of implementing a communication mechanism ($Comm$) and implementing an object that uses the communication mechanism (Obj). We write this as

$$\frac{Comm\ Obj}{CSystem: W, S \vdash R} [SolInt] \ll OG \gg \quad (4)$$

which, means that we used the engineering expertise in the Object Group pattern (represented as $\ll OG \gg$) to justify a solution interpretation (represented by the rule [$SolInt$]) from $CSystem$ to $Comm$ and Obj.

But there is a problem. Equation (4) implies that if we have a solution to $Comm$ and Obj then we also have a solution to $CSystem$. Having a communication mechanism that allows for reliable multicast communication and objects capable of communicating that way may be sufficient to argue that the solution can satisfy the group communication (Req. 1) and fault tolerance requirements (Req. 2); but the solution does not address the requirement that the objects be capable of degraded mode operation (Req. 3).

We can add to our solution as many transformations as necessary. We can add to (4), a transformation justified by the Explicit Interface pattern [33].

$$\frac{Comm\ \frac{Intf\ Node}{Obj} [SolInt] \ll EI \gg}{CSystem: W, S \vdash R} [SolInt] \ll OG \gg \quad (5)$$

The Explicit Interface pattern describes how to achieve separation between an object and its environment [33]. We can use that separation to argue that the nodes in our design can function even when disconnected from each other.

Equation (5) is a solution tree with $CSystem$ at the root. Two problem transformations extend the tree upward toward the leaves $Comm$, $Intf$, and $Node$. The equation structures an argument whose adequacy is established by the conjunction of all justifications – in this case by the engineering expertise contained in the Object Group pattern $\ll OG \gg$ and the engineering expertise contained in the explicit interface pattern $\ll EI \gg$. A solved problem is written with a bar over it; for example, if the Object Group pattern were sufficient to convince us that we have an adequate communication mechanism, then we could rewrite (5) as follows

$$\frac{\overline{Comm}\ \frac{Intf\ Node}{Obj} [SolInt] \ll EI \gg}{CSystem: W, S \vdash R} [SolInt] \ll OG \gg \quad (6)$$

where the bar over $\overline{Comm}$ indicates that we have sufficient justification to consider that problem solved. A complete and fully-justified problem tree means that all leaf problems – for (5), the leaves are $Comm$, $Intf$, and $Node$ – have been solved. We complete the problem tree in (5) by adding transformations and justifications sufficient to solve all leaf problems.

$$\frac{\overline{Recvr}}{Comm} [SolInt] \ll J_1 \gg \quad \frac{\overline{PPStrat}}{Intf} [SolInt] \ll J_2 \gg \quad \frac{\overline{PPNode}}{Node} [SolInt] \ll J_3 \gg \quad (7)$$

Equation (7) continues the solution from (5) by providing solutions for all leaf problems in (5). In (7) the problems $Recvr$, $PPStrat$, and $PPNode$ correspond to the *Receiver*, *PushPullStrategy* and *PushPullNode* (from **Figure 2**) respectively. Each leaf problem from (7) is a design implementation of the patterns chosen in (5). The $Recvr$is an implementation of the communication mechanism prescribed by the Object Group pattern, $PPStrat$ is an implementation of the interface prescribed by the Explicit Interface pattern, and $PPNode$ is an implementation of the domain object prescribed by the Explicit Interface pattern. By considering (5) in combination (7), we can conclude that, given sufficient justification (J_1, J_2, and J_3), we can consider our original problem ($CSystem$ from (5)) solved. In other words, once we find J_1, J_2, and J_3, we will have a complete proof-of-correctness argument for the design described in Section III.

C. Our Approach: Practical Calculations

So far, we have a general argument for how to use software design patterns to solve our problem, but it isn't clear how this general argument relates to our specific design. In this section, we introduce a method of calculation – based on Fuzzy Inference [34] – that connects our more general argument to the specific design decisions represented by the *Receiver*, *PushPullStrategy* and *PushPullNode* elements of **Figure 2**. We use the calculation results as the justification (J_1, J_2, and J_3) needed to complete the proof-of-correctness argument for (5) and (7).

Fuzzy inference is based on a generalized modus ponens [34] where arguments take the form:

$$\begin{array}{c} If\ A\ Then\ B \\ A' \\ Therefore\ B' \end{array} \quad (8)$$

For example, suppose we accepted the general rule that: *if the Object Group pattern were well implemented as part of our collaborative system, then the group communication of*

our system would be good. If we knew that, in our system, the Object Group pattern were implemented poorly, then fuzzy inference would allow us to conclude that the group communication of the system would also be poor.

We apply fuzzy inference to statements about the use of software design patterns to create a technique for calculating the results of software design decisions. In works such as [35], [36], and [37] fuzzy logic has been used in combination with design patterns to reverse engineer a design from source code. A software design pattern can have several different implementations. These works use fuzzy inference to determine if an existing solution, known to satisfy certain requirements, matches a general design pattern. We apply this same idea, but in reverse: for a given design pattern, we use fuzzy inference to determine if a particular implementation of that pattern will lead to a solution that we can trust will satisfy particular requirements. For all fuzzy logic operations (such as creating fuzzy input variables, performing fuzzy inference, and visualizing fuzzy output variables), we used Mathematica's Fuzzy Logic Environment [38].

We begin our calculation by creating fuzzy rules [34] that represent the design constraints introduced by (5) and (7):

Rule 1. If the object group pattern is implemented then group communication will be good

Rule 2. If the object group pattern is not implemented then fault tolerance will be low

Rule 3. If the explicit interface pattern is not implemented then degraded-mode operation will not be enabled

Rule 4. If the push pull node communicates statically then degraded-mode operation will not be enabled

Rule 5. If the push pull node communicates dynamically and the explicit interface pattern is implemented then degraded-mode operation will be enabled

Rule 6. If the push pull node communicates dynamically and the object group pattern is implemented then group communication will be good and fault tolerance will be high.

Each rule makes statements concerning input and output variables. Each variable has membership functions [34] that allow the inference engine to turn the numeric values of the variables into the more intuitive concepts used in Rules 1-6. For example, **Figure 4** shows the three membership functions (Poor, Good, and Moderate) for the Group Communication output variable. From the shape of the membership functions, we can see that a Group Communication variable with a value of 0.7 would be considered mostly moderate, slightly good, and not at all poor.

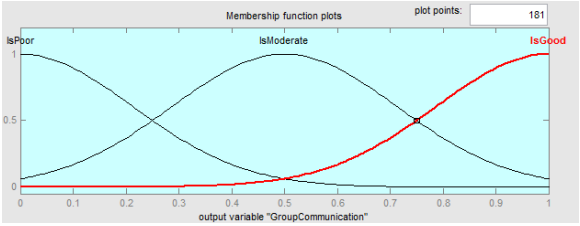


Figure 4: Membership function for the Group Communication output variable

The fuzzy rules capture our understanding of how the software engineering expertise contained in $\ll OG \gg$ and $\ll EI \gg$ (from (5)) relates to the original requirements R of (1). In our calculation, $Recvr$, $PPStrat$, and $PPNode$ (from (7)) are represented using input fuzzy variables and Req. 1, Req. 2, and Req. 3 (from (1)) are represented using output fuzzy variables. As shown in **Figure 5**, input variables representing our implementation choices are fed into an inference engine which, has been loaded with Rules 1-6. The inference engine produces values for the fuzzy output variables which, represent the results of our calculation.

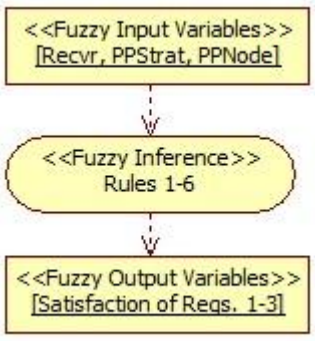


Figure 5: Process flow of the simulation.

We calculate the design choices made in (7) by assigning specific values to the fuzzy input variables $Recvr$, $PPStrat$, and $PPNode$. Because JGroups provides a faithful implementation of the Object Group pattern, the $Recvr$ provides an almost-complete implementation (0.949) of the Object Group pattern's communication mechanisms. The $PPNode$ is a reasonably good approximation (0.762) of the Object Group's node element; but the communication strategy provided by the $PPStrat$ is not a very good representation (0.584) of intent of the Explicit Interface pattern. The $PPStrat$ object separates from $PPNode$ the details of group communication, but, unlike a true explicit interface, still requires $PPNode$ to select an appropriate instance of $PPStrat$ based on the current circumstances.

The results of the calculation predict that the design decisions described in (7) will result in a collaborative system that satisfies Req. 1-3 (see **Figure 6**). The results of the calculation are that the system will have good group communication (0.833), good fault tolerance (0.815), and will operate well in degraded mode (0.807).

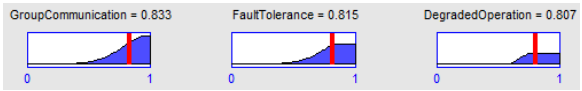


Figure 6: The results of collaborative system calculation.

If we compare every possible design choice to its corresponding calculated result, we get a design space that shows how design choices affect the quality of the system. **Figure 7** shows the design space for achieving the desired fault tolerance. There are a number of design choices for the *Recvr* and *PPNode* elements (shaded in yellow) that will result in acceptable (0.7 or greater) fault tolerance for the system. The design space shows that fault tolerance is most dramatically affected (indicated by the surface's steep drop-off) by the design of the *Recvr* – which, makes sense because that portion of the design determines the group communication capabilities.

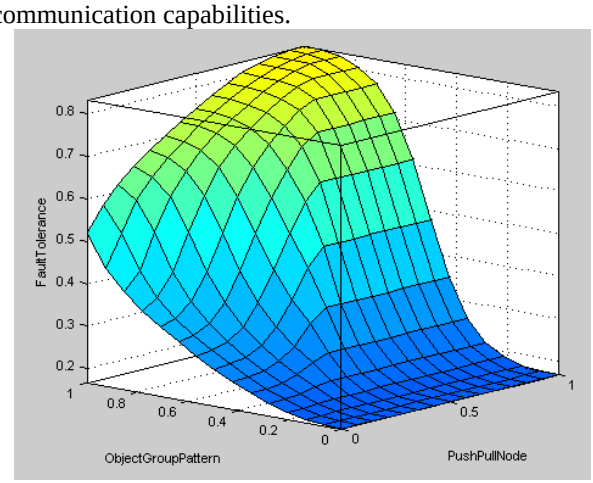


Figure 7: Fault tolerance design space

Figure 8 shows the design space for achieving the desired degraded-mode operation for each node in the collaborative system. The number of acceptable design choices for the *PPStrat* and *PPNode* are more limited than the choices available in the design space of **Figure 7**. The choice of design for *PPNode* seems to be slightly more influential to degraded-mode operation than the design choices for *PPStrat*.

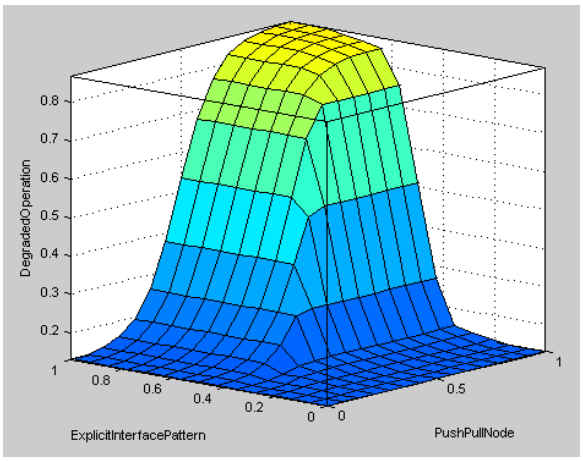


Figure 8: Degraded-mode operation design space

The positive calculation results (which, are also confirmed by our analysis of the design spaces) provides the justification (J_1, J_2, and J_3), needed to complete the proof-of-correctness argument for the design of **Figure 2**.

The argument is complete, but is it trustworthy? Can we expect the method of argument described here to be sufficient to build consensus among practicing software engineers that our design meets its requirements? In the remainder of this paper, we perform a critical analysis with the goal of answering these questions.

V. ANALYSIS AND CONCLUSIONS

We have proposed a method of proof-of-correctness for software design. Keep in mind that by proof-of-correctness, we mean some method for convincing our audience that a design meets its requirements. We started with the real-world problem of designing a collaborative system. We used POAD Theory to create a general argument; we used software design patterns to justify the argument; and we used calculation to apply the general argument to our specific design. Our goal was to introduce a cost-effective method for getting consensus among practicing software engineers. We analyze whether or not our method accomplishes our goal by considering the following questions: is our proposed method trustworthy, is it convincing, is it practical?

Is our method trustworthy? We can consider our method trustworthy if we can show that it is sound: given premises that can be trusted, our method will produce conclusions that can be trusted. Our method consists of a general argument based on POAD Theory and specific calculations based on Fuzzy Logic. POSE transformations – the basis of POAD Theory – are sound. Premise problems can only be interpreted as conclusion problems given sufficient justification for doing so. The original problem is considered solved only after all leaf-level sub-problems are known to be solved. In POAD Theory a solved problem is made only of known-solved sub-problems; and the break-

down of problems into constituent sub-problems is fully-justified. Generalized modus ponens – the basis of fuzzy inference – is also sound in that its conclusions are true if the premises are true [39]. We can trust the results of our calculation as long as we trust the rules that we establish for governing the simulation.

Is our method convincing? Whether or not the particular argument given by (5), (7) and the justification from Section IV, Subsection C is convincing will depend on the results of a social process among practicing software engineers. The argument will have to generate interest and credibility among some initial group of engineers. It will have to be circulated among a wider audience, polished and refined. A truly convincing argument will be internalized by engineers. That is, practicing software engineers may attempt to use parts or all of the argument to justify designs of their own; or the design itself will be routinely copied and used in other working IT software systems. We can, however, determine if our general proof-of-correctness method is capable of producing convincing arguments. We can compare the methods described here to the method of proof used in mathematics – the social process of scrutinizing humanly understandable (as opposed to purely formal) arguments [40]. Therefore, we focus the analysis of whether or not our method is convincing by asking, instead: *does our method encourage the creation and collective scrutiny of understandable arguments?*

Our method is, essentially, an application of analogical reasoning – one of the basic patterns of human reasoning [41]. Our method makes arguments understandable by replacing the more difficult task of predicting the consequences of a design with the much easier task of comparing a design with known software design patterns. The calculations of Section IV, Subsection C draw a comparison between the design of Section III and the interaction of software design patterns given by (5) and (7). We reason that the closer our design is to the solutions described in the design patterns, the closer our results will be to the consequences described in the design patterns. Our calculation tells us just how close our design needs to be in order to produce satisfying results. POAD Theory allows us to record the argument so that it can be read, circulated, and scrutinized (as evidenced by this publication). Further, using software design patterns, we build on the processes of collective scrutiny and feedback already in existence in the pattern community [26].

Is our method practical? With a relatively small amount of effort (roughly the same amount of time it took to create the original design), we were able to use math and calculation to discover things about the design that are not obvious. With Eq. 1-7 and the associated explanatory text, we were able to create a mathematical model that had a meaningful correspondence to the collaborative system design in Section III. We were able to use those equations to structure an argument for the design's adequacy and to predict that: given the argument structure defined by (5) and (7); and the engineering expertise contained in the Object Group and Explicit Interface patterns; all we needed to validate the design of Section III was to find justifications J_1, J_2, and J_3. Using Rules 1-6; fuzzy variable membership function definitions (the membership function for the Group Communication output variable is shown in **Figure 4**); and fuzzy inference; we were able to simulate the effect that the design choices of (7) would have on the resulting system qualities (shown in **Figure 6**).

Although we are able to argue that the method we describe here is capable of practical proof-of-correctness, we consider it to be a greater accomplishment to demonstrate that particular arguments based on this method are convincing. That is, our ultimate goal is to produce arguments that are trusted enough to become the infrastructure for a particular field of endeavor in software engineering. We recognize that gaining consensus and confidence in an argument will likely requirement more than just argument creation and discussion. We recognize the need to empirically demonstrate the ability of our proposed methods. At CSC, we are currently using this method to predict and manage risk in large-scale data center migration. Effectively managing risk for such large-scale endeavors requires high levels of consensus and coordination among migration teams. The methods described in this research are being used to identify ideas that are most likely to result in a better understanding and mitigation of the risk factors involved. We are exploring whether or not our methods are effective in identifying a set of ideas that a community of CSC engineers can rely on to improve performance in some of our most complex projects.

ACKNOWLEDGMENT

We would like to thank Dariusz W. Kaminski of the Marine Scotland directorate of the Scottish Government for his insightful review and commentary.

REFERENCES

[1] J. Overton. *Practical Math and Simulation in Software Design.* Proceedings of the Third International Conferences on Pervasive Patterns and Applications (Computation World 2011). 2011.

[2] R. Hersh. *What is Mathematics, Really?* Oxford University Press, New York, Oxford, 1997.

[3] R. De Millo, R. Lipton, and A. Perlis. *Social Processes and Proofs of Theorems and Programs*, Communications of the ACM, Volume 22, Number 5, 1979.

[4] G. Holzmann. *Trends in Software Verification.* Proceedings of the Formal Methods Europe Conference (FME'03). 2003.

[5] D. Jackson. *Lightweight Formal Methods.* FME 2001: Formal Methods for Increasing Software Productivity, Lecture Notes in Computer Science, Volume 2021, 2001

[6] J.P Bowen. *Formal Methods in Safety-Critical Standards*. In Proceedings of 1993 Software Engineering Standards Symposium

(SESS'93), Brighton, UK, IEEE Computer Society Press, pages 168-177, 1993.

[7] J. Overton, J. Hall, L. Rapanotti, and Y. Yu. *Towards a Problem Oriented Engineering Theory of Pattern-Oriented Analysis and Design*. In Proceedings of 3rd IEEE International Workshop on Quality Oriented Reuse of Software (QUORS), 2009.

[8] J. Overton, J. G Hall, and L. Rapanotti. *A Problem-Oriented Theory of Pattern-Oriented Analysis and Design.* 2009, Computation World: Future Computing, Service Computation, Cognitive, Adaptive, Content, Patterns, pages 208-213, 2009.

[9] W. Adrion, M. Branstad, and J. Cherniavsky. *Validation, Verification, and Testing of Computer Software.* ACM Computing Surveys, Vol. 14, No.2, pages 159-192, June 1982.

[10] E. Hehner. *A Practical Theory of Programming.* Springer-Verlag, New York, 1993.

[11] D. Latella, I. Majzik and M. Massink. *Towards A Formal Operational Semantics of UML Statechart Diagrams.* Third International Conference on Formal Methods for Open Object-Oriented Distributed Systems, pages 331-347, Kluwer Academic Publishers, 1999.

[12] D. Alexandre, M. Moller, and W. Yi. *Formal Verification of UML Statecharts with Real-time Extensions*. Fundamental Approaches to Software Engineering, 5th International Conference, FASE 2002, volume 2306 of LNCS, pages 218–232. Springer–Verlag, 2002.

[13] G. Kwon. *Rewrite Rules and Operational Semantics for Model Checking UML Statecharts*. Proceedings of the 3rd International Conference on UML, Lecture Notes Comp. Sci. 1939, pages 528–540, 2000

[14] D. Scott. *Domains for Denotational Semantics.* In Proceedings of ICALP, 1982.

[15] R. Keller. *Formal Verification of Parallel Programs.* Communications of the ACM Volume 19, No. 7 pages 371-384. 1976

[16] M. Sirjani, A. Movaghar, A. Shali, and F. de Boer. *Modeling and Verification of Reactive Systems using Rebeca*. Fundamenta Informaticae, pages 385–410, 2004.

[17] S. Abramsky, D. R. Ghica, A. S. Murawski, and C.-H. L. Ong. *Applying Game Semantics to Compositional Software Modeling and Verification*. In TACAS'04, volume 2988 of Lecture Notes in Computer Science, pages 421–435, 2004.

[18] C. Hoare. *An Axiomatic Basis for Computer Programming.* Communications of the ACM, Volume 12, No. 10, pages 576-583, 1969.

[19] J. G. Hall, L. Rapanotti, and M. Jackson. *Problem-Oriented Software Engineering: Solving the Package Router Control Problem.* IEEE Trans. Software Eng., 2008. doi:10.1109/TSE.2007.70769

[20] T. Taibi and D. Ngo. *Formal Specification of Design Pattern Combination Using BPSL*, Information and Software Technology 45, Elsevier, pages 157–170, 2002.

[21] N. Soundarajan and J. Hallstrom. *Responsibilities and Rewards: Specifying Design Patterns*, Proceedings of the 26th International Conference on Software Engineering (ICSE'04), pages 666-675, May 2004.

[22] P. Alencar, D. D. Cowan, and C. J. P. Lucena. *A Formal Approach to Architectural Design Patterns*, Proceedings of the 3rd International Symposium of Formal Methods Europe, pages. 576-594, 1995.

[23] D. J. Ram, P. J. K. Reddy, and M. S. Rajasree. *An Approach to Estimate Design Attributes of Interacting Patterns.* http://dos.iitm.ac.in/djwebsite/LabPapers/JithendraQAOOSE2003.pdf, Last Accessed: 30 January 2011.

[24] J. Paakki, A. Karhinen, J. Gustafsson, L. Nenonen, and A. Verkamo. *Software Metrics by Architectural Pattern Mining.* In Proceedings of the International Conference on Software: Theory and Practice (16th IFIP World Computer Congress), peges 325–332, 2000.

[25] P. Tonella and G. Antoniol. *Object Oriented Design Pattern Inference.* In Proceedings of the IEEE International Conference on Software Maintenance. IEEE Computer Society Washington, DC, USA, 1999.

[26] N. Harrison. *The Language of Shepherds.* http://hillside.net/plop/plop99/proceedings/harrison/shepherding4.pdf, Last Accessed: 06/21/2012.

[27] T. Clouqueur, K.K. Saluja, and P. Ramanathan. *Fault Tolerance in Collaborative Sensor Networks for Target Detection*. IEEE Transactions on Computers. Vol. 53, No. 3, pages 320-333, March 2004.

[28] http://www.citysense.net, Last Accessed: 1/27/2012

[29] J. Overton. *Collaborative Fault Tolerance using JGroups*. Object Computing Inc. Java News Brief, 2007, http://jnb.ociweb.com/jnb/jnbSep2007.html, Last Accessed 02/05/2012.

[30] The JGroups Project. http://www.jgroups.org/. Last Accessed 06/21/2012

[31] F. Buschmann, K. Henney, and D. Schmidt. *Pattern-Oriented Software Architecture: On Patterns and Pattern Languages*, Volume 5. John Wiley & Sons, West Sussex, England, 2007.

[32] S. Maffeis. *The Object Group Design Pattern.* In Proceedings of the 1996 USENIX Conference on Object-Oriented Technologies, (Toronto, Canada), USENIX, June 1996.

[33] F. Buschmann, K. Henney, and D. Schmidt. *Pattern-Oriented Software Architecture: A Pattern Language for Distributed Computing (Wiley Software Patterns Series)*, Volume 4. John Wiley & Sons, 2007.

[34] K. Tanaka. *An introduction to Fuzzy Logic for Practical Application.* Berlin: Springer, 1996.

[35] J. Niere. *Fuzzy Logic Based Interactive Recovery of Software Design*. Proceedings of the 24th International Conference on Software Engineering, Orlando, Florida, USA, pages 727-728, 2002.

[36] C. De Roover, J. Brichau, and T. D'Hondt. *Combining Fuzzy Logic and Behavioral Similarity for Non-strict Program Validation.* In Proceedings of the 8th Symposium on Principles and Practice of Declarative Programming, pages 15–26, 2006.

[37] I. Philippow, D. Streitferdt, M. Riebisch, and S. Naumann. *An Approach for Reverse Engineering of Design Patterns.* Software Systems Modeling, pages 55–70, 2005.

[38] http://www.wolfram.com/products/applications/fuzzylogic/, Last Accessed: 06/23/2012

[39] S.J. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Englewood Cliffs, NJ: Prentice Hall, 1995.

[40] W. Thurston. *On Proof and Progress in Mathematics.* Bulletin of the American Mathematical Society. Volume 30, pages 161-177, 1994.

[41] G. Polya. *Mathematics and Plausible Reasoning: Volume II, Patterns of Plausible Inference.* Princeton University Press. 1968.

Concept, Design and Evaluation of Cognitive Task-based UAV Guidance

Johann Uhrmann & Axel Schulte
Institute of Flight Systems
Universität der Bundeswehr München
Munich, GERMANY
{johann.uhrmann|axel.schulte}@unibw.de

***Abstract*—This paper discusses various aspects of automation for the integration of multiple, detached, unmanned sensor platforms into a military helicopter scenario. The considered scenario incorporates operating over unknown, potentially unsafe terrain including ad-hoc mission orders issued to the crew even during flight. Unmanned sensor platforms provide mission-relevant real-time reconnaissance and surveillance information to the crew and therefore lead to an increase in mission performance. To achieve this, the UAVs (Uninhabited Aerial Vehicles) shall be automated beyond the level of commonly used systems, i.e., autopilots and waypoint guidance. Instead the human operator shall be enabled to transfer authority to the unmanned platforms in a well-defined manner just like in tasking human subordinates. Automatic task execution is achieved by installing knowledge-based and goal-driven agents based on artificial cognition on the unmanned platforms for planning and decision-making. These agents allow the human operator to assign tasks to the UAVs on an abstraction level which is comparable to the supervision of human subordinates within a mission. This paper presents the concept and design of such artificial cognitive agents. A novel views on levels of automation will be discussed. The required knowledge driving the cognitive automation will be explained and the results of the evaluation of the system with subject matter experts will be discussed. The results, which include measures of the overall mission performance, operators' interaction, behaviour, workload, situation awareness and acceptance ratings, indicate that task-based UAV guidance is feasible, accepted and beneficial in military helicopter operations.**

Keywords - task-based guidance; goal-driven behaviour; artificial cognititive units; artificial cognition; level of automation

I. INTRODUCTION

The utilization of UAVs (Uninhabited Aerial Vehicles) as detached sensor platforms of a manned helicopter in a military scenario promises to enhance mission safety and effectiveness by allowing the crew to deploy sensors in dynamic and uncertain environments without exposing personnel to potential threats more than needed. Using unmanned vehicles for this purpose requires a change in the UAV guidance paradigm that enables a single human operator to control one or even multiple UAVs while being the commander of a manned aircraft. If those detached platforms were controlled by humans, a commander would just assign tasks referring to the mission context and the current situation and leave the details of task execution as well as the application of domain knowledge generating local tactical behaviours to the human subordinate. A way to incorporate this leadership concept in the guidance of uninhabited vehicles using such knowledge in a machine agent and the evaluation of a corresponding experimental system are first time described in [1]. A final evaluation experiment as described in [1] took place in May 2011. This paper extends the findings provided by [1] and describes the overall concept of task-based guidance, the system architecture, the knowledge base and the evaluation in more detail.

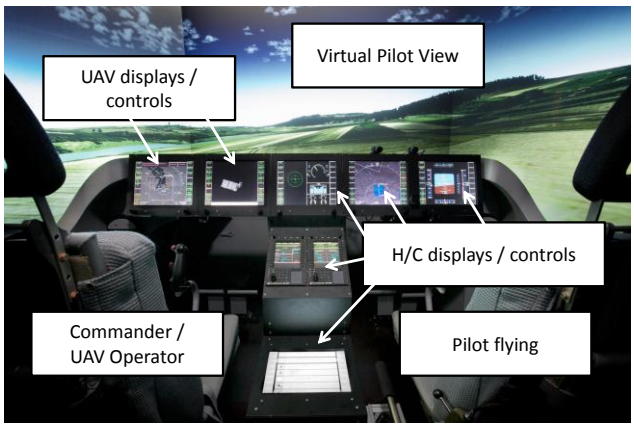


Figure 1. Helicopter simulator of the Institute of Flight Systems

Some current research approaches concerning UAV guidance allow the definition of scripts or plays [2] to define action sequences for one or multiple UAVs. Moreover, some of these systems also react to changes in the situation like a new threat along a flight route [3]. However, the resulting behaviours of these systems are solely defined at design-time. The underlying goals of the UAVs are not explicitly expressed in the system but are implicitly encoded in the implementation of the behaviours. With implicit goals, the system "simply makes guesses – statistically plausible guesses based on the designer's observations and hunches." [4]. This paper describes the system architecture that avoids most of the "guessing" by the application of knowledge and goals driving task-based, cooperative and cognitive UAV automation. Furthermore, various metrics that can be applied to automation of UAVs are presented. The resulting type of supervisory control shall avoid at least some of the issues of conventional automation by taking a step towards human-centred automation [5]. The resulting laboratory prototype has been integrated in the helicopter research flight simulator of the Institute of Flight Systems at the Universität der Bundeswehr München, which is shown in Figure 1, and evaluated in experiments with experienced German Army aviators. In these experiments, the pilots had to perform several, dynamic troop transport missions including an unscheduled combat recovery task

with the support of the manned helicopter and three tactical UAVs.

The following sections present related work in the field of UAV guidance and the concepts behind the task-based guidance approach in general as well as its application to UAVs. Section IV illustrates different measures of automation in the domain of unmanned vehicles. Section V presents the system architecture of a simulation prototype including a short introduction into the concept of artificial cognitive units (ACUs) and a description of the knowledge base of a UAV. Finally, Section VI contains the description, measures and results of an experimental evaluation of the concept of task-based guidance.

II. Related Work

Most current research projects in the area of UAV guidance and mission management focus on solving problems in the field of trajectory generation [6] and management and the achievement of what is mostly referred to as "full autonomy" by the application of control algorithms [7].

This research concentrates on optimizing mission effectiveness, e.g., time or fuel consumption, within a given constraint set. However, such constraint sets and parameters are either static or the definition is left to the human operator or the experimenter. If the handling and monitoring of the control algorithms of multiple UAV is allocated to the commander of a manned helicopter, then the result is error-prone behaviour and high workload for the operator [8, 9]. Therefore, we present a system that integrates flight management, payload control and data links into one entity of automation. This entity uses its knowledge about the situation, the mission, the vehicle and its capabilities to provide an interface to the human operator that allows UAV guidance on a situation adaptive task level rather than sub-system handling. Instead of optimising isolated algorithms or use-cases, this approach aims for the integration of multiple unmanned vehicles into a highly dynamic military mission while allowing the commander of a manned helicopter to use the UAV capabilities at an abstraction level similar to commanding human subordinates, i.e., additional manned helicopters.

Previous publications focus on the requirements engineering [9] and global system design and test environment including the integration of assistant systems [10–12]. Moreover, [13] provides a detailed description of the software framework used in this work. This framework is currently undergoing a major redesign to reflect the feedback from various applications [14]. The main contribution of this paper consists of a discussion of the foundations of task-based guidance, its implementation for UAV guidance, the resulting levels of automation on various scales, a detailed description of the knowledge base as well as the experimental evaluation of the system.

III. Task-based Guidance

The concept of task-based guidance by sharing authority and common goals was first described by the military strategist Sun Tzu [15] around 500 BC. He noted the importance of sharing and pursuing common goals among all ranks to be successful. Consequently, the guidance of subordinates should not just consist of instructions but also include the reason and the objectives.

A. Concept of tasks

In this paper we define task as the combination of (1) a goal to achieve and (2) a transfer of authority to a subordinate in order to achieve that goal. Therefore, issuing tasks to a subordinate (who may be human or artificial agents) has several implications and requirements to the subordinate as well as to the supervisor.

Miller [16] lists six requirements for delegation relationships:

1. "The supervisor retains overall responsibility for the outcome of the work…" as well as the overall authority.
2. "if the supervisor wishes to provide detailed instructions, s/he can; when s/he wishes to provide only loose guidelines … s/he can do as well…"
3. "… the subordinate must have substantial knowledge about and capabilities within the domain."
4. A supervisor has to know the limitations of the subordinate.
5. A common representation of tasks and goals has to be shared between supervisor and subordinate to communicate about tasks, goals and constraints.
6. "The act of delegation will itself define a window of control authority within which the subordinate may act."

Based on those requirements the following consequences for designing an artificial subordinate can be derived:

- Following the first requirement, a subordinate must not be "fully autonomous", i.e., a subordinate must not alter the goal to achieve. According to [16], a truly autonomous system would neither be ethical nor be useful, because it takes away responsibility and control from the human supervisor. Therefore, an artificial subordinate must not violate its "window of control authority" (requirement 6).
- Requirement 3 demands that an artificial subordinate shall be designed and implemented as knowledge based system. In combination with requirement 5, this leads to a *symbolic knowledge representation* which allows to use explicit knowledge for processing and communication.
- Requirement 5 as well as our definition of the term "task" leads to a *goal-driven system*, i.e., the overall behaviour of the system shall be defined by the goals pursued.
- To address requirement 2, the supervisor may choose to provide only tasks considered relevant to him or her. Consequently, it is the responsibility of the system, to *maintain a consistent task agenda*. This is accomplished by inserting missing tasks as required the mission to be accomplished, general domain knowledge and causality, e.g., knowing that a start procedure is required to be airborne.

It is obvious that a technical system capable of fulfilling those requirements and the above-mentioned conclusions is a very complex technical system by design. Billings [18] listed several negative characteristics of systems where humans have to supervise complex automation in general. Complexity in this context means that the system cannot fully be understood by the human

operator in every state of the system or in every workload situation that may arise. The characteristics described by Billings are:

- *Brittleness* – The design of the automation limits its use cases to those defined by the designer. Outside of these limits, the behaviour of the automation is not defined. Further information about the relation between designer and operator of complex systems can be found in [19].
- *Opacity* – The operator of complex automation may have a wrong or incomplete model of the automation. The automation does not provide sufficient information to support a correct and complete model in every situation.
- *Literalism* – The automation does not have any knowledge about goals and intents of the operator, but executes its functions defined at design time. It does not and cannot check if those functions support the achievement of the operator's goals.

While automation complexity is inherent to the introduction of a new automation layer, the approach of task-based guidance attempts to reduce those negative effects. This is achieved by the following techniques in the design of task-based guidance systems:

- To address *brittleness*, domain specific practices and regulations, e.g., air space regulations in the aviation domain, shall be known to the automation and shall be pursued as goals rather than executed as hard wired functions. Due to the inherent knowledge-processing characteristics of cognitive automation, the system will strive to follow the regulations even in situations not foreseen by its designer.
- *Opacity* can be reduced by providing feedback on the abstraction level of task description. Using this abstraction level during task assignment, task processing, task execution and in the feedback about current and future tasks allows the human operator to build a mental model about the current and future state of the automation.
- In contrast to conventional, procedure-based automation, task-based guidance follows explicit and abstract objectives. This counteracts *literalism*, because the automation chooses functions (action alternatives) that pursue the task objectives with respect to the currently observed situation.

The following section describes the application of this concept to the guidance of UAVs.

B. Application to UAV guidance

Task-based UAV guidance aims at integrating multiple unmanned vehicles into a manned helicopter mission in a similar manner as integrating additional manned helicopters into the scenario. Therefore, the guidance of unmanned vehicles should be on an abstraction level that allows the allocation of a series of tasks to each UAV. These tasks are issued by the human operator and request the achievement of goals, e.g., the request of reconnaissance information about a landing site. The interpretation of the tasks and the use of on-board systems to fulfil these tasks are left to the UAV. The series of tasks is on a similar abstraction level as tasks assigned to a pilot during mission briefing in a conventional, manned helicopter mission.

Moreover, just like a human pilot, UAVs should also use opportunities of supporting the mission, e.g., by getting sensor information of nearby objects, without a direct command from the operator.

This implies UAV guidance and mission management on a level where one or more UAVs are controlled by tasks that use mission terms instead of waypoints and the request of results rather than in-detail configuration of flight control functions and sensor payload. The latter should be generated aboard the UAV by its on-board automation.

The tasks currently implemented in the experimental setup are:

- a ***departure*** task that respects basic air traffic regulations of the airfield and makes the UAV take off and depart via a given, named departure location.
- a ***transit*** task that causes a flight to a specific, named location. While being in transit, the UAV configures the on-board camera into forward looking mode. Known threats will be automatically avoided, if possible.
- a ***recce route*** (short for "route reconnaissance") task that causes the UAV to fly a route to a named destination. The sensor payload will be configured to provide reconnaissance information about the flight path, i.e., information about locations of sensor readings that indicate armed vehicles and hostile air defence. If the UAV possesses knowledge about another UAV also tasked with a recce of the same route, it will modify its flight path to maximize sensor coverage.
- a ***recce area*** task that causes the UAV to gather recce information about a named area. The camera will be used to provide ortho-photos of the area.
- an ***object surveillance*** task. While working on this task, the UAV will use the payload control to deliver a continuous video stream of a named location.
- a ***cross corridor*** task makes the UAV fly through a transition corridor between friendly and hostile territory. It consists in avoiding friendly fire and ease cooperation with the own ground based air defence; this crossing is modelled as separate task. Moreover, it is the only task allowed to cross the border between friendly and hostile territory.
- a ***landing*** task causes the UAV to take an approach route to an airfield and to land at that airfield.

The capability to understand these tasks at mission level consists in knowledge of several domains, i.e., artificial situation awareness, planning capabilities and using the air vehicle and its payload. This requires an automation that incorporates certain sub-functions as found in cognitive behaviour of a human [10, 14], i.e., creating cognitive behaviour of the automation. The following sections discuss issues concerning the levels of automation and describe the architecture and information processing of a so-called Artificial Cognitive Unit (ACU).

IV. LEVELS OF AUTOMATION

Currently, UAV systems operate on a wide range of different guidance modes. That modes cover the whole range from direct manual control [20], flight control based [9], scripted behaviours [2] up to above-mentioned task-based guidance [10]. These guidance modes form a stack of *abstraction layers* as depicted in Figure 2. In this figure, "R/C pilot" refers to remotely controlled piloted systems like model airplanes. FMS stands for Flight Management Systems capable of following pre-programmed waypoint lists.

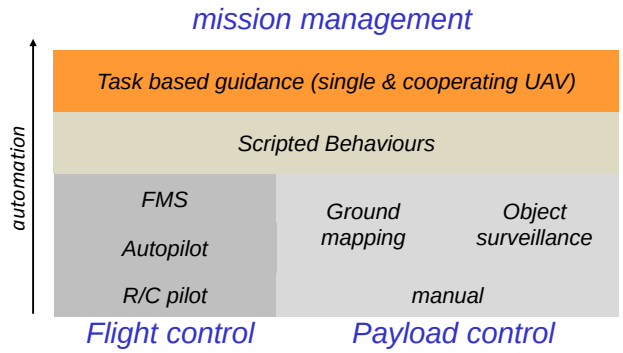


Figure 2. Levels of abstraction in UAV guidance [1]

Sheridan and Verplank [21] describe a different view of levels of automation. These levels are mostly independent from the chosen abstraction layer but set the focus on task allocation and *authority sharing* between the human and the automation. They range from manual control (level 1) to automation that does neither allow intervention from the human operator nor provide information about the action taken (level 10). In the design of current UAV guidance systems, various levels of automation can be found, e.g., in waypoint based guidance systems, the definition of waypoints may be the sole responsibility of the human operator. No automation, in this case, is provided to support that task. However, automatic flight termination systems, e.g., may not allow the human to veto on the decision of the automation but merely report the flight termination after its execution, i.e., level 7 according to Sheridan and Verplank: "computer does the whole job and necessarily tells the human what it did" [21].

Another view to automation focuses on capabilities and the *interoperability* with the control station provided by the system. A prominent example for this kind of automation scale is defined in [22] as *Levels of Interoperability (LoI)*:

- Level 1: Indirect receipt of UAV data
- Level 2: Direct receipt of UAV data
- Level 3: Level 2 plus control and monitoring of the payload
- Level 4: Control and monitoring of the UAV, less launch and recovery
- Level 5: Level 4 plus launch and recovery

The task-based guidance approach described in this paper introduces an additional dimension in the levels of automation. The operator can choose to provide different tasks to the UAV. The UAV will check the tasks for consistency and may insert additional tasks to warrant a consistent task agenda. The consistency check and completion of the task agenda is based on a planning scheme, which behaves deterministic with respect to the current tactical situation and the task elements known so far. Therefore, the operator may choose to specify only task elements relevant to him or her and leave the specification of other tasks to the UAV. This particular type of adaptable automation allows the specification of strict or tight task agendas, i.e., the human operator defines every task of the UAV. However, also loose task agendas may be defined, i.e., the human operator only defines the most important tasks and leaves the details to the UAV. Therefore, this level of automation defines a varying *tightness* of UAV control.

Moreover, this kind of automation also can reduce the chance of human errors, because unintentionally omitted tasks are also completed by the automation.

Table I shows an overview of the aforementioned dimensions of automation in UAV guidance. In the design of a UAV guidance system, each automation level may be fixed, e.g., a system may provide task-based guidance (abstraction) with management where the system offers a complete set of action alternatives, i.e., authority sharing on level 2 [21] including launch and recovery (interoperability LoI 5) where every single task has to be specified by the human operator (strict tightness).

Despite of having a fixed level of automation on the four scales, a system can allow the human operator to adapt the abstraction level, the sharing of authority and the tightness level. Moreover, the automation can change the sharing of authority, i.e., it can be designed as adaptive system.

TABLE I. DIMENSIONS OF AUTOMATION

Dimension	fixed	adaptable	adaptive
Abstraction	•	•	
Authority sharing	•	•	•
Interoperability	•		
Tightness	•	•	

As the focus of this work is on the task-based guidance and the tightness of the UAV guidance, our prototype and evaluation environment uses a fixed abstraction level (task-based guidance), a fixed sharing of authority (depending on the automated function) and operates on LoI 5. The tightness can be implicitly adapted by the human operator. For every task the operator assigns to the UAV the authority of task refinement is passed from the operator to the UAV. The amount of the required refinement defines the degree of tightness in the UAV guidance.

V. SYSTEM ARCHITECTURE

With respect to implementing the desired machine behaviours, this section will provide an overview of the design principles and information processing architecture enabling task-based guidance capabilities.

A. Design of knowledge-based Artificial Cognitive Units

Based on models of cognitive capabilities of human pilots, Artificial Cognitive Units (ACUs) were designed. As depicted in Figure 3, these units become the sole mediator between the human operator and the vehicle [23] in the work system [17]. This additional automation allows

the desired shift in the guidance paradigm from the subsystem level, i.e., separate flight guidance and payload management, to commanding intelligent participants in the mission context (also refer to [24]).

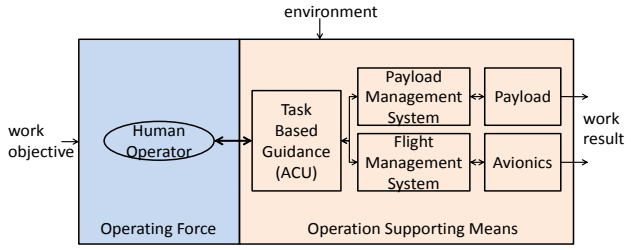


Figure 3. Work system "UAV guidance"

To understand and execute tasks with respect to the current situation, the ACU requires relevant parts of the knowledge and cognitive capabilities of human pilots. That knowledge can be grouped into system management, understanding and evaluating mission objectives in the context of the current scenario as well as knowledge to interact with the human operator [25]. This knowledge is derived by formalization of domain specific procedures defined in documents like the NATO doctrine for helicopter use in land operations [26]. Furthermore, interviews with experienced helicopter pilots revealed relevant knowledge. The interviews and the additional evaluation of recordings of training missions used the Cognitive Process Method [13]. For every phase, the human's objective is evaluated. Moreover, all possible and hypothetic action alternatives to pursue the objective are determined. Furthermore, all environmental knowledge is gathered, which is used to select a particular action alternative or which influences the execution of a chosen action. In our laboratory prototype, this knowledge is used to select a particular action alternative over another, thereby avoiding state space explosions and reducing planning time. At last, the procedural knowledge to execute the actions is evaluated and transformed into machine readable instruction models.

B. Human-machine interface

To support the guidance of multiple UAVs from a manned helicopter, the human-machine interface (HMI) has to be integrated into the manned helicopter. Considering an audio interface, i.e., speech recognition to guide the UAVs, was rejected by a majority of the interviewed pilots due to the already high radio traffic that has to be handled by the helicopter crew.

Therefore, a graphical interface was chosen to interact with the UAV. This interface is integrated into two identical multifunctional displays available to the commander of the manned helicopter. Figure 4 depicts the implemented multifunctional display format.

On the lower left of the multifunctional keyboard, the operator can switch between UAV control and the displays of the manned helicopter (A/C / UAV). Above, the current UAV can be selected. On the top left, the operator can select three different modes: CAM, TASKS, and ID. The right multifunctional soft keyboard shows the context sensitive options for the current mode chosen on the left.

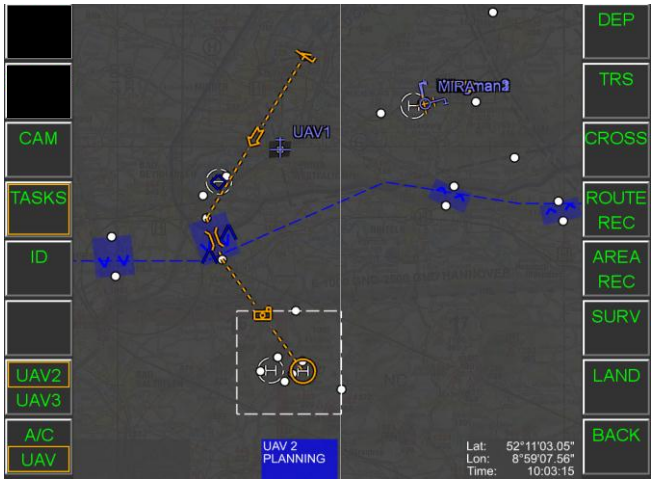


Figure 4. UAV tasking interface

CAM provides a live video stream from the camera of the currently selected UAV.

TASKS can be used to monitor the current tactical situation and to manipulate the displayed task elements of the currently selected UAV. The currently active task is highlighted in yellow. A task can be inserted into the task agenda of the UAV by choosing the task type as shown on the right in Figure 4, optionally choosing the predecessor of the task on the map and selecting the target position of the task. A task can be selected for immediate execution. This functionality can be used to start the execution of the first task as well as for skipping tasks, i.e., the human operator chooses to cancel one or more task to give priority to a more important task. Additionally, tasks can be deleted and moved, i.e., the target area description of the task is altered. If tasks are added, deleted or modified, the UAV will maintain a consistent task agenda by inserting missing tasks depending on the current tactical situation. As long as this planning is in progress, it is indicated on the bottom of the display as shown for UAV number 2 in Figure 4. To prevent immediate re-insertion of deleted task elements, the consistency checks are delayed after the operator deletes a task element. This allows further modifications of the task agenda by the human operator without being interrupted by the UAV.

The ID display mode is used to review photos taken by the UAV and to classify the objects on the images into pre-defined types (car, military vehicle, ground based air defence) and hostility, i.e., neutral, friend or foe. Those classifications are also reflected in the tactical situation shown in the task mode as well as the electronic map displays available to the pilot flying. Furthermore, those classifications will be transmitted to the UAVs in order to support reaction to the changed tactical environment, e.g., to plan flight routes around hostile air defence.

The combination of those display functionalities shall allow the human operator to guide the UAVs to support a military air assault mission that involves operation over hostile areas and support of infantry troops. Moreover, by tasking the UAVs using mission terms, e.g., by selection of "area reconnaissance of the primary landing site", the control of three UAVs shall be feasible and enhance mission safety by providing valuable information about mission relevant areas and routes without risking exposure

of own troops to threats like ground based air defence and other opposing forces.

C. Information processing

The implementation of artificial cognitive units is based on the Cognitive System Architecture (COSA) framework [13]. This framework is based upon Soar [28] and adds support for object-oriented programming as well as stereotypes for structuring the knowledge into environment models, desires, action alternatives and instruction models.

This (a-priori) knowledge constitutes the application specific part of the Cognitive Process, which is described in detail by Putzer and Onken [13] as well as Onken and Schulte [17]. Information and knowledge processing as well as interfacing with the environment is depicted in Figure 5. The inner ellipse represents the static, a-priori knowledge of the system. This knowledge is defined at design-time. Input data and instances of the a-priori knowledge constitute the situation knowledge, which is depicted in light grey in Figure 5. The arrows indicate the information flow in the cognitive process. Every processing step modifies one specific area of situation knowledge, but may read from all areas of knowledge.

The following describes the information processing steps using examples of the knowledge of the UAVs' on-board ACUs.

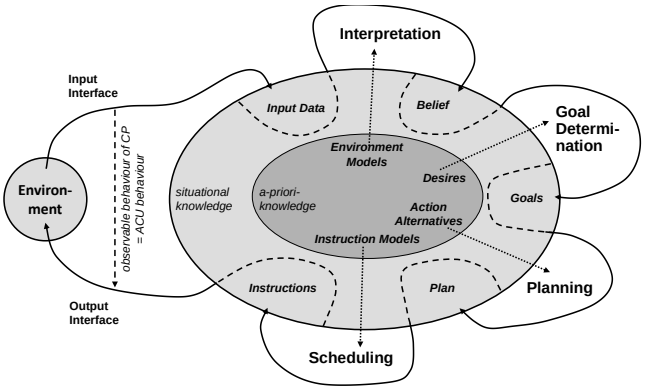


Figure 5. Knowledge processing in the Cognitive Process [17]

Input data are retrieved from the environment by input interfaces. There are three types of input interfaces: (1) reading sensor information from the sensors of the UAV, (2) reading information from the communication link of the UAV and (3) providing results from on-board automation, e.g., information about flight routes generated by an external route planner.

The *environment models* of the a-priori knowledge of the ACU drive the interpretation of input data into instances of semantic concepts. Those concepts form an understanding of the current tactical environment including knowledge about existence and positions of threats, areas, bases, landing sites, routes, waypoints etc. Due to the nature of the cognitive system architecture, environment models continuously monitor all the input data and other knowledge of the cognitive process and react with instantiation, modification or removal of corresponding beliefs. All instances of environment models, i.e., *beliefs*, form the representation of the current situation of the UAV. *Desires* describe world states the UAV should maintain. Every desire contains declarative knowledge about the detection of violation of the state, i.e., it contains rules that continuously monitor the situation for facts that indicate a violation of the desire. If a violation is detected, an instance of the desire is created, i.e., the desire creates an active goal. Desires may contain knowledge that derives priorities from the current situation, e.g., the desire of executing task modifications issued by the human operator takes precedence over the desire of having a consistent task agenda. The motivation is to avoid fixing agendas that are currently modified by the operator.

Action alternatives provide ways to support active goals. They instantiate if a corresponding goal is active, but only if the current situational knowledge allows the selection of the action alternative. If more than one action alternative can be proposed, then the action alternatives model selection knowledge to prefer one alternative over the other. For example, the action alternatives "transit flight" and "route reconnaissance" may both support the goal of reaching a specific location. If both action alternatives are feasible, the fitness and selection of the alternative depends on the type of area that has to be crossed.

After the action alternative has been chosen, the *instruction models* become active and support the action alternatives by generating instructions on the output interface of the ACU. Those instructions are read by the output interface and cause the transmission of radio messages, configuration changes at the flight control system or the payload system or activate on-board automation, e.g., a route planner.

In combination, all those processing steps depicted in Figure 5 generate purely goal-driven behaviour that allows reasoning over the tactical situation and the task elements entered by the human operator to provide situation-dependent actions, which are consistent with tactical concepts of operations. Unlike procedure-based architectures, the Cognitive Process is not bound to pre-defined algorithms, which are affected by unforeseen changes in the environment or may be unable to deal with concurrent events.. Instead, the situation is continuously analysed with respect to explicitly encoded domain knowledge. Furthermore, the open world assumption of COSA allows dealing with "… incomplete information, which is essential taking sensor data into account." [27].

D. Knowledge Base

While the information processing of COSA is domain independent, the knowledge base defines the domain knowledge models. The knowledge of the ACU is grouped into packages which may refer to each other. Each package defines knowledge of one subdomain:

- environment
- supervisory control
- mission
- cooperation
- task synthesis for loose vs. tight control
- task scheduling
- role management

Every package consists of knowledge models, which are represented in CPL (Cognitive Programming Language) [25], which is based on Soar [28]. For every

type of knowledge described in this section, there is a separate knowledge model encoded in CPL.

The knowledge models of the prototype focus on mission management, cooperation of UAVs and task-based guidance in general and are mostly vehicle independent. However, [24] presents an architecture that allows the integration of high-level UAV mission tasking and vehicle specific knowledge.

1) Environment knowledge

The environment package contains all knowledge models that allow the ACU to build an internal, symbolic representation of the current environment including the state of the UAV. This knowledge may be considered machine situation awareness comparable to human situation awareness on level 2 (understanding of the current situation) according to Endsley [29].

This knowledge covers models about the existence of the UAV and other UAVs in the team. Moreover, knowledge about ground forces, air spaces, positions in general and relation between positions is represented. Information about the sensor system is covered by a model of the on-board sensors. Information about sensor photos that may be reviewed by the human operator for classification is represented by a corresponding knowledge model.

```
class <belief> hotspot
{
  attributes:
    string name := |hotspot|;

    // location of the hotspot (WGS84)
    double lat;
    double lon;
    double alt;

  behaviour:
    sp { create*from-sensor-input
    : o-support
      (state <s> ^io.input-link.sensor.thermal-detector <sensinput>)
      (<sensinput> ^lat <lat> ^lon <lon> ^alt <alt> )
    -->
      (elaborate <i>)
      (<i> ^lat <lat> ^lon <lon> ^alt <alt> )
  }
};
```

Figure 6. Example of an environment model

Fig. 6 shows a short example of an environment model of the UAV. This particular environment model represents sensor readings of the automated target recognition system (ATR), which indicate possible threats at a defined position. The stereotype "belief" makes the knowledge model an environment model. The behavior "create*from-sensor-input" consists of a condition part and an action part. The conditions are matched against the current knowledge and check for the existence of a "thermal-detector" node in the sensor input data. If the condition is met, an instance of the knowledge model is created ("elaborate") and the coordinates of the sensor input are copied from the input data into the newly created instance. The keyword "o-support" makes the instance permanent, i.e., the general truth maintenance property of COSA is not applied and the instance will not disappear if the input data disappears. The syntax used here is an object oriented extension [13] of Soar [28].

2) Supervisory control knowledge

Knowledge about supervisory control covers knowledge necessary for the task assignment to the UAV. It contains a model of the instruction sent from the human operator to the UAV. For every task type available, there is an instruction model derived from that base instruction model. Furthermore, there are models for the messages that request the execution of a specific task as well as one model for the request to stop or delete a task.

Another knowledge model in this package represents the current guidance mode of the UAV. This model is responsible for the representation of the task-based guidance as such. It detects overrides to the task-based guidance, e.g., the aforementioned request to stop a task, and is responsible for granting or revoking access to the flight management system to the ACU as such. The ACU always initializes with those privileges being revoked, i.e., it is the sole authority of the human operator to transfer the authority over the flight management system to the ACU.

3) Mission knowledge

The mission knowledge represents the models used to execute the tasks assigned to the ACU. Most of the behavior of the ACU is defined by its desire to comply with the assigned tasks. This knowledge model makes the ACU strive to fulfill the current task at hand.

Furthermore, the mission knowledge contains the desire to use opportunities for retrieving additional recce information which may be unrelated to the current task. Therefore, the ACU combines its knowledge about the type of sensor information, i.e., "unidentified sensor-hotspot", the availability of its sensors, the availability of sensor information from its own sensors and from other UAVs, and its relative position to the unidentified force. This combination of knowledge enables the UAV to safely detect and use the chance of getting more information about the location. Moreover, the UAVs also behave cooperative as the decision to generate additional sensor information is suppressed if another UAV has generated that sensor information from a similar angle to the unidentified force.

The action alternatives of this knowledge package model ways to achieve active goals. Moreover, additional desires model prerequisites for action alternatives, e.g., to make the action alternative of crossing an airspace corridor feasible, the aircraft shall be near the entry point of the airspace corridor.

Instruction models contain the knowledge about how to execute the chosen action alternative, i.e., how to interact with the on-board automation and the environment.

4) Cooperation knowledge

The cooperation package contains all models which

- represent knowledge about current and future tasks of the own UAV as well as tasks of other UAVs.
- represent knowledge about the task at hand and the sequence of future tasks. This knowledge also includes strategies for determining the current task at hand.
- determine the information needs of all teammates, i.e., other UAVs, and generates the information feedback to the human operator. Furthermore, action alternatives exist to fulfill those information needs.

There is a common base model for all task elements that defines the common knowledge and common behavior of all tasks. Derived from that model, there is a model for every task type available, i.e.:

- *recce-route* is the task to get reconnaissance information about a flight route to a specified destination.
- *recce-area* is the task to get information about a specified area and its surroundings.
- *surveillance* delivers a continuous video stream of a specified location or a designated force.
- *transit* is the task to fly a safe transit to a specified target location.
- *departure* is the task to execute a departure procedure with compliance to the departure rules of the current location.
- *landing* is the task to execute an approach and landing procedure with compliance to the approach rules of the specified landing location.
- *cross-flot* is the task to cross airspace boundaries, i.e., the so-called forward line of own troops (FLOT), at a specified airspace corridor.

An *agenda* models the sequence of the tasks of a UAV. In order to know the current task at hand, the cooperation package defines three action alternatives. First, if there is an instruction from the human operator requesting a task for immediate execution, then this task becomes the current task. If this alternative is available, then it is preferred over other strategies. Second, the ACU may select the successor of the last completed task as the new current task. This is the default strategy. If neither strategy is applicable, the ACU may choose the first non-completed task from the agenda.

To model the information needs of the team mates, a *knowledge monitor* tracks instantiation, change and destruction of relevant knowledge models. The relevance for team mates is implemented as additional model attribute that can be evaluated at runtime. The model of the desire to keep the team informed is activated, if an instance of "knowledge monitor" detects a change in the monitored instances. As a consequence, action alternatives are activated and propose to communicate the change in the knowledge to the team. There are multiple action alternatives to model different serializations of knowledge, i.e., to address different communication channels.

5) Knowledge about task synthesis

To model the variable tightness described in Section IV, the ACU possesses a desire to *have a consistent task agenda*. This desire activates into an active goal, if one of the following rules for consistent agendas is violated:

1. Every task except departure requires the UAV to be airborne.
2. If there is an approach route for a landing site, then it shall be used by the landing task.
3. The tasks "recce-area" and "surveillance" require the UAV to be near the area or the named location respectively.
4. The task "cross-flot" should start at an airspace corridor.
5. If there are designated entry/exit points for an operation area, then these shall be used by the UAV.
6. If there are airspace corridors connecting airspaces, then those corridors shall be used.

To detect those violations, there is a knowledge model representing the state of the UAV *after completion* of a task. An additional instance of that knowledge model refers to the *current state* of the UAV. Furthermore, there is a knowledge model whose instances represent the preconditions of *future tasks*. Violations of the rules can be detected by comparing the prerequisites of one task with the predicted state after completion of its predecessor.

An example of a violation is depicted in Figure 4. The route reconnaissance on the lower half of the image, which is shown by a stippled, orange line with a camera symbol, crosses the boundaries of the operation area (white stippled rectangle). However, that area shall be entered only via its designated entry points (white dots). Therefore, this leads to an activation of "have a consistent task agenda". As this activation is considered relevant knowledge to the team, it is transmitted to the operator and shown as "UAV planning" in Figure 4.

If there are multiple, concurrent violations of rules, the violations are scheduled according to a "divide-and-conquer" scheme, e.g., if rule 6 is violated, this violation is addressed first to divide the agenda into parts operating only in a single airspace.

The action alternatives supporting the goal of having a consistent agenda are the creation and insertion of additional tasks into the task agenda. Furthermore, existing tasks may be altered, e.g., to ensure that a cross-flot task starts on the right side of the airspace corridor. Action alternatives are selected based on the current or projected tactical situation, e.g., the resolution of a violation of rule 4 depends on the type of the terrain, i.e. a "transit" task is inserted when operating over safe terrain and a "recce route" task is inserted to reach the corridor while operating over unsafe terrain.

As mentioned in Section IV, the human operator can make use of this behaviour of the ACU by skipping tasks on purpose and thereby shifting the completion and specification of missing tasks to the ACU.

6) Task scheduling

The human machine interface allows the operator to insert new tasks at a certain position of the agenda after having specified the predecessor of the new task. However, the operator may also define tasks without specifying where to insert the task into the existing agenda. Therefore, the ACU follows a desire to know the task insertion point. For new tasks without specified insertion point that the behavior of that desire creates an active goal.

The action alternatives for determining the insertion point are to insert the task at the end of the task agenda or to insert the task in a way that minimizes the detour based on the existing task agenda. The latter alternative is not available for departures and landings.

7) Role management

Knowledge about role management supports the cooperation of multiple actors (UAVs) working on a common task. The term role is used as defined in social sciences [30]. Biddle states that roles in the symbolic interactionist role theory are *"…thought to reflect norms, attitudes, contextual demands, negotiation, and the evolving definition of the situation as understood by the actors."* [30]

To bring the concept of roles to the UAV, the ACU has a knowledge model describing its roles and the roles of the teammates. Furthermore, another knowledge model defines the desire of *having a unique role per task*. This

desire is activated into an active goal if there is no role assigned to a task or if there is knowledge available, that the assigned role is also assigned to a teammate for the same task.

There are models about role configurations that define which roles are available depending on the task type and the number of UAVs working on the common task. Depending on those configurations, all possible roles are proposed for a task.

To support the goal of having a unique role per task, there are two action alternatives available. Firstly, the ACU may assign an available role to the common task. The selection can be random or based on the application of selection knowledge like preferring to stick to a role in subsequent tasks of the same type. Secondly, the ACU may drop an assigned role as result of a role conflict.

Therefore, the general outline of role-based cooperation is:

1. UAVs communicate assigned tasks to each other.
2. Each UAV detects that it participates in a task also assigned to another UAV, i.e., a common task requiring cooperation.
3. Every UAV proposes a role describing how to participate in the task.
4. Conflicting and missing role assignments are detected and resolved.
5. Roles affect the way a task is executed by the UAV.

Because of the knowledge-based nature of the ACU, the arbitration of roles is immune to race conditions like simultaneously changing roles and environment. Any invalid role assignment triggers the activation of the goal of having a valid assignment and consequently causes the correction of the role configuration. The following example illustrates how a team of UAVs benefits from this property in a dynamic situation.

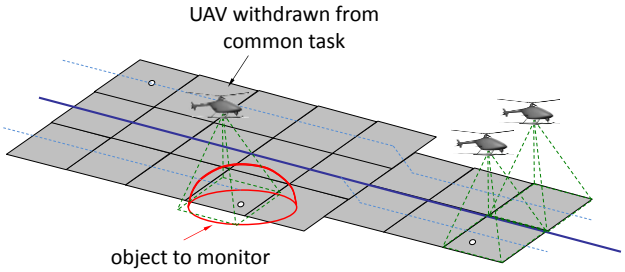


Figure 7. Multiple UAVs working on common task

If there are three UAVs with a common recce-route task, the corresponding role configuration and roles will become active and propose the roles to “fly to the left of the track”, “fly to the right of the track” and “fly on track” in order to maximize the sensor coverage. When the ACU plans its route for the route reconnaissance, the assigned role affects the instruction model responsible for the route planning. Therefore, the route planner is instructed to add an offset to the track while planning. This leads to a formation-like flight of the three UAVs. The left half of Figure 7 depicts the resulting flight paths of three UAVs flying a common recce-route task. The grey patches illustrate the coverage of the sensor images taken by the UAVs.

Once a UAV is withdrawn from the common task, the roles available may change. This leads to a reassignment of the roles and to a change in the task execution as depicted in the right half of Figure 7, i.e., one UAV is being withdrawn from the common task causing the other UAVs to change their roles and flight paths.

VI. Experimental Setup and Results

Experiments were conducted with experienced German Army helicopter pilots in order to evaluate the task-based guidance approach. The simulator cockpit shown in Figure 1 has been used to perform military transport helicopter missions. The simulation of the UAVs consists of a simple kinematic model of a generic helicopter. The elementary flight performance envelope of this model is comparable to the model of the manned helicopter, i.e., the maximum speed is about 120 knots. However, the concept of task-based UAV guidance is independent from specific UAV platform types or dynamics, and is also applicable to fixed-wing aircraft. The kinematic flight model was fitted with an autopilot, waypoint tracking capabilities, and interfaces to the ACU. Together with an electro-optical sensor simulation, this simulates the flight control and payload control as depicted in Figure 2. In the simulation setup, the LoI is fixed at level 5, i.e., the operator has full control of UAV including payload, launch and recovery (cf. Section IV) and only the task-based layer is available to the human operator, i.e., the abstraction layer is fixed in the experiment.

The objective of the missions was to pick up troops from a known location and to carry them to a possibly threatened destination. According to the briefing, three UAVs should be used to provide reconnaissance information about the flight routes and landing sites in order to minimize exposure of the manned helicopter to threats. In addition to the tasks to perform in previous baseline experiments without task-based guidance [9], in this experiment an unscheduled combat recovery task was commanded to the crew as soon as the main mission objective had been accomplished.

Prior to the measurements, every test person had been given one and a half day of education and training on the system. The test persons acted as pilot flying and commander. This configuration was chosen to evaluate the effects of the UAV guidance to crew cooperation and crew resource management.

The following data were recorded during the experiment:

- Interaction of the operator with the system
- Commands sent to the UAV via data link
- Resulting task agendas of the UAVs
- Helicopter and UAV flight paths
- Sensor coverage

This data was used to retrieve measures in the categories *performance*, *behaviour* and *subjective ratings*. Performance covers the mission success as such, including different aspects of reconnaissance and UAV flight guidance. Behavioural measures include the attention demand of UAV guidance, the distribution of interaction with the UAV guidance system over the mission phases and the tightness of the UAV guidance (see Section IV). Subjective ratings cover the perceived workload and the system ratings from the test persons.

In the following, the measures and results of the commander of the manned helicopter, who is also the UAV operator at the same time, are presented. Results for the complete flight crew may be found in [31].

A. Performance

One key aspect when measuring performance in military missions is the overall mission success. The test persons managed to accomplish the mission including the additional combat recovery in every simulation run.

Another figure is the gain in mission safety and security achieved by the deployment of detached sensor platforms. This can be estimated by the sensor coverage of the flight path of the manned helicopter. In the experiment, the manned helicopter operated within the terrain mapped by ortho-photos 94.5% of the time in hostile areas.

It is the responsibility of the commander to use the UAVs in a way that maximizes tactical advantages. According to the test persons, this consisted in having sufficient information about the flight path of the helicopter to support mission-critical decisions. Moreover, the army aviators emphasized the importance of having forces near the helicopter to react to unforeseen events.

To evaluate the tactical advantages, a scoring was developed to measure the quality of the reconnaissance achieved with the UAVs.

TABLE II. SCORES FOR RECONNAISSANCE PERFORMANCE

	yes	***no***
Reconnaissance data of helicopter route available in time?	2	0
Reconnaissance data of primary landing site available in time?	2	0
Classification of UAV sensor data in time? (only 1 point, if pilot flying had to request classification)	2	0

Table II lists the applied criteria for the reconnaissance performance and the corresponding scoring. The video recordings of the simulations were analysed to apply the conditions listed. To get the full score of 2 points, the listed requirement has always to be fulfilled during the mission. Otherwise, the criterion was assessed 0 points. The availability of reconnaissance data was considered not "in time", if the manned helicopter had to slow down in order to wait for UAV data or classification or if the helicopter operated near unknown or not-located forces. In the experiment, an average score of 88.3% (n=16) of the maximum of 6 points was reached.

Furthermore, most of the commanders used the capabilities of the UAVs to get reconnaissance information that could have been useful in alternate outcomes of the missions, i.e., information about alternate flight routes and information about alternate landing sites. In some cases this led to delays in the mission progress as UAVs were busy getting information of alternate routes and landing sites.

TABLE III. SCORES FOR ADDITIONAL TACTICAL BENEFITS

	yes	***partial***	***no***
Reconnaissance data of alternate flight routes available?	2	1	0
Reconnaissance data of alternate landing sites available?	2	1	0
Delays in the mission progress because of missing reconnaissance data?	0	n/a	2

Table III provides a scoring for these additional benefits, the commanders got from the deployment of the UAVs. On average (n=16), 60.5% of the maximum score was reached in this scale.

The results in this section indicate, that task-based guidance is a way of UAV guidance which supports the overall mission success as well as mission safety. Moreover, the test persons used the UAVs as force multiplier to get additional sensor data of alternate sites and routes.

B. Fan-Out

Supervision of the UAVs places extra work demands on the human operator. To get a measure of the demands of multiple UAV guidance using task-based guidance, the maximum number of UAVs the operator can handle shall be estimated. This estimation is based on the operator's attention required by one UAV. Goodrich and Olsen [32] introduce the concept of Robot Attention Demand (RAD) which can be calculated as

$$\mathrm{RAD} = \frac{\mathrm{IT}}{\mathrm{IT+NT}} \quad (1)$$

where IT denotes the Interaction Time, i.e., the time the operator actually interacts with a multi-robot system. NT is the Neglect Time, i.e., the amount of time a robot can be neglected before its performance drops below a certain threshold [32].

For multi-robot systems like multi-UAV guidance, it can be assumed, that the human operator uses NT to interact with additional robots. Therefore, the inverse of RAD gives an upper bound for the number of robots that the human operator can handle. This measure is called Fan-Out (FO) [32].

To further improve the estimate of the Fan-Out, Cummings et al. [33, 34] introduce the concept of Wait Time (WT). Wait times occur, if the human operator should interact with a robot, but fails to do so because he is busy with another robot (wait time caused by interaction – WTI), because of task switching delays (wait time in decision making queue – WTQ) or he lacks the situation awareness to recognize the need for interaction (WTSA). With wait times, the Fan-Out can be calculated as

$$\mathrm{FO} = \frac{\mathrm{NT}}{\mathrm{IT+WT}} + 1 \quad (2)$$

To apply the measurement of IT, NT and WT to the experiment, which models a complex military scenario, the following criteria are used to distinguish interaction time, neglect time and wait time:

Wait Time (WT) occurs in the following cases:

- At least one UAV is idle, i.e., it has completed all of its tasks.

- There is at least one mission relevant object in the sensor images of the UAV waiting for classification by the human operator, e.g., a UAV has taken an image of hostile ground forces but the operator has not yet evaluated that image.
- A UAV enters the range of hostile air defence.

Interaction Time (IT) occurs, if none of the conditions for wait times are fulfilled and at least one of the following conditions is true:

- The operator prepares the human-machine interface for interacting with a UAV.
- The operator defines, modifies or removes a task of a UAV.
- The operator evaluates UAV sensor data or prepares the human-machine interface to do so.
- The operator interacts with the human-machine interface to monitor the current position and task of a UAV. This is equivalent to "robot monitoring and selection" as defined by Olsen [35].

All other time spans are considered Neglect Time (NT).

The times were measured by evaluating the interactions of the operator with the overall system instead of measuring per UAV. Therefore, the resulting Fan-Out is relative to the initial number of UAVs, which is three.

The average neglect time measured is 57% (n=16) of the overall mission time. The mean of the wait times is 6.5% (n=16).

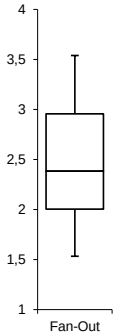


Figure 8. Distribution of the Fan-Out (FO) in the experiment

Hence, the average Fan-Out is computed to 2.49 (n=16).

The average share of NT used for interactions with the systems of the manned helicopter as well as interacting with the pilot flying is only 19%.

This result and the high neglect time indicate that task-based guidance of three UAVs is feasible and the human operator still has sufficient resources to remain in his role of being the mission commander and pilot in command of the manned helicopter.

C. Task Instructions per Mission Phase

To evaluate if the concept of task-based guidance is also applicable in situations unforeseen by the human operator, all missions of the experiment were divided into four phases:

- Phase A begins with the start of the experiment and ends with the take-off of the manned helicopter. This phase is not time-critical, i.e., it is assumed that the crew can start the preparation of the mission as early as required, although in the real application there might be some organisational and military constraints to that.
- Phase B begins with the take-off of the manned helicopter and ends with the successful completion of the main mission objective, e.g., if the main mission objective is to transport troops, phase B ends when the troops leave the helicopter at the remote landing site.
- Phase C starts after phase B with the assignment of an additional mission objective which was unknown to the crew prior to the experiment, e.g., to rescue the crew of a crashed aircraft. Phase C ends with the successful completion of the additional mission assignment.
- Phase D starts after phase C and covers the egress to the home base.

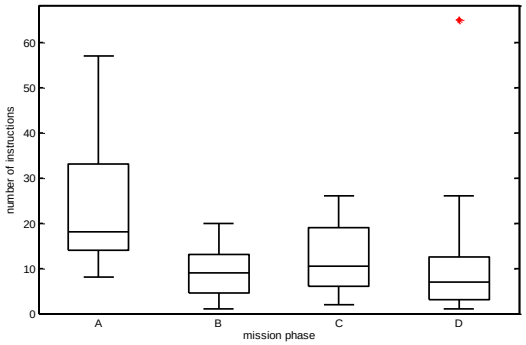


Figure 9. Number of task modifications per mission phase

Figure 9 depicts the number of task instructions, i.e., instructions to insert, alter or remove tasks from the agenda, issued by the operator per mission phase. The mission phase with the most instructions to the UAVs was the time-wise uncritical preparation phase A. If there are only minor changes to the situation, i.e., changes that can be foreseen by experienced mission commanders, only a small number of changes to the UAV task agenda are necessary (phases B and D in Figure 9).

However, if there is a fundamental change to the mission objective including locations and goals which were unknown to the helicopter crew, this can also be handled with a relative small number of changes to the UAV agenda. This is expressed by a small increase of tasking instructions in phase C compared with phases B and D.

Therefore, task-based guidance as evaluated in this experiment shows two qualities:

1. It allows the human operator to shift interactions from mission critical phases to the mission preparation.
2. Even unforeseen situations can be handled with an adequate amount of interactions that is not significantly larger than the number of interactions in known situations.

Just like in conventional, pre-planned missions with only little flexible mission management approaches, most of the interactions are performed in the planning and preparation phase. Nevertheless, the flexibility of the task-based guidance approach is demonstrated, because the unknown secondary task and, hence, the required re-planning activities could be handled with minimum effort.

D. Tightness Level

The tightness level in the task-based UAV guidance can be expressed as the ratio of the number of task elements assigned by the human operator versus the number of synthesized tasks.

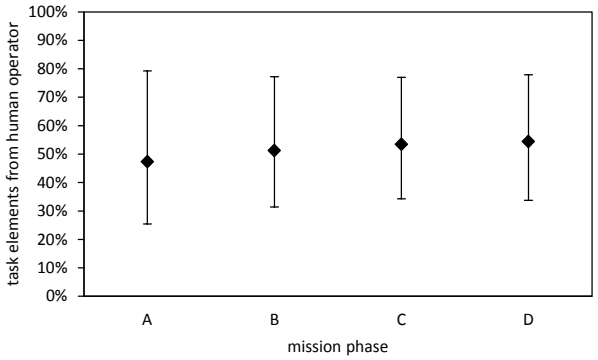


Figure 10. Average thightness in UAV guidance per mission phase

In the experiment, 51% (n=8) of the elements in the task agendas of the UAVs were inserted by the human operator. The remaining 49% of the task elements were automatically inserted by the UAVs to establish a consistent task agenda. Figure 10 depicts the share of task elements assigned by the human operator. In the experiment, this observed tightness level is mostly independent from the mission phase. However, it may vary depending on the individual human operator, as depicted in Figure 11 which shows the figures for two different operators.

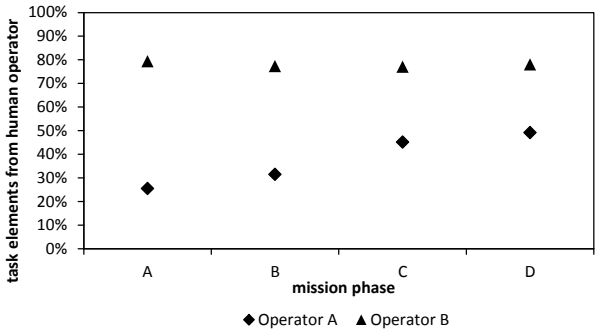


Figure 11. Individual tightness in UAV guidance

Operator A preferred to allow the UAV a higher degree of authority and defined only 38% of all task elements entries during the mission. While being faced with an unknown situation (phase C), the operator took back some of the authority by specifying the new tasks in more detail.

Operator B specified 78% of all tasks elements and did not change that tight guidance level during the course of the mission.

E. Subjective Measures

In every simulation run, the simulation had been halted twice, i.e., in the ingress and during a demanding situation while the helicopter is near the hostile target area, to get measures of the operator's workload using NASA TLX [36]. During the simulation halt, all displays and the virtual pilot view were blanked and the intercom between pilot flying and pilot non-flying was disabled. To get an indication of the test persons' situation awareness, the test persons were simultaneously questioned about the current tactical situations, system settings, e.g., radio configuration, and the upcoming tasks of the UAV and the manned helicopter. Furthermore, commander and pilot had to mark the positions of the manned helicopter, the UAVs and known ground forces in an electronic map. The specified positions were compared with the actual positions of the objects. This measure is an adaption of the SAGAT technique [29]. The test persons achieved a score of 100% for deviations less than 0.75 nm, 50% for deviations up to 1.5 nm and 0% for larger distances or if the object was missing. Only hostile ground forces objects were counted, because neutral ground forces are considered irrelevant to the mission progress [31].

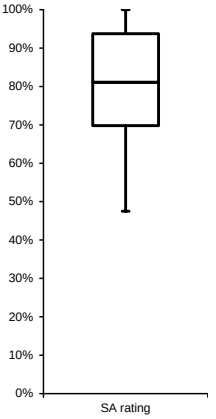


Figure 12. Overall SAGAT measures

Figure 12 depicts the distribution of this score. The commanders got an average score of 80% in this test.

After every mission, a debriefing follows which includes questions about the system acceptance, system handling, interface handling as well as feedback about the degree of realism of the simulation environment.

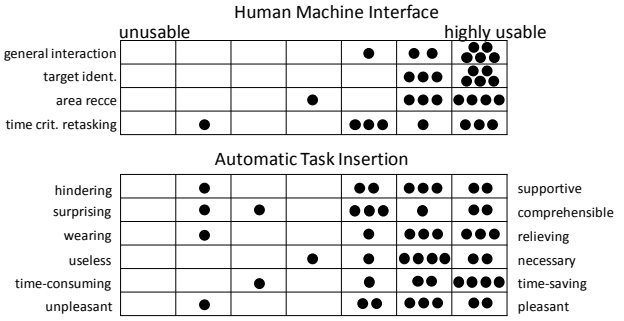


Figure 13. Subjective Pilot Ratings for HMI / Consistency Management

Figure 13 shows the subjective ratings of the test persons concerning the human machine interface and the automatic maintenance of a consistent task agenda, i.e., the automatic insertion of tasks. The representation of tasks as graphic elements on an interactive map was considered suitable for task monitoring and task manipulation. As depicted in Figure 13, one operator missed interfaces for time critical modification of tasks, especially a way to quickly assign low-level commands, e.g., heading and speed, to the UAV. The chosen type of human-machine interface and the automatic insertion of task elements to maintain a consistent agenda are generally accepted by all test persons.

The test persons stated that handling the UAVs consumed an average of 62% of the time while 34% remained for acting as commander of the manned helicopter. This indicates that the test persons felt the UAV guidance twice as demanding as supporting the pilot flying. However, the test persons also experienced the UAVs as highly supportive element for mission accomplishment, which outweighs the additional demands of guiding the unmanned aircraft.

TLX measures of the commander range from 23% of subjective workload during the ingress over friendly territory up to a value of 60% during time-critical re-planning of multiple UAV in the target area.

VII. CONCLUSION

This paper shows a way how operational knowledge can be encoded into an artificial cognitive system to allow the guidance of multiple UAV from the commander of manned helicopter. Task-based guidance, being the guidance concept advertised in this paper, shows high potential for embedding unmanned assets not just as additional complex automation but as artificial subordinates.

The experiment provided evidence, that task-based guided UAVs can increase the overall mission performance and provide tactical advantages. The behavioural measures show that task-based guidance consumes only a moderate share of the operator's mental resources, which allows him to remain in his role of the commander of the manned helicopter. Furthermore, the introduced adaptable tightness of UAV control is found to be intuitively used by the operators to balance the authority between the human and the UAV.

Subjective measures and ratings indicated a manageable workload, a sufficient level of situation awareness as well as a good acceptance of task-based guidance.

Fields that shall be addressed in future work are the handover and shared use of UAV capabilities. Thereby, UAVs could remain airborne over the operation area and human crews can request UAV services on demand. Furthermore, as reaction to emergencies, the human operator should use varying levels of automation, e.g., bypassing task-based guidance to directly set heading and altitude of a UAV. For that case, a methodology shall be developed that defines when and how the authority over the UAV can be reassigned to the artificial cognitive unit. The introduction of varying levels of automation may also incorporate the guidance of teams of UAVs [37], i.e., a single task can be assigned to multiple unmanned systems and those will define and distribute subtasks within the team.

REFERENCES

[1] J. Uhrmann and A. Schulte, "Task-based guidance of multiple UAV using cognitive automation," in COGNITIVE 2011 - The Third International Conference on Advanced Cognitive Technologies and Applications, T. Bossomaier and P. Lorenz, Eds, Rome, Italy, 2011, pp. 47–52.

[2] C. Miller, H. Funk, P. Wu, R. Goldman, J. Meisner, and M. Chapman, "The Playbook™ approach to adaptive automation," Ft. Belvoir: Defense Technical Information Center, 2005.

[3] M. Valenti, T. Schouwenaars, Y. Kuwata, E. Feron, J. How, and J. Paunicka, "Implementation of a manned vehicle-UAV mission system", 2004.

[4] D. Norman, "The design of future things," Basic Books, 2007.

[5] D. D. Woods and N. B. Sarter, "Learning from automation surprises and 'going sour' accidents: Progress on human-centered automation," Ohio State University, Institute for Ergonomics, Cognitive Systems Engineering Laboratory; National Aeronautics and Space Administration; National Technical Information Service, distributor, 1998.

[6] I. Kaminer, O. Yakimenko, A. Pascoal, and R. Ghabcheloo, "Path generation, path following and coordinated control for timecritical missions of multiple UAVs," in American Control Conference, 2006, pp. 4906–4913.

[7] S. Wegener, S.S. Schoenung, J. Totah, D. Sullivan, J. Frank, F. Enomoto, C. Frost, and C. Theodore, "UAV autonomous operations for airborne science missions", in Proceedings of the American Institute for Aeronautics and Astronautics 3rd "Unmanned...Unlimited" Technical Conference, Workshop, and Exhibit, 2004.

[8] A. Schulte and D. Donath, "Measuring self-adaptive UAV operators' load-shedding strategies under high workload," in EPCE'11 Proceedings of the 9th international conference on Engineering psychology and cognitive ergonomics, 2011, pp. 342–351.

[9] J. Uhrmann, R. Strenzke, A. Rauschert, and A. Schulte, "Manned-unmanned teaming: Artificial cognition applied to multiple UAV guidance," in NATO SCI-202 Symposium on Intelligent Uninhabited Vehicle Guidance Systems, 2009.

[10] J. Uhrmann, R. Strenzke, and A. Schulte, "Task-based guidance of multiple detached unmanned sensor platforms in military helicopter opertations," in COGIS 2010, 2010.

[11] R. Strenzke and A. Schulte, "The MMP: A mixed-initiative mission planning system for the multi-aircraft domain," in Scheduling and Planning Applications woRKshop (SPARK) at ICAPS 2011, 2011.

[12] D. Donath, A. Rauschert, and A. Schulte, "Cognitive assistant system concept for multi-UAV guidance using human operator behaviour models," in Conference on Humans Operating Unmanned Systems (HUMOUS'10), 2010.

[13] H. Putzer and R. Onken, "COSA - A generic cognitive system architecture based on a cognitive model of human behavior," Cognition, Technology & Work, vol. 5, no. 2, pp. 140–151, 2003.

[14] S. Brüggenwirth, W. Pecher, and A. Schulte, "Design considerations for COSA2," in Intelligent Agent (IA), 2011 IEEE Symposium on Intelligent Agents, 2011, pp. 1–8.

[15] R. D. Sawyer,."The seven military classics of ancient china (history and warfare)," Basic Books, 2007.

[16] C. Miller, "Delegation architectures: Playbooks and policy for keeping operators in charge," Workshop on Mixed-Initiative Planning and Scheduling, 2005.

[17] R. Onken and A. Schulte, "System-ergonomic design of cognitive automation: Dual-mode cognitive design of vehicle guidance and control work systems," Heidelberg: Springer-Verlag; 2010.

[18] C. E. Billings, "Aviation automation: The search for a human-centered approach," Mahwah, N.J: Lawrence Erlbaum Associates Publishers, 1997.

[19] H. Wandke and J. Nachtwei, "The different human factor in automation: the developer behind versus the operator in action," in Human factors for assistance and automation, D. de Waard, F. Flemisch, B. Lorenz, H. Oberheid, and K. Brookhuis, Eds, Maastricht, the Netherlands: Shaker Publishing, 2008, pp. 493–502.

[20] H. Eisenbeiss, "A mini unmanned aerial vehicle (UAV): system overview and image acquisition," International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences, vol. 36, no. 5/W1, 2004.

[21] T. B. Sheridan and W. L. Verplank, "Human and Computer Control of Undersea Teleoperators," Ft. Belvoir: Defense Technical Information Center, 1978.

[22] Standard interfaces of UAV control system (UCS) for NATO UAV interoperability, STANAG 4586, NATO, 2007.

[23] J. Uhrmann, R. Strenzke, and A. Schulte, "Human supervisory control of multiple UAVs by use of task based guidance," in Conference on Humans Operating Unmanned Systems (HUMOUS'10), 2010.

[24] M. Kriegel, S. Brüggenwirth, and A. Schulte, "Knowledge Configured Vehicle - A layered artificial cognition based approach to decoupling high-level UAV mission tasking from vehicle implementations," in AIAA Guidance, Navigation, and Control Conference 2011, 2011.

[25] G. Jarasch, S. Meier, P. Kingsbury, M. Minas, and A. Schulte, "Design methodology for an Artificial Cognitive System applied to human-centred semi-autonomous UAV guidance," in Conference on Humans Operating Unmanned Systems (HUMOUS'10), 2010.

[26] Use of helicopters in land operations, NATO doctrine 49(E).

[27] S. Puls, J. Graf, and H. Wörn, "Design and Evaluation of Description Logics based Recognition and Understanding of Situations and Activities for Safe Human-Robot Cooperation," in International Journal On Advances in Intelligent Systems, vol. 4, no. 3 & 4, 2011.

[28] J. Laird, A. Newell, and P. S. Rosenbloom, "Soar: An architecture for general intelligence," Stanford, CA: Dept. of Computer Science, Stanford University, 1986.

[29] M. R. Endsley, "Situation awareness global assessment technique (SAGAT)," in Aerospace and Electronics Conference, 1988, pp. 789–795.

[30] B. J. Biddle, "Recent Developments in Role Theory," Annual Review of Sociology, vol. 12, no. 1, pp. 67–92, 1986.

[31] R. Strenzke, J. Uhrmann, A. Benzler, F. Maiwald, A. Rauschert, and A. Schulte, "Managing cockpit crew excess task load in military manned-unmanned teaming missions by Dual-Mode Cognitive Automation approaches," in AIAA Guidance Navigation and Control GNC Conference, 2011.

[32] M. Goodrich and D. Olsen, "Seven principles of efficient human robot interaction," in SMC'03 Conference Proceedings. 2003 IEEE International Conference on Systems, Man and Cybernetics. Conference Theme - System Security and Assurance (Cat. No.03CH37483): IEEE, 2003, pp. 3942–3948.

[33] M. L. Cummings, C. E. Nehme, J. Crandall, and P. Mitchell, "Predicting operator capacity for supervisory control of multiple UAVs," in Studies in Computational Intelligence, Innovations in Intelligent Machines - 1, J. Chahl, L. Jain, A. Mizutani, and M. Sato-Ilic, Eds.: Springer Berlin / Heidelberg, 2007, pp. 11–37.

[34] M. L. Cummings and P. Mitchell, "Predicting controller capacity in supervisory control of multiple UAVs," Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on, vol. 38, no. 2, pp. 451-460, 2008.

[35] D. R. Olsen Jr. and S. B. Wood, "Fan-out: measuring human control of multiple robots," in Proceedings of the SIGCHI conference on Human factors in computing systems, 2004, pp. 231-238.

[36] S. G. Hart and L. E. Staveland, "Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research," Human mental workload, vol. 1, pp. 139-183, 1988.

[37] A. Schulte, C. Meitinger, and R. Onken, "Human factors in the guidance of uninhabited vehicles: oxymoron or tautology?," Cogn Tech Work, vol. 11, no. 1, pp. 71–86, 2009.

Bringing Context to Intentional Services for Service Discovery

Salma Najar
Centre de Recherche en Informatique-Université Paris1
90, rue de Tolbiac 75013 Paris - France
Salma.Najar@malix.univ-paris1.fr

Manuele Kirsch-Pinheiro, Carine Souveyet
Centre de Recherche en Informatique-Université Paris1
90, rue de Tolbiac 75013 Paris - France
Manuele.Kirsch-Pinheiro@univ-paris1.fr,
Carine.Souveyet@univ-paris1.fr

***Abstract*—In service-orientation, the notion of service is studied from different point of views. On the one hand, several approaches have been proposing services that are able to adapt themselves according to the context in which they are used. On the other hand, some researches have been proposing to consider user intentions when proposing business services. We believe that these two views are complementary. An intention is only meaningful when considering the context in which it emerges. Conversely, context description is only meaningful when associated with a user intention. In order to take profit of both views, we propose to extend the Ontology Web Language for services description (OWL-S). We include on it both the specification of context associated with the service and the intention that characterize it. This extended description is experimented in a semantic registry that we built for service discovery purposes. Such registry considers a matching algorithm, which exploits the extended description. Then, we present experimental results of this matching algorithm that demonstrates the advantages one may have on using the proposed descriptor. Thus, we propose a new vision of service orientation taking into account the notion of intention and context. This new vision is based on the extended semantic descriptor, which is necessary in order to enhance transparency of the system by proposing to the user the most appropriate service.**

Keywords-OWL-S; SOA; Intentional Service; Context-Aware Service; Service Discovery

I. INTRODUCTION

Service-Oriented Architecture (SOA) is a computing paradigm lying on the notion of service. This notion is represented as fundamental element for developing software applications [25]. Besides, service stands to independent entities, with well-defined interfaces that can be invoked in a standard way. This does not require, from the user, knowledge about how the service actually performs its tasks [10].

SOA can be viewed through multiple lenses, from the IT perspective up to business leaders [37]. The notion of service is used on different abstraction levels. Technically, it refers to a large variety of technologies (Web Services, ESB [31], OSGI [24], etc.). On a business level, services are proposed as a way to respond to high-level user requirements.

One of the essential challenges in service orientation is *how to find a set of suitable service candidates with regard to a user request and needs*?

On the one hand, we can observe a tendency to context-awareness and adaptation on services. Several authors [15][34][35][36] have been proposing adaptable services to the context in which they are used. These services are usually called *context-aware services* [15]. Their importance is growing with the development of pervasive and mobile technologies. Context-aware services focus on service adaptation considering the circumstances in which it is requested. However, considerations such as why context is important and what is its impact to the user needs remain underestimated.

On the other hand, research has pointed out the importance of considering user requirements on service orientation. Several works [12][18][25][28] proposed to take into account user intentions when proposing business services. According to these works, a service is supposed to satisfy a given user intention.

However, even when considering high-level services, as business services, one should consider variability related to context. Several authors have been considering the influence of context information on business process [30][32]. This influence remains whenever such processes are implemented through business services. Such services still have to cope with the context in which they are called.

Therefore, we have two separated views of service orientation. First, we have an extremely technical view. It focuses on technical issues needed to execute and to adapt service in highly dynamic environments. In the opposite, we have a high level view. This view focuses on user requirements. The latter considers why a service is needed, without necessarily considering how it is executed, neither in which circumstances it is performed. More than the execution context, this high level view ignores the context in which user intentions emerge. Besides, technical view passes over user intentions behind service and observed context information.

We believe that these two views are complementary and should not be isolated from each other. Fully potential of service orientation will not be reached if we do not consider both points of view: *intention-based services* and *context-aware services*. A new vision of service orientation is necessary in order to enhance transparency of the system. In our opinion, an intention is only meaningful when considering it in a given context. Moreover, a context description is only meaningful when associated with a user intention.

Therefore, services should not only be realistic. They should also be described in sufficient detail to allow meaningful semantic discovery.

In order to explore such a complementary views, we should: (1) be able to represent user *intention* in order to be aware of the real use of a service; and (2) capture user *context* in order to choose the best strategy to reach user intention.

Thus, a closely relation can be observed between the concepts of intention, service and context. First, a user intention is defined as "*user requirement representing the intention that a user wants to be satisfied by a service without saying how to perform it*". Then, the context information is defined as "*any information that can be used to characterize the situation of an entity (a person, place, or object)* [7]". We believe that both concepts should be considered in service orientation. We advocate that the selection of the service satisfying user intention is valid only in a given context. For us, a context plays an important role influencing the manner to fulfill user intention and the execution of the service that satisfies this intention (Figure 1).

A user does not require a service because he is under a given context. He requires a service because he has an intention that a service can satisfy in this context. However, this intention is not a simple coincidence; it emerges because he is under a given context.

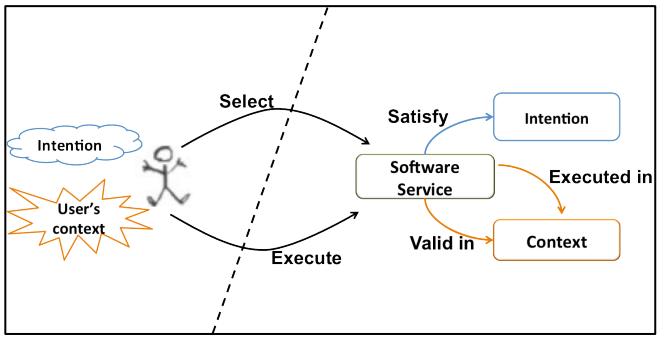


Figure 1. Context and Intention in Service Orientation

In a previous paper [21], we start exploring these ideas, by proposing a semantic description of services. This description encompasses the description of the intention service can satisfy and the context in which this intention is meaningful.

In the present paper, we go further on this semantic description. We propose a semantic description of services that fully describes service intentionality, contextual conditions and intentional composition of these services. We propose to enrich service registry by developing a complete semantic service descriptor based on our OWL-S extension. Then, we propose a service discovery process based on a matching algorithm guided by user and service context and intention. The matching algorithm is based on the implementation of our semantic service descriptor in order to find the most appropriate service according to the user request. This service discovery is implemented and evaluated in order to demonstrate the feasibility of our algorithm.

This paper is organized as follows: Section II presents an overview on related work. Section III presents a motivating scenario. The Section IV introduces the notion of intention and context as preliminary concepts. In Section V, we present our proposition of a semantic descriptor for intentional and context-aware services. Besides we present, in this section, the implementation of our enriched service discovery and the matching algorithm. In Section VI, we discuss our evaluation of the discovery process. And finally, we conclude in Section VII.

II. Related Work

A service can be seen as an independent and easily composed application that can be described, discovered and invoked by other applications and humans. In the last decade, the notion of service has evolved, from simple Web services to semantic Web services [17]. Indeed, we could observe an important tendency for semantically describing services, in order to handle potentially ambiguous service descriptions [17]. Such semantic description is based on richer representation languages, such as OWL-S [16]. OWL-S provides a comprehensive specification of a service.

From the one side, a semantic description is one of the building blocks of context-aware services. These context-aware services can be defined as services which description is associated with contextual properties. We can notice that, an important change has been performed on the way we work and on the way technology support us. We pass from a quite static model, in which people use to interact with business process only during their "work time" to "mobile worker" model [20]. With the evolution of mobile technologies, and notably smartphones, the static model does not fit anymore. Thus, Systems should now consider not only the tasks a user can (or must) perform, but also the context in which such user finds him when performing an action.

In this context, Taylor et al. [35] have considered enriching service with context information. Such works have considered using semantic Web technologies for describing context-aware services. These authors define context-aware services as services that are able to adapt themselves (their composition as well as the content they supply) according to the context in which they are used.

Next, several authors [34][36] have been proposing context-aware services, whose importance is growing with the development of pervasive technologies. An illustration of this phenomenon is given by [34], who propose improving service modeling, based on OWL-S, with context information (user information, service information and environment information). Suraci *et al.* [34] consider that user should be able to specify contextual requirements corresponding to the service he is looking for (availability, location, etc.). Furthermore, this user should be able to specify the context provided by the environment (wireless connection, etc.).

Other authors, such as [36], also advocate for representing context requirements when describing context-aware services. Toninelli *et al.* [36] consider that, in pervasive scenarios, users require context-aware services. These services are tailored to their needs, current position, execution environments, etc. Therefore, service modelling should be improved, including contextual information.

Moreover, Ben Mokhtar *et al*. [2] propose a context aware semantic matching of services based on ontologies. This is expressed in OWL-based languages for enriching service description. In order to support efficient, semantic and context-aware service discovery, they present EASY. From the one side, EASY provides a language for semantic specification of functional and non-functional service properties named EASY-L. From the other side, it provides EASY-M, a corresponding set of conformance relations. These authors [2] propose the use of ontologies in order to automatically and unambiguously discover such services.

Then, Xiao *et al.* [38] are interested on context-aware service and especially on the dynamicity of the environment. These authors propose a context modelling approach. This approach can dynamically handle various context types and values. They use ontologies to enhance the meaning of a user context values and automatically identify the relations among different context values. Based on the relations among context values, they discover and select the potential services that the user might need.

From the other side, several authors have considered a direct participation of the end user on service specification.

Brnsted et al. [4] illustrate this tendency by observing several approaches allowing end users to actively interact with service composition specification. However, these authors do not consider whether terminology used by these tools correspond to the user current vocabulary. The question that emerges here is the following: are these users technical people, who are familiarized with service-oriented technology? Or, are these users business actors who are totally unaware of technical considerations?

A different point of view is given by [4][12][18], which highlight the importance of considering user requirements on service orientation. According to them, a service is supposed to satisfy a given user intention, which becomes central to service definition.

For example, Web Service Modelling Ontology (WSMO) [44] provides a conceptual framework. This framework describes semantically the core element of semantic web services. It is well known by its intention-driven approach. This approach assumes that a user is looking for a service in order to satisfy a specific intention (goal). According to Roman et al. [29] an intention (goal) *describes aspects related to user desires with respect to the requested functionality*. Then, Keller et al. [6] present a mechanism for WSMO web service discovery. This mechanism is based on a matching process between the user goal and the web service capabilities. This information is represented as a set of objects referring to ontologies. The ontologies used in this service discovery mechanism capture general knowledge about a specific domain.

Moreover, WSMO is used in [43] in order to raise the business process management (BPM) from the IT level (Technical) to the user level (Business). In this project [43], the notion of intention is used in order to specify processes and tasks for which the most appropriate web services can be discovered dynamically.

Thus, in WSMO the user intention and the service capabilities are not formulated according to a specific template. As we mentioned, this information is only represented as a set of object. Therefore, they do not identify the real role that plays each object in the intention specification. Consequently, they do not exploit the semantic of verbs, targets and parameters that can represent an intention.

Besides, in WSMO we do not consider the contextual information that can influence the service execution. This element is not clearly defined in the service description. Thus, they neglect the influence that can have the context on the satisfaction of the user intention.

Another works such as [12] and [28] propose a service oriented architecture based on an intentional perspective. Such architecture proposes the notion of *intentional service*. This represents a service focusing on the intention that allows satisfying rather than the functionality it performs. Therefore, the introduction of intentional services is an alternative for bridging the gap between low level, technical software-service descriptions and high level, strategic expressions of business needs for services.

Then, Aljoumaa *et al.* [1] propose an approach for building the Intentional Services Model (ISM) proposed by [12][28]. These authors [1] present an ontological based solution to help user discovering and formulating his needs. They propose a mechanism for matching user needs formulated in business terms as intentions with the intentions of services published in an extended registry.

Moreover, Mirbel e*t al.* [18] also adopt ontology and semantic web technologies for proposing intention-based service discovery mechanisms. They propose a semantic approach guided by the user intentions. In this approach, user requests are expressed using semantic Web technologies.

Then, Olson *et al.* [23] believe that by using intentions, services can be described on any arbitrary and useful level of abstraction. According to these authors, through an intention refinement algorithm, intentions can be used not only for describing services, but also for improving the performance of service discovery.

In addition, Baresi et al [2] propose an innovative intention-based approach to represent requirements and adaptation capabilities for service composition.

None of these works considers the notion of context, contrary to Bonino et al. [4]. These authors [4] propose an intention-based dynamic service discovery and composition framework that uses context information. Nevertheless, context information is used only for filtering the input of the user request.

All these works represent two different views of service orientation: (i) one view proposing a context-aware based approach. This view focuses on the adaptation of services according to the context information; and (ii) a second view focusing on an intention-based approach, proposing high level services. This view focuses on user intentions. The first view focuses on service discovery and composition on a highly dynamic environment. It does not consider why service is needed. More, it focuses especially on the context on which a service is valid or can be executed rather than the real use of the service and the purpose of the user.

The latter considers this question without considering the context in which this need emerges. The user requests a service with a specific intention. Although, this intention is more significant when considered in a specific context that can influence its satisfaction.

Questions such as "*why a service is useful in a given context*?" or "*in which circumstances a service need raises*?" remain unexplored. Thusn in order to explore both views; we have first to represent them in a semantic way. Thus, we propose a semantic descriptor of services that encompasses notions of context and intention. This description will enrich the service discovery and will improve the selection of the most appropriate service.

III. MOTIVATING SCENARIO

Bob works as a commercial in a company. He is responsible for preparing customer proposals. His company offered him different manners to prepare his proposals. When he is in the company, he has a direct access to the enterprise resources planning (ERP). He uses the service *prepare proposal* that allow him to write the proposal and send it to the customer via an e-mail. When he is outside, he needs first of all, to make a VPN connection that will allow him to access the ERP. Then, he has to write the proposal and finally send it via fax or e-mail to the customer. In this situation, Bob needs to know how to prepare his proposal depending on his context (if he needs a VPN connection, if he has an Internet connection, if he has a fax next to him, etc.). The information system provides several implementations that Bob needs to know. Such technical details seem too complicate for Bob, who would prefer just a service to prepare his proposal. Actually, Bob needs a transparent access to the service he is looking for, without any technical details concerning which implementation is available in a given context. In order to handle this problem, we propose to describe and to search for him services based on the intention they are supposed to satisfy, which is easier to understand for Bob than technical details about available implementations.

IV. PRELIMINARY CONCEPTS: INTENTION & CONTEXT

Before presenting details about our proposition, some concepts should be introduced, essentially the notion of *context* and *intention*. In this paper, we exploit the close relation between these two concepts. This is in order to enrich the service description and enhance the service discovery mechanism.

In the next part, we will introduce the notion of intention, define intentional services and present the intentional composition.

A. Intentional service at the glance

The term intention has several different meanings. According to [11], an intention is an "optative" statement expressing a state that is expected to be reached or maintained. The intention represents the goal that we want to achieve without saying how to perform it [11]. Bonino *et al.* [4] defines an intention as a goal to be achieved by performing a process presented as a sequence of intentions and strategies to the target intention. Even if they differ, all these definitions let us consider an intention as a *user requirement representing the intention that a user wants to be satisfied by a service without saying how to perform it* [22].

This intention represents the user request when he is looking for a service satisfying his needs. Aljoumaa et al. [1], present a mechanism, based on ontologies, that guide user when he formulates his intention. They present a methodology that help user to discover his needs and formulate it consequently.

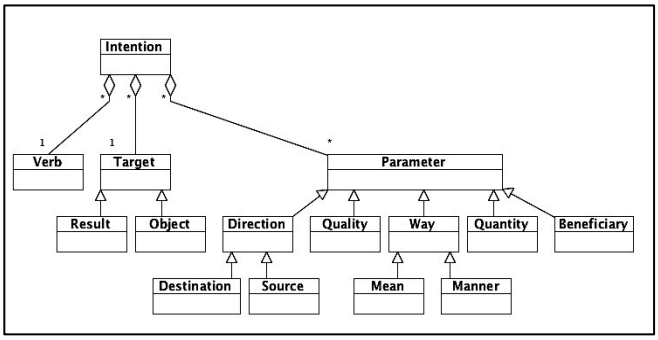


Figure 2. Intention template based on [28]

Thus, to ensure a powerful intention matching, we formulate the intention according to a specific template [12][28]. This template is defined based on linguistic approach [26]. This approach is inspired by the Fillmore case grammar [9] and its extensions by Dick [8]. It represents user and service requirements. In this template, an intention is expressed by a *verb*, a *target* and a set of optional *parameters,* as illustrated in Figure 2. The verb and target are mandatory, while the other parameters are optional and play specific roles with respect to the verb.

First, the *verb* exposes the action allowing the realization of the intention. Then, the *target* represents either the *object* existing before the achievement of the intention, or the *result* resulting from the intention satisfaction. The parameters are useful to clarify the intention and to express additional informational such as: direction, ways, quality, etc. The *direction* parameter characterizes the *source* and *destination* of the entities. From the one side, the *destination* identifies the location of the entities produced by the intention satisfaction. From the other side, the *source* identifies the initial location of the entities. In addition, the intention template represents the *ways* parameter. This parameter refers to the instrument of the intention satisfaction. It represents the *mean* and the *manner*. The *mean* describes the entity that serves as an instrument to achieve the intention, while the *manner* identifies an approach in which the intention can be satisfied. Finally, the *quality* parameter defines a property that must be reached or maintained [20].

In addition to intention template proposed on [12][28], we also consider the sense of a verb. The intention formalism is based on the verb as an element that expresses the actions, the states, the activities, etc.

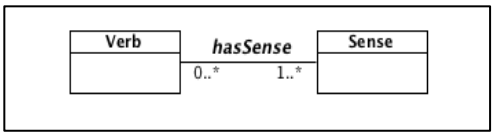


Figure 3. Sense of the verb

The same verb can have different senses depending on his use. For every intention verb, we attribute a set of senses, as illustrated in Figure 3. These senses indicate the meanings of this verb. For example the verb "*reserve*" has different senses such as: "*give or assign a resource to a particular person or cause*", "*arrange for and reserve (something for someone else) in advance*", etc.

1) Intentional services definition

As we mentioned above, an intentional service is represented as a service captured at a high-level, in business comprehensible terms. This service is described by the intention it can satisfy, i.e., according to an intentional perspective. A model of this intentional service is presented in [12][28]. These authors [12][28] present an intentional service model (ISM) that associate to each service an intention it can satisfy. ISM is composed of 4 facets, represented in Figure 4, namely the *service interface*, the *service behaviour*, the *service composition* and the Q*oS*.

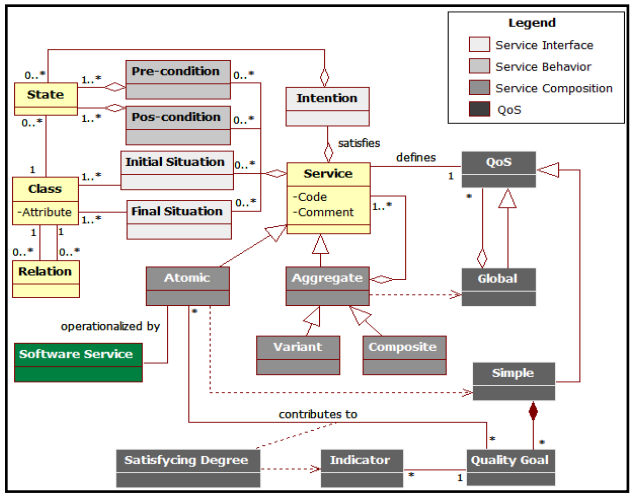


Figure 4. Intentional Service Model (ISM) [28]

First, the *service interface* represents the service that permits the fulfilment of an intention. This is based on an initial situation and terminating in a final situation. Then, the *service behaviour* specifies the pre and post conditions. The pre condition represents the sets of initial states required by the service for the intention achievement. The post condition represents the set of final states resulting from intention achievement. Next, the *service composition* represents the possibility of composing more complex intentions by combining lower abstraction level intentions. Next section gives more precisions about service composition. Finally, the *QoS* introduces the non-functional dimension of service. It represents the quality requirements associated with intentional services.

2) Intentional services composition

The intentional service model emphasises variability on the satisfaction of its corresponding intention. It allows the variability through the service composition. In the ISM model, an intentional service can be *aggregate* or *atomic*. First, aggregate services represent high-level intentions. These intentions can be decomposed in lower level one, helping business people to better express their strategic/tactical intentions.

Intentional composition admits two kind of aggregate services: a *composite* and a *variant*. While composite services reflect the precedence or succession relationship between two intentions, variant service correspond to the different manner to achieve an intention. This needs for variability is justified by the need to introduce flexibility in intention achievement [12][28].

According to [28], atomic services are related to operationalized intentions and can be fulfilled by SOA functional services. Atomic intentional services are then operationalized by software services. In contrast, aggregate services have high-level intentions that need to be decomposed in lower level ones till atomic intentional services are found.

Nevertheless, we advocate that this vision does not consider the evolution of service technology. This evolution can stand now for small pieces of software. This software encapsulates reusable functionalities, as well as for large legacy systems, whose complex process are hidden by technologies such as Web Services or ESB [31].

By considering that only atomic services can be operationalized by software service, ISOA architecture limits the reuse of such legacy systems under an intentional approach. Actually, legacy systems often encompass complex services. These systems subsume the satisfaction of multiple intentions or an intensive variability on their satisfaction. Moreover, such systems can be compared to aggregate intentions, but they cannot be assimilated to simple atomic intentions.

In this paper, we extend the vision originally proposed by [12][28]. We consider that both atomic and aggregate intentional services can be operationalized by software service, which can be also atomic or composite. As a consequence, both technical and intentional compositions are possible independently, allowing more powerful constructions. Besides, contrarily to [28], we do not consider that intentional service should be seen as a separate entity from technical service. Such separation leads to poor technical descriptions that are semantically incomplete, since they do not include an intentional description. We propose in this paper a full semantic descriptor, which considers service as a single entity with multiple dimensions: intentional, technical and contextual dimensions.

B. Context information description

Context information corresponds to a very wide notion. As we mentioned earlier, it is usually defined as any

information that can be used to characterize the situation of an entity (a person, place, or object considered as relevant to the interaction between a user and an application) [7]. The notion of context is central to context-aware services that use it for adaptation purposes. Context information can stand for a plethora of information, from user location, device resources [27], up to user agenda and other high level information [13]. Nevertheless, in order to perform such adaptation processes, context should be modelled appropriately. The way context information is used depends on what it is observed and how it is represented. The context-adaptation capabilities depend on the context model [19].

Thus different kinds of formalism for context representation have been proposed. Nevertheless, an important tendency can be observed on most recent works: the use of ontology for context modelling [19]. According to [19], different reasons motivate the use of ontologies, among them their capability of enabling knowledge sharing in a non-ambiguous manner and its reasoning possibilities. This tendency follows the evolution of context-aware services, which adhere, in their majority, to a semantic description of such services. In this paper, we also adhere to this tendency, adopting an ontology-based context modelling based on [27].

Reichle *et al.* [27] define context information based on three main concepts: 1) the ***entity*** referring to the element to which the context information refers; 2) the ***scope*** identifying the exact attribute of the selected entity that it characterizes; and 3) the ***representation*** used to specify the internal representation used to encode context information in data-structures.

According to this context model, we directly associate the scope that we observe with the entity that the context element refers to. This let us consider that, in order to have the value for a given scope, we have to observe his corresponding entity. However, this represents an ambiguity since some scopes are not directly related to a precise entity. For example, if we want to represent the humidity around a given user, this information can not be captured by observing the *user* but rather the *environment*.

Therefore, in order to make this context model more meaningful, we believe that we must separate clearly the notion of *entity* that we want to represent from the *property* that we want to observe.

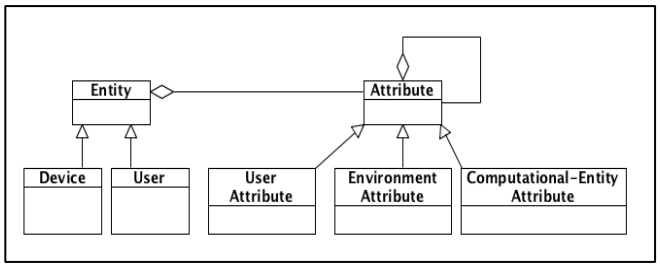


Figure 5. Context Model

The Figure 5 illustrates our proposed context model. Each context information is identified by two important concepts, the *entity* and the *attribute*. The distinction between these two concepts is adopted in order to not mix up the *entity* to which the context information refers to (e.g., user, device, etc.) with the *attribute* that characterize the property that we want to observe. The attribute represents a piece of context information about the environment (location, time...), a user (profile, role...) or a computational entity (resource, network....).

Moreover, this context model is based on a multi-level ontology representing knowledge and describing context information (Figure 6). It provides flexible extensibility to add specific concepts in different domains. All these domains share common concepts that can be represented using a general context model, but they differ in some specific details.

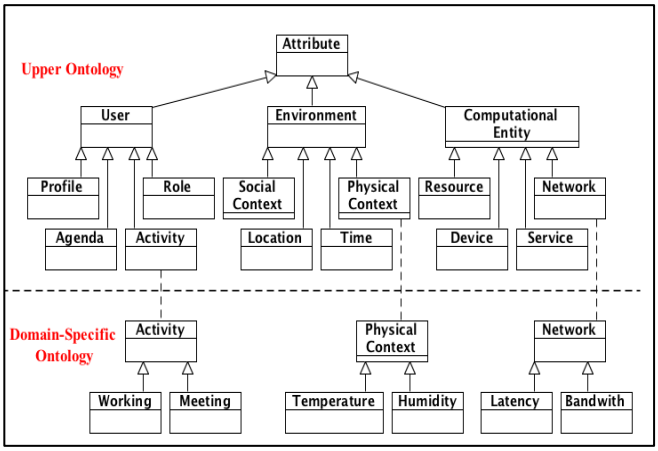


Figure 6. Multi-Level Context Ontology

According to [19], this context model presents context according to three main categories: (i) *environment context* representing contextual information about user location, time, social context, etc.; (ii) *user context* that represents user profile, agenda, Role, activity, etc.; (iii) *computational entity context* including contextual information related resource, network, etc.

This two-level ontology consists in an upper level, defining general context information (e.g., profile, activity, location, network, etc.), and a lower level, with more specific context information (temperature, latency, etc.). Therefore this separation enhances the reuse of general context information and provides flexibility to add domain-specific knowledge.

Besides, in our opinion, context information does not have all the same importance. It can differ from a user to another according to their preferences. Consequently, we propose to associate with context attributes the notion of '*weight*'. Our purpose is to clarify the importance of a context attribute according to the domain and to the user preferences. The profile context model, presented in the Figure 7, consider this by allowing to each entity to have a profile specifying this weight.

A *profile* represents the user preferences regarding context information. These user preferences are represented as a *profile* assigned to each *entity*. It allows the definition of a *weight* that the profile owner allocates to each context

attribute. This weight, whose value is between 0 and 1, represents the importance of an attribute to a given entity.

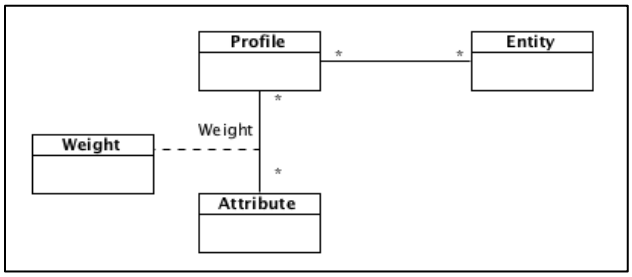


Figure 7. Profile Context Model

The purpose here is to highlight the real importance of a context attribute according to user preferences. The importance of the attribute is proportional to its weight. It decreases if the value affected decreases, and it grows if this value grows. The *weight* can then be used for matching purposes, and notably during the matching between the user current context and service context conditions. By proposing this profile model, we intend to promote context attributes that are seen as most relevant ones for a given user. For example, by considering this profile, a context attribute having a lower weight (i.e., that is not particularly interesting for the user) will be less influent for calculating the context matching score, than another attribute with higher weight. Even if this context attribute participates in the matching process, the weight assigned to it will decrease its importance, and consequently the context score will be calculated according to user preferences.

Therefore, a user (or even a system administrator) may define, for an entity, a set of profiles representing his preferences. Through this notion of profile, it is possible to enhance this selection of the most appropriate service that can interest the user.

The next section describes how all dimensions can coexist in the proposed service semantic descriptor.

V. PROPOSITION: PUTTING EVERYTHING TOGETHER: CONTEXT-AWARE INTENTIONAL SERVICES

The latest research in service oriented computing recommends the use of the OWL-S for semantically describe services [34]. Even if OWL-S is tailored for Web services, it is rich and general enough to describe any service [34]. OWL-S [16] defines web service capabilities in three parts representing interrelated sub-ontologies named service profile, process model and grounding. The *service profile* expresses what the service does. It gives a high-level description of a service, for purposes of advertising, constructing service requests and matchmaking. The *process model* answers to the question: how is it used? It represents the service behaviour as a process and describes how it works. Finally, the *grounding* maps the constructs of the process model onto detailed specification of message formats, protocols and so forth (often WSDL).

The OWL-S represents a flexible and extensible language, as demonstrated by works such as [14][34]. Similar to these works, we propose to extend service description in OWL-S by including information concerning both context and intention that characterize a service.

A. Describing context-aware intentional service in OWL-S

In this section, we present our extension of OWL-S. This extension includes: (i) intentional information about services; and (ii) contextual information about services conditions of execution.

In the following part, we present the intentional extension of OWL-S.

1) Describing service intentions

According to an intentional perspective, a user requires a service because he has an intention that the service is supposed to satisfy. Hence, the importance of considering user intentions emerges on service orientation. Such new dimension is central to service definition.

Thus, we propose to enrich OWL-S service description with the intention associated to it. We extend OWL-S, including on it the intention that a service can satisfy. This is done by adding a new sub-ontology, which describes the intentional information of the service.

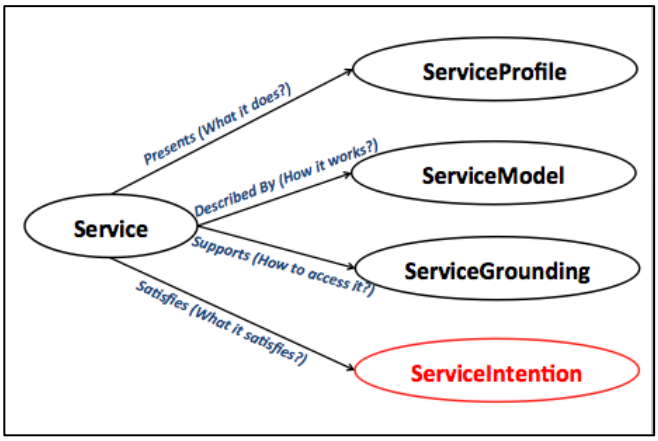


Figure 8. Service intention Description in OWL-S

The Figure 8 illustrates our intentional extension of owl-s. The proposed property *satisfies* is a property of *Service*. The class *ServiceIntention* is the respective range of this property. Each instance of *Service* will *satisfy* a *ServiceIntention* description. The *ServiceIntention* provides the information needed to discover the appropriate service in order to satisfy a specific *intention*. The service intention presents "what the service satisfies", in a way that is suitable to determine whether the service fulfills user intention. This part of the service description presents the principal intention of the service. This intention is formulated according to a specific template [28].

This description differs from the last service intention description presented in [20]. One can notice that the service intention description presented in this present paper is defined in a separate sub-ontology. It is related to the service description instead of describing it as a service parameter in the service profile description. This change is due to several reasons: (i) since the service intention, in our proposition,

represents an important aspect of service definition. It will affect the service discovery process, and is more meaningful and clear to describe it in a separate block; (ii) the evaluation performed on both descriptions demonstrated that the analysis of a service description with separately intention description on a sub-ontology is more efficient than the proposal presented in [20]; (iii) the analysis of a service description by a matching algorithm that ignores intention description is easiest: if this description is separated from the rest (in other words, extended services remain compatible with old registry).

```
1   ...
2   <service:Service rdf:ID="PREPARE_PROPOSAL_SERVICE">…
    </service:Service>
3   <profile:Profile rdf:ID="PREPARE_PROPOSAL_PROFILE">
4   ...
5       <eprofile:context
        rdf:resource="http://193.55.98.54/iSOA/
6       ExtensionOWL-S/ContextDescription.xml#condition1"/>
7       <iprofile:hasintention
        rdf:ID="INTENTION_PREPARE_PROPOSAL_INTENTION"/>
8   ...
9   </profile:profile>
10  <intention:Intention
    rdf:ID="INTENTION_PREPARE_PROPOSAL_INTENTION">
11      <intention:Verb rdf:resource="http://
        www.crinfo.univ-paris1.fr/iSOA/
        ExtensionOWL-S/Intention.owl#concept.intention.verb
        ">
12              prepare
13      </intention:Verb>
14      <intention:Target rdf:resource="http://
        www.crinfo.univ-paris1.fr/iSOA/ExtensionOWL-S/
        Intention.owl#concept.intention.target">
15          <intention:Object rdf:resource="http://
            www.crinfo.univ-paris1.fr/iSOA/ExtensionOWL-
            S/Intention.owl#concept.intention.target.object
            ">
16              proposal
17          </intention:Object>
18      </intention:Target>
19  </intention:Intention>
```

Figure 9. Example of describing service intention in OWL-S

In the Figure 9, a service is associated with the intention "prepare a proposal" (line 7, *<iprofile:hasintention>*). This intention *(<intention:Intention>)* is then described according the template "verb, target, parameters" (see lines 10-19), using the extended OWL-S elements. In this example, the verb *(<intention:Verb>)* is "prepare" and the target (*<intention:Target>*) represents the object "proposal" (*<intention:Object>*).

In the next section, we will present the extension of OWL-S including service contextual information.

2) Describing contextual information

An intention that a user wants to satisfy emerges in a given context. In our opinion, it has a closely relation between the notion of context and intention. This relation should be exploited. Thus, the user intention becomes less important and less significant if we did not take it with its context of use. According to this, we propose to extend the service profile. Our purpose is to allow service provider to define context information that characterize an intentional service.

For instance, let us consider the intention *prepare proposal* (described in the Section III). For this intention different implementations are available enabling users to search, prepare and send a proposal to the customer in different situations. This service can be particularly executed considering client *location*, type of the used *device* and type of the *network*.

A first implementation can be proposed considering the user is in the company (location). He writes his proposal from his personal computer (device) and sends it via fax to the customer. This user is connected via the Ethernet of the company (network).

A second implementation of the same service can be executed when the user is outside (location). He accesses, via his smart phone (device) with a 3G connexion (network), to a specific application allowing him to write a proposal and then send it via mail to the customer.

Each one of these implementations can be associated with a different context description. By considering such a description and the user current context, it is possible to select the most appropriate implementation in a transparent way for the user.

For example, the Figure 10 and Figures 11 illustrates an example of a context conditions description that can be associated to the first implementation of the *prepare proposal* intentional service.

```
1   <ctx:context
    xmlns:ctx="http://www.citypassenger.com/services/ContextSchema.
    xsd"
2   ...
3   <ctx:condition>
4       <ctx:contextElement>
5           <ctx:hasEntity
            resource="http://www.citypassenger.com/services/
            ContextModel.owl#concept.Entity.Person"/>
6           <ctx:hasAttribut
            resource="http://www.citypassenger.com/services/
            ContextModel.owl#concept.
            Attribut.Environment.Location"/>
7           <ctx:contextValueSet>
8               <ctx:contextValue>
9                   <ctx:hasAttribut
                    resource="http://www.citypassenger.com/services/
                    ContextModel.owl#concept.Attribut.Environment.
                    Location.PredefinedLocation"/>
10                  <ctx:valueSet>
11                      <ctx:valueElement>
12                          <ctx:operator
                            resource="http://www.citypassenger.com/
                            services/ContextModel.owl#Concept.
                            Operator.Equal"/>
13                          <ctx:value>Company</ctx:value>
14                      </ctx:valueElement>
15                  </ctx:valueSet>
16              </ctx:contextValue>
17          </ctx:contextValueSet>
18      </ctx:contextElement>
```

Figure 10. User context Description: Condition 1

First, the Figure 10 represents the condition that the "location" of the user is the "company". The *user* represents the *entity* to which the context refers (<ctx:hasEntity>). The *location* represents attribute that characterize the observed property of the context (<ctx:hasAttribute>). And the

company represents the observed value of the attribute (<ctx:value>).

Next, the Figure 11 represents another context condition. This illustrates that the "network" (attribute) of the "device" (entity) is "Ethernet" (value).

```
<ctx:contextElement>
    <ctx:hasEntity
    resource="http://www.citypassenger.com/services/
    ContextModel.owl#concept.Entity.Device"/>
    <ctx:hasAttribut
    resource="http://www.citypassenger.com/services/
    ContextModel.owl#concept.
    Attribut.ComputationalEntity.Network"/>
    <ctx:contextValueSet>
        <ctx:contextValue>
            <ctx:hasAttribut
            resource="http://www.citypassenger.com/services/
            ContextModel.owl#concept.Attribut.
            ComputationalEntity.Network.Connection"/>
            <ctx:valueSet>
                <ctx:valueElement>
                    <ctx:operator
                    resource="http://www.citypassenger.com/
                    services/ContextModel.owl#Concept.
                    Operator.Equal"/>
                    <ctx:value>Ethernet</ctx:value>
                </ctx:valueElement>
            </ctx:valueSet>
        </ctx:contextValue>
    </ctx:contextValueSet>
</ctx:contextElement>
```

Figure 11. User context Description: condition 2

Thus, and according to [20], contextual information can then be considered as part of the service description, since it indicates situations to which the service is better suited. However according to [14], context information cannot be statically stored on the service profile due to its dynamic nature. Context properties related to service execution can evolve, whereas service profile is supposed to be a static description of the service.

Thus, in order to handle dynamic context information on static service description, we adopt the approach [14]. This approach enriches OWL-S service profile with a *context* attribute, which represents a URL pointing to context description file. Since context information is dynamic, we opt to describe context element in an external file. Thus, this will allow service provider to easily update such context information related to the service description itself. The context description of a service describes, from the one side, the situation status of the requested service (environment in which the service is executed), and from the other side, the contextual conditions (requirements) to execute the service. Both information can be used for service discovery purposes. In the next section, we will describe briefly the composition of intention.

3) Composing intentions

Intention and context attributes described above intent to expose both aspects of a service notably for discovery purpose. Thanks to the OWL-S extension we propose, a service can be discovered either by intention it can satisfy, or by the context associated with this intention. In addition to these aspects, a third aspect should be exposed: the service variability. Such variability is expressed, in the intentional perspective, by the composition of intentional services. This indicates the decomposition of the service intention on lower level intentions. Thus, according to [20], (i) the technical composition of a service, described in OWL-S process model, represents software components. These components are combined to supply service operations; (ii) the intentional composition represents not only lower level intentions necessary to satisfy service intention, but also different possibilities the service offers for satisfying this intention [20].

The technical composition supplies technical elements necessary for service execution. Then, the intentional composition provides an understanding, from final user point of view, of the service and the diverse forms of satisfying service intention. Thus, we propose to extend OWL-S process model by including the specification of an intentional service process, as illustrated in Figure 12.

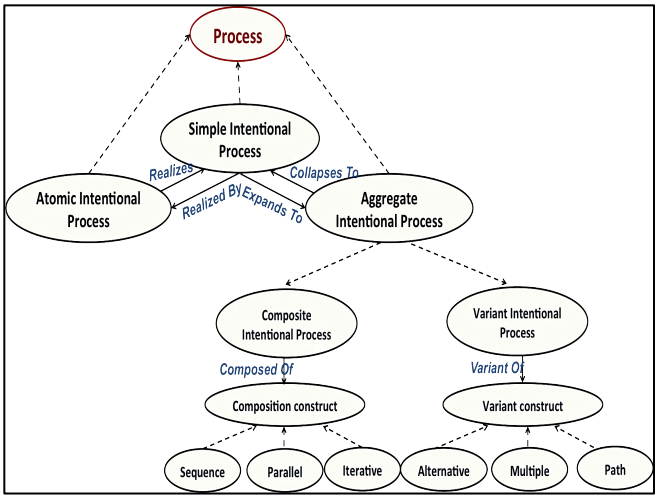


Figure 12. Composing intentions in OWL-S process Model

The Figure 12 presents the extension we propose for the process model. This extension considers two kinds of process: the atomic intentional process and the aggregate intentional process. It considers also a simple intentional process, which is used to provide an abstracted view that can be atomic or aggregate. A simple intentional process is realized by an atomic intentional process and expands into an aggregate intentional process. An aggregate intentional process can be either a composite intentional process or a variant intentional process.

First, the composite intentional processes reflect the precedence/succession relationship between their intentions. Such relationships are specified using composition constructs such as *Sequence*, *Parallel* and *Iterative*. The composition represents a *sequence* in which there is a sequential order between component processes, or a *parallel* in which components can run in parallel. The *iterative* construct is used when the satisfaction of an intention may require iterative execution of a given set of actions.

Then, the variability is represented by the variant intentional process, which uses constructs such as *multiple*,

alternative and *path*. The *multiple* construct offers a non-exclusive choice in the realization of the intention. It groups multiple simple processes, among them, at least one will be chosen. The *alternative* construct represents a process with an alternative choice. It regroups several simple processes that are mutually exclusive. Then, it builds a new process of the same level of abstraction but of higher granularity. And finally, the *path* construct offers a choice in how to achieve the intention of the aggregate process. It offers composite processes that are mutually exclusive.

```
<eiprocess:CompositeIntenionalProcess rdf:ID=
"http://www.crinfo.univ-paris1.fr/iSOA/ExtensionOWL-S/services/
PrepareProposal">
    <eiprocess: CompositeIntenionalProcessID>
        PrepareProposal
        </eiprocess: CompositeIntenionalProcessID>
    <eiprocess: CompositeIntenionalProcessName>
        prepare a proposal
    </eiprocess: CompositeIntenionalProcessName>
    <eiprocess: ComposedOf>
        <eiprocess: Sequence>
            <process: Components rdf:parseType="Collection">
                <eiprocess: AtomicIntenionalProcess
                rdf:ID="http://www.crinfo.univ-paris1.fr/iSOA/
                ExtensionOWL-S/services/LaunchVPNConnection">
                <eiprocess: AtomicIntenionalProcess
                rdf:ID="http://www.crinfo.univ-paris1.fr/iSOA/
                ExtensionOWL-S/services/WriteProposal">
                <eiprocess: VariableIntenionalProcess
                rdf:ID="http://www.crinfo.univ-paris1.fr/iSOA/
                ExtensionOWL-S/services/SendProposal">
            </process: Components>
        </eiprocess: Sequence >
    </eiprocess: ComposedOf>
</eiprocess: CompositeIntenionalProcess>
```

Figure 13. Example of OWL-S Intentional composition: Composite Intentional Process

For instance, let us consider the example of the service (described in the Section III) satisfying the intention *prepare proposal*. This service is offered by Bob ERP. It allows him to write a proposal and send it to the customer.

This service is described as a composite service (<eiprocess:CompositeIntentionalProcess> in Figure 13). It represents a *sequence* (<eiprocess:Sequence>) between the atomic intentions (<eiprocess:AtomicIntentionalProcess>) *launch VPN connection* and *write proposal*, and the variant intention (<eiprocess:VariableIntentionalProcess>) *send proposal,* as illustrated in Figure 13.

This latest variant represents a path (<eiprocess:Path>) between the atomic intentions *send proposal by mail* and *send proposal by fax* , as illustrated in Figure 14. From a technical point of view, this service is composed by multiple ERP functionalities, described in OWL-S process model. Such description is beyond the scope of this paper, since no modification has been proposed for technical composition on OWL-S process model.

In our vision, an aggregate intentional service, which is composed of other intentional services, can be associated with a software service. This software service can be also composite of other technical services. This extends the vision of [1][12][28] that consider that only atomic intentional service can be operationalized by a software service.

According to them, aggregate intentional service need to be decomposed till an atomic intentional service is found. These authors do not take into account, for example, the software encapsulating reusable functionalities.

```
<eiprocess: VariableIntenionalProcessrdf:ID=
"http://www.crinfo.univ-paris1.fr/iSOA/ExtensionOWL-S/services/
SendProposal">
    <eiprocess: VariableIntenionalProcessID>
        SendProposal
    </eiprocess: VariableIntenionalProcessID>
    <eiprocess: VariableIntenionalProcess Name>
        Send Proposal
    </eiprocess: VariableIntenionalProcessName>
    <eiprocess: VariantOf>
        <eiprocess: Path>
            <process:Components rdf:parseType="Collection">
                <eiprocess: AtomicIntenionalProcess
rdf:ID="http://www.crinfo.univ-paris1.fr/iSOA/ExtensionOWL-S/services/
SendProposalByMail">
                <eiprocess: AtomicIntenionalProcess
rdf:ID="http://www.crinfo.univ-paris1.fr/iSOA/ExtensionOWL-S/services/
SendProposalByFax">
            </process: Components>
        </eiprocess: Path>
    </eiprocess: VariantOf>
</eiprocess:VariableIntenionalProcess >
```

Figure 14. Example of OWL-S Intentional composition: Variant Intentional Process

Thanks to the OWL-S extension proposed here, we enable the description of intentional composition, from final user point of view. This extension exposes the variability representing different manners to satisfy user intentions. The intentional composition description allows a service discovery guided by intention, presented at a high level.

4) Describing Service Resource

A service, with an intentional description, can be seen as an *intentional service*. Each intentional service acts as a fragment of process implemented by the software service. It handles input information in order to satisfy its corresponding intention and resulting in some output information. Input and output of an intentional service describes, respectively, an initial and a final situation. These situations are expressed as set of states over resources handled by the service. Such initial and final situations are important for intentional composition. This is because they are supposed to guide the satisfaction of high-level intention associated with the aggregate service.

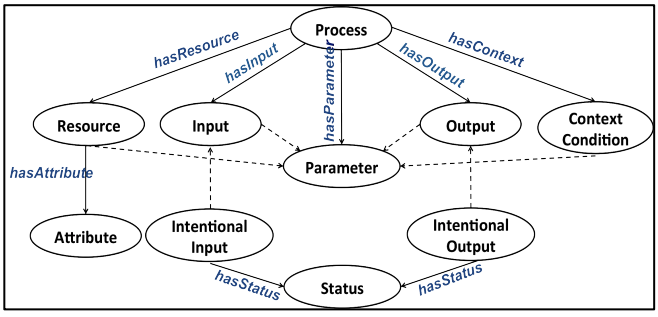


Figure 15. Resource description in OWL-S process model

According to this, we introduced the notion of resource on OWL-S, as presented in Figure 15. A *resource* represents a class of objects, with its corresponding attributes, that are manipulated by an intentional service. For instance, a service implementing the intention "*prepare proposal*" will manipulate a "*proposal*" resource, with a "*preparation state*" attribute. Then, when the resource is used as intentional input or output parameter, a *state* can be assigned to the resource. The element "*state*" allows then attaching values to each resource attribute.

The Figure 16 illustrates the resource "*proposal*" used as intentional output with a state defined by the attribute "*preparation state*" with the value "*done*".

```
<process:hasOutput>
    <eiprocess:IntentionalOutput
    rdf:ID="http://193.55.96.54/iSOA/service.owl#
    resource"/>
        <eiprocess:IntentionalOutputObject
        rdf:ID="http://193.55.96.54/iSOA/service.owl#
        resource.proposal">
        proposal
        </eiprocess:IntentionalOutputObject>
        <eiprocess:hasState>
            <eiprocess:StateAttribute
            rdf:ID="http://193.55.96.54/iSOA/service.
            owl#resource.attribute.preparation">
                <eiprocess:StateName> Preparation
                </eiprocess:StateName>
                <eiprocess:StateValue> Done
                </eiprocess:StateValue>
            </eiprocess:StateAttribute>
        </eiprocess:hasState>
    </eiprocess:IntentionalOutput>
</process:hasOutput>
```

Figure 16. Intentional output in OWL-S process model

Besides, we believe that variability on intention achievement may depend on external factors. These factors concern context information. Each variant may have context conditions in which it is most appropriate to use it. For each variant, we attribute a context conditions description. This context description represents in which circumstances it is most appropriate to use it.

Thus, in order to consider context influence on intentional variants, we propose including context information also on the process variability description. We associate contextual conditions to each process variant described at the intentional level. This context description will enhance the variability dynamic of intentional process.

Thus, such extension can help to choose the variant according to context conditions. Thus, we extend OWL-S process model by including on it a contextual condition through the element "*context*" (<eprofile:context> in Figure 17). Similar to context element associated with service profile, this element points out to an external file containing context description (see Section V.A.2), referring to context conditions that apply to a given variant.

```
<process:CompositeProcess rdf:ID="Prepare Proposal">
    <process:hasContestCondition
    rdf:ID="http://www.crinfo.univ-paris1.fr/iSOA/
    ExtensionOWL-S/service.owl#resource">
        <eprofile:context
        rdf:resource="http://www.crinfo.univ-paris1.fr/iSOA/
        ExtensionOWL-S/ContextDescription.xml#conditions1"/>
    </process:hasContestCondition>
</process:CompositeProcess>
```

Figure 17. Context Condition on OWL-S process model

For example, the he Figure 17 illustrates this OWL-S process model extension through a contextual condition pointing out the context description file.

B. *Context-Aware Intentional Service registry: Implementation*

In this section, we will present the implementation of our semantic service descriptor. Then, we will introduce our service discovery.

1) *Overview*

In this paper, we present a semantic enriched service descriptor. In order to demonstrate feasibility, we implemented a semantic service registry. This takes into account an enriched service descriptor based on the extension of OWL-S described in this paper. This description provides comprehensive specifications of a service. This specification is based on the intention it satisfies and the context conditions in which it is valid and executed. This extended service description is then tested and used by a context-aware intentional service discovery process (detailed in the Section V.B.3). The purpose is to find the most appropriate service according to a given request.

The Figure 18 shows the architecture of our enriched registry application [33]. The interface *ServiceManager* represents the entry point to the application. It offers a set of methods allowing ontology management and service discovery and selection. The implementation of this interface holds two references of the *PersistenceManager* and the *SearchEngine* interfaces. Both implementations use the Strategy Pattern in order to provide a flexible change of the strategy. Then, this can facilitate the addition of new persistence or/and matchmaker implementations. To load the right strategy, the application uses a properties file in which it is stated the strategy class to use.

The *SerachEngine* uses a *MatcherFacade* interface that acts as a façade between the SearchEngine and the API to operate service descriptions. The *PersistenceFacade* interface acts as a façade between the PersistenceManager and the database to access service and ontology descriptions.

Thus, our proposed semantic service registry can be divided into two core parts. The first one is the *persistence* package. It handles ontologies and service descriptions. The second one is the *search engine* package. It is in charge of searching an appropriate service for a given request, based on the extended service description.

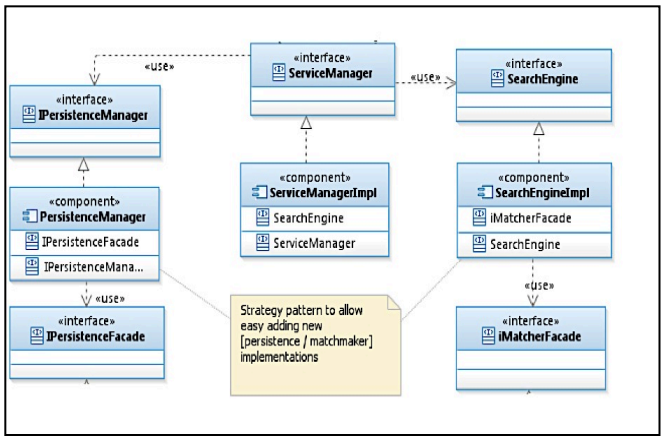


Figure 18. Semantic Service Registry Architecture

In the next sections, we explain the different used ontologies and models. They are implemented in order to create the enriched semantic service registry and to implement the extended OWL-S descriptor proposed here. Then, we explain how this extended service description is used by the service discovery manager. And how it can find the most appropriate service that fits user request in his current context.

2) Implemented models for a semantic service descriptor

The OWL-S description is based on a set of ontologies. This ontologies supplies service providers with a core set of constructs. It describes the properties and the capabilities of their services. In order to enhance this description and implement the semantic descriptor proposed in this paper, we extend OWL-S ontologies: (i) *Service.owl* (ontology providing the service profile) and (ii) *Profile.owl* (ontology providing a service grounding). Then, we add a new ontology, called *Intention.owl.* This ontology contains elements and concepts necessary to describe the intention that a service is able to satisfy. Then a new ontology named *ExtendedService.owl* brings together service and intention ontologies on this new service descriptor. The *ExtendedProfile.owl* defines the structure of the elements describing the service profile and the intention this service satisfies. In this description, we add an ObjectProperty *has_Intention* and the context information by adding a new DataProperty *context* (pointing to the context description file).

Based on the OWL-S API Mindswap [41], we develop an OWL-S extension API in Java. This extension implements the service description according to his intention and context. In order to evaluate the proposed implementation, the service retrieval test collection OWLS-TC2 [42] is used. Although, OWLS-TC contains only basic service descriptions based on OWL-S. It does not have any information about the service intention and context conditions. We preferred this test collection because it provides a large number of services from several domains, test queries and relevant ontologies. The collection is intended to support the evaluation of the performance of OWL-S service matching algorithms. OWLS-TC [42] provides 576 semantic Web services written in OWL-S 1.0 and OWL-S 1.1 from 7 domains (education, medical care, food, travel, communication, economy, weapon).

Based on this collection, we have implemented our presented semantic service descriptor. We use it in order to extend OWLS-TCS service descriptions with intentional and contextual description. Besides, we develop an interface called OWL-S extended API. This interface allows us to load from the OWLS-TC a service description and enrich it with contextual and intentional.

Thus, all this models and service descriptions are then used for a context-aware intentional service discovery process. The purpose is to search the most interesting service according to user request and context. This is detailed in the next section.

3) Context-Aware Intentional Service Discovery

With the variability and the diversity of services that are potentially available to the user in a pervasive environment, we propose a mechanism for services discovery. This mechanism take into account the user context and intention. It is based on the presented semantic descriptor of services and on a matching algorithm that we detail below. The purpose of presenting this algorithm here is to illustrate potential application of the semantic description we propose for service discovery.

The service discovery process is launched when the user sends his request representing the intention he wants to be satisfied. Once the request submitted, we load the collected user current context file. The Service Discovery Matcher loads all the semantic description of the available services (described using our proposed extension of OWL-S). Then, launches the matching process on all the available services.

The matching is a two-step process, illustrated on Figure 19. First of all, we match the user intention with the intention that the service satisfies. Second, all service context conditions are matched with the user current context (step 1.2). Finally, we calculate the degree of match between the user request and the provided service (the sum of intention and context degree of matching). Next, we add the service with its obtained score in a list (step 1.3). These steps are done for all the available services. Then, from the resulted list, we select the service having the highest score (step 2 on Figure 18).

The most important steps in this matching process are thus, the *intention matching* and the *context matching* (step 1.1 and 1.2 on Figure 19).

The intention matching process compares semantically the *verb* and the *target* of the user intention with those from service intention. From the one side, we compare the user intention verb with the service intention verb. This semantic comparison is based on a verb ontology. This verb ontology contains a set of verbs, relations between them (synonym, hyponym, hypernym) and their meanings in a specific domain. From the other side, we compare semantically the user intention target with the service intention target based on domain-specific ontologies. This domain-specific

ontology represents a set of possible targets in a specific domain.

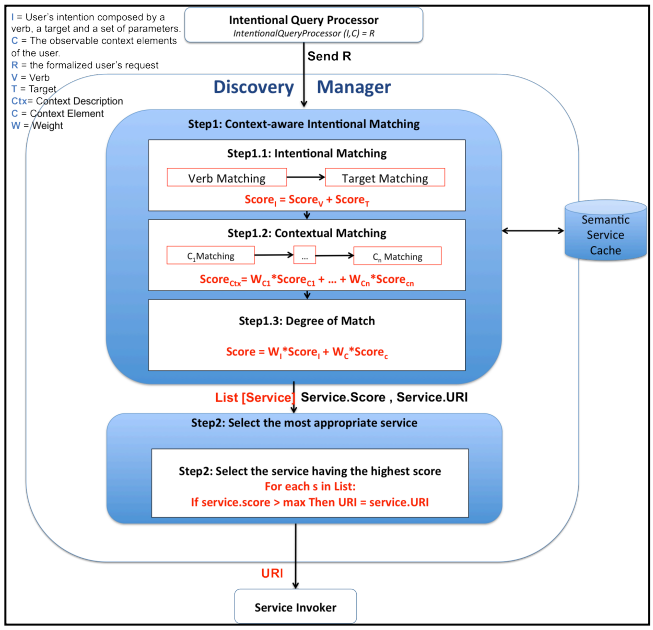


Figure 19. The context-aware intentional service discovery

Then, the context matching process, based on the context model, matches the different context element values. These values represent the context conditions of the service with those from the user current context.

This service discovery mechanism taking into account the notion of context and intention is well detailed in [22].

In order to evaluate our proposition, we first implement our semantic descriptor. Then, we propose to users an interface that allows them to load from the OWLS-TC a service description and enrich it with contextual and intentional information.

In our architecture, the *ServiceManager* module (Figure 18) represents the entry point for our discovery process. It supplies search methods for client applications. Two different search methods are offered. The first one proposes only the best ranked service. The second one proposes a list of all suitable services with their matching scores. This method is interesting because it allows the requester to observe the score of the different services and then decide whether the service really fits the request. Furthermore, the complete list allows the requester to make his own choice of which service is the best for him [33]. For example, if there is more than one service with a top score, the requester could examine each of them and decide by himself which one he wants to use. The *searchService*, on the other hand, is a simple method, that returns only the top service without any score. This method is interesting whether the client is not interested by interacting with a list and just needs a simple and fast result.

The *searchService* is based on a *Matchmaker* interface representing a discovery service matching algorithm. In our implementation, the matchmaker is easily replaceable in order to support multiple discovery processes. Thus, to select the matchmaker that should be launched with the application, we have to set up the properties file. This can be done by adding the name of the needed matcmaker.

Two main implementations of matchmaker have been proposed. First, we implement a basic matching algorithm that we called the *BasicMatchmaker*. This matchmaker is based on the input and output information. According to the user input and output, the *BasicMatchmaker* will be in charge of searching and selecting the best service [33]. Then, we implement our proposed context-aware intentional service matching algorithm (Figure 19) using OWL-S extended API, Jena [40] and the reasoner Pellet [39]. ***Jena*** [40] is a Java framework for building Semantic Web applications. It provides a programmatic environment for RDF, RDFS and OWL, SPARQL and includes a rule-based inference engine. We mainly use this framework for persistence purposes: reading, writing and verifying ontologies. Second, *Mindswap OWL-S API* [41] provides a Java API for programmatic access to read, execute and write OWL-S service descriptions. The extension of this API (OWL-S extended API) is used to operate on the service descriptions, such as getting the intention and the context of each service in order to calculate its match score. Finally, *Pellet* [39] is an OWL reasoner for Java. We adopt Pellet mainly due to its good performance and widely usage.

The implemented context-aware intentional service algorithm is called the *ContextIntentionMatchmaker*. It calculates a score based only on the user and service intention and context. This matchmaker is based on two classes *ContextMatching* and *Intentionmatching*, which are in charge of calculating respectively the context and intention scores. By separating score computation, it is possible to easily disable one of these classes in order to evaluate separately the impact of context or intention matching on the core.

The *ContextIntentionMatchmaker* demonstrates how the search method works and how a more sophisticated matchmaker can be implemented and used in the application.

In order to evaluate the validity of our context-aware service discovery algorithm and the impact of the enriched service descriptor, we present the results experiments and the evaluation of our proposition in the next section.

VI. EVALUATION

As mentioned earlier in this paper, we generate a semantic repository. This repository contains a set of extended service descriptions based on the extended OWLS-TC2 [42]. We choose the Travel domain for the test. It represents about 200 service descriptions. We enrich those descriptions by intentional and contextual information related to each service. Then, the evaluation has been performed under an Intel Core 2 Duo 2,26 GHz CPU with 2Gb of main memory.

The purpose of our experiments is to evaluate the validity of our algorithm and the feasibility of our extended service descriptor. Our objective is to evaluate 1) whether the processing time is reasonable: *scalability;* 2) whether the

algorithm effectively select the most appropriate services: *result quality*.

A. Scalability

We measure the scalability of our service discovery algorithm with respect to the number of services and the capabilities of the laptop, by measuring the average processing time.

We implement a bash script that runs our implemented service discovery algorithms. This script returns the response time in second. We launch it several times and then calculate the average response time.

The result of the service discovery performance is shown in Figure 20. We evaluate the response time of three types of service discovery: (i) input/output service discovery (actually the BasicMatchFacade) (IO); (ii) intentional service discovery (implemented by the Context Intention MatchMaker with the context matching score disabled) (I) and (iii) context-aware intentional service discovery (IC). The average response time is measured with different quantity of services (10, 30, 60, 100 and 200 files).

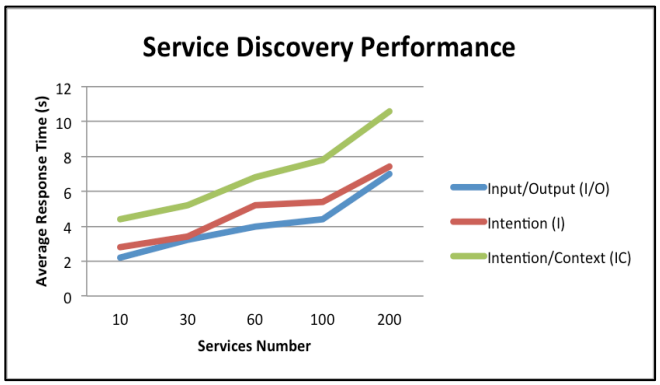


Figure 20. The evaluation of the response time of different service discovery

For example, for 60 services, the (IO) takes 4 s to select the desired services, while (I) takes 5,2 s and (IC) takes 6,8 s. The graph clearly illustrates that the context-aware intentional service discovery algorithm (IC) takes longer to process services. However, this difference on execution time does not represent a serious time difference from a user perspective (from 7s, for the faster algorithm, up to 10,6 s for the slower for 200 services). Actually, we notice that the service discovery based on input and output goes faster for selecting services. But, the response time of our algorithm still represents a reasonable processing time for selecting the most appropriate service.

As the next section demonstrates, our algorithm offers better results than input/output algorithm, since we proceed to service selection also according to user context and intention.

B. Result Quality

In order to measure the quality of the result, we cover the two most useful evaluations: *precision* and *recall*. These two metrics are defined in terms of a set of retrieved services and a set of relevant services. The *precision* represents the proportion of retrieved services that accurately matches user intention in a given context. It is calculated according to the formula (1). The *recall* represents the proportion of relevant services that are retrieved. It is calculated according to the formula (2).

$$Precision = \frac{|\{relevant\ services\} \cap \{retrived\ services\}|}{|\{retrieved\ services\}|} \quad (1)$$

$$Recall = \frac{|\{relevant\ services\} \cap \{retrived\ services\}|}{|\{relevant\ services\}|} \quad (2)$$

Thus, to evaluate this two metrics, we formulate five user requests. These requests represent the user intention in a given context, as illustrated in Table I. These requests are relatives to the travel domain. They are formulated and inspired from the available service descriptions.

We choose to represent some requests that can be accurately matched with available services (Exact). Besides, we add some requests that can have a good matching score with the available services (Not Exact).

These requests are formulated, as mentioned above, in order to validate the correctness of our IC algorithm. Moreover, our purpose is to verify if it really returns what it is used to return.

For evaluation purposes, we adopt a scenario from the *travel* domain corresponding to the service set used for testing. In this scenario, the user wants to practice a sport during his holiday. He is looking for surfing or hiking sport. Thus, he searches a destination where he can practice such sports. Then he wants to reserves a hotel or a BedAndbreakfast room for the period. Based on this scenario, we propose five requests with different context elements in order to evaluate the result quality of our implemented context aware intentional service discovery.

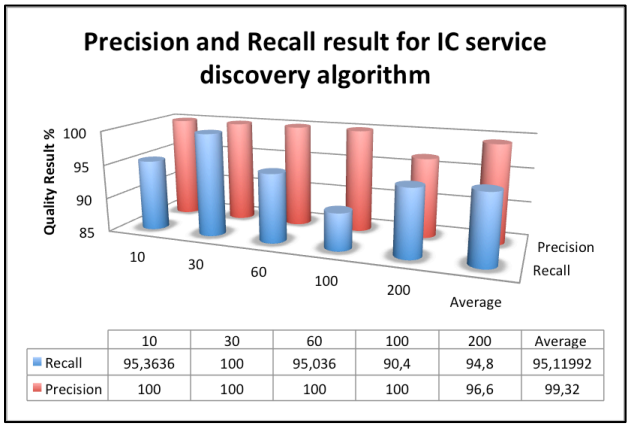
TABLE I. USER REQUEST WITH CURRENT CONTEXT

Intention	Context
Reserve Hotel	- Age >=18
Reserve BedAndBreakfast	- Age >= 18 - Season = Summer
Locate Sport Destination	- Age >= 18 - Season = Summer - City = Germany
Search Surfing Destination	- Age >= 18 - Season = Summer - Surfing Level = Beginner - Weather = not disturbed
Search Hiking	Age >= 18 - Hiking Level = Confirmed - Weather = not disturbed - Health = Good

Through the experiments, we could observe that the precision and recall is most important when considering the user intention and service context in the service discovery.

The result in Figure 21 shows that we obtain about 99 % of precision and about 95 % of recall for the 5 randomly chosen requests. These results are then compared to those obtained by IO service discovery algorithm illustrated on Figure 22. This figure shows that we could obtain an interesting recall 95,2% but a lower precision, which is about 50%.

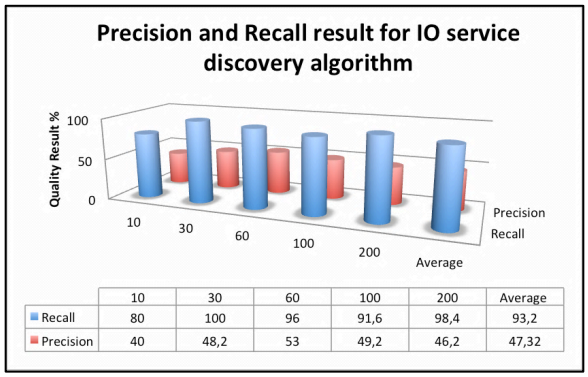
From the one side, the 95% of recall obtained by the IO algorithm is circumstantial since this algorithm is not able to select a service adapted neither to user context nor to his intention. In fact, the IO service discovery algorithm can only return all the service related to the request with a high rate of "false-positive" (indicated by precision).



	10	30	60	100	200	Average
Recall	95,3636	100	95,036	90,4	94,8	95,11992
Precision	100	100	100	100	96,6	99,32

Figure 21. The result quality of the context-aware intentional service discovery

The comparison between the Figure 21 and the Figure 22 illustrates that our proposed IC service discovery algorithm presents a more interesting result and a high level quality of results. This good result is due to 1) the use of an intention that describes the user real need; and 2) the use of context that makes the service selection most appropriate to the user (by selecting only the services that are valid and that can be executed in the user current context).



	10	30	60	100	200	Average
Recall	80	100	96	91,6	98,4	93,2
Precision	40	48,2	53	49,2	46,2	47,32

Figure 22. The result quality of the input/output service discovery

Thus, these results demonstrate that the IC algorithm is able to find all or almost services that can fulfills user intention in a given context, with the lowest rate of "false-positive"

VII. CONCLUSION AND FUTURE WORKS

In this paper, we considered context-aware and intention-based service orientation as complementary approaches that should not be isolated from each other. We explain our belief that an intention is only meaningful when considering it in a given context. Moreover, we believe that a context description is only meaningful when associated with other intention. We proposed, consequently, to enrich OWL-S service description. This extension includes the description of the intentions a service can satisfy. It includes also the context in which this intention is meaningful, context in which service is (or can be) executed.

From the one side, we enriched service description with knowledge about intentions and composition of intentions that are meaningful for final users, who request the service. From the other side, we enriched this service description with context information necessary for adapting such service.

By proposing such a semantic descriptor of service, we enable the expression of services that can adapt themselves to context of use and that represent a formulated user requirements. The service discovery process will exploit this extended description in order to enhance the satisfaction of the user request. By exposing both aspects of a service, we could develop a context-aware intentional service registry. From the one side we implement different models and ontologies needed by our proposed semantic service descriptor. From the other side, we implement a context aware intentional service discovery that illustrates how the semantic descriptor we propose can be exploited for discovery purposes.

The evaluation of our implementation demonstrates that our extension of the service description (by adding the context and intention information) makes the description more meaningful and the service discovery more precise and appropriate to the user needs.

In order to progress on this sense, our next step is to improve the implementation and analysis the results of our proposed semantic descriptor and context-aware intentional service discovery mechanism. Then, we expect to evaluate our service discovery mechanism in a more interesting real world scenario. Besides, these experiments will be tested on a more important number of services.

Based on these results, our efforts will be then focused particularly on the service prediction. Given the large amount of existing services and user needs, our purpose is to help users. We opt to propose them personalized services, without their demand, that can interest them according to their history and current context.

REFERENCES

[1] K. Aljoumaa, S. Assar, and C. Souveyet, "Reformulating User's Queries for Intentional Services Discovery Using an Ontology-Based Approach", 4th IFIP Int. Conf on New Technologies, Mobility and Security (NTMS), Paris, France, pp. 1-4, 2011

[2] L. Baresi and L. Pasquale, "Adaptive Goals for Self-Adaptive Service Compositions," IEEE Int Conf on Web Services (ICWS), pp. 353-360, 2010

[3] S. Ben Mokhtar, D. Preuveneers, N. Georgantas, V. Issarny, and Y. Berbers, "EASY: Efficient semAntic Service discoverY in pervasive computing environments with QoS and context support", Journal of Systems and Software 81(5), pp. 785-808, 2008

[4] L.O. Bonino da Silva Santos, G. Guizzardi, L.F. Pires, and M. Van Sinderen, "From User Intentions to Service Discovery and Composition," Proceeding ER'09 Proceedings of the ER 2009 Workshops (CoMoL, ETheCoM, FP-UML, MOST-ONISW, QoIS, RIGiM, SeCoGIS) on Advances in Conceptual Modeling – Challenging Perspectives, pp. 265-274, 2009

[5] J. Brnsted, K. Hansen, and M. Ingstrup, "Service Composition Issues in Pervasive Computing," IEEE Pervasive Computing, vol. 9(1), pp. 62 -70, 2010

[6] U. Keller, R. Lara, A. Polleres, I. Toma, M. Kifer, D. Fensel, "D5.1v0.1 WSM Web Service Discovery", available in http://www.wsmo.org/ 2004/d5/d5.1/v0.1/20041112, November 2004

[7] A. Dey, "Understanding and using context," Journal Personal and Ubiquitous Computing, vol 5(1), pp. 4-7, 2001

[8] S.C. Dik, "The theory of functional grammar," Foris publications, Dodrecht, Nederthlands, 1989

[9] C.J. Fillemore, "The case for case, in Universals in linguistic theory," Holt, Rinehat and Winstonc inc, E.Bach/R.T.Harms (eds), 1968

[10] V. Issarny, M. Caporuscio, and N. Georgantas, "A Perspective on the Future of Middleware-based Software Engineering," In: Briand, L. and Wolf, A. (Eds.), Future of Software Engineering 2007 (FOSE), ICSE (Conf on Software Engineering), IEEE-CS Press, 2007

[11] M. Jackson, "Software Requirements and Specifications: A lexicon of practice, principles and prejudices," Addison Wesley Press, 256, 1995

[12] R.S. Kaabi and C. Souveyet, "Capturing intentional services with business process maps," 1st IEEE International Conference on Research Challenges in Information Science (RCIS), pp. 309-318, 2007

[13] M. Kirsch-Pinheiro, J. Gensel, and H. Martin, "Representing Context for an Adaptative Awareness Mechanism," G.-J. de Vreede; L.A. Guerrero, G.M.Raventos (Eds.), LNCS 3198 - X Workshop on Groupware (CRIWG 2004), 2004, pp. 339-348

[14] M. Kirsch-Pinheiro, Y. Vanrompay, and Y. Berbers, "Context-aware service selection using graph matching," 2nd Non Functional Properties and Service Level Agreements in Service Oriented Computing Workshop (NFPSLA-SOC'08), ECOWS. CEUR Workshop proceedings, vol. 411, 2008

[15] Z. Maamar, D. Benslimane, and N.C. Narendra, "What can context do for web services?," Communication of the ACM, vol. 49(12), 2006, pp. 98-103

[16] D. Martin, M. Paolucci, S. Mcllraith, M. Burstein, D. McDermott, D. McGuinness, B. Parsia, T. Payne, M. Sabou, M. Solanki, N. Srinivasan, and K. Sycara, "Bringing Semantics to Web Services: The OWL-S Approach," Cardoso, J. & Sheth, A. (Eds.), SWSWPC 2004, LNCS 3387, Springer, 2004, pp. 26-42

[17] S.A. Mcllraith, T.C. Son, and H. Zeng, "Semantic Web Services," IEEE Intelligent Systems, vol. 16, pp. 46-53, 2001.

[18] I. Mirbel and P. Crescenzo, "From end-user's requirements to Web services retrieval: a semantic and intention-driven approach," J.-H. Morin, J. Ralyte, M. Snene, "Exploring service science", First International Conference, IESS 2010, LNBIP 53, Springer, pp. 30-44, 2010

[19] S. Najar, O. Saidani, M. Kirsch-Pinheiro, C. Souveyet, and S. Nurcan, "Semantic representation of context models: a framework for analyzing and understanding," J. M. Gomez-Perez, P. Haase, M. Tilly, and P. Warren (Eds),1st Workshop on Context, information and ontologies (CIAO 09), European Semantic Web Conference (ESWC), ACM, pp. 1-10, 2009

[20] S. Najar, M. Kirsch-Pinheiro, and C. Souveyet, "The influence of context on intentional service," 5th Int. IEEE Workshop on Requirements Engineerings for Services (REFS'11) - IEEE Conference on Computers, Software, and Applications (COMPSAC), Munich, Germany, pp. 470–475, 2011

[21] S. Najar, M. Kirsch-Pinheiro, and C. Souveyet, "Bringing context to intentional services," 3rd Int. conf on Advanced Service Computing, Service Computation'11, Rome, Italy, pp. 118-123, 2011

[22] S. Najar, M. Kirsch-Pinheiro, C. Souveyet, L.A. Steffenel, "Service Discovery Mechanisms for an Intentional Pervasive Information System", Proceedings of 19th IEEE International Conference on Web Services (ICWS 2012), Honolulu, Hawaii, 2012, pp. 24-29

[23] T. Olsson, M.Y. Chong, B. Bjurling, and B. Ohlman, "Goal Refinement for Automated Service Discovery", 3rd Int. Conf on Advanced Service Computing, Service Computation'11, Rome, Italy, pp. 46-51, 2011

[24] OSGi Alliance, http://www.osgi.org/ : January, 2011

[25] M.P. Papazoglou, P. Traverso, S. Dustdar, and F. Leymann, "Service-Oriented Computing: A Research Roadmap," Int. J. Cooperative Inf. Syst. vol 17 n° 2, 2008, pp. 223-255

[26] N. Prat, "Goal formalisation and classification for requirements engineering," In Proc. of the 3rd International Workshop on Requirements Engineering: Foundations of Software Quality (REFSQ'97). E.Dubois, A.L.Opdahl, K.Pohl. (eds), Presses Universitaires de Namur, 1997

[27] R. Reichle, M. Wagner, M. Khan, K. Geihs, L. Lorenzo, M. Valla, C. Fra, N. Paspallis, and G.A. Papadopoulos, "A Comprehensive Context Modeling Framework for Pervasive Computing Systems," In 8th IFIP Conf on Distributed Applications and Interoperable Systems (DAIS), Springer, 2008

[28] C. Rolland, M. Kirsch-Pinheiro, C. Souveyet, "An Intentional Approach to Service Engineering," IEEE Transactions on Service Computing, vol. 3(4), 2010, pp. 292-305

[29] D. Roman, U. Keller, H. Lausen, J. de Bruijn, R. Lara, M. Stollberg, A. Polleres, C. Feier, C. Bussler, D. Fensel, "Web Service Modeling Ontology", Applied Ontology, vol. 1(1), 2005, pp. 77- 106

[30] M. Rosemann, J. Recker, and C. Flender, "Contextualization of Business Processes," Int. J. Business Process Integration and Management, vol. 1(1/2/3), 2007

[31] W. Roshen, "SOA-Based enterprise integration: a step-by-step guide to services-based application integration," McGraw Hill, 2009

[32] O. Saidani and S. Nurcan, "Towards Context Aware Business Process Modeling," 8th Workshop on Business Process Modeling, Development, and Support (BPMDS'07), CAiSE'07, 2007

[33] S. Schulthess, "Construction of a registry for searching web service," Master Thesis, EFREI, Engineering School Paris, 2011

[34] V. Suraci, S. Mignanti, and A. Aiuto, "Context-aware Semantic Service Discovery," 16th IST Mobile and Wireless Communications Summit, pp. 1-5, 2007

[35] N. Taylor, P. Robertson, B. Farshchian, K. Doolin, I. Roussaki, L. Marshall, R. Mullins, S. Druesedow, and K. Dolinar, "Pervasive Computing in Daidalos," Pervasive Computing, vol. 10(1), 2011, pp. 74 -81

[36] A. Toninelli, A. Corradi, and R. Montanari, "Semantic-based discovery to support mobile context-aware service access," Computer Communications, vol.31(5), 2008, pp. 935-949

[37] R. Welke, R. Hirschheim, and A. Schwarz, "Service-oriented architecture maturity," IEEE Computer, vol. 44(2), pp. 61-67, 2011.

[38] H. Xiao, Y. Zou, J. Ng, and L. Nigul, "An Approach for Context-aware Service Discovery and Recommendation", IEEE Int. Conf on Web Services (ICWS), Miami, pp. 163-170, 2010

[39] http://www.mindswap.org/2003/pellet/ : March, 2011

[40] http://sourceforge.net/projects/jena/files/ : March, 2011

[41] http://www.mindswap.org/2004/owl-s/api/ : January, 2011

[42] http://semwebcentral.org/projects/owls-tc/ : February, 2012

[43] http://www.ip-super.org/ : June, 2012

[44] http://www.wsmo.org : June, 2012

Designing Indicators to Monitor the Fulfillment of Business Objectives with Particular Focus on Quality and ICT-supported Monitoring of Indicators

Olav Skjelkvåle Ligaarden*†, Atle Refsdal*, and Ketil Stølen*†
* *Department for Networked Systems and Services, SINTEF ICT*
PO Box 124 Blindern, N-0314 Oslo, Norway
E-mail: {olav.ligaarden, atle.refsdal, ketil.stolen}@sintef.no
† *Department of Informatics, University of Oslo*
PO Box 1080 Blindern, N-0316 Oslo, Norway

***Abstract*—In this paper we present our method ValidKI for designing indicators to monitor the fulfillment of business objectives with particular focus on quality and ICT-supported monitoring of indicators. A set of indicators is valid with respect to a business objective if it measures the degree to which the business or relevant part thereof fulfills the business objective. ValidKI consists of six main steps. We demonstrate the method on an example case focusing on the use of electronic patient records in a hospital environment.**

***Keywords*-Indicator, business objective, quality, ICT-supported monitoring, electronic patient record**

I. Introduction

Today's companies benefit greatly from ICT-supported business processes, as well as business intelligence and business process intelligence applications monitoring and analyzing different aspects of a business and its processes. The output from these applications may be indicators which summarize large amounts of data into single numbers. Indicators can be used to evaluate how successful a company is with respect to specific business objectives. For this to be possible it is important that the indicators are valid. A set of indicators is valid with respect to a business objective if it measures the degree to which the business or relevant part thereof fulfills the business objective. Valid indicators facilitate decision making, while invalid indicators may lead to bad business decisions, which again may greatly harm the company.

In today's business environment, companies cooperate across company borders. Such co-operations often result in sharing or outsourcing of ICT-supported business processes. One example is the interconnected electronic patient record (EPR) infrastructure. The common goal for this infrastructure is the exchange of EPRs facilitating the treatment of the same patient at more than one hospital. In such an infrastructure, it is important to monitor the use of EPRs in order to detect and avoid misuse. This may be achieved through the use of indicators. It may be challenging to identify and compute good indicators that are valid with respect to business objectives that focus on quality in general and security in particular. Furthermore, in an infrastructure or system stretching across many companies we often have different degrees of visibility into how the cooperating parties perform their part of the business relationship, making the calculation of indicators particularly hard.

In [1] we presented the method *ValidKI* (Valid Key Indicators) for designing indicators to monitor the fulfillment of business objectives with particular focus on quality and ICT-supported monitoring of indicators. ValidKI facilitates the design of a set of indicators that is valid with respect to a business objective. In this paper we present an improved version of the method.

We demonstrate ValidKI by applying it on an example case targeting the use of EPRs. We have developed ValidKI with the aim of fulfilling the following characteristics:

- **Business focus:** The method should facilitate the design and assessment of indicators for the purpose of measuring the fulfillment of business objectives with particular focus on quality and ICT-supported monitoring of indicators.
- **Efficiency:** The method should be time and resource efficient.
- **Generality:** The method should be able to support design of indicators for systems shared between many companies or organizations.
- **Heterogeneity:** The method should not place restrictions on how indicators are designed.

The rest of the paper is structured as follows: in Section II we introduce our basic terminology and definitions. In Section III we give an overview of ValidKI and its six main steps. In Sections IV – IX we demonstrate our six-step method on an example case addressing the use of EPRs in a hospital environment. In Section X we present related work, while in Section XI we conclude by characterizing our contribution and discussing the suitability of our method.

II. Basic terminology and definitions

Hammond et al. defines indicator as *"something that provides a clue to a matter of larger significance or makes perceptible a trend or phenomenon that is not immediately detectable"* [2]. For example, a drop in barometric pressure

may signal a coming storm, while an unexpected rise in the traffic load of a web server may signal a denial of service attack in progress. Thus, the significance of an indicator extends beyond what is actually measured to a larger phenomenon of interest.

Indicators are closely related to metrics. [3] defines metric as *"a quantitative measure of the degree to which a system, component, or process possesses a given attribute,"* while it defines attribute as *"the specific characteristic of the entity being measured."* For the web server mentioned above, an example of an attribute may be availability. An availability metric may again act as an indicator for denial of service attacks, if we compare the metric with a baseline or expected result [4]. As we can see, metrics are not that different from indicators. For that reason, indicators and metrics are often used interchangeably in the literature.

Many companies profit considerably from the use of indicators [5] resulting from business process intelligence applications that monitor and analyze different aspects of a business and its processes. Indicators can be used to measure to what degree a company fulfills its business objectives and we then speak of key indicators. Some business objectives may focus on business performance, while others may focus on risk or compliance with laws and regulations. We will in the remainder of the paper refer to indicators as key indicators, since we focus on indicators in the context of business objectives.

A. The artifacts addressed by ValidKI

The UML [6] class diagram in Figure 1 relates the main artifacts addressed by ValidKI. The associations between the different concepts have cardinalities that specify how many instances of one concept that may be associated to an instance of the other concept.

As characterized by the diagram, one or more key indicators are used to measure to what extent a business objective is fulfilled with respect to a relevant part of the business. Each key indicator is calculated based on data provided by one or more sensors. The sensors gather data from the relevant part of the business. A sensor may gather data for more than one key indicator.

B. The models/descriptions developed by ValidKI

As illustrated by Figure 2, performing the steps of ValidKI results in nine different models/descriptions each of which describes one of the artifacts of Figure 1 from a certain perspective.

A specification, at a suitable level of abstraction, documents the relevant part of the business in question.

Business objectives are typically expressed at an enterprise level and in such a way that they can easily be understood by for example shareholders, board members, partners, etc. It is therefore often not completely clear what

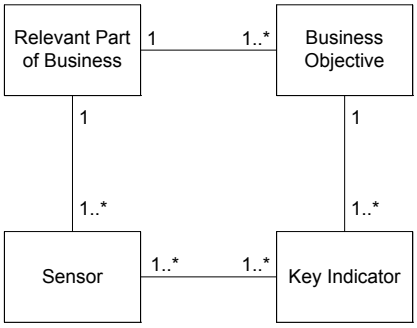


Figure 1. The artifacts addressed by ValidKI

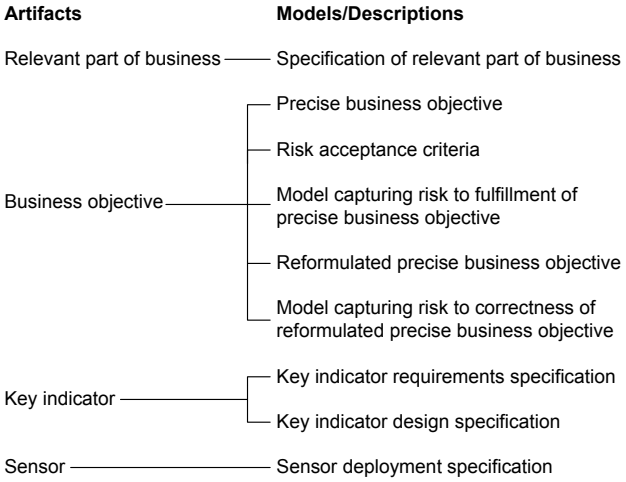


Figure 2. The models/descriptions developed by ValidKI

it means to fulfill them. This motivates the need to capture each business objective more precisely.

The fulfillment of a precise business objective may be affected by a number of risks. We therefore conduct a risk analysis to capture risk to the fulfillment of the precise business objective. To evaluate which risks that are acceptable and not acceptable with respect to the fulfillment of the precise business objective, we use risk acceptance criteria. It is the risks that are not acceptable that we need to monitor. The acceptable risks may be thought of to represent uncertainty we can live with. In other words, their potential occurrences are not seen to significantly influence the fulfillment of the business objective.

The degree of fulfillment of a precise business objective is measured by a set of key indicators. To measure its degree of fulfillment there is a need to express each precise business objective in terms of key indicators. We refer to this reformulation as the reformulated precise business objective. Moreover, the correctness of key indicators will be affected if they are not implemented correctly. This may again lead to new unacceptable risks that affect the fulfillment of the precise business objective. Since the reformulated precise

business objective is the precise business objective expressed in terms of key indicators, we need to analyze risks to the correctness of the reformulated precise business objective.

The computation of key indicators relies on different kinds of data. To collect the data, sensors need to be deployed in the relevant part of business. Thus, there is a need to specify the deployment of different sensors.

For each key indicator we distinguish between two specifications: the key indicator requirements specification and the key indicator design specification. The first captures requirements to a key indicator with respect to the sensor deployment specifications, while the second defines how the key indicator should be calculated.

C. Validity

[7] defines validation as *"confirmation, through the provision of objective evidence, that the requirements for a specific intended use or application have been fulfilled."* Since an indicator is basically a metric that can be compared to a baseline/expected result, the field of metric validation is highly relevant. There is however no agreement upon what constitutes a valid metric [8]. In [8], Meneely et al. present a systematic literature review of papers focusing on validation of software engineering metrics. The literature review began with 2288 papers, which were later reduced to 20 papers. From these 20 papers, the authors extracted and categorized 47 unique validation criteria. The authors argue that metric researchers and developers should select criteria based on the intended usage of the metric. Even though the focus in [8] is on validation of software engineering metrics, a number of the validation criteria presented are general, thus not specific to software engineering. In particular, following [8] we define a set of key indicators to be valid with respect to a business objective if it is valid in the following two ways:

1) **internal validity** – the precise business objective expressed in terms of the key indicators correctly measures the degree to which the business objective is fulfilled; and
2) **construct validity** – the gathering of the sensor measurements of each key indicator is suitable with respect to its requirements specification.

III. Overview of ValidKI

Figure 3 provides an overview of the ValidKI method. It takes as input a business objective and delivers a set of key indicators and a report arguing its validity with respect to the business objective received as input. When using ValidKI in practice we will typically develop key indicators for a set of business objectives, and not just one which we restrict our attention to here. It should be noticed that when developing key indicators for a set of business objectives, we need to take into account that key indicators (i.e., software or infrastructure) developed for one business objective may affect the validity of key indicators developed for another.

Input:	A business objective
Step 1:	**Establish target**
Step 1.1:	Express business objectives more precisely
Step 1.2:	Describe relevant part of business
Step 2:	**Identify risks to fulfillment of business objective**
Step 2.1:	Specify risk acceptance criteria
Step 2.2:	Risk identification and estimation
Step 2.3:	Risk evaluation
Step 3:	**Identify key indicators to monitor risks**
Step 3.1:	Deploy sensors to monitor risks
Step 3.2:	Specify requirements to key indicators wrt deployed sensors
Step 4:	**Evaluate internal validity**
Step 4.1:	Express business objective in terms of key indicators
Step 4.2:	Evaluate criteria for internal validity
Step 5:	**Specify key indicator designs**
Step 6:	**Evaluate construct validity**
Output:	A set of key indicators and a report arguing its validity with respect to the business objective received as input

Figure 3. Overview of ValidKI

In the following we offer additional explanations for each of the six main steps of the ValidKI method.

A. Establish target

The first main step of ValidKI is all about understanding the target, i.e., understanding exactly what the business objective means and acquiring the necessary understanding of the relevant part of business for which the business objective has been formulated. We distinguish between two sub-steps. In the first sub-step we characterize the business objective more precisely by formulating constraints that need to be fulfilled. In the second sub-step we specify the relevant part of the business.

B. Identify risks to fulfillment of business objective

The second main step of ValidKI is concerned with conducting a risk analysis to identify risks to the fulfillment of the business objective. We distinguish between three sub-steps. In the first sub-step the risk acceptance criteria are specified. The criteria classify a risk as either acceptable or unacceptable based on its likelihood and consequence. In the second sub-step we identify how threats may initiate risks. We also identify vulnerabilities and threat scenarios leading up to the risks, and we estimate likelihood and consequence. During the risk analysis we may identify risks that pull in the same direction. Such risks should be combined into one risk. The individual risks may be acceptable when considered in isolation, while the combined risk may be unacceptable.

In the third sub-step we evaluate the identified risks with respect to the specified risk acceptance criteria.

C. Identify key indicators to monitor risks

The third main step of ValidKI is concerned with identifying key indicators to monitor the unacceptable risks identified in the previous step. We distinguish between two sub-steps. In the first sub-step we specify how sensors should be deployed in the relevant part of business. The key indicators that we identify are to be calculated based on data gathered by the sensors. In the second sub-step we specify our requirements to the key indicators with respect to the deployed sensors. The two sub-steps are typically conducted in parallel.

D. Evaluate internal validity

The fourth main step of ValidKI is concerned with evaluating whether the set of key indicators is internally valid with respect to the business objective. We distinguish between two sub-steps. In the first sub-step we reformulate the precise business objective by expressing it in terms of the identified key indicators. This step serves as an introductory step in the evaluation of internal validity. In the second sub-step we evaluate whether the set of key indicators is internally valid by showing that the reformulated precise business objective from Step 4.1 correctly measures the fulfillment of the precise business objective from Step 1.1.

Internal validity may be decomposed into a broad category of criteria [8]. In the following we list the criteria that we take into consideration. For each criterion, we first provide the definition as given in [8], before we list the papers on which the definition is based.

- **Attribute validity:** *"A metric has attribute validity if the measurements correctly exhibit the attribute that the metric is intending to measure"* [9][10]. In our case, the key indicator needs to correctly exhibit the risk attribute (likelihood or consequence) of the risk that it is measuring. In addition, the key indicator is of little value if it can only produce values that always result in the risk being acceptable or unacceptable.
- **Factor independence:** *"A metric has factor independence if the individual measurements used in the metric formulation are independent of each other"* [11]. This criterion applies especially to composite key indicators that are composed of basic key indicators. A composite key indicator has factor independence if the basic key indicators are independent of each other, i.e., if they do not rely on the same measurements.
- **Internal consistency:** *"A metric has internal consistency if "all of the elementary measurements of a metric are assessing the same construct and are interrelated""* [12]. This criterion also applies especially to composite key indicators that are composed of basic key indicators. If the basic key indicators measure things that are not conceptually related, then the composite key indicator will not have internal consistency. For instance, let us say that we have a composite key indicator that is composed of two basic key indicators. The first basic key indicator measures the code complexity of a software product, while the second measures the cost of shipping the software product to the customers. In this case, the composite key indicator does not have internal consistency, since the two basic key indicators are not conceptually related.
- **Appropriate continuity:** *"A metric has appropriate continuity if the metric is defined (or undefined) for all values according to the attribute being measured"* [10]. An example of a discontinuity is fraction calculations when the denominator is zero. To avoid discontinuity, the key indicator should be defined for that case.
- **Dimensional consistency:** *"A metric has dimensional consistency if the formulation of multiple metrics into a composite metric is performed by a scientifically well-understood mathematical function"* [10][13]. Under dimensional consistency, no information should be lost during the construction of composite key indicators. Loss of information may be experienced if different scales are used for the basic and composite key indicators.
- **Unit validity:** *"A metric has unit validity if the units used are an appropriate means of measuring the attribute"* [10][14]. For instance, the unit fault rate may be used to measure the attribute program correctness [10].

If the set is not internally valid, then we iterate by re-doing Step 3.

E. Specify key indicator designs

In the fifth main step of ValidKI we specify the designs of the identified key indicators. Each design specifies how the key indicator should be calculated. The design also shows how sensors, actors, and different components interact.

F. Evaluate construct validity

In the sixth main step of ValidKI we evaluate whether the set of key indicators has construct validity with respect to the business objective. As with internal validity, construct validity may be decomposed into a broad category of criteria [8]. In the following we list the criteria that we take into consideration. For each criterion, we first provide the definition as given in [8], before we list the papers on which the definition is based.

- **Stability:** *"A metric has stability if it produces the same values "on repeated collections of data under similar circumstances""* [12][15][16]. A key indicator whose calculation involves decisions made by humans, may for example result in different values and thus lack of stability.

- **Instrument validity:** *"A metric has instrument validity if the underlying measurement instrument is valid and properly calibrated"* [10]. In our case, this criterion concerns the sensors that perform the measurements.
- **Definition validity:** *"A metric has definition validity if the metric definition is clear and unambiguous such that its collection can be implemented in a unique, deterministic way"* [11][15][16][17][18]. This criterion concerns the implementation of the key indicators. To implement a key indicator correctly, the key indicator's design specification needs to be clear and unambiguous.

To evaluate the different criteria, we re-do the risk analysis from Step 2.2 with the precise business objective replaced by the reformulated precise business objective, which is the precise business objective expressed in terms of key indicators. For each key indicator we identify risks towards the correctness of the reformulated precise business objective that are the result of threats to criteria for construct validity that the key indicator needs to fulfill. If the risk analysis does not result in any new unacceptable risks, then we have established construct validity for each key indicator. If the set does not have construct validity, then we iterate. We will most likely be re-doing Step 5, but it may also be the case that we need to come up with new key indicators and new sensors. In that case, we re-do Step 3. If the set of key indicators is both internally valid and has construct validity with respect to the business objective, then we have established that the set is valid.

IV. Establish target

In the following we assume that we have been hired to help the public hospital Client H design key indicators to monitor their compliance with Article 8 in the European Convention on Human Rights [19]. The article states the following:

Article 8 – Right to respect for private and family life

1) Everyone has the right to respect for his private and family life, his home and his correspondence.
2) There shall be no interference by a public authority with the exercise of this right except such as is in accordance with the law and is necessary in a democratic society in the interests of national security, public safety or the economic well-being of the country, for the prevention of disorder or crime, for the protection of health or morals, or for the protection of the rights and freedoms of others.

Client H needs to comply with Article 8 since it is a public authority. The consequence for Client H of not complying with Article 8 may be economic loss and damaged reputation. One example [20] of violation of Article 8 is from Finland. A Finnish woman was first treated for HIV at a hospital, before she later started working there as a nurse. While working there she suspected that her co-workers had unlawfully gained access to her medical data. She brought the case to the European Court of Human Rights in Strasbourg which unanimously held that the district health authority responsible for the hospital had violated Article 8 by not protecting the medical data of the woman properly. The district health authority was held liable to pay damages to the woman. Client H has therefore established the following business objective:

Business objective BO-A8: Client H complies with Article 8 in the European Convention on Human Rights.

Client H wants to make use of key indicators to monitor the degree of fulfillment of BO-A8, and now they have hired us to use ValidKI to design them. In the rest of this section we conduct Step 1 of ValidKI on behalf of Client H with respect to BO-A8.

A. Express business objectives more precisely (Step 1.1 of ValidKI)

Article 8 states under which circumstances a public authority can interfere with someone's right to privacy. One of these circumstances is *"for the protection of health,"* which is what Client H wants us to focus on. In the context of Client H this means to provide medical assistance to patients. The ones who provide this assistance are the health-care professionals of Client H.

The medical history of a patient is regarded as both sensitive and private. At Client H, the medical history of a patient is stored in an electronic patient record (EPR). An EPR is *"an electronically managed and stored collection or collocation of recorded/registered information on a patient in connection with medical assistance"* [21]. The main purpose of an EPR is to communicate information between health-care professionals that provide medical care to a patient. To protect the privacy of its patients, Client H restricts the use of EPRs. In order to comply with Article 8, Client H allows a health-care professional to interfere with the privacy of a patient only when providing medical assistance to this patient. Hence, the dealing with EPRs within the realms of Client H is essential.

For Client H it is important that every access to information in an EPR is in accordance with Article 8. A health-care professional should only access a patient's EPR if he/she provides medical assistance to that patient, and he/she should only access information that is necessary for the medical assistance provided to the patient. The information accessed can not be used for any other purpose than providing medical assistance to patients. Accesses to information in EPRs not needed for providing medical assistance would not be in accordance with Article 8. Also, employees that are not health-care professionals and work within the jurisdiction of Client H are not allowed to access

EPRs. Based on the constraints provided by Client H, we decide to express BO-A8 more precisely as follows:

Precise business objective PBO-A8: $C_1 \wedge C_2 \wedge C_3$

- **Constraint** C_1**:** Health-care professionals acting on behalf of Client H access:
  - a patient's EPR only when providing medical assistance to that patient
  - only the information in a patient's EPR that is necessary for providing medical assistance to that patient
- **Constraint** C_2**:** Health-care professionals acting on behalf of Client H do not use the information obtained from a patient's EPR for any other purpose than providing medical assistance to that patient.
- **Constraint** C_3**:** Employees that are not health-care professionals and that work within the jurisdiction of Client H do not access EPRs.

As indicated by PBO-A8's definition, all three constraints must be fulfilled in order for PBO-A8 to be fulfilled.

B. Describe relevant part of business (Step 1.2 of ValidKI)

To design key indicators to monitor BO-A8 we need to understand the part of business that is to comply with BO-A8 and therefore is to be monitored. "Public hospital *Client H*" has outsourced some of its medical services to two private hospitals. These two are referred to as "Private hospital *X-ray*" and "Private hospital *Blood test analysis*" in Figure 4. The first hospital does all the X-ray work for Client H, while the second hospital does all the blood test analyses. Client H is not only responsible for its own handling of EPRs, but also the outsourcing partners' handling of EPRs, when they act on behalf of Client H.

As can be seen in Figure 4, Client H outsources medical tasks to the two private hospitals, and gets in return the results from performing these tasks. All three health-care institutions employs some kind of EPR system for handling the EPRs. An EPR system is *"an electronic system with the necessary functionality to record, retrieve, present, communicate, edit, correct, and delete information in electronic patient records"* [21]. These systems use EPRs provided by different health-care institutions. As shown in Figure 4, these systems are only of interest when they handle EPRs where Client H is responsible for their handling.

At the three health-care institutions, most of the medical tasks that a health-care professional conducts during a working day are known in advance. It is known which patients the professional will treat and what kind of information the professional will need access to in order to treat the different patients. Client H and the two outsourcing partners maintain for each health-care professional an authorization list documenting which patients the professional is treating and what kind of information the professional needs for this purpose. These lists are used by the EPR systems and they are updated on a daily basis by the medical task management systems. Many of these updates are automatic. For instance, when Client H is assigned a new patient, then this patient is added to the lists of the health-care professionals who will be treating this patient.

Each EPR is owned by a patient, which is natural since the information stored in the EPR is about the patient in question. As already mentioned, the content of a patient's EPR is both considered sensitive and private. Moreover, some of the EPRs may contain information that is considered highly sensitive and private. Such information may for instance describe medical treatment received by a patient in relation to:

- the patient being the victim of a crime (e.g., rape, violence, etc.);
- sexual transferable diseases or abortion; and
- mortal or infectious mortal diseases.

Information classified as highly sensitive and private is handled with even more care than information that is just classified as sensitive and private. To raise awareness of the criticality of such information and to enable monitoring of its use, the EPR systems at the three health-care institutions tag highly sensitive and private information in EPRs based on predefined rules.

Accesses to information in EPRs can be classified as *authorized* or *unauthorized* based on the authorization lists of health-care professionals. An access is classified as authorized if the professional needs the information to do a planned task. Otherwise, the access is classified as unauthorized. If an access is classified as unauthorized then it is possible to check in retrospect whether the access was necessary. In an emergency situation, for instance when a patient is having a heart attack, a health-care professional often needs access to information in an EPR that he/she was not supposed to access. By checking in retrospect whether unauthorized accesses were necessary it is possible to classify the unauthorized accesses into two groups; one for accesses that were necessary, and one for those that were not. The first group is called *approved* unauthorized accesses, while the second group is called *not approved* unauthorized accesses. All accesses that are classified as not approved unauthorized accesses are considered as *illegal* accesses.

At Client H and the two outsourcing partners, health-care professionals use smart cards for accessing information in EPRs. If a card is lost or stolen, the owner must report it as missing, since missing cards may be used by other health-care professionals or others to access EPRs illegally. When the card has been registered as missing it can no longer be used. When reporting it as missing, the last time the card owner used it before noticing that it was missing is recorded. All accesses to EPRs that have occurred between this time and the time it was registered as missing are considered as illegal accesses.

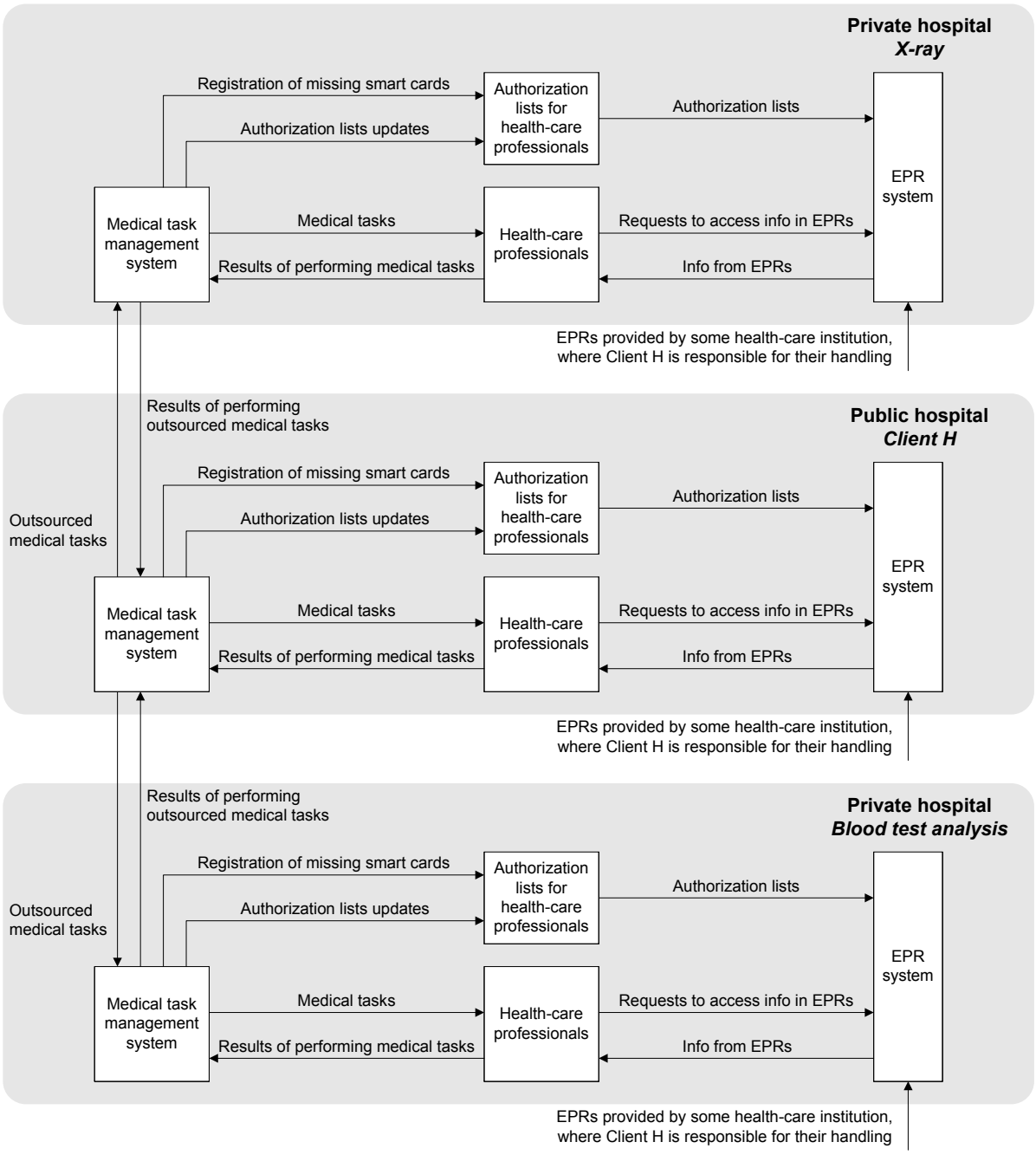


Figure 4. Specification of relevant part of business

Table I
CONSEQUENCE SCALE FOR THE ASSET "FULFILLMENT OF PBO-A8" (TOP) AND LIKELIHOOD SCALE (BOTTOM)

Consequence	Description
Catastrophic	Law enforcement agencies penalize Client H after having been notified about the incident
Major	Health authorities penalize Client H after having been notified about the incident
Moderate	Health authorities are notified about the incident
Minor	Head of hospital is notified about the incident
Insignificant	Head of department is notified about the incident
Likelihood	**Description**
Certain	Five times or more per year $[50, \infty\rangle$: 10 years
Likely	Two to five times per year $[20, 49]$: 10 years
Possible	Once a year $[6, 19]$: 10 years
Unlikely	Less than once per year $[2, 5]$: 10 years
Rare	Less than once per ten years $[0, 1]$: 10 years

V. IDENTIFY RISKS TO FULFILLMENT OF BUSINESS OBJECTIVE

For the sake of simplicity, we only show how we identify risks to the fulfillment of constraint C_1. Thus, we assume that the precise business objective PBO-A8 only consists of the constraint C_1.

A. Specify risk acceptance criteria (Step 2.1 of ValidKI)

Before we specify the risk acceptance criteria, we need to establish scales for measuring likelihood and consequence. Table I presents these scales. We view "Fulfillment of PBO-A8" as the asset to be protected. In Table II the risk acceptance criteria for the asset "Fulfillment of PBO-A8" are expressed in terms of a risk evaluation matrix. Risks whose values belong to the white area of the matrix are acceptable, while risks whose values belong to the gray area are unacceptable.

B. Risk identification and estimation (Step 2.2 of ValidKI)

Based on the information provided by the representatives of Client H, we identify and estimate risk. For this purpose we use the CORAS methodology [22]. However, other approaches to risk analysis may be used instead. Using CORAS we identify how threats may initiate risks that harm the asset "Fulfillment of PBO-A8" if they occur.

The CORAS threat diagram in Figure 5 documents different risks that may harm the fulfillment of the precise business objective PBO-A8. The CORAS threat diagram contains two human threats; one accidental (the white one) and one deliberate (the black one). The accidental human threat "Health-care professional" may initiate the threat scenario "Unauthorized access to information in a patient's EPR" with likelihood "Likely" by exploiting the vulnerability "No restrictions on what EPRs a health-care professional can access." We can also see that the deliberate human threat "Health-care professional" may initiate this threat scenario, and that the threat scenario occurs with likelihood "Certain." If the threat scenario occurs then it leads to the threat scenario "Unauthorized access to sensitive and private information" with conditional likelihood "0.7." This threat scenario leads to the risk "R1: Not approved unauthorized access to sensitive and private information in an EPR, where the owner of the EPR is a patient of the accessor" with conditional likelihood "0.6" if it occurs. The risk occurs with likelihood "Likely." It impacts the asset "Fulfillment of PBO-A8" with consequence "Insignificant" if it occurs.

The diagram documents that a health-care professional can accidentally perform unauthorized accesses. It also documents that a health-care professional can deliberately perform unauthorized accesses, or use lost/stolen smart cards to access information in EPRs. As can be seen in the diagram, we distinguish between not approved unauthorized accesses to information in EPRs where the owner of the EPR is a patient and not a patient of the accessor. Client H finds it most serious if the owner of the EPR is not a patient of the accessor. We also distinguish between not approved unauthorized accesses to sensitive and private information and not approved unauthorized accesses to highly sensitive and private information. Naturally, Client H finds not approved unauthorized accesses to the latter type of information the most serious.

C. Risk evaluation (Step 2.3 of ValidKI)

The risk evaluation consists in plotting the risks into the risk evaluation matrix according to their likelihoods and consequences. As indicated in Table III, two out of the six risks namely $R4$ and $R6$ are unacceptable with respect to the fulfillment of the precise business objective PBO-A8.

VI. IDENTIFY KEY INDICATORS TO MONITOR RISKS

A. Deploy sensors to monitor risks (Step 3.1 of ValidKI)

Figure 6, which is a detailing of the target description in Figure 4, specifies the deployment of sensors in the relevant part of business. This specification corresponds to the sensor deployment specification referred to in Figure 2. An antenna-like symbol is used to represent each sensor in Figure 6. The different sensors monitor data messages exchanged within the relevant part of business. The results from the monitoring are to be used in the computation of key indicators.

The following sensors are deployed in the relevant part of business:

- $S_{\text{CH-REG-MIS-SC}}$, $S_{\text{BTA-REG-MIS-SC}}$, and $S_{\text{XR-REG-MIS-SC}}$ monitor data messages related to the registration of missing smart cards at Client H, Blood test analysis, and X-ray, respectively.
- $S_{\text{CH-AUTH-LIST}}$, $S_{\text{BTA-AUTH-LIST}}$, and $S_{\text{XR-AUTH-LIST}}$ monitor data messages related to the authorization lists employed by the EPR systems at Client H, Blood test analysis, and X-ray, respectively.

Table II
RISK EVALUATION MATRIX FOR THE ASSET "FULFILLMENT OF PBO-A8"

Likelihood \ Consequence	Insignificant	Minor	Moderate	Major	Catastrophic
Rare					
Unlikely					
Possible					
Likely					
Certain					

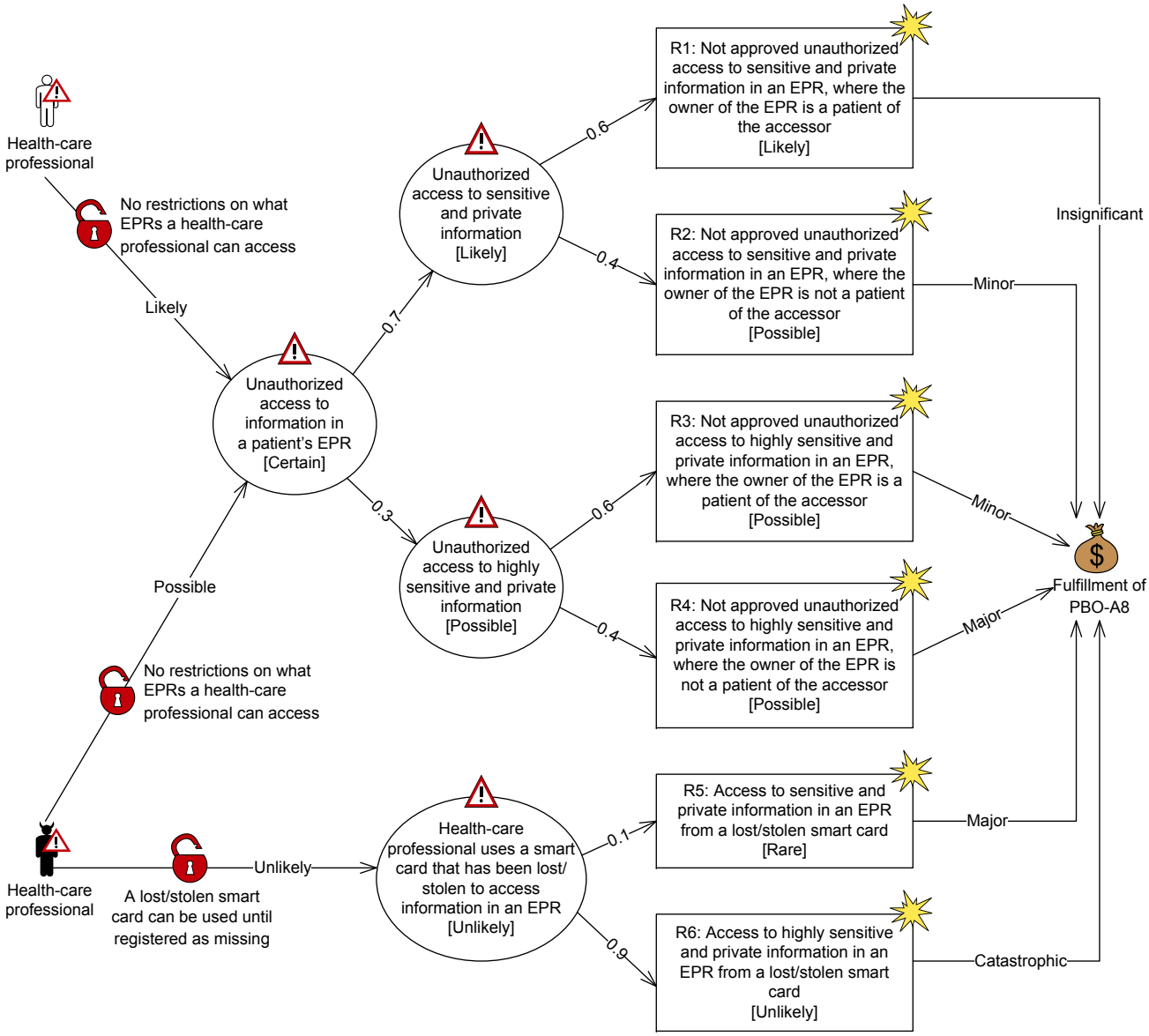


Figure 5. CORAS threat diagram documenting the result of the risk identification and estimation

Table III
THE RISK EVALUATION MATRIX FROM TABLE II WITH THE ACCEPTABLE AND UNACCEPTABLE RISKS INSERTED

Likelihood \ Consequence	Insignificant	Minor	Moderate	Major	Catastrophic
Rare				$R5$	
Unlikely					$R6$
Possible		$R2$, $R3$		$R4$	
Likely	$R1$				
Certain					

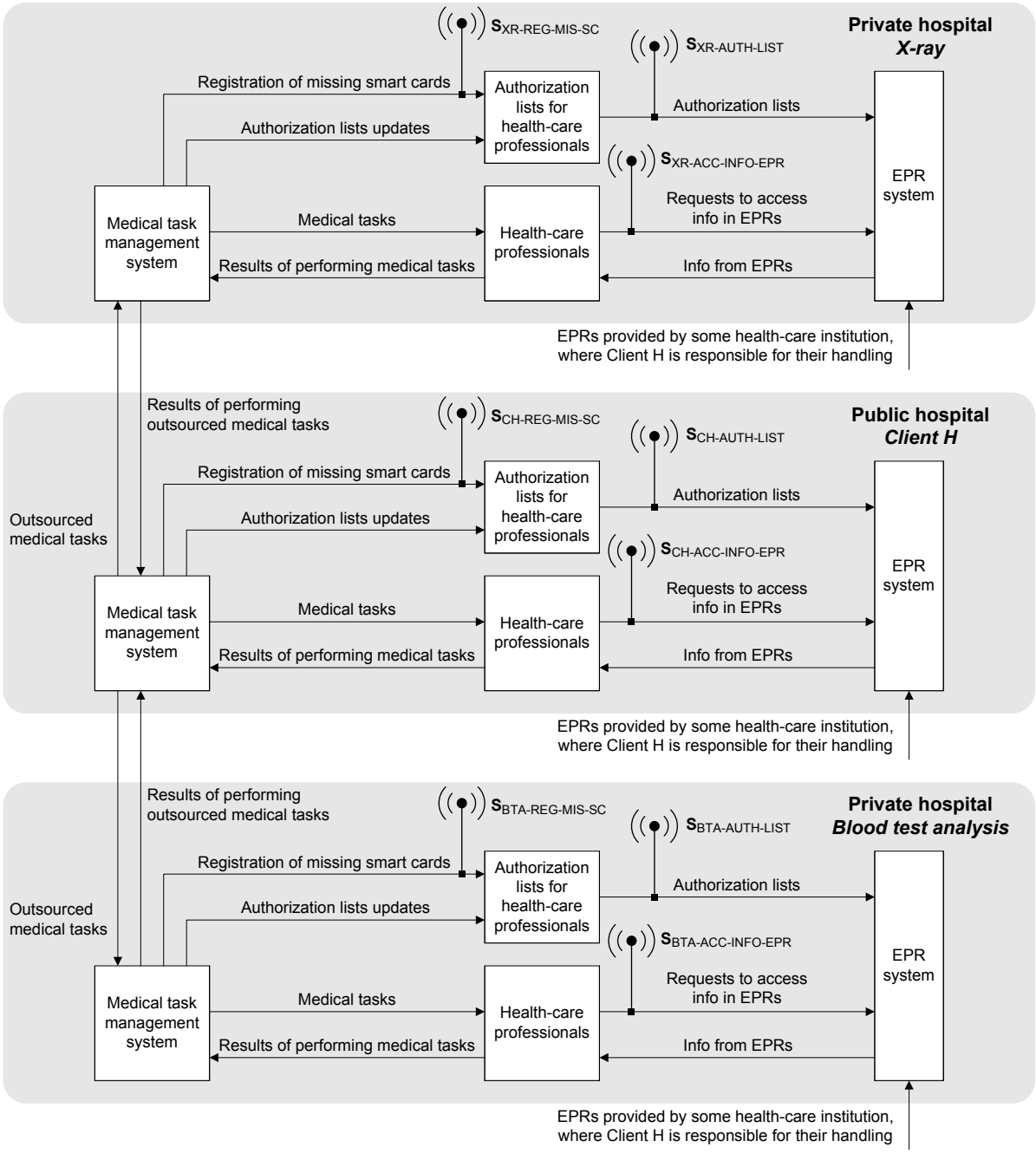


Figure 6. Deployment of sensors in the relevant part of business

Table IV
KEY INDICATOR REQUIREMENTS SPECIFICATIONS FOR $K_{\text{CH-NOT-APP-UNAUTH-ACC}}$, $K_{\text{BTA-NOT-APP-UNAUTH-ACC}}$, AND $K_{\text{XR-NOT-APP-UNAUTH-ACC}}$

Requirements for $K_{\text{X-NOT-APP-UNAUTH-ACC}}$, where $X \in \{\textbf{CH}, \textbf{BTA}, \textbf{XR}\}$
In:
$S_{\text{X-AUTH-LIST}} : M^*$
$S_{\text{X-ACC-INFO-EPR}} : M^*$
Out:
$K_{\text{X-APP-UNAUTH-ACC}} : \mathbb{N}$
Description:
$K_{\text{X-NOT-APP-UNAUTH-ACC}}$ = "The number of not approved unauthorized accesses at X since the monitoring started to highly sensitive and private information in EPRs, where the owners of the EPRs are not patients of the accessors"

Table V
KEY INDICATOR REQUIREMENTS SPECIFICATIONS FOR $K_{\text{CH-ILL-ACC-SC}}$, $K_{\text{BTA-ILL-ACC-SC}}$, AND $K_{\text{XR-ILL-ACC-SC}}$

Requirements for $K_{\text{X-ILL-ACC-SC}}$, where $X \in \{\textbf{CH}, \textbf{BTA}, \textbf{XR}\}$
In:
$S_{\text{X-REG-MIS-SC}} : M^*$
$S_{\text{X-ACC-INFO-EPR}} : M^*$
Out:
$K_{\text{X-ILL-ACC}} : \mathbb{R}$
Description:
$K_{\text{X-ILL-ACC-SC}}$ = "The number of illegal accesses at X since the monitoring started to highly sensitive and private information in EPRs from lost/stolen smart cards"

- $S_{\text{CH-ACC-INFO-EPR}}$, $S_{\text{BTA-ACC-INFO-EPR}}$, and $S_{\text{XR-ACC-INFO-EPR}}$ monitor data messages where each message is a request issued by health-care professional to access information in an EPR at Client H, Blood test analysis, and X-ray, respectively. It is not necessary to monitor the actual information received, since health-care professionals will always get the information they request.

B. Specify requirements to key indicators wrt deployed sensors (Step 3.2 of ValidKI)

Two key indicators $K_{\text{NOT-APP-UNAUTH-ACC}}$ and $K_{\text{ILL-ACC-SC}}$ are identified to monitor the likelihood values of the two unacceptable risks $R4$ and $R6$, respectively. In Tables VI and VII their requirements are given. The two key indicators calculate likelihoods with respect to a ten year period, because the likelihoods in the likelihood scale in Table I are defined with respect to a ten year period. Both key indicators are composed of basic key indicators. Table IV presents the requirements to the basic key indicators that $K_{\text{NOT-APP-UNAUTH-ACC}}$ is composed of, while Table V presents the requirements to the basic key indicators that $K_{\text{ILL-ACC-SC}}$ is composed of.

Table VI
KEY INDICATOR REQUIREMENTS SPECIFICATION FOR $K_{\text{NOT-APP-UNAUTH-ACC}}$

Requirements for $K_{\text{NOT-APP-UNAUTH-ACC}}$
In:
$S_{\text{CH-AUTH-LIST}}, S_{\text{BTA-AUTH-LIST}}, S_{\text{XR-AUTH-LIST}} : M^*$
$S_{\text{CH-ACC-INFO-EPR}}, S_{\text{BTA-ACC-INFO-EPR}}, S_{\text{XR-ACC-INFO-EPR}} : M^*$
Out:
$K_{\text{NOT-APP-UNAUTH-ACC}} : \mathbb{R}$
Description:
$K_{\text{NOT-APP-UNAUTH-ACC}} = (10 \cdot (K_{\text{CH-NOT-APP-UNAUTH-ACC}} + K_{\text{BTA-NOT-APP-UNAUTH-ACC}} + K_{\text{XR-NOT-APP-UNAUTH-ACC}})) /$ *Number of years since the monitoring started*

Table VII
KEY INDICATOR REQUIREMENTS SPECIFICATION FOR $K_{\text{ILL-ACC-SC}}$

Requirements for $K_{\text{ILL-ACC-SC}}$
In:
$S_{\text{CH-REG-MIS-SC}}, S_{\text{BTA-REG-MIS-SC}}, S_{\text{XR-REG-MIS-SC}} : M^*$
$S_{\text{CH-ACC-INFO-EPR}}, S_{\text{BTA-ACC-INFO-EPR}}, S_{\text{XR-ACC-INFO-EPR}} : M^*$
Out:
$K_{\text{ILL-ACC}} : \mathbb{R}$
Description:
$K_{\text{ILL-ACC-SC}} = (10 \cdot (K_{\text{CH-ILL-ACC-SC}} + K_{\text{BTA-ILL-ACC-SC}} + K_{\text{XR-ILL-ACC-SC}})) /$ *Number of years since the monitoring started*

For each key indicator we specify required sensor data. All of the key indicators rely on sequences of data messages (M^*) gathered by the different sensors. We also specify the output type and requirements to output. For a key indicator K we refer to its requirement description as $Req(K)$.

VII. EVALUATE INTERNAL VALIDITY

A. Express business objective in terms of key indicators (Step 4.1 of ValidKI)

The precise business objective PBO-A8' is a reformulation of the precise business objective PBO-A8 expressed in terms of key indicators.

$$\begin{aligned}\text{PBO-A8'} = {} & K_{\text{NOT-APP-UNAUTH-ACC}} \in [0,5] \wedge \\ & K_{\text{ILL-ACC-SC}} \in [0,1] \wedge \\ & Req(K_{\text{NOT-APP-UNAUTH-ACC}}) \wedge \\ & Req(K_{\text{ILL-ACC-SC}})\end{aligned}$$

The precise business objective PBO-A8 is fulfilled if the likelihood values of the two unacceptable risks $R4$ and $R6$ change in such a way that the two risks becomes acceptable. The risks $R4$ and $R6$ become acceptable if the likelihoods change from "Possible" to "Unlikely" or "Rare" and from "Unlikely" to "Rare," respectively. The likelihoods will change in such a way if the two composite key indicators $K_{\text{NOT-APP-UNAUTH-ACC}}$ and $K_{\text{ILL-ACC-SC}}$, monitoring

these likelihoods, are contained in the interval $[0, 5]$ (interval capturing both "Rare: $[0, 1]$: 10 years" and "Unlikely: $[2, 5]$: 10 years") and $[0, 1]$ ("Rare: $[0, 1]$: 10 years"), respectively. Moreover, the two composite key indicators need to measure the likelihoods correctly in order to measure the fulfillment of PBO-A8. This can be determined based on the requirements ($Req(K_{\text{NOT-APP-UNAUTH-ACC}})$ and $Req(K_{\text{ILL-ACC-SC}})$) to the two composite key indicators.

The reformulated precise business objective can also be used to determine to what degree the precise business objective is fulfilled. For instance, if $K_{\text{NOT-APP-UNAUTH-ACC}}$ equals 6 while $K_{\text{ILL-ACC-SC}}$ equals 0, then PBO-A8 is close to being fulfilled. On the other hand, if $K_{\text{NOT-APP-UNAUTH-ACC}}$ equals 10 instead, then PBO-A8 is far from being fulfilled.

B. Evaluate criteria for internal validity (Step 4.2 of ValidKI)

To evaluate the internal validity of the set of key indicators, we need to show that the reformulated precise business objective PBO-A8' measures the fulfillment of the precise business objective PBO-A8. We evaluate the internal validity of each composite key indicator based on the criteria given in Section III-D. To evaluate attribute validity we need to compare the definitions of the two risks $R4$ and $R6$ in Figure 5 with the requirements of the two composite key indicators, which are given by $Req(K_{\text{NOT-APP-UNAUTH-ACC}})$ and $Req(K_{\text{ILL-ACC-SC}})$. In both cases there is a match between the definition of the risk and the requirements to the composite key indicator. We therefore conclude that the composite key indicators measure the likelihoods of the two risks. In addition, based on the requirements specified for the two composite key indicators it is clear that the two composite key indicators are not restricted to only producing values that are always contained or not contained in the intervals mentioned above. Thus, both acceptable and unacceptable risks can be detected.

Moreover, both of the composite key indicators have factor independence. Each composite key indicator is calculated based on three basic key indicators. These are independent of each other, since they are computed by three different health-care institutions. The two composite key indicators do also have internal consistency, since the three basic key indicators employed by each composite key indicator measure the same thing, but at different health-care institutions. The three basic key indicators are therefore conceptually related.

We continue the evaluation of internal validity by evaluating whether the composite key indicators have appropriate continuity. Both are discontinuous if *"Number of years since the monitoring started"* equals zero. Client H does not consider this to be a problem, since the denominator will in both cases be a real number that is never zero. We also show that the two composite key indicators have dimensional consistency. Each composite key indicator adds three likelihoods, where each is for the period of *"Number of years since the monitoring started"* years, and transforms the resulting likelihood into a likelihood which is for a period of ten years. Thus, no information is lost when constructing the composite key indicators from their respective basic key indicators. The two composite key indicators do also have unit validity. Both use the unit "likelihood per ten years," which is appropriate for measuring the two likelihood attributes of the risks.

Based on the evaluation of the different internal validity types of criteria above, we conclude that the set of key indicators is internally valid. When the precise business objective PBO-A8 is fulfilled, we get the risk evaluation matrix in Table VIII. In this situation, both of the risks $R4$ and $R6$ are acceptable, and the risk $R4$ will either have the likelihood "Rare" ($R4'$) or "Unlikely" ($R4''$).

VIII. Specify key indicator designs

For the sake of simplicity, we only specify the design of the key indicator $K_{\text{NOT-APP-UNAUTH-ACC}}$ and the basic key indicators that it is composed of. We use the UML [6] sequence diagram notation for the key indicator design specifications, but one may of course also use other languages depending on the problem in question. The sequence diagram in Figure 7 specifies how the key indicator $K_{\text{NOT-APP-UNAUTH-ACC}}$ is calculated. Each entity in the sequence diagram is either a component, a sensor, or an employee at Client H, and it is represented by a dashed, vertical line called a lifeline, where the box at its top specifies which entity the lifeline represents. The entities interact with each other through the transmission and reception of messages, which are shown as horizontal arrows from the transmitting lifeline to the receiving lifeline. We can also see that a lifeline can be both the sender and receiver of a message. The sequence diagram contains one reference (ref) to another sequence diagram. This reference can be replaced by the content of the sequence diagram that it refers to. The reference refers to the sequence diagram given in Figure 8, which describes the calculation of the basic key indicator $K_{\text{CH-NOT-APP-UNAUTH-ACC}}$. We do not present sequence diagrams describing the calculations of the two other basic key indicators, since these calculations are performed in the same way as the calculation of $K_{\text{CH-NOT-APP-UNAUTH-ACC}}$, and since these calculations involve the same types of sensors, actors, and components as the ones described in Figure 8. For the two other basic key indicators we only show that they are sent to "Component for calculating $K_{\text{NOT-APP-UNAUTH-ACC}}$," and that they are used in the calculation of $K_{\text{NOT-APP-UNAUTH-ACC}}$.

The sequence diagram in Figure 8 shows that the basic key indicator $K_{\text{CH-NOT-APP-UNAUTH-ACC}}$ is updated each week. The first thing that happens is that "Component for calculating $K_{\text{CH-NOT-APP-UNAUTH-ACC}}$" sends the value that was computed for the basic key indicator in the previous week to "Employee at Client H." Afterwards, the component identifies

Table VIII
THE RISK EVALUATION MATRIX WHEN THE PRECISE BUSINESS OBJECTIVE PBO-A8 IS FULFILLED

Likelihood \ Consequence	Insignificant	Minor	Moderate	Major	Catastrophic
Rare				$R4'$, $R5$	$R6$
Unlikely				$R4''$	
Possible		$R2$, $R3$			
Likely	$R1$				
Certain					

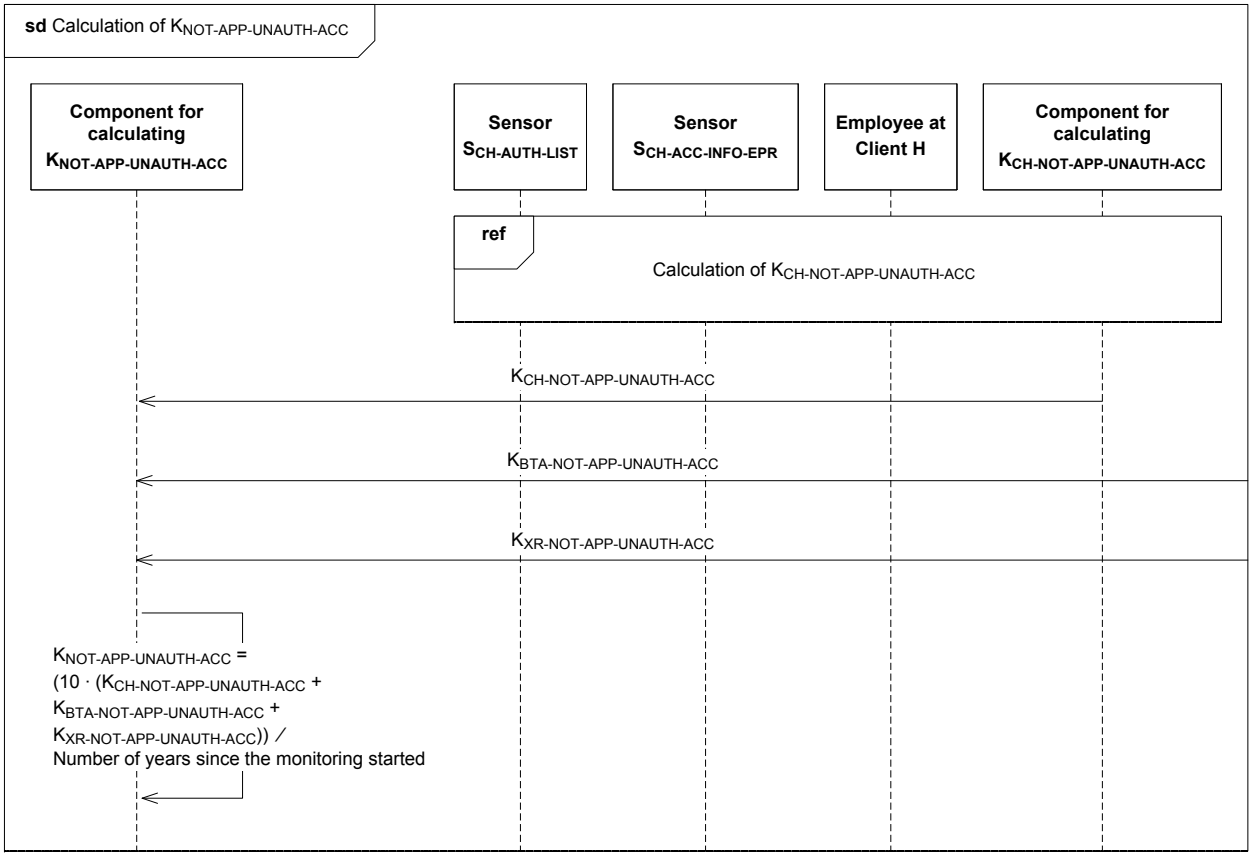


Figure 7. The sequence diagram "Calculation of $K_{\text{NOT-APP-UNAUTH-ACC}}$"

"All unauthorized accesses at Client H in the period of one week backwards to highly sensitive and private information in EPRs, where the owners of the EPRs are not patients of the accessors" based on input from the entities representing the sensors. The "Employee at Client H" performs a manual inspection of each of these unauthorized accesses, and classifies each as approved or not approved. If the unauthorized access is classified as not approved, then the basic key indicator is incremented by one. After all the unauthorized accesses have been inspected and classified, "Employee at Client H" sends the basic key indicator to the component which stores it. Afterwards, the component sends the basic key indicator to "Component for calculating $K_{\text{NOT-APP-UNAUTH-ACC}}$," as illustrated in the sequence diagram in Figure 7.

IX. EVALUATE CONSTRUCT VALIDITY

To evaluate whether the key indicator $K_{\text{NOT-APP-UNAUTH-ACC}}$ has construct validity, we re-do the risk analysis from Step 2.2 with the asset "Fulfillment of PBO-A8" replaced by the asset "Correctness of PBO-A8'." We have established that the monitoring infrastructure described in Step 2–4 is suitable for monitoring the relevant part of business. With the designs of the key indicators specified in the previous step, we want to identify in this step whether the proposed implementation of the monitoring infrastructure results in any new unacceptable

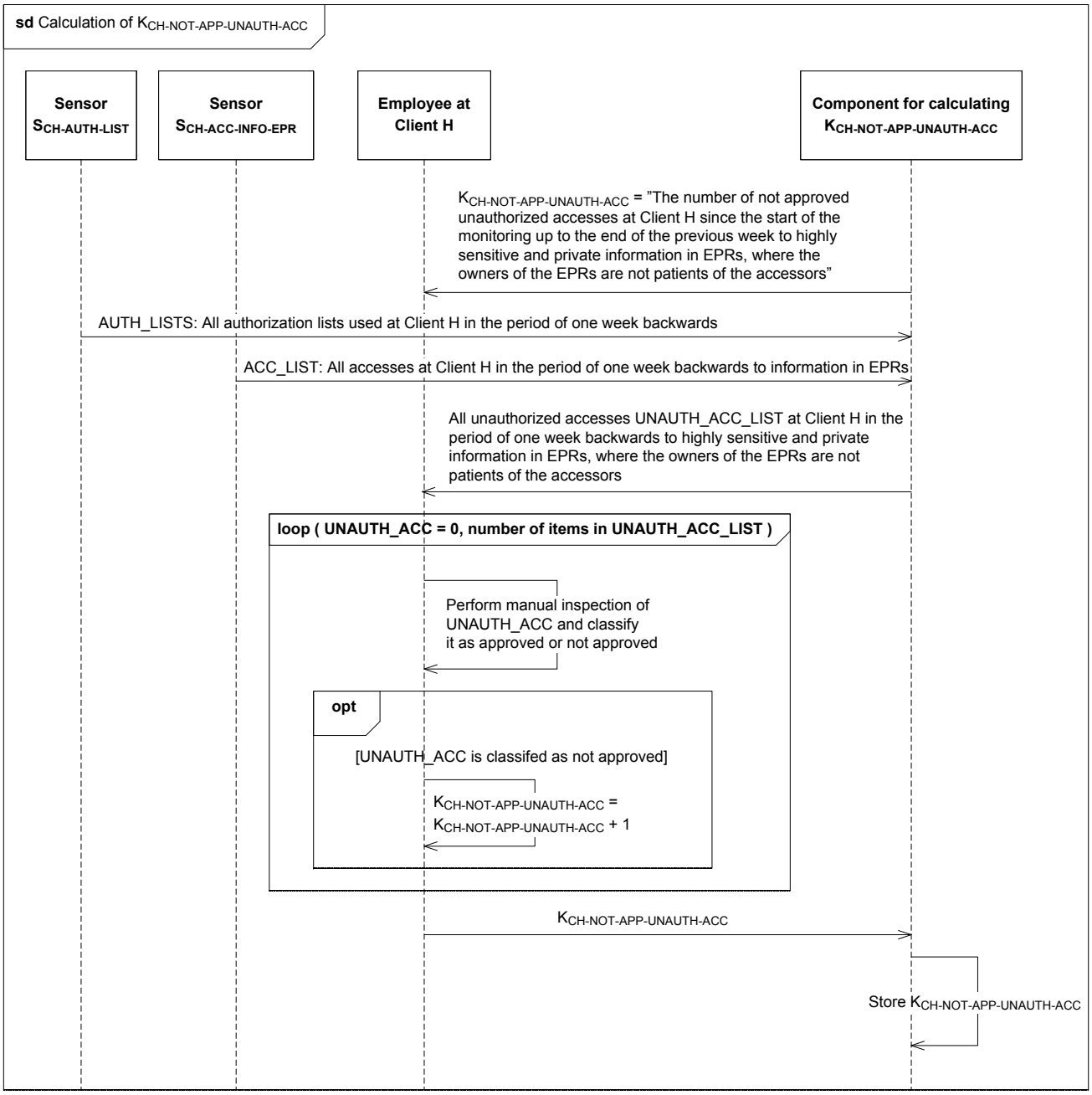


Figure 8. The sequence diagram "Calculation of $K_{\text{CH-NOT-APP-UNAUTH-ACC}}$"

risks. More precisely, we want to identify unacceptable risks towards the correctness of the reformulated precise business objective that are the result of threats to criteria for construct validity that $K_{\text{NOT-APP-UNAUTH-ACC}}$ needs to fulfill.

The result of the risk analysis is given in the CORAS threat diagram in Figure 9. We evaluate the construct validity of the composite key indicator based on the criteria given in Section III-F. Client H is of the opinion that the correctness of the key indicator $K_{\text{NOT-APP-UNAUTH-ACC}}$ referred to in the reformulated precise business objective PBO-A8' may be affected if the employees who classify unauthorized accesses as approved or not approved at X-ray and Blood test analysis are incompetent and fraudulent, respectively. Both these cases are examples of violation of the stability criterion, since the classification of unauthorized accesses as approved or not approved involves human decisions.

Moreover, Client H is worried that the sensor

$S_{\text{CH-ACC-INFO-EPR}}$ (represented as a non-human threat) may be unstable with respect to logging of accesses to information in EPRs. This is an example of violation of the instrument validity criterion. Besides the stability and instrument validity criteria, definition validity should also be evaluated. In our case, we say that a key indicator has definition validity if its design is clear and unambiguous so that the key indicator can be implemented correctly. The only thing that is not clear and unambiguous with respect to the design of $K_{\text{NOT-APP-UNAUTH-ACC}}$ is how unauthorized accesses should be classified as approved or not approved. Since this has already been covered during the evaluation of the stability criterion, we do not pursue this issue further.

The different types of behavior affect the correctness of the key indicator $K_{\text{NOT-APP-UNAUTH-ACC}}$, which again affects the correctness of PBO-A8'. In Table IX, two new risks $R7$ and $R8$ have been plotted according to their likelihoods and consequences. As we can see from the table, none of the new risks are unacceptable. We therefore conclude that the key indicator $K_{\text{NOT-APP-UNAUTH-ACC}}$ has construct validity.

X. Related work

To the best of our knowledge, there exists no other method for the design of valid key indicators to monitor the fulfillment of business objectives with particular focus on quality and ICT-supported monitoring of key indicators. There is a tool-framework called Mozart [23] that uses a model-driven approach to create monitoring applications that employs key performance indicators. We do not focus on the implementation of key indicators, but we specify what is needed for implementing them. The work in [23] also differs from our work by not designing indicators from scratch, but by mining them from a data repository during the design cycle.

An important part of our method is the assessment of the validity of the key indicators we design. Our approach to assessing validity is inspired by research conducted within the software engineering domain. As previously explained, there is however no agreement upon what constitutes a valid software metric [8]. A number of the software metrics validation approaches advocate the use of measurement theory [24][25][26] in the validation (see e.g., [9][27][28]). Measurement theory is a branch of applied mathematics that is useful in measurement and data analysis. The fundamental idea of this theory is that there is a difference between measurements and the attribute being measured. Thus, in order to draw conclusions about the attribute, there is a need to understand the nature of the correspondence between the attribute and the measurements. [29] is an example of an approach that relies on measurement theory for the validation of indicators. In [29], measurement theory is used to validate the meaningfulness of IT security risk indicators.

Measurement theory has been criticized of being too rigid and restrictive in a practical measurement setting. Briand et al. [27] advocate a pragmatic approach to measurement theory in software engineering. The authors show that even if their approach may lead to violations of the strict prescriptions and proscriptions of measurement theory, the consequences are small compared to the benefits. Another approach that takes a pragmatic approach to measurement theory is [28]. Here, the authors propose a framework for evaluating software metrics. The applicability of the framework is demonstrated by applying it on a bug count metric.

There exist also approaches that assess the validity of specific sets of key indicators. For instance, in [30] the validity of indicators of firm technological capability is assessed, while the validity of indicators of patent value is assessed in [31].

There are several approaches that focus on measuring the achievement of goals. One example is COBIT [32], which is a framework for IT management and IT governance. The framework provides an IT governance model that helps in delivering value from IT and understanding and managing the risks associated with IT. In the governance model, business goals are aligned with IT goals, while metrics, in the form of leading and lagging indicators [33], and maturity models are used to measure the achievement of the IT goals. In our approach we do not focus on the value that the use of IT has with respect to the business objectives. On the other hand, the risk that the use of IT has with respect to the business objectives is important. In our context, IT is relevant in the sense of providing the infrastructure necessary for monitoring the part of business that needs to fulfill the business objectives. In Step 6 of our method we identify risks that may result from the use of the monitoring infrastructure with respect to the business objectives.

Another way to measure the achievement of goals is by the use of the Goal-Question-Metric [34][35] (GQM) approach. Even though GQM originated as an approach for measuring achievement in software development, it can also be used in other contexts where the purpose is to measure achievement of goals. In GQM, business goals are used to drive the identification of measurement goals. These goals do not necessarily measure the fulfillment of the business goals, but they should always measure something that is of interest to the business. Each measurement goal is refined into questions, while metrics are defined for answering each question. No specific method, beyond reviews, is specified for validating whether the correct questions and metrics have been identified. The data provided by the metrics are interpreted and analyzed with respect to the measurement goal in order to conclude whether it is achieved or not. One of the main differences between our method and GQM is that we characterize precisely what it means to achieve a goal/objective. In GQM, however, this may be a question of interpretation.

In the literature, key indicators are mostly referred to

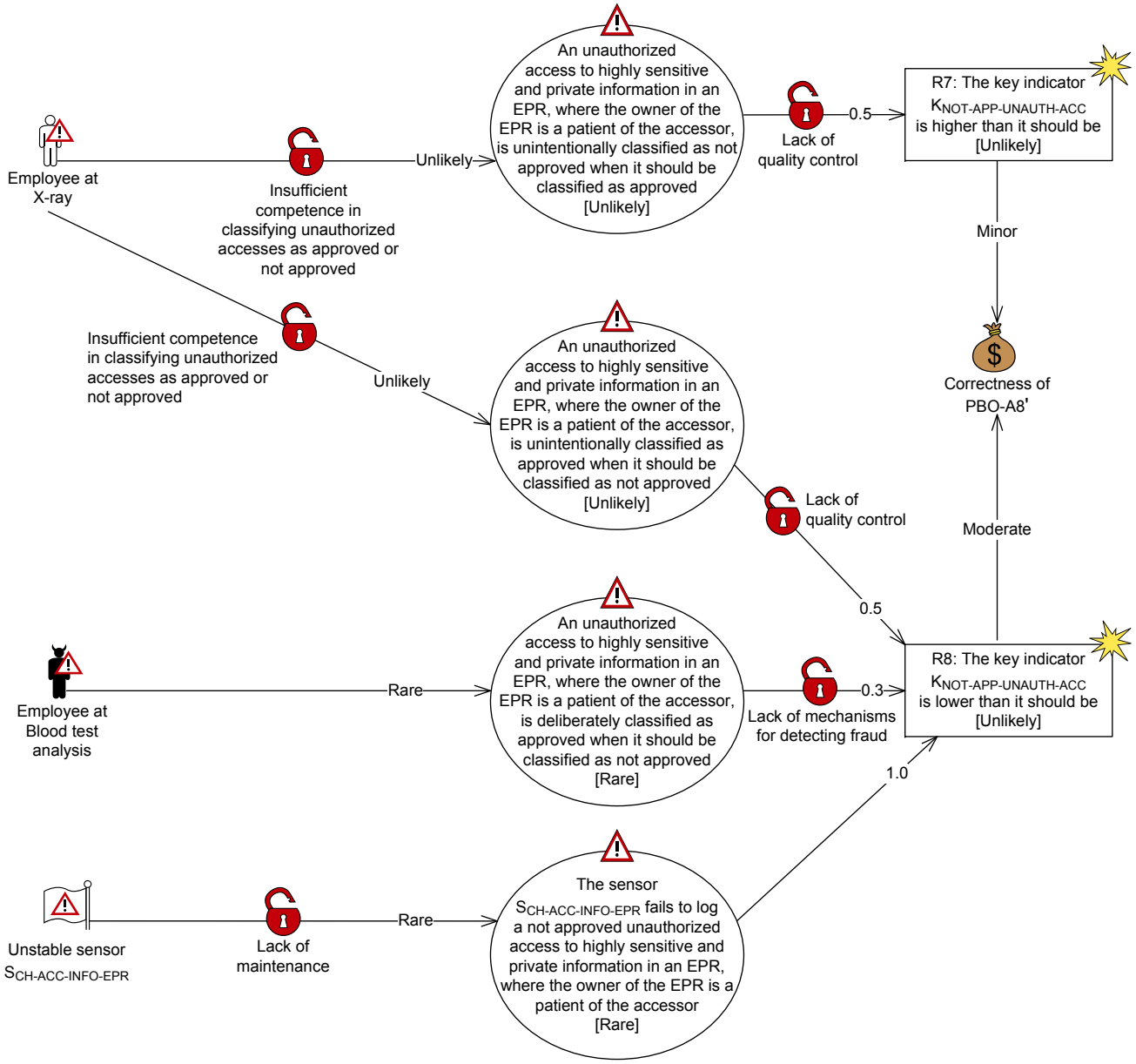


Figure 9. CORAS threat diagram documenting risks resulting from the proposed implementation of the monitoring infrastructure for the composite key indicator $K_{\text{NOT-APP-UNAUTH-ACC}}$

Table IX
THE RISK EVALUATION MATRIX FROM TABLE VIII WITH THE RISKS $R7$ AND $R8$ INSERTED

Consequence / Likelihood	Insignificant	Minor	Moderate	Major	Catastrophic
Rare				$R4'$, $R5$	$R6$
Unlikely		$R7$	$R8$	$R4''$	
Possible		$R2$, $R3$			
Likely	$R1$				
Certain					

in the context of measuring business performance. There exist numerous approaches to performance measurement. Some of these are presented in [36]. Regardless of the approach being used, the organization must translate their business objectives/goals into a set of key performance indicators in order to measure performance. An approach that is widely used [37] is balanced scorecard [5]. This approach translates the company's vision into four financial and non-financial perspectives. For each perspective a set of business objectives (strategic goals) and their corresponding key performance indicators are identified. However, the implementation of a balanced scorecard is not necessarily straight forward. In [38], Neely and Bourne identify several reasons for the failure of measurement initiatives such as balanced scorecards. One problem is that the identified measures do not measure fulfillment of the business objectives, while another problem is that measures are identified without putting much thought into how the data must be extracted in order to compute the measures. The first problem can be addressed in Step 4 of our method, while the second problem can be addressed in Step 3 and Step 5 of our method. In Step 3 we identify the sensors to be deployed in the relevant part of business, while in Step 5 we present the kinds of data that needs to be extracted from these sensors in order to compute the measures.

Much research has been done in the field of data quality. The problem of data quality is also recognized within the field of key indicators [39][40]. In [41] a survey on how data quality initiatives are linked with organizational key performance indicators in Australian organizations is presented. This survey shows that a number of organizations do not have data quality initiatives linked to their key indicators. Data quality should be taken into account when designing key indicators, since the use of key indicators based on poor quality data may lead to bad business decisions, which again may greatly harm the organization.

In [42][43] the problem of key indicators computed from uncertain events is investigated. The motivation for this work is to understand the uncertainty of individual key indicators used in business intelligence. The authors use key indicators based on data from multiple domains as examples. In these papers a model for expressing uncertainty is proposed, and a tool for visualizing the uncertain key indicators is presented.

XI. Conclusion

In [1] we presented the method *ValidKI* (Valid Key Indicators) for designing key indicators to monitor the fulfillment of business objectives with particular focus on quality and ICT-supported monitoring of key indicators. ValidKI facilitates the design of a set of key indicators that is valid with respect to a business objective. In this paper we have presented the improved and consolidated version of the method.

To the best of our knowledge, there exists no other method for the design of valid key indicators to monitor the fulfillment of business objectives with particular focus on quality and ICT-supported monitoring of key indicators. The applicability of our method has been demonstrated on an example case addressing the use of electronic patient records in a hospital environment.

Even though ValidKI has been demonstrated on a realistic example case there is still a need to apply ValidKI in a real-world industrial setting in order to evaluate properly to what extent it has the characteristics specified in the introduction and to what extent it can be used to design key indicators for systems shared between many companies or organizations. By applying ValidKI in such a setting we will also gain more knowledge regarding whether it is time and resource efficient.

Acknowledgments

The research on which this paper reports has been carried out within the DIGIT project (180052/S10), funded by the Research Council of Norway, and the MASTER and NESSoS projects, both funded from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreements FP7-216917 and FP7-256980, respectively.

References

[1] O. S. Ligaarden, A. Refsdal, and K. Stølen, "ValidKI: A Method for Designing Key Indicators to Monitor the Fulfillment of Business Objectives," in *Proceedings of First International Conference on Business Intelligence and Technology (BUSTECH'11)*. Wilmington, DE: IARIA, 2011, pp. 57–65.

[2] A. Hammond, A. Adriaanse, E. Rodenburg, D. Bryant, and R. Woodward, *Environmental Indicators: A Systematic Approach to Measuring and Reporting on Environmental Policy Performance in the Context of Sustainable Development*. Washington, DC: World Resources Institute, 1995.

[3] International Organization for Standardization, International Electrotechnical Commission, and Institute of Electrical and Electronics Engineers, "ISO/IEC/IEEE 24765 Systems and Software Engineering – Vocabulary," 2010.

[4] B. Ragland, "Measure, Metrics or Indicator: What's the Difference?" *Crosstalk: The Journal of Defense Software Engineering*, vol. 8, no. 3, 1995.

[5] R. S. Kaplan and D. P. Norton, "The Balanced Scorecard – Measures That Drive Performance," *Harvard Business Review*, vol. 70, no. 1, pp. 71–79, 1992.

[6] Object Management Group, "Unified Modeling Language Specification, Version 2.0," 2004.

[7] International Organization for Standardization and International Electrotechnical Commission, "ISO/IEC 9126 Information Technology – Software Product Evaluation – Quality Characteristics and Guidelines for their Use," 1991.

[8] A. Meneely, B. Smith, and L. Williams, "Software Metrics Validation Criteria: A Systematic Literature Review," Department of Computer Science, North Carolina State University, Raleigh, NC, Tech. Rep. TR-2010-2, 2010.

[9] A. L. Baker, J. M. Bieman, N. E. Fenton, D. A. Gustafson, A. Melton, and R. W. Whitty, "A Philosophy for Software Measurement," *Journal of Systems and Software*, vol. 12, no. 3, pp. 277–281, 1990.

[10] B. Kitchenham, S. L. Pfleeger, and N. Fenton, "Towards a Framework for Software Measurement Validation," *IEEE Transactions on Software Engineering*, vol. 21, no. 12, pp. 929–944, 1995.

[11] J. M. Roche, "Software Metrics and Measurement Principles," *ACM SIGSOFT Software Engineering Notes*, vol. 19, no. 1, pp. 77–85, 1994.

[12] B. Curtis, "Measurement and Experimentation in Software Engineering," *Proceedings of the IEEE*, vol. 68, no. 9, pp. 1144–1157, 1980.

[13] B. Henderson-Sellers, "The Mathematical Validity of Software Metrics," *ACM SIGSOFT Software Engineering Notes*, vol. 21, no. 5, pp. 89–94, 1996.

[14] N. E. Fenton, "Software Measurement: A Necessary Scientific Basis," *IEEE Transactions on Software Engineering*, vol. 20, no. 3, pp. 199–206, 1994.

[15] K. El-Emam, "A Methodology for Validating Software Product Metrics," National Research Council of Canada, Ottawa, ON, Tech. Rep. NCR/ERC-1076, 2000.

[16] J. P. Cavano and J. A. McCall, "A Framework for the Measurement of Software Quality," in *Proceedings of the Software Quality Assurance Workshop on Functional and Performance Issues*. New York, NY: ACM Press, 1978, pp. 133–139.

[17] R. Lincke and W. Lowe, "Foundations for Defining Software Metrics," in *Proceedings of 3rd International Workshop on Metamodels, Schemas, Grammars, and Ontologies (ateM'06) for Reverse Engineering*. Mainz: Johannes Gutenberg-Universität Mainz, 2006.

[18] M. E. Bush and N. E. Fenton, "Software Measurement: A Conceptual Framework," *Journal of Systems and Software*, vol. 12, no. 3, pp. 223–231, 1990.

[19] Council of Europe, "Convention for the Protection of Human Rights and Fundamental Freedoms," 1954.

[20] European Court of Human Rights, "Press Release – Chamber Judgments 17.07.08," 17. July 2008.

[21] Helsedirektoratet, "Code of Conduct for Information Security – The Healthcare, Care, and Social Services Sector," http://www.helsedirektoratet.no/publikasjoner/norm-for-informasjonssikkerhet/Publikasjoner/code-of-conduct-for-information-security.pdf, Accessed: 2012-06-21, 2. June 2010.

[22] M. S. Lund, B. Solhaug, and K. Stølen, *Model-Driven Risk Analysis: The CORAS Approach*, 1st ed. Berlin/Heidelberg: Springer-Verlag, 2010.

[23] M. Abe, J. Jeng, and Y. Li, "A Tool Framework for KPI Application Development," in *Proceedings of the IEEE International Conference on e-Business Engineering (ICEBE'07)*. Los Alamitos, CA: IEEE Computer Society, 2007, pp. 22–29.

[24] D. H. Krantz, R. D. Luce, P. Suppes, and A. Tversky, *Foundations of Measurement, Vol. I: Additive and Polynomial Representations*. New York, NY: Academic Press, 1971.

[25] P. Suppes, D. H. Krantz, R. D. Luce, and A. Tversky, *Foundations of Measurement, Vol. II: Geometrical, Threshold, and Probabilistic Representations*. New York, NY: Academic Press, 1989.

[26] R. D. Luce, D. H. Krantz, P. Suppes, and A. Tversky, *Foundations of Measurement, Vol. III: Representation, Axiomatization, and Invariance*. New York, NY: Academic Press, 1990.

[27] L. Briand, K. El-Emam, and S. Morasca, "On the Application of Measurement Theory in Software Engineering," *Empirical Software Engineering*, vol. 1, no. 1, pp. 61–88, 1996.

[28] C. Kaner and W. P. Bond, "Software Engineering Metrics: What Do They Measure and How Do We Know?" in *Proceedings of 10th International Software Metrics Symposium (METRICS'04)*. Los Alamitos, CA: IEEE Computer Society, 2004.

[29] A. Morali and R. Wieringa, "Towards Validating Risk Indicators Based on Measurement Theory," in *Proceedings of First International Workshop on Risk and Trust in Extended Enterprises*. Los Alamitos, CA: IEEE Computer Society, 2010, pp. 443–447.

[30] T. Schoenecker and L. Swanson, "Indicators of Firm Technological Capability: Validity and Performance Implications," *IEEE Transactions on Engineering Management*, vol. 49, no. 1, pp. 36–44, 2002.

[31] M. Reitzig, "Improving Patent Valuations for Management Purposes – Validating New Indicators by Analyzing Application Rationales," *Research Policy*, vol. 33, no. 6-7, pp. 939–957, 2004.

[32] IT Governance Institute, "COBIT 4.1," 2007.

[33] W. Jansen, *Directions in Security Metrics Research*. Darby, PA: DIANE Publishing, 2010.

[34] V. R. Basili and D. M. Weiss, "A Methodology for Collecting Valid Software Engineering Data," *IEEE Transactions on Software Engineering*, vol. SE-10, no. 6, pp. 728–738, 1984.

[35] R. V. Solingen and E. Berghout, *The Goal/Question/Metric method: A Practical Guide for Quality Improvement of Software Development*. New York, NY: McGraw-Hill International, 1999.

[36] A. Neely, J. Mills, K. Platts, H. Richards, M. Gregory, M. Bourne, and M. Kennerley, "Performance Measurement System Design: Developing and Testing a Process-based Approach," *International Journal of Operation & Production Management*, vol. 20, no. 10, pp. 1119–1145, 2000.

[37] T. Lester, "Measure for Measure," http://www.ft.com/cms/s/2/31e6b750-16e9-11d9-a89a-00000e2511c8.html#axzz1ImHJOLmg, Accessed: 2012-06-21, 5. October 2004.

[38] A. Neely and M. Bourne, "Why Measurement Initiatives Fail," *Measuring Business Excellence*, vol. 4, no. 4, pp. 3–6, 2000.

[39] S. M. Bird, D. Cox, V. T. Farewell, H. Goldstein, T. Holt, and P. C. Smith, "Performance Indicators: Good, Bad, and Ugly," *Journal Of The Royal Statistical Society. Series A (Statistics in Society)*, vol. 168, no. 1, pp. 1–27, 2005.

[40] D. M. Eddy, "Performance Measurement: Problems and Solutions," *Health Affairs*, vol. 17, no. 4, pp. 7–25, 1998.

[41] V. Masayna, A. Koronios, and J. Gao, "A Framework for the Development of the Business Case for the Introduction of Data Quality Program Linked to Corporate KPIs & Governance," in *Proceedings of the 2009 Fourth International Conference on Cooperation and Promotion of Information Resources in Science and Technology (COINFO'09)*. Los Alamitos, CA: IEEE Computer Society, 2009, pp. 230–235.

[42] C. Rodríguez, F. Daniel, F. Casati, and C. Cappiello, "Computing Uncertain Key Indicators from Uncertain Data," in *Proceedings of 14th International Conference on Information Quality (ICIQ'09)*. Potsdam/Cambridge, MA: HPI/MIT, 2009, pp. 106–120.

[43] C. Rodríguez, F. Daniel, F. Casati, and C. Cappiello, "Toward Uncertain Business Intelligence: The Case of Key Indicators," *Internet Computing*, vol. 14, no. 4, pp. 32–40, 2010.

From Linked Data and Business Intelligence to Executable Reality

Vagan Terziyan
Department of Mathematical Information Technology,
University of Jyvaskyla
P.O. Box 35 (Agora), 40014, Jyvaskyla, Finland
e-mail: vagan@jyu.fi

Olena Kaykova
Industrial Ontologies Group, Agora Center,
University of Jyvaskyla
P.O. Box 35 (Agora), 40014, Jyvaskyla, Finland
e-mail: olena@cc.jyu.fi

Abstract—**This paper presents the concept of "Executable Knowledge", which is based on Linked Data and in addition to traditional subject-predicate-object semantic triplet model it contains also subject-predicate-query triplets. Actual values for such "executable" properties are supposed to be queried or/and computed whenever requested "on-the-fly" from/by some internal or external information source or computational capability provider at the right time and place according to the dynamic user context. We discussed two possible applications of this concept. One, which we named "Executable Reality", will enhance emergent (Mobile) Augmented and Mixed Reality concepts within two dimensions: utilization of Linked Data and Business Intelligence on top of it. Executable Reality will provide a real-time context-aware analytics related to various real-life objects selected by the users from their terminals. Other executable knowledge benefits are shown in the context of educational quality assurance and related to personalized online quality evaluation and ranking of various academic resources (people, departments, universities, etc.). It is demonstrated that the special Quality Assurance Portal for higher education may automatically utilize business analytics on top of Linked Data in the form of executable knowledge.**

Keywords- Business Intelligence; Linked Data; Mixed Reality; Executable Reality; Executable Knowledge; Quality Assurance

I. INTRODUCTION

Business intelligence (BI) can be considered as a set of methods, techniques and tools utilized on top of business data to compute (acquire, discover) additional (implicit) analytics out of it and to present it in a form suitable for decision-making, diagnostics and predictions related to business. Taking into account that "business data" is becoming highly heterogeneous, globally distributed (not only in the Internet space but also in time), huge and complex, extremely context sensitive and sometimes subjective, the ways the BI is utilized have to be qualitatively changed. Semantic (Web) Technology [1][2][3][4] is known to be a suitable approach to enable more automation within BI-related data processing. The vision of BI 2.0 [5] includes also issues related to Service-Oriented Architecture (SOA), mobile access, context handling, social media, etc. All these issues will also definitely benefit from adding semantics [6][7]. It is however a known fact that there is not much semantically annotated data available for BI. We have to live with data sets created independently, according to different schemas or even data model types. The realistic role of Semantic Technology for such data would be linking related "pieces" of it with some semantic connections and by doing this transforming the original data into the Linked Data.

There are no doubts that such semantically interlinked "islands" of data have a lot of hidden (implicit) and potentially interesting information that none of the separate data sets has alone. Now the challenge would be to utilize BI on top of Linked Data to be able to get all the benefits from semantic enhancement of the data.

Another trend is related to fast development of technology for better delivery and visualization of information. Among those there are Augmented Reality [12] and Mixed Reality [13] technologies and their mobile versions [14][15][16]. They are based on automated interlinking of various Web-based digital data collections with the real-time data from sensors about physical world and presenting it in a useful form for a user. An interesting topic would be considering these technologies in the context of business data or even BI-provided analytics. This may inspire more professional use of Augmented and Mixed Reality in addition to public use of it.

In this paper we propose "Executable Reality" as an enhancement of the "Mixed Reality" concept within two dimensions (utilization of Linked Data and BI on top of it). We present "Executable Knowledge" as a tool to enable Linked Reality and "Executable Focus" to control it by a user. Executable knowledge in addition to subject-predicate-object semantic triplet model (in ontological terms) contains also subject-predicate-query triplets ("executable properties"). Actual value for the properties based on a new triplet will be computed "on-the-fly" (based on user request navigated by executable focus) by some online BI service or other computational capability provider at the right time and place and according to the dynamic user context.

The rest of the paper is organized as follows: in Section II we discuss Linked Data issues and its enhancement by context-sensitive similarity links; in Section III we present (Mobile) Augmented and Mixed Reality technology and challenges; In Section IV we show how these technologies can be further developed towards "Executable Reality" on top of enhanced Linked Data and BI services (there we also present the concept of "Executable Knowledge"); we discuss Related Work in Section V; show one implementation of Executable Knowledge related to educational quality assurance in Section VI, and we conclude in Section VII.

II. LINKED DATA, CO-REFERENCE AND SEMANTIC SIMILARITY

Linked Data is a concept closely related to the Semantic Web yet providing some specific facet to it. According to Tim Berners-Lee "The Semantic Web is not just about putting data on the web. It is about making links, so that a person or machine can explore the web of data. With linked data, when you have some of it, you can find other, related, data" (http://www.w3.org/DesignIssues/LinkedData.html). The so called "5 stars" advice from Tim Berners-Lee to enable Linked Data includes: making data available on the web (whatever non-proprietary format) as machine-readable structured data, utilizing open standards from W3C (RDF and SPARQL) to identify things and finally linking the data to other people's data to provide context.

According to Kingsley Idehen (OpenLink Software CEO), due to development of Semantic Technology, meshing (or natural data linking) will replace mashing (brute-force data linking) and therefore mesh-ups can be considered as a next step comparably to the mash-ups in the sense to merge and integrate different data sources and processing devices to provide new information services.

Linked Data can be considered as an outcome of the technology, which semantically interconnects heterogeneous data "islands" (e.g., as shown in Figure 1). Even if the original sources of data are highly heterogeneous (not just only different schema of data within the same data model type but also different data model types), still it is possible to build some "bridges" between entities from these data sources utilizing semantic technology. The traditional Semantic Web approach would be: (a) creating a semantic model of the domain (ontology), (b) replacing original data from each source with full semantic (RDF) representation of its resources in terms of the ontology. Of course such an approach enables seamless integration of the original data and simplifies the usage of it. However with distributed and dynamic sources of data, which are managed and constantly updated independently, it would be difficult to provide such "semantic synchronization" (updating metadata and mapping it to the ontology) in real time. Therefore Linked Data would be less ambitious and the more practical approach would be: data sources are managed independently as they used to be; semantic connections between appropriate resources from different sources will be either automatically discovered or manually created whenever appropriate. Usage experience and usability performance for each separate data source will be preserved. The usability of such "virtually integrated" data storages will increase with the increase of the amount of the semantic "bridges".

According to [8] there are three important types of RDF links within Linked Data:

(a) "relationship links" that point at related things in other data sources (like "object properties" in terms of OWL: *owl:ObjectProperty*);

(b) "identity links" that point at URI aliases used by other data sources to identify the same real-world object or abstract concept (e.g., *owl:sameAs*, *owl:sameIndividualAs*, *owl:equivalentClass*);

(c) vocabulary links that point from data to the definitions of the vocabulary terms that are used to represent the data (like "datatype properties" in terms of OWL: *owl:DatatypeProperty*).

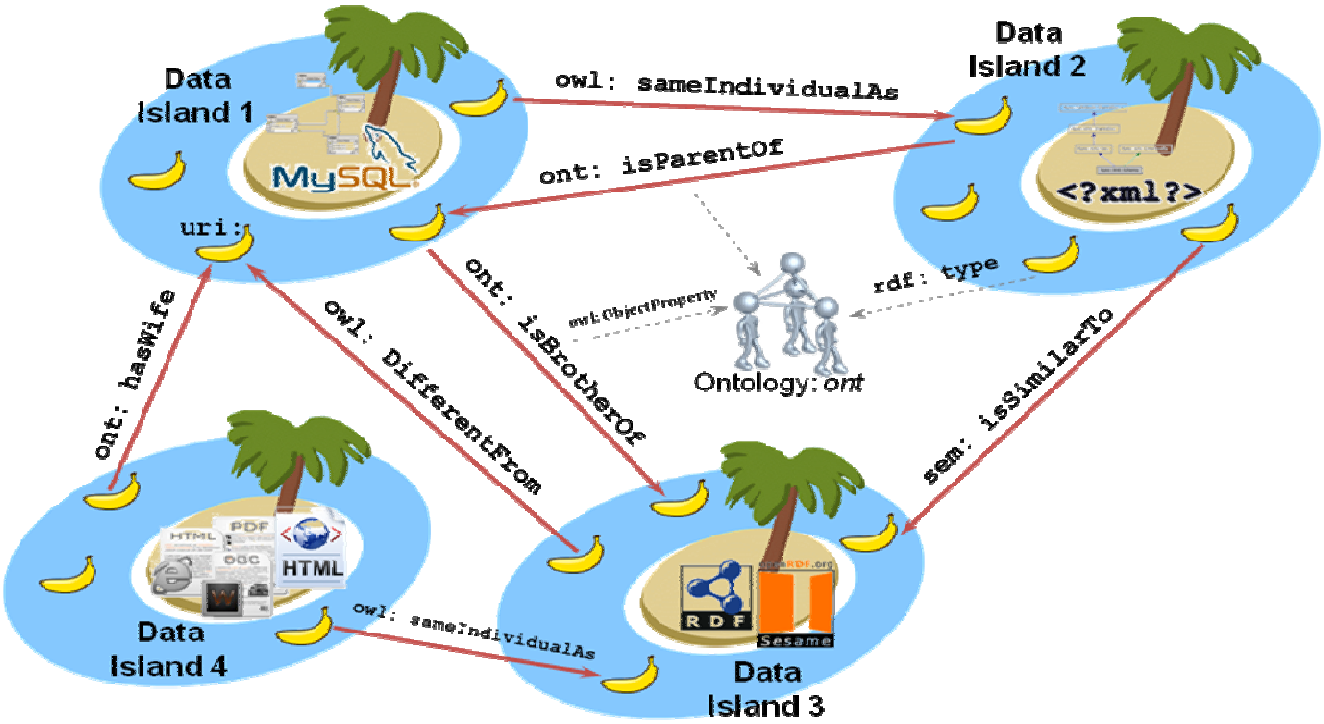


Figure 1. Linked Data: "bridges" between heterogeneous "islands" of data

We think it would be reasonable to extend traditional explicit semantic links within Linked Data with the implicit ones, e.g., those, which could be automatically derived by various reasoners. Among those special attention should be paid to the "semantic similarity" links. Usually, when someone queries a data, she looks for the resource(s), which are "the same" as the one specified in the query. However often there are none of such found. In many cases there is a sense to find "similar" resources to the target one. Similarity search was always a big issue within many disciplines and it is especially important for the Linked Data. The reason is related to the fact that actually same resources in different "islands" of data may have different URIs and often quite extensive work should be done to recognize same resources. Usually first we see that some resources look similar and therefore in practice could be the same ones and then we perform some check on the identity of the resources.

Semantic similarity search is not a trivial task because it should take into account heterogeneity of data types representing properties of the resources being compared, i.e., it should have a special distance calculator for each particular data type, then some normalization function for component distances (for each attribute of the compared resources) and finally some aggregation function of the component distances into the final distance. Also, depending on the context, different attributes of the resources may have different importance (weight) for the final distance aggregation function. Sometimes the context may influence not only the importance of an attribute but even the choice of the distance function itself.

Different sources of information (even isolated ones) are using the same words (concepts) from the real world to refer to particular groups of objects, people, events, etc. To enable automated information processing and interoperability, the providers are using URIs to distinguish between different instances of the same concept and trying to guarantee that once defined URI for something will stay the same throughout the whole set of documents coming from the same data source across all its history. It is obvious however that multiple sources of Linked Data cannot afford mutual awareness and sharing of the URIs, which results in URIs ambiguity in a global scale. In the context of Linked Data, the problem of determination of equivalent URIs referring to the same concept or entity is commonly known as "co-reference resolution" [26]. The problem is not that simple and traditional approaches to connect appropriate instances of data with *owl:sameAs* relation are not always working as shown in [27]. Also in [28] authors argue that in some contexts the comprehensive inference based on *owl:sameAs* relation for co-referenced entities is not possible due to hidden variations of the *owl:sameAs* semantics.

One of the well-known areas, in which co-reference becomes a major problem, is in author disambiguation [29]. There are many authors who share the same name and distinguishing between them is a vital part of any digital library or citation system. At the same time not only authors share the same names but variation in the spelling of names also leads to a single author having multiple identities (see example in Figure 2). This example is related to the academic career history of some female researcher. First stage of her career happens to be in USSR (Russian-speaking environment) and therefore first records on her identity (name, affiliation, etc.) and academic record (degrees, publications, projects, etc.) appear in the Web in Russian (Cyrillic letters used as it can be seen from downside of the Figure). Later she got an international passport where her name was transliterated from Cyrillic "Кайкова" into Latin "Kaikova". Her publication record since then has been indexed by Google Scholar according to this new identity and therefore Кайкова and Kaikova start to exist as two different persons (different URIs). Later, when time comes to change old international passports to the new ones, the transliteration rules changed and the new version of the name appear to be "Kaykova". After that all new publications of the researcher has been indexed by Google Scholar based on this new identity, which means that for this Web service there exist 3 different persons, which in reality is the same one (Figure 2). Assume that some Web application (e.g., the one making BI-driven academic quality summary report on some university) is going to automatically check citation index (e.g., h-factor) of this person and makes automatic query to Google Scholar with her current identity (Kaykova). It will get a number (e.g., h=10) based on incomplete publication record. Even if to manually make all 3 queries for "Кайкова", "Kaikova" and "Kaykova" and get 3 outcomes, e.g., h1=5, h2=18, h3=10, there is no way to automatically compute overall h-factor without analyzing content of all 3 publication sets. The problem actually is more complicated because all the 3 names may also belong to some other persons.

Summarizing the co-reference problem: 1) The same resource (e.g., a person, which has some record published in the Web) may have different URIs in different Web documents or databases; 2) Different resources may happen to be represented in some Web records with the same URIs due to similar identities; 3) BI-based computing reports being made separately on top of the records belonging to the same resource may not be integrated afterwards easily (if at all); 4) Explicit co-reference knowledge on similarity among resources, e.g., *owl:sameIndividualAs* network among distributed URIs, would be helpful; 5) Automated discovery of same resources in distributed records is not a trivial task; 6) Creating globally shared repositories of all Web resources with their identities is not a trivial task either (if realistic at all); 7) Relations like *owl:sameIndividualAs* may have different hidden semantics in different contexts (time, location, goal, preferences, etc.) and therefore should be carefully analyzed against the context when applied; 8) The co-reference problem handling is the one of the great importance for future potential of Business Intelligence, which is expected to be automatically applied for the Linked Data utilization.

Figure 2. Co-reference problem visualized

OKKAM (www.okkam.org), as a Large Scale Integrating Project (January 2008 - June 2010) co-funded by the European Commission, was looking for a scalable and sustainable solution for systematic and global identifier reuse in decentralized information environments enabling users to get and create globally shared URIs [30]. Created so far OKKAM repository of about 7.5 million entities cannot however solve co-reference problem at a full scale to be used by Linked Open Data community.

Other use of similarity measure (than co-reference resolution) is when one is looking for a capability-providing resource (e.g., a service), cannot find exactly the one she wants but still will be satisfied by finding a resource with similar functionality. Therefore it would be reasonable to have some explicit similarity links between stored data entities obtained as a result of appropriate similarity search procedures. The two major challenges here are: (a) a resource within one data "island" may have very different model of description when compared to some resource within another data "island" (e.g., a human documented in a relational database will not be easily compared with a human from some XML storage or from some html document); (b) some resources being very different in one particular context could be considered as similar ones in some other context.

We consider three types (sub-properties in terms of OWL) of semantic similarity based on common ternary object property relation named *isSimilarGivenContext* and they are: (1) *isSimilarGivenQuery*; (2) *isSimilarGivenGoal*; and (3) *isSimilarGivenRole*.

The first type of similarity assumes that two resources can be considered as similar ones (in the context of some semantic query, e.g., SPARQL query) if this query, being applied over the locations of these two resources, returns both of them as a result. See example in Figure 3(a). Here the resources "Mikhail S. Gorbachev" and "George W. Bush" are shown to be inferred as the similar ones, given query "Former president, male with at least one daughter".

The second type of similarity applies to the resources which can be replaced with each other as input parameters needed to perform some function (action) or utilize some external capability (product or service) for achieving some goal without affecting expected outcome. For example, a "Bugatti Veyron" car would be a similar resource to e.g., "Expensive diamond ring" as an "input" ("making wedding present"), given goal "To make the girlfriend happy" as it is shown in Figure 3(b).

The third type of similarity assumes that two resources will be used as similar ones if they both can fill some slot in a business process with the specified role. Consider the example in Figure 3(c). Here some resource (instance of class "Lamp") has been computed as similar one to another one (instance of class "Candle"), given role "Lightening".

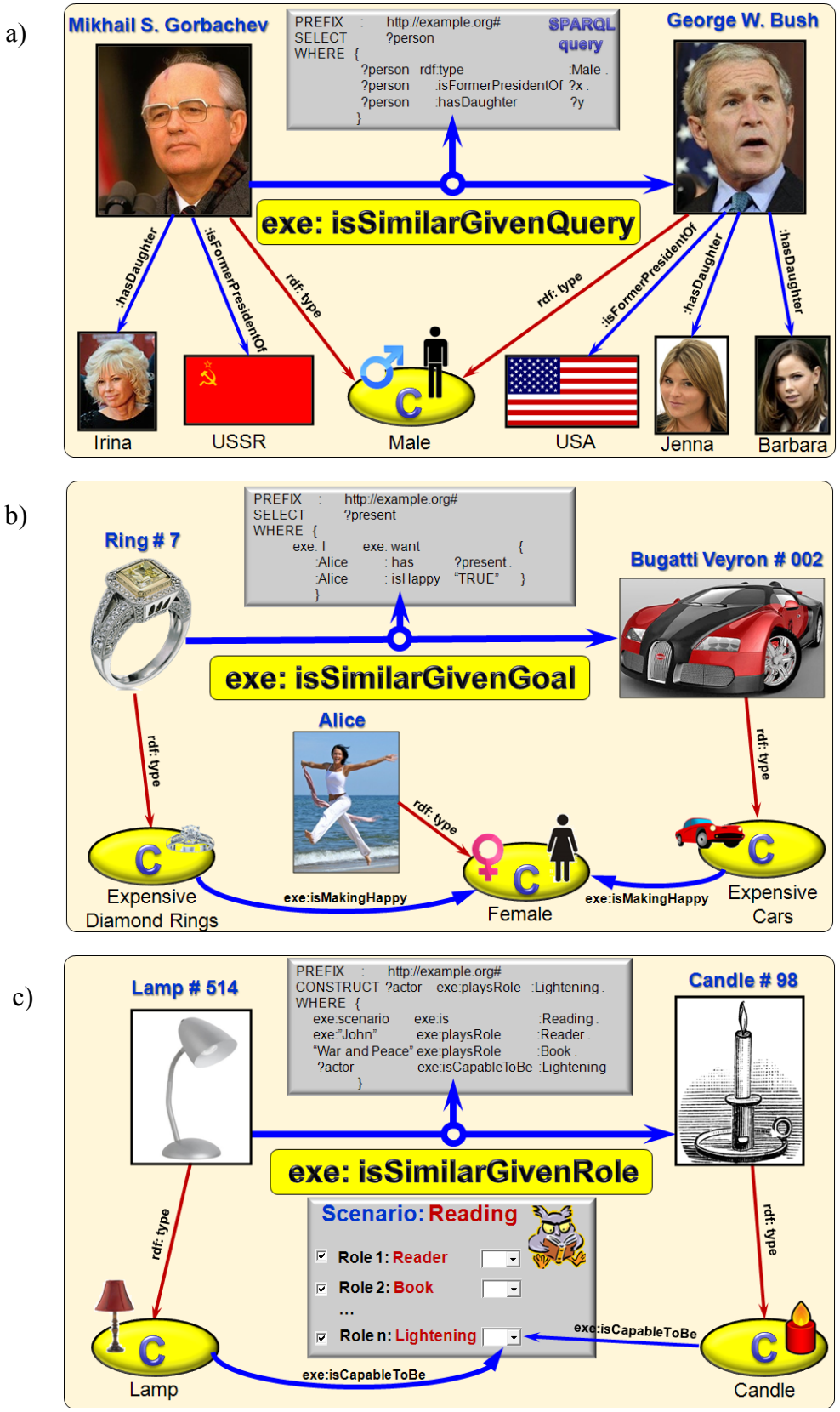


Figure 3. Three types of similarity relations: a) similar in the same context; b) similar to be used to reach the same goal; c) similar when playing the same role

More information about our approach for defining context in various practical applications and semantic similarity search within context can be found in [9][10][11].

The major challenge is how to provide support for automated utilization of Linked Data, which in fact remains heterogeneous, and how to get added value of additional semantic connections between data components. Anyway we claim that providing similarity links, in addition to traditional types of RDF links described in [8], can be very helpful for practical utilization of Linked Data and we will try to show this in the following sections of the paper.

III. AUGMENTED AND MIXED REALITY

Augmented Reality (AR) [12] is a technology aimed to enhance the traditional perception of a reality (real-world environment), which elements are augmented by computer-generated sensory input (e.g., data, sound or graphics). AR enriches real world for the user rather than replaces it. By contrast, *virtual reality* replaces the real-world with a simulated one. Emerging development of mobile computing has naturally resulted to growing interest towards Mobile AR [15] and also to Ubiquitous Mobile AR [14] for successfully bridging the physical world and the digital domain for mobile users. The AR concept has been further developed to Mixed Reality [13], which means merging of real and virtual worlds to produce new environments and visualizations where physical and digital objects co-exist and interact in real time. In June 2009, Nokia Research Center announced the vision [16] of Mobile Mixed Reality, according to which a phone becomes a "magic lens" (smart and context-aware), which lets users look through the mobile display at a world that has been supplemented with information about the objects that it sees. The users simply point their phone's camera, and look "through" the display. Objects of interest visible in the current view will be gathered from existing Point-Of-Interest databases or created by the user and will be highlighted. They can be associated with physical objects or featureless spaces like squares and parks. Once selected, objects provide access to additional information from the Internet and hyperlinks to other related objects, data, applications and services. Context-awareness is guaranteed by various rich sensors that are being incorporated into new phones (GPS location, wireless sensitivity, compass direction, accelerometer movement, sound and image recognition, etc.). Therefore the new technology is going to actively utilize acquired dynamic context to better filter and select relevant information about surrounding real-world objects for a user.

In the following section we further develop the concept of (mobile) mixed reality within two dimensions: the first one is related to Linked Data utilization and the second one will be related with the utilization of Business Intelligence through "Executable Knowledge".

IV. TOWARDS "EXECUTABLE" REALITY

The concept of "Executable Reality" and associated technology, which we are offering, should be considered as an extension of the (Mobile) Mixed Reality concept and the technology. If the traditional technology assumes that the explicitly available relevant data about some real-world object will be taken from some database and delivered to the user on demand (based on her attention focus pointed to this object), the Executable Reality in addition is able to provide online BI computation based on similar demand (we call it "Executable Focus") and present to the user results of computed analytics adapted for the current context.

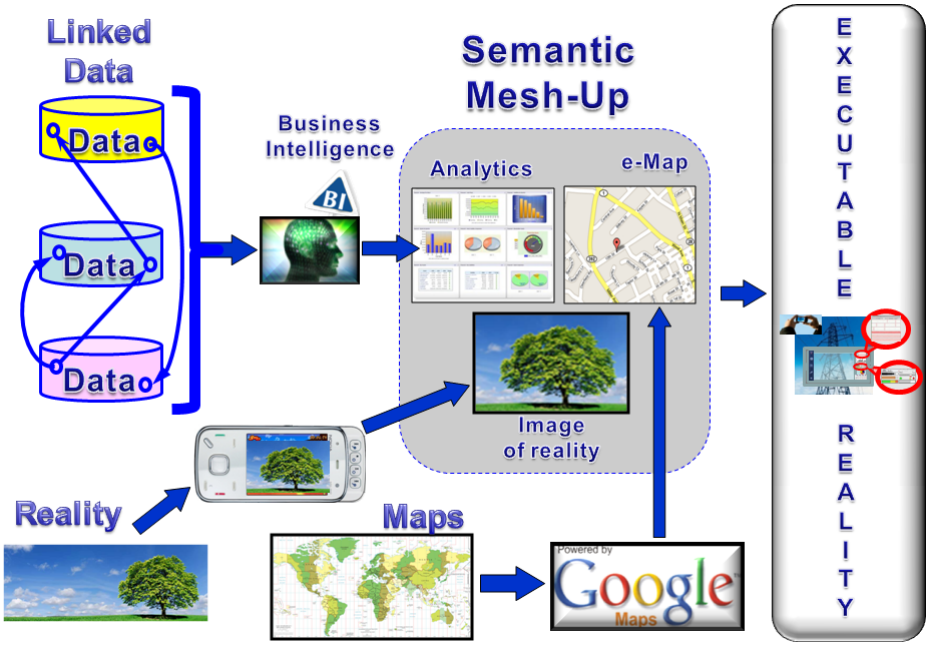


Figure 4. Executable-Reality-related process illustrated

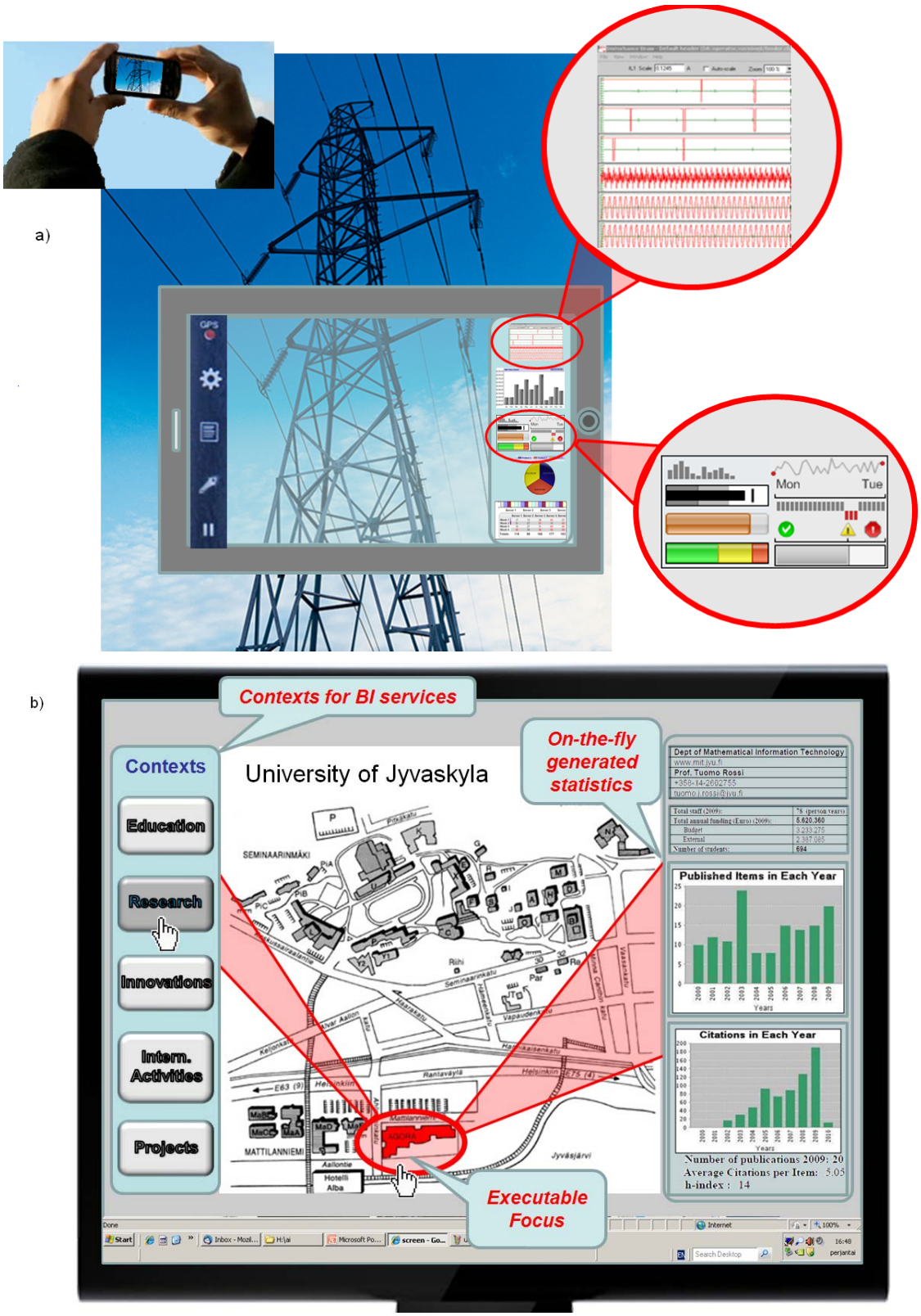


Figure 5. Executable Reality use case examples: (a) on-the-fly computed statistics about power line performance is delivered to mobile terminal of the maintenance engineer on implicit demand; (b) research performance statistics is delivered to the user based on chosen unit (click on the building where the university department is located and selecting context "research" for filtering appropriate data from the unit needed for research performance calculations)

Consider an architecture of a typical Executable Reality process in Figure 4. The content of the three information channels has been mixed in a kind of "mesh-up" (assuming that similar resources from all the channels are recognized, identified and semantically mapped together). The first channel presents the "image of reality" taken through the focus of some sensor (e.g., mobile phone camera). The second channel presents the geographic information (e.g., received online via Google Maps service). Linking the content of these two channels together provides a typical case of the Mobile Mixed Reality. Let us however consider the third channel as shown in Figure 4. The initial sources of its content are assumed to be heterogeneous, distributed and interconnected according to the Linked Data layer of semantics. In contrast to the Mixed Reality, the Executable Reality processes are not only capable to map the actual content from the Linked Data original sources into the structured image of Reality, but they are also capable to map the analytics (outcomes of BI software-as-a-services) taken automatically from the Linked Data into the image of Reality. A typical interface from an Executable Reality application will look as follows. Some sensor (e.g., mobile phone camera) gives to a user a snapshot of the Reality. The user clarifies a focus of her attention (e.g., selects certain object highlighted on the snapshot), "clicks" on it (directly or through special additional controls) and as a result, instead of getting just traditional (e.g., Mobile Mixed Reality) information output, she will get also some analytics associated with the selected object based on information queried and processed on-the-fly from the Linked Data.

Two use-case scenarios for the Executable Reality are shown in Figure 5 (a, b). In the first one, the user (maintenance engineer of the power network company) is putting the executable focus (smart phone camera) into the direction of the power line and by doing this makes implicit request (associated with the profession of the user, knowledge about the context and the type of resource recognized) for, e.g., the last 24 hours performance statistics of this power line. The query will go further to the server; appropriate BI service will be selected (based on semantic comparison of the query and available service descriptions) and automatically invoked; a resulting page with numbers, graphics (and sounds if appropriate) will be generated and delivered back to the terminal and shown in the appropriate window of the screen as shown in Figure 5 (a).

Another scenario in Figure 5 (b) shows that a user observes the campus map of some university and selects the building where a particular department is located. The user also selects the context in which she wants to get performance statistics of the unit, e.g., "research". Chosen object and the context together form the executable focus, which will automatically generate the query for the required computation. Then the process goes in a similar way as with previous scenario and the user will get "fresh" statistics (assuming that some remote Web-services, i.e., some public citation indexes collectors will be automatically queried and processed) concerning performance of the department.

To enable such scenarios we have to find an effective way to utilize Linked Data, which is a natural source for online BI computations, and also to enable BI functionality as semantically annotated Web services. We propose to organize Linked Data in form of "Executable Knowledge".

Executable Knowledge can be considered as distributed (or organized as a cloud) set of heterogeneous data storages and computational services (e.g., BI) interconnected with semantic (RDF) links. The major feature here is that, in addition to the traditional ("subject-predicate-object" or "resource-property-value") triplet-based semantics of an RDF link (either "datatype" property, where "value" is a literal; or "object" property, where "value" is another resource), the new model will have new property type named "executable property" with the structure: "subject-predicate-query". It is supposed that reasoners, engines, etc., working with such knowledge will execute the query within the target triplet and treat the obtained result as a value for the property. Two immediate advantages of this extension are: (a) the triplet will always implicitly keep knowledge about most recent value for the property because the query to some data storage or to some BI function will be executed only on demand when needed and the latest information will be delivered; (b) the query may be written according to different standards, data representation types, models and schemas so that heterogeneity of original sources of data and capabilities will not be a problem. Therefore distributed data collections can be maintained independently (autonomously) and "queried" in real time by executable RDF links.

Consider an example in Figure 6. Here the executable statement in RDF (N3 syntax) means: "If you want to know with whom John is currently in love, execute the query Q1". The query Q1 (prefix "exe:" points to Executable Knowledge ontology and indicates that the RDF statement is executable) in this case is semantically described as a SPARQL query to the RDF data storage and it means: "Select the girl from the current database of staff, who is colleague of John, has red hair and is 25 years old". When the SPARQL query engine executes the query and finds that "Mary" fits it, the executable RDF statement is transformed into the traditional one (reference to the query "exe:Q1" is replaced with actual value "Mary"). Notice that, if the same knowledge will be explored after 1 year, then the same executable statement will be transformed into: "John is in love with Anna", because staff data (separate source) will be autonomously updated (Anna becomes 25 years old) and RDF connections (semantic layer of Linked Data) on top will be automatically updated when executed.

Consider similar example in Figure 7 and notice that here we have an SQL query to some relational database as implicit value of executable RDF statement. The query Q2 means request for computing average journal papers' publication performance of young (< 30) PhD students. The original executable RDF statement means: "If you want to know average performance of young doctoral students in AI Department, execute query Q2". When the query returns computed value, the executable RDF statement is transformed into the traditional one (reference to query "exe:Q2" is replaced with actual value "7", which means that the "executable" RDF property is replaced with the "datatype" property).

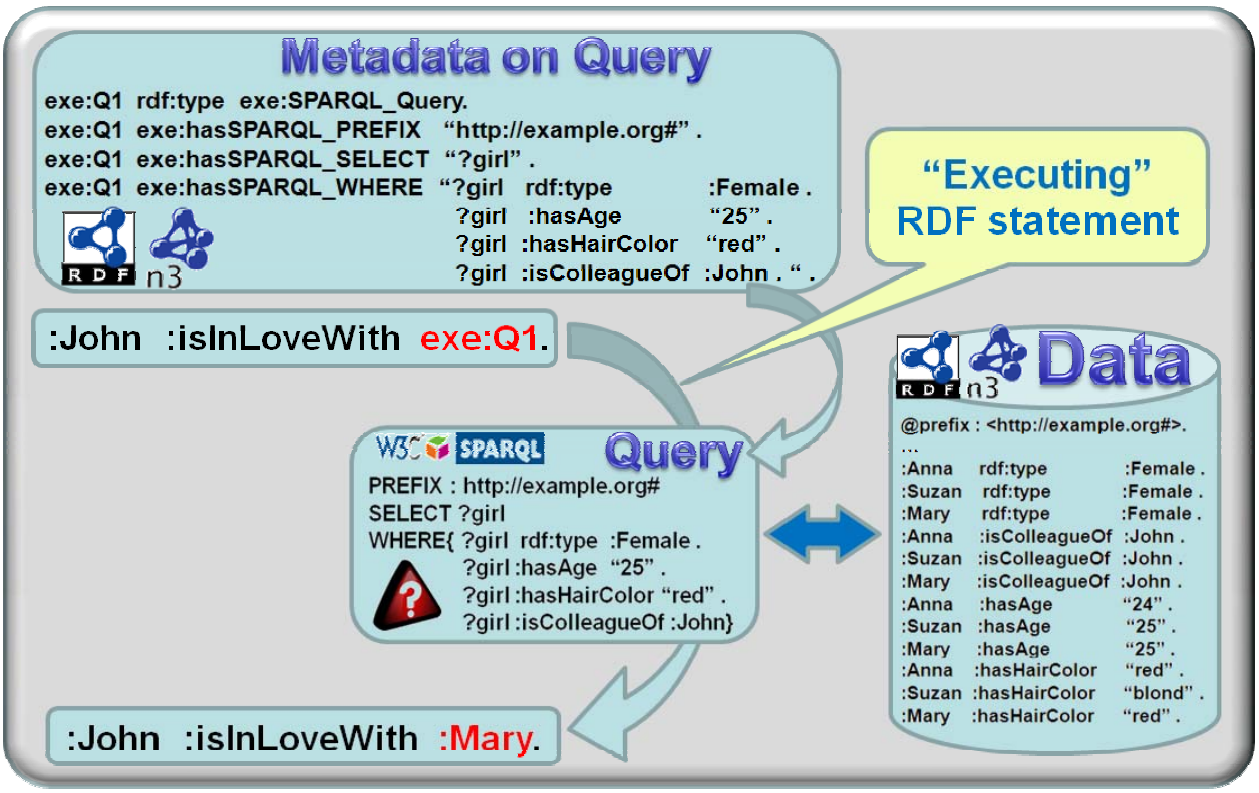


Figure 6. Example of processing executable RDF statement, which contains implicit value as SPARQL query to the RDF storage

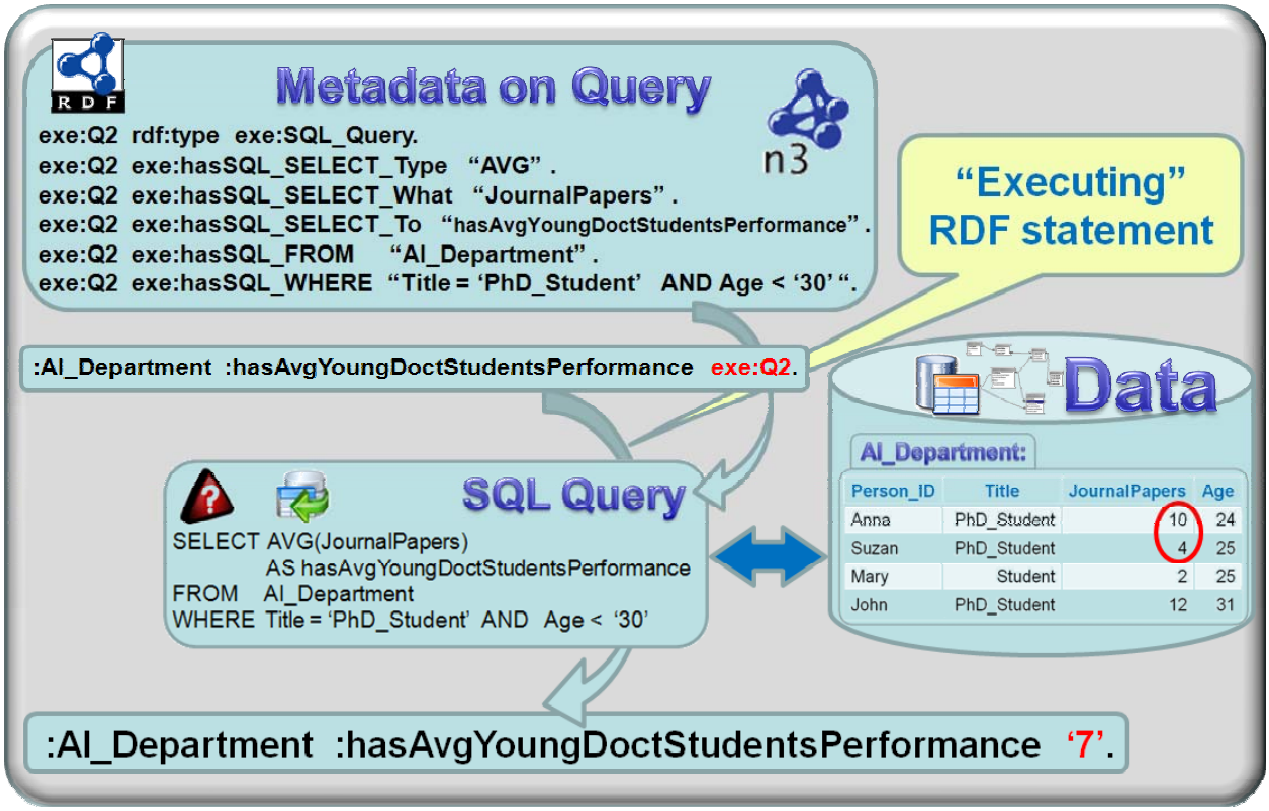


Figure 7. Example of processing executable RDF statement, which contains implicit value as SQL query to a relational database

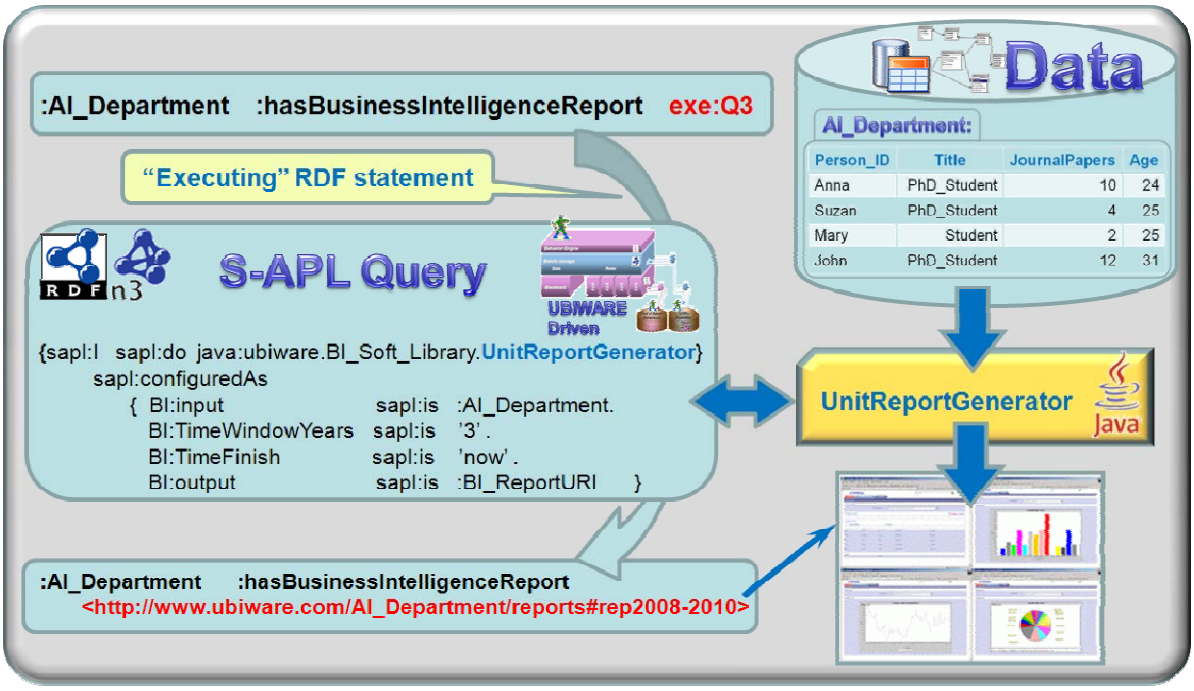


Figure 8. Example of processing executable RDF statement, which contains implicit value as S-APL query to BI software as a service

Consider the example in Figure 8. Here we have an executable RDF statement that can be interpreted like: "If you want to get basic BI-statistics report for the AI Department for the last 3 years, execute query Q3". Behind this query there is a Java software module "UnitReportGenerator" provided as-a-service from online software library. The query itself is written in S-APL (Semantic Agent Programming Language [17]) used for UBIWARE-based applications [18]. S-APL is a RDF-based language for multi agent systems, in which both data and actions are described semantically. UBIWARE [19] ("Smart Semantic Middleware for Ubiquitous Computing") has been developed by Industrial Ontologies Group (http://www.cs.jyu.fi/ai/OntoGroup). It is a software technology and a tool to support design and installation, autonomic operation and interoperability among complex, heterogeneous, dynamic and self-configurable distributed systems, and to provide a coordination, collaboration, interoperability, data and process integration service. UBIWARE platform is used actually to deal with "Executable Knowledge" and its utilization for "Executable Reality" services. In the example, when the BI software is executed, it generates the html page where all analytics are visually presented with different BI widgets. The URI for such page will replace the implicit "exe:Q3" value from the original RDF statement and creates traditional RDF statement with object property connecting two resources (AI Department URI and BI statistics Report URI).

There is also a possibility to compute semantic similarity between resources from different data storages and automatically create appropriate RDF connections for similar (same) instances. As it was shown in Section II, some instances can be considered as similar ones in one context and can be considered otherwise in another context. Therefore, similarly to the examples above, the "Executable Knowledge" supports also RDF statements with implicit similarity search queries, in which needed query parameters are automatically taken from the current context. Change of context can be considered as implicit query (if appropriate setup is made) to re-compute similarity links, which makes the RDF graph on top of Linked Data very dynamic. Our approach for context-sensitive semantic similarity computing and its implementation is discussed in [11].

Mixed Reality is just one possible way to utilize Executable Knowledge concept. There should be definitely other application areas for it. Generally many industrial applications, which require dynamic self-configurable solutions, applications and architectures, will benefit from the flexible Executable Knowledge, as our experience with UBIWARE industrial cases demonstrates [19].

V. Related Work

Concepts of "virtual", "augmented", "mixed", etc., realities discussed in Section III are being actively developed into various services for the public. There are many other relevant concepts and activities, which have many common features with the above, having however some specifics. One such abstraction is so-called "Mirror World" [20], which is a representation of the real world in digital form mapped in a geographically accurate way. Mirror worlds can be seen as an autonomous manifestation of digitalized reality including virtual elements. Another relevant concept is "Metaverse" (http://metaverseroadmap.org), which is the convergence of virtually-enhanced physical reality and physically persistent

virtual space being a fusion of both. The "Second Life" (http://secondlife.com/) is a 3D virtual world enhanced by social networks and communication capabilities. "Lifelogging" [21] is continuous capturing from a human and sharing through the Web various data, events and activities collected by various devices, sensors, cameras, etc. Other slightly different concept is "Lifeblog" [http://europe.nokia.com/support/product-support/nokia-photos], which is also known as a popular service for collecting and putting into a timeline (mobile) user activities and creating data in the form of complex multimedia diary.

Our intention was to find out reasonable services out of these concepts suitable not just for public use but mostly for professionals. We explored the possibility to utilize BI as an additional capability for that purpose. Preliminary information on the interesting relevant effort named "Augmented BI" has appeared in the Web [22] quite recently. Augmented BI is considered in [22] as a process of using a mobile device to scan an image or a barcode and overlaying metrics and or charts onto the image. This supposes to facilitate the process of a store manager moving around a retail store, who would like to get more information about certain products' sales performance. See Figure 9, which demonstrates a possible use case for the Augmented BI. There are some evident similarities with our use-cases from Figure 5, however our implementation benefits from the Linked Data utilization and allows context-sensitive view to the BI-enhanced reality.

Figure 9. Demonstration of possible Augmented BI usage scenario [22].

Our solution related to BI-enhanced mixed reality (or Executable Reality) is based on the Executable Knowledge concept. The Executable Knowledge inherits some features from a Dynamic Knowledge (see, e.g., [23]), which is actually dynamically changing knowledge and according to (www.imaginatik.com) providing on-demand, in-context, timely, and relevant information. Issues related to such knowledge include power and expressive tools and languages (such as, e.g., LUPS [24]) for representing such knowledge and proper handling of conflicting updates as addressed in [23]. Given an initial knowledge base (as a logic program) LUPS will sequentially update it.

Since executable knowledge is definitely a kind of dynamic knowledge, other issue would be whether it is declarative or procedural knowledge. A procedural knowledge (or knowledge on how to do something) is known to be a knowledge focused on obtaining a result and exercised in the accomplishment of a task, unlike declarative knowledge (propositional knowledge or knowledge about something) [25]. Procedural knowledge is usually represented as finite-state machine, computer program or a plan. It is often a tacit knowledge, which means that it is difficult to verbalize it and transfer to another person or an agent. The opposite of tacit knowledge is explicit knowledge.

The concept of executable knowledge can be actually considered as a kind of hybrid of declarative and procedural knowledge. For similar purpose, Marvin Minsky in [37] suggested to use so called "demons" within frame models already in 1975. Demons are supposed to be attached to some slots in a frame to cause execution of some procedure when accessed. Since that, however, demons have never been supported by the RDF data model. As it can be seen from the examples in Section IV, by "executing" knowledge, one actually transforms tacit (procedural) knowledge into explicit (declarative one). Therefore an executable knowledge contains explicit procedural (meta-) knowledge on *how to acquire* (or compute) declarative knowledge. Such capability means that the executable knowledge is naturally self-configurable knowledge (or more generally – self-managed knowledge). We use S-APL (Semantic Agent Programming Language [17]) for its representation, which is based on RDF (N3) syntax and which is equally suitable to manage declarative and procedural knowledge.

Our implementation of the executable knowledge on top of UBIWARE [18][19] agent-driven platform allows UBIWARE agents autonomously "execute" knowledge by following explicit procedural instructions for BI services execution and therefore updating (or making explicit) appropriate declarative beliefs.

The basic architecture of UBIWARE as a cloud-based platform is shown in Figure 10. It is supposed that any user of UBIWARE will be able to design and upload her own "application" via friendly and simple Web interface and this application can be executed and run continuously at the UBIWARE cloud. An application supposed to be designed in accordance with the SOA principles and it looks like semantic specification (in S-APL) of needed components (capabilities and knowledge as-a-service) and semantic specification (in S-APL) of desired business logic to connect these components. The components needed for the application can be internal (i.e., available in the cloud, semantically annotated, searchable and executable internally) and external (Web services, databases, etc.), for automatic utilization of which special semantic interfaces (adapters) are needed. The layers of knowledge between the application and the components in the Figure 10 are actually playing role of such adapters. These knowledge layers are organized as executable knowledge, which means that whenever a running application needs to refer to some internal or external component it simply sends semantic query to the executable knowledge and then the actual query (formal information or service request) will go from the executable knowledge to intended components discovered on-the-fly.

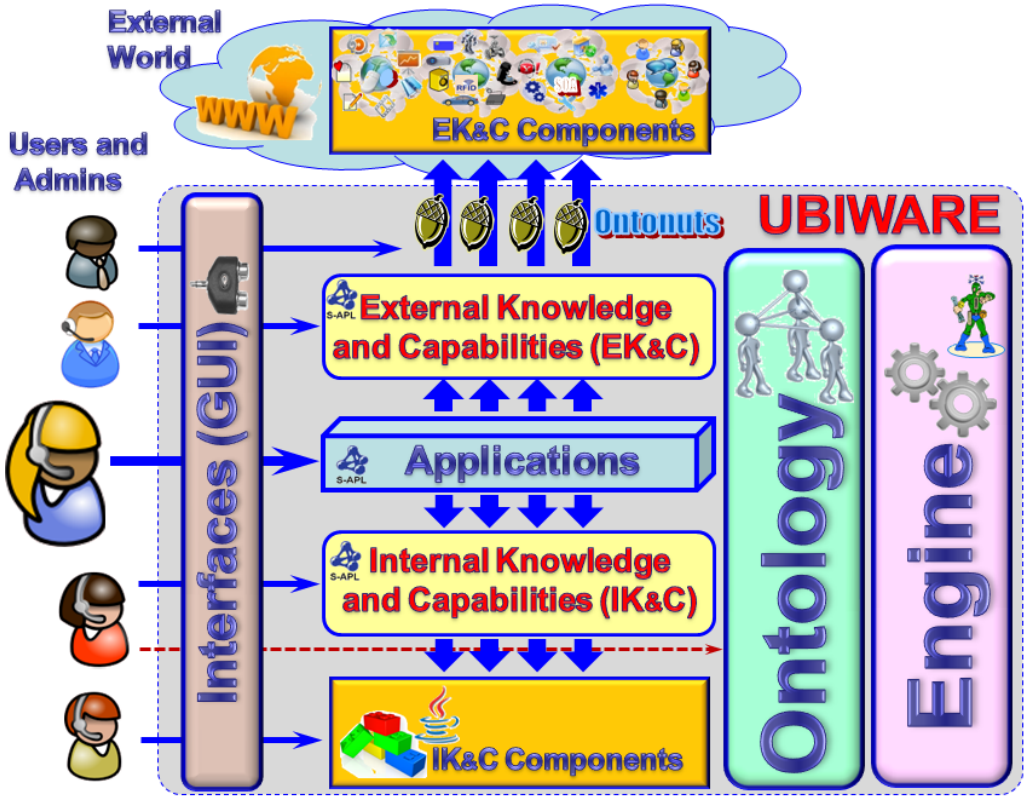


Figure 10. Architecture of the UBIWARE platform

The process of executing an application with discovered on-the-fly components is supported by special UBIWARE ontology and special agent-driven engine working as "Knowledge Processor" for knowledge execution.

Some external components, which are not software-as-a-service, may require special additional semantic interfaces to their APIs to be used automatically. In UBIWARE such interfaces (semantic software components) are named as *Ontonuts* and they are capable to facilitate the presentation of modular scripts and plans related to the external world utilization within the UBIWARE platform [36].

VI. Executable Knowledge in Action: Quality Assurance Case

One of the ongoing activities, which is actively utilizing the Executable Knowledge concept, is the EU Tempus-IV Project TRUST: "Towards Trust in Quality Assurance Systems" (516935-TEMPUS-1-2011) [31][32][33][34][35]. The overall goal of the TRUST project is to support the reforms of Ukrainian Higher Education (HE) by introducing a common, comprehensive and transparent Quality Assurance (QA) framework for all HE institutions (HEI) and QA organizations. The framework is based on the knowledge triangle ("education-research-innovation") and is open to stakeholders. An ecosystem of solutions is developed that enables, supports and automates the QA activities and transactions between HEIs, different national and international QA actors, students and different stakeholders and supports various forms of information exchange and knowledge sharing. The framework is assumed to guarantee trust between all QA players and society by ensuring that all QA procedures will be based on credible, transparent and relevant sources of information and explainable decision-making techniques documented in a common portal. Impact of each dimension of the knowledge triangle will be taken into consideration and the most independent and therefore credible sources from each dimension will be included into the system of quality criteria. Education quality is proposed to be assessed by both EU students who have taken courses or got degrees in Ukrainian HEIs and return back to EU and Ukrainian graduates moving to work or to continue study to EU. Research can be evaluated by official sources of international scientific citation indices mediated by Web-services. EU companies are to be involved into evaluation of innovation potential of Ukrainian HEIs. A trusted QA system should be based as much as possible on external objective evaluations. However because it is difficult to immediately utilize expensive experience of external evaluators in Ukrainian QA system one may (as the first step) make the academic data, metadata, quality indicators and QA processes available and transparent to national and

international academic community and combine it with other publicly available information within an ecosystem based on a web portal which enables external assessment of the academic performance.

These objectives are supported by a flexible and powerful instrument – Portal [34][35][32], which is a work-in-progress, providing a set of solutions that (on the basis of Executable Knowledge) enable, support and automate the activities, information flows and transactions within the ecosystem of individuals, HEIs, and QA organizations. The core system is extended with mechanisms allowing consideration of flexible multidimensional and multicontextual quality indicators, which will reflect constantly variable contexts (caused by political, economical, etc., reasons) at the different user-dependent levels (international, European, national, local) in all aspects of HEIs processes. These enable each HEI or national QA organization to develop its own appropriate QA strategies, HEI evaluations, rankings, etc., and provide capabilities for self-proof of decisions. Provided IT-support of QA enables: machine-processable QA-related information; management of globally distributed and heterogeneous QA-related data collections and Web-services; QA-related automated knowledge transfer through intelligent information retrieval, extraction, sharing, reuse and integration. To achieve this, the knowledge needed for QA is organized according to the Executable Knowledge concept and it is augmented in several dimensions:

• To allow anybody to easily add her own QA technique (a "Quality Calculator") or evaluation criteria (i.e., executable properties as described in Section IV) to the knowledge base and to get a personalized view on the quality status (in absolute or relative scales) of any educational organization or any educational outcome. As a result such executable knowledge becomes in a way a "Smart Knowledge" (i.e., enable ranking, evaluation, etc., formulas, QA procedures and techniques to be proactive knowledge instances, to be self-descriptive, extendable, self-managed and reusable);

• To make the results more transparent and trustful such executable knowledge must also be a "Cross-Validated Knowledge" (i.e., providing Service-Oriented Architecture for the portal enabling automatic update of the values of various quality indicators by taking them from external Web-based sources (portals, databases, etc.), such as, e.g., ISI Web of Knowledge, Google Scholar, etc., externalizing and internationalizing various quality monitoring activities);

• To help the user see the reasons behind good or bad performance we need our knowledge to "behave" as a "Self-Explanatory Knowledge" (that provides automated support for detailed explanation of every calculated or inferred value of any quality indicator used in QA activities);

• To automate the interpretation of the values of various quality indicators in different situations we need such executable knowledge to perform also as a "Context-Aware Knowledge" (i.e., utilization of formalized knowledge about context (local, regional, national, international, etc., for providing more grounded evaluations in a particular context).

The multidimensional and multilayered formalized model of the executable knowledge (including smart, self-explanatory, cross-validated and context-aware knowledge) about QA domain (resources, parameters, values, activities, etc.) collected or linked during the TRUST project is called QA Ontology. It includes (a) core layer (the one, which specifies concepts and properties related to knowledge triangle: education, research and innovation and which supposed to be a required part for various quality evaluations); (b) customized layer (the one which every organization can flexibly adapt to a local context or every user can adapt to own preferences); (c) system of values (which defines weights for various quality indicators in various contexts); (d) QA processes (i.e., formally specified internal or cross-organizational processes to enable QA execution monitoring).

The essential components of the TRUST portal are presented in Figure 11. One can see that the information about educational resources (e.g., universities, departments, academic personnel, etc.) is automatically collected from remote but trusted sources of data and interlinked with the metadata layer based on co-reference resolution and according to the QA ontology. It is assumed that any time a user can query the quality evaluation of any educational resource on her choice or order some comparative evaluation (e.g., ranking list) of chosen set of resources. After the query is done, the user will be asked, according to which method ("Quality Calculator") she wants to get the evaluation. She will be offered the list of available calculators (different from each other by set of quality indicators and their relevant weights of importance, information about creators and context, in which it has been or has to be applied). If the choice is made, then the information about selected resource and its quality indicators will be queried from the remote sources by "executing" properties from the metadata layer. After that the returned data will be normalized, weighted and computed in accordance with the chosen quality calculator, and finally the user will get the required and personalized evaluation report.

The more interesting case would be if the user is not satisfied with any of the available quality calculators and wants to create the new own one to be used for quality calculations in that particular case and also in the future. The Portal allows the user to design (through the Web interface) her own quality calculator and therefore contribute to the executable knowledge creation. There are two options here depending on the user experience:

(1) The new quality calculator is based on quality indicators (executable properties, similar to ones described in Figures 6-8) already supported by the portal, i.e., there is some external source of data with the interface to it from the portal, which can be queried to get the value for each indicator. In this case the user only specifies her preferences on importance of each indicator for final quality calculation;

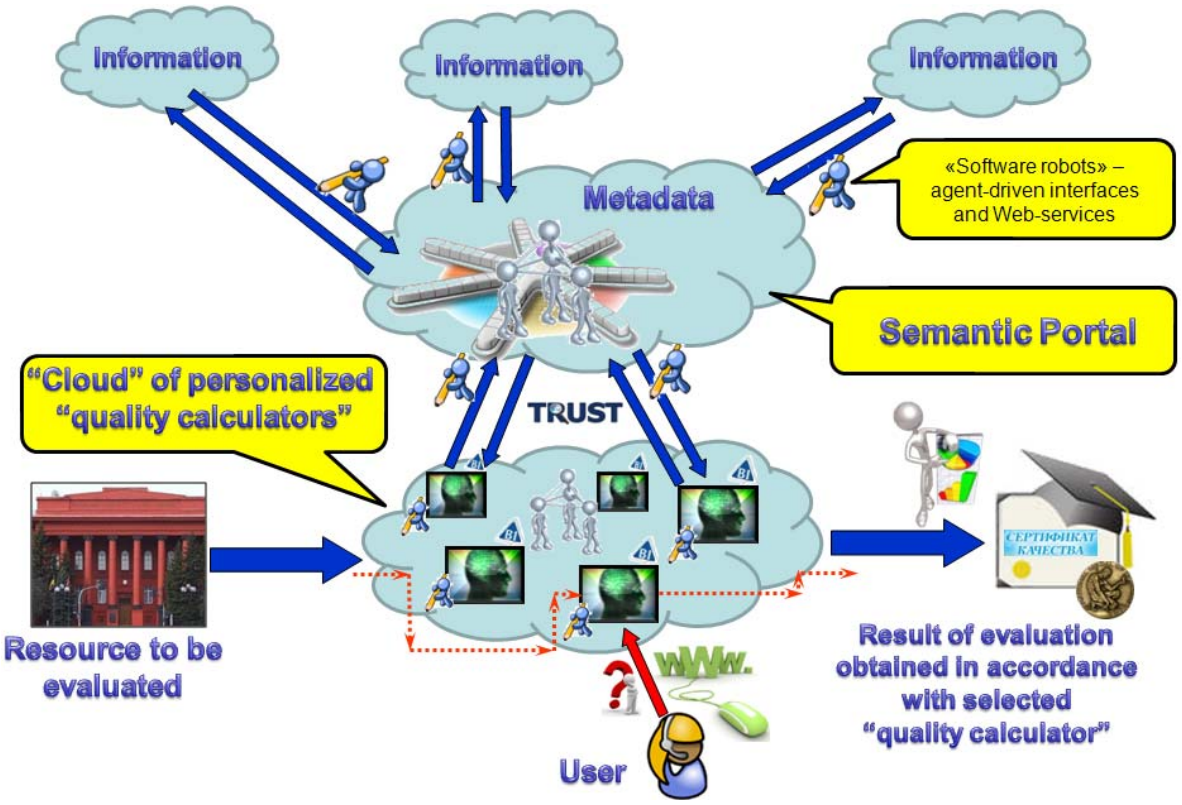


Figure 11. Executable-Knowledge-Based Architecture of the TRUST portal for quality assurance and personalized quality evaluation (a user is capable to choose or to create her own "Quality Calculator" to be applied for measuring quality of chosen resource) [32].

(2) The (advanced) user wants and is capable to add some new quality indicator (not supported by the portal yet), which means creating a new executable property and providing interface (an Ontonut) to the new source of data. Only after all the necessary and new quality indicators will be specified and remote information sources for getting values for these indicators will be available, the user may design the quality calculator itself like in the case (1).

VII. Conclusion and Future Work

In this paper, we presented the concept of Executable Knowledge, which is based on Linked Data and in addition to traditional subject-predicate-object semantic triplet model it contains also subject-predicate-query triplets (Executable Properties). We have demonstrated that data heterogeneity problem in distributed systems can be handled by the executable knowledge, which semantic (RDF) links include explicit queries to data or to (BI) services and other capabilities based on various data models and the context.

We have shown one way (named Executable Reality) on how Linked Data can be automatically processed by various BI services; and also how the results of BI processing can be requested, delivered and presented to the user through similar to the (Mobile) Mixed Reality technology interfaces.

Other executable knowledge benefits have been shown in the context of educational quality assurance and related to online quality evaluation and ranking of various academic resources. It is shown how the special Quality Assurance Portal enables executable knowledge and allows a user not only to evaluate particular academic resource based on Linked Data from external information sources, but also create her own "Quality Calculator", according to which personalized evaluations or rankings will be computed.

In the near future we are going to extend the current solutions and fully implement a powerful domain independent tool to build and execute systems on the basis of executable knowledge and to investigate new domains where such knowledge will provide an evident added value.

This paper is an extended version of conference paper [1] accepted for journal publication.

Acknowledgements

We are grateful to our international teams from UBIWARE and TRUST projects for fruitful collaboration in research and software developments; also to Tekes (Finnish Funding Agency for Industry-Driven Research) for the UBIWARE project support during 2007-2010; and to Tempus Program of the European Commission for the funding provided for the TRUST project starting from 2011.

REFERENCES

[1] Terziyan, V., and Kaykova, O., Towards "Executable Reality": Business Intelligence on Top of Linked Data, In: Proceedings of the First International Conference on Business Intelligence and Technology, September 25-30, 2011, Rome, Italy, pp. 26-33.

[2] Berners-Lee, T., Hendler, J., and Lassila, O., The Semantic Web. Scientific American, 284(5), 2001, pp. 34–43.

[3] Sheth, A. and Ramakrishnan, C., Semantic (Web) Technology in Action: Ontology Driven Information Systems for Search, Integration and Analysis, IEEE Data Eng. Bulletin, 26(4), 2003, pp. 40-48.

[4] Hitzler, P., Krötzsch, M. and Rudolph, S., Foundations of Semantic Web Technologies, Chapman&Hall/CRC, 2009, 455 pp.

[5] Nelson, G., Business Intelligence 2.0: Are we there yet? In: Proceedings of the SAS Global Forum 2010, Seattle, USA, 11-14 April, 2011, paper 040-2010.

[6] Domingue, J., Fensel, D., and González-Cabero, R., SOA4All, Enabling the SOA Revolution on a World Wide Scale, In: Proceedings of the 2nd IEEE International Conference on Semantic Computing (ICSC 2008), August 4-7, 2008, Santa Clara, California, USA, IEEE CS Press, 2008, pp. 530-537.

[7] Sell, D., Cabral, L., Motta, E., Domingue, J. and Pacheco, R., Adding Semantics to Business Intelligence, In: Proceedings of 16th International Workshop on Database and Expert Systems Applications, Copenhagen, 26 August, 2005, pp. 543 – 547.

[8] Heath, T., and Bizer, C., Linked Data: Evolving the Web into a Global Data Space. Synthesis Lectures on the Semantic Web: Theory and Technology, Morgan & Claypool, 2011, 136 pp.

[9] Khriyenko, O., and Terziyan, V., A Framework for Context-Sensitive Metadata Description, International Journal of Metadata, Semantics and Ontologies, 1(2), 2006, Inderscience Publishers, pp. 154-164.

[10] Terziyan, V., Predictive and Contextual Feature Separation for Bayesian Metanetworks, In: B. Apolloni et al. (Eds.), Proceedings of KES-2007 / WIRN-2007, Vietri sul Mare, Italy, September 12-14, Vol. III, Springer, LNAI 4694, 2007, pp. 634–644.

[11] Khriyenko, O., Terziyan, V., Similarity/Closeness-Based Resource Browser, In: J.J. Zhang (Ed.), Proceedings of the Ninth International Conference on Visualization, Imaging and Image Processing (VIIP-2009), July 13-15, 2009, Cambridge, UK, pp. 184-191.

[12] Azuma, R., A Survey of Augmented Reality, Presence: Teleoperators and Virtual Environments 6 (4), 1997, MIT Press, pp. 355-385.

[13] Ohta, Y., Tamura, H. (Eds.), Mixed Reality: Merging Real and Virtual Worlds, 1999, Springer, 418 pp.

[14] Hollerer, T., Feiner, S., Terauchi, T., Rashid, G., Hallaway, D., Exploring MARS: Developing Indoor and Outdoor User Interfaces to a Mobile Augmented Reality System, Computers & Graphics, 23, 1999, Elsevier, pp. 779-785.

[15] Henrysson, A., Ollila, M., UMAR: Ubiquitous Mobile Augmented Reality, In: Proceedings of the Third International Conference on Mobile and Ubiquitous Multimedia (MUM-2004), College Park, Maryland, USA, 2004, pp. 41-45.

[16] Mobile Mixed Reality: The Vision, Nokia Technology Insights Series, Nokia Research Center, June 2009, 4 pp., Available online in: http://research.nokia.com/files/insight/NTI_MARA_-_June_2009.pdf.

[17] Katasonov, A. and Terziyan, V., SmartResource Platform and Semantic Agent Programming Language (S-APL), In: P. Petta et al. (Eds.), Proceedings of the 5-th German Conference on Multi-Agent System Technologies (MATES'07), 24-26 September, 2007, Leipzig, Germany, Springer, LNAI 4687 pp. 25-36.

[18] Katasonov, A., Terziyan, V., Implementing Agent-Based Middleware for the Semantic Web, In: Proceedings of the Second IEEE International Conference on Semantic Computing (ICSC-2008) / International Workshop on Middleware for the Semantic Web, August 4-7, 2008, Santa Clara, USA, IEEE CS Press, pp. 504-511.

[19] UBIWARE: http://www.cs.jyu.fi/ai/OntoGroup/UBIWARE.htm, Project Web Site, Industrial Ontologies Group, 2008-2011.

[20] Gelernter, D., Mirror Worlds: or the Day Software Puts the Universe in a Shoebox...How It Will Happen and What It Will Mean, 1st ed., Oxford University Press, 1992.

[21] O'Hara, M., Tuffield, M., Shadbolt, N., Lifelogging: Privacy and Empowerment with Memories for Life, In: Identity in the Information Society, Vol. 1, No.1, Springer, pp. 155-172.

[22] Husting, P., Augmented Business Intelligence in Retail, In: Microsoft BI Collaboration and Community Blog, 3 March 2011, Available online in: http://blog.extendedresults.com/2011/03/03/augmented-business-intelligence-in-retail/ (accessed 15 June 2011).

[23] Alferes, J., Pereira, L., Przymusinska, H., Przymusinski, T., Quaresma, P., Dynamic Knowledge Representation and its Applications, In: Proceedings of the 9th International Conference on Artificial Intelligence: Methodology, Systems, and Applications (AIMSA '00), Varna, Bulgaria, September 20-23, 2000, LNCS, Vol. 1904, Springer, pp. 1-10.

[24] Alferes, J., Pereira, L., Przymusinska, H., Przymusinski, T., LUPS - A Language for Updating Logic Programs, In: Proceedings of the LPNMR'99, El Paso, Texas USA, December 2-4, 1999, LNAI, Vol. 1730, Springer, pp. 162-176.

[25] Berge, T., Hezewijk, R., Procedural and Declarative Knowledge. An Evolutionary Perspective, Theory & Psychology, 1999, Sage Publications, Vol. 9(5), pp. 605–624.

[26] Jaffri, A., Glaser, H., and Millard, I., URI Disambiguation in the Context of Linked Data, In: Proceedings of the LDOW-2008 Workshop: Linked Data on the Web, April 22, 2008, Beijing, China.

[27] Glaser, H., Jaffri, A., and Millard, I., Managing Co-reference on the Semantic Web, In: Proceedings of the LDOW-2009 Workshop: Linked Data on the Web, April 20, 2009, Madrid, Spain.

[28] Halpin. H., Hayes, P., McCusker, J., McGuinness, D., and Thompson, H., When owl:sameAs Isn't the Same: An Analysis of Identity in Linked Data, In: Proceedings of the 9th International Semantic Web Conference (ISWC 2010), Shanghai, China, November 7-11, 2010, Springer, LNCS, Vol. 6496/2010, pp. 305-320.

[29] Jaffri, A., Glaser, H., and Millard, I., URI Identity Management for Semantic Web Data Integration and Linkage, In: Proceedings of the Workshop on Scalable Semantic Web Systems, 2007, Vilamoura, Portugal, Springer.

[30] Bouquet, P., Stoermer, H., Niederee, C., and Mana, A., Entity Name System: The Backbone of an Open and Scalable Web of Data, In: Proceedings of the IEEE International Conference on Semantic Computing (ICSC 2008), IEEE CS Press, August 2008, pp. 554-561.

[31] Tiihonen, T., How to Create Trust, In: Tempus Information Days Helsinki, Finland, November 22, 2011, Available online in: http://www.cimo.fi/instancedata/prime_product_julkaisu/cimo/embeds/cimowwwstructure/22799_TRUST_Tempus_Timo_Tiihonen.pdf.

[32] Terziyan, V., TRUST: Towards Trust in Quality Assurance Systems. Brief Introduction of the Project Idea, In: TRUST (516935-TEMPUS-1-2011) Project Coordination Meeting, Kiev, Ukraine, October 20, 2011, Available online in: http://www.cs.jyu.fi/ai/Quality-2.ppt .

[33] Web Site of TRUST Project "Towards Trust in Quality Assurance Systems" (516935-TEMPUS-1-2011), 2011, Available Online in: http://www.dovira.eu.

[34] Klymova, M., Ontology-Based Portal for National Educational and Scientific Resources Management, In: Universities Nationwide IT Days (IT-2007), Jyvaskyla, Finland, November 1, 2007, Available online in: http://www.cs.jyu.fi/ai/OntoPortal-2007.ppt.

[35] Web Site of the Ukrainian National Ontology-Based Portal for Management of Educational and Scientific Resources, 2009-2011, Available online in: http://ailab.kture.kharkov.ua/site/index.html.

[36] Nikitin, S., Katasonov, A., and Terziyan, V., Ontonuts: Reusable Semantic Components for Multi-Agent Systems, In: R. Calinescu et al. (Eds.), Proceedings of the ICAS 2009, April 21-25, 2009, Valencia, Spain, IEEE CS Press, pp. 200-207.

[37] Minsky, M., A Framework for Representing Knowledge, In: P. Winston (ed.), The Psychology of Comp. Vision, McGraw-Hill, 1975.

How to Switch IT Service Providers: Recommendations for a Successful Transition

Matthias Olzmann
Business Solutions
noventum consulting
Muenster, Germany
e-mail: Matthias.olzmann@noventum.de

Martin Wynn
School of Computing & Technology
University of Gloucestershire
Cheltenham, UK
e-mail: MWynn@glos.ac.uk

***Abstract*—Although IT outsourcing is a growing industry and a common topic in the literature, there is limited research which critically analyses and assesses the switching of IT outsourcing providers – in particular the factors contributing to success are under-researched. This article explores this growing area of management and consultancy activity by analyzing the existing literature in the field. This allows the identification of critical success factors that are pertinent to the switching of providers and provides recommendations for a successful transition.**

Keywords - service providers; outsourcing; IT outsourcing; ITO; switching providers; changing providers; transition; exit management; critical success factors; checklist for success.

I. INTRODUCTION

When companies outsource their IT for the first time, it can be assumed that the majority of IT experts will transfer from the client company to the IT outsourcing (ITO) provider. Together with the IT experts, the client specific knowledge is transitioned to the provider. This reduces the negative performance impact. In contrast, when providers are switched, it cannot be anticipated that the majority of IT experts (together with the client specific knowledge) will transition from the incumbent provider to the new provider [1].

It can be assumed that the leaving provider has only marginal interest in actively supporting the incoming provider, for example with knowledge transition. This results in major challenges for the tripartite relationship (client, incumbent provider, new provider).

A main building block in switching ITO providers is the transition. Transition is a complex, risky, and challenging building block of strategic importance which begins after the contract is signed and ends with service delivery. Two thirds of all issues can be tracked to the transition [2, 3]. Despite growing interest in topics such as sourcing the IT back in-house or switching providers [4-6], no studies have holistically focused on how successful ITO transitions are performed for clients switching service providers.

The factors contributing to a successful transition from the incumbent provider to the new provider are not fully understood. Yet understanding the factors contributing to a successful transition is vitally important. For the client, these factors determine on the one hand the success or the failure of the whole outsourcing endeavour; and on the other hand, ultimately the survival of the overall business, as it is linked to the successful switch of the ITO providers. This is exemplified by the following quotes from three ITO researchers:

1. "To our knowledge, no work has suggested strategies that managers should employ during the process of transitioning from one vendor to another" [7].
2. "However, all of this extant literature focuses on the decision to switch a vendor or include a new vendor in the supplier portfolio rather than manage the change-over. The implication is that the outsourcing literature provides little insight about managing the switching process from a long-lived prior vendor relationship to a new vendor relationship" [8].
3. "Relatively little work has focused on the area of switching vendors and bringing previously outsourced activities back in-house (backsourcing) (Lacity and Willcocks, 2000). Even less has been done specifically in the context of planning for the possibility of either of these two events" [9].

This article sets out to review available literature related to this topic and draws conclusions regarding critical success factors for achieving the switching of service providers. In the following section, a wide range of literature related to ITO is systematically reviewed. This leads to a discussion of critical success factors in section III, focusing on both the pre-delivery phase and the critical transition process. Section IV then makes some concluding remarks related to the analysis of existing literature, and highlights a conceptual framework for future work in this field.

This literature review and analysis will provide the basis for recommendations to guide practitioners involved in the switching of ITO providers, and also act as a platform for subsequent research in this field.

II. INITIAL LITERATURE REVIEW

"Outsourcing can be defined as turning over all or part of an organizational activity to an outside vendor" [10]. In contrast to other types of outsourcing, ITO affects the complete organisation – IT "is pervasive throughout the organization"[11]. Reference [5] suggests that in an ITO deal, the IT is either partly or fully turned over to "…one or more external service providers". As such, in the

context of this paper, the scope of ITO is likely to be more than just one of the possible elements depicted in Fig. 1.

A. ITO History and Market Development

Even though large scale modern ITO began in 1989 with the Kodak outsourcing deal [11, 12], some researchers argue that ITO "is still at the early stages of the profession itself" [12]. Kodak was not the first ITO deal in history although other deals had only received scarce attention. "It was not until Kathy Hudson, the Kodak CIO, announced to the world that Kodak had entered into a 'strategic alliance' with its IS partners, led by IBM but also including DEC and Businessland, did the world sit up and take notice" [11].

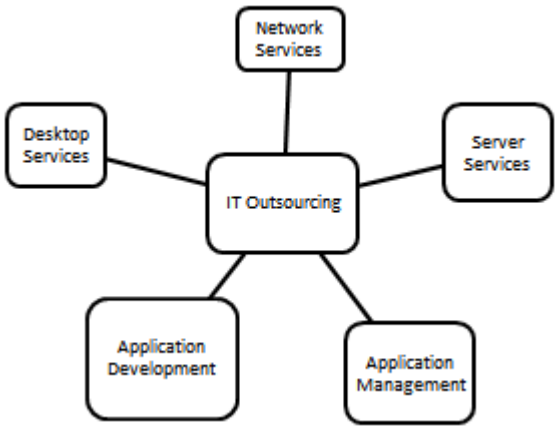


Figure 1. The scope of IT Outsourcing (ITO)

Many scholars and practitioners forecast further growth of the ITO market [12-14]. Reference [15] emphasizes that: "on conservative estimates, looking across a range of reports and studies, global ITO revenues probably exceeded $270 billion in 2010; it is very clear that, with its 20-year history, outsourcing of IT and business services is moving into becoming an almost routine part of management, representing in many major corporations and government agencies the greater percentage of their IT expenditure". All reports (Gartner, Everest, NASSCOM, and IDC) reviewed have indicated a global growth of ITO in the range of 5-8% per year [15].

B. Reasons for ITO

Research findings indicate that the main reasons for IT outsourcing are driven by the goal of cost reduction [16, 17], the focus on core capabilities and a desire to access resources of the provider such as superior capabilities, expertise and technology [10, 15].

The primary reason for outsourcing in 90% of the reviewed literature indicated the motivation of cost reduction [15]; but not all researchers agree that the goal of cost reduction and performance improvement will automatically be achieved - no matter how the outsourcing endeavour is managed. Reference [10] argues that "this overly optimistic view of outsourcing derives from the fact that most articles about outsourcing are written during the so called 'honeymoon' period i.e., just before or after the contract is signed". Hirschheim and Lacity [17] warn that cost reduction and service reduction frequently go hand in hand. Company executives often strive for cost cutting while company employees strive for a better service [17]. Outsourcing strategies therefore need to be deliberate to increase the companies' overall performance.

From the perspective of the ITO provider, *long-term revenue* is the primary reason to enter outsourcing arrangements. Reference [11] points out that "long-term outsourcing arrangements help stabilize vendor business volume and revenue, making planning more predictable, and increase shareholder's comfort levels".

The typical length of ITO contracts is generally 5-10 years and "thus, both client and vendor have come to expect that during the life of the contract, some form of renegotiations will be likely" [11]. The rapid growth and the complex nature of ITO have not been without impact. Recently a number of outsourcing deals have experienced both serious problems and the premature discontinuation of contracts [3-6, 11, 13, 18]. This leads companies to re-consider sourcing options and strategies. The discontinuation of contracts results in several strategic options. Regarding ITO contracts, "as much as 50%" of these are ended for other options such as switching the provider, or IT backsourcing" [5]. Other researchers have found that most clients stay with the incumbent provider [12, 13]. Reference [13] estimates that 25% of contracts will be awarded to new providers and merely 10 % will be back-sourced. Reference [4] notes the reasons for changing ITO providers as follows:

- "Dynamic changes in the customer landscape (e.g. the client organization may have outgrown the supplier)
- A shift in management's risk tolerance
- Changes in the supply market (e.g., emergence of new or specialized players)
- Supplier rationalization (e.g., consolidation to enhance bargaining power)".

C. Factors Influencing Sourcing Options

What factors influence sourcing option decisions when contracts are re-evaluated? Switching costs play a vital role in sourcing decisions – they are a good indicator for understanding and predicting clients' outsourcing decisions after re-evaluating sourcing options [19]. After the client has initially outsourced the IT and has transferred employees and capabilities to the provider, it is difficult to bring the services back in-house [20] .

"In sum, the literature defines operationalized switching costs in terms of economic (i.e., monetary) expenditures and intangible (i.e., psychological or relational) costs associated with changing an exchange relationship" [19]. Reference [5] argues that "the greater the information transfer/setup costs, the more likely that outsourcing continuation will be the strategic choice, vendor switching will be the intermediate choice, and backsourcing will be avoided". The researchers warn that

"high switching costs might entrap the customer organization into a 'no change situation', forcing it to continue outsourcing IT work to the same vendor". "Two factors amplify these latent risks. First, when firms outsource processes that require the transfer of a large amount of tacit knowledge, they have to invest time and effort in training providers' employees. Second, some processes take a long time to stabilize when companies offshore them. In both cases, the cost of switching from existing providers is very high. That accentuates the risk that over time, vendors will dictate terms to buyers" [20]. Although customer entrapment has been noted - not much has been written in the academic literature about how to avoid or adequately address it.

In contrast to high switching costs, if companies anticipate low switching costs and the option to choose from many vendors, there is "no real advantage in recontracting with the same vendor" [10]. Despite the significance of switching costs, the measurement of these costs remains unclear [19].

A study analyzing the influencing factors of sourcing options found that firms which decided to switch providers or to backsource typically experienced high service quality and low relationship quality [6]. They acknowledged that "relationship quality plays on important role in the decision to switch vendors. Of our three groups, those that switched vendors had the lowest perception of trust, commitment, culture, and communication in relation to their vendors…hence, the building of trust between an outsourcer and a firm is far more a socio-emotional condition than it is a matter of providing excellent product and/or service" [6].

The importance of relationship for staying with the current provider has been highlighted in a previous study [10], where the researchers found a high interest in staying with the same provider if relationship specific investments have been made. Reference [21] concludes that when there is low trust in the capabilities of the provider to manage the outsourcing deal and the relationship qualities are also low that this brings the client to consider backsourcing or switching providers. The risk of losing knowledge and the potential service operation distortions prevents companies from switching ITO providers [8]. Reference [8] argues that the "switching of IT vendors is seen to impose too much short-term operational risk to justify the financial savings and quality improvements that could accrue from a relationship with a new vendor".

D. ITO Success

There are contrasting conclusions on the contributing success factors for ITO success [22]. It is not clear if this is due to the lack of a generally accepted construct of a success definition or because "ITO success is so idiosyncratic that one must assess it against each organization's own, different criteria" [22]. Reference [11], in a widely cited (more than 500 times according to Google scholar) literature survey and analysis, notes that "outsourcing success is usually viewed as the attainment of economic, technological or business-related benefits. Satisfaction with the benefits attained is often used as an indicator of outsourcing success". Reference [23] found in their literature review on critical success factors that the research is typically divided between research on the success or "on the failure of economic activity".

Companies outsource their IT for different reasons, as previously noted. For example one company outsources to gain access to superior IT capabilities, another to focus on core competences, and another to reduce costs. This means that outsourcing success is dependent on the overall context. Thus, it is plausible that "any attempt to assess ITO success in terms of more detailed criteria, such as cost savings or focusing on core business, requires identification of the different criteria relevant to each organization for each different contract at the time of the study" [22].

Therefore it appears to be important to define factors contributing to outsourcing success before the contract is signed [24]. Reference [22] argues that success should be assessed by:

1. Defining most important outcomes before they actually materialise during the lifecycle of the contract
2. Measuring the extent to which the outcomes have been achieved.

Can outsourcing be considered as a standardised activity of everyday management with readily defined solutions? Reference [15] disputes this and concludes that "our review of 20 years of research establishes the common denominator that, for management and operational staff, outsourcing is far from easy". Reference [25] found that even skilled organizations don't work in a proactive mode and are hurt by slow organizational learning. Therefore, in order to reduce learning curves, it is important to understand how success can be defined and what the contributing factors are. Reference [24] suggests a more abstract description of success factor such as:

- "Use 'best outsourcing practices' as major references for corporate outsourcing decision.
- Clearly understand the goals, objective, scope, budget, and the duration of IS outsourcing project….
- Select a reputable vendor and then communicate well on the corporate outsourcing plan.
- Realize the legal issues related to contract negotiations and signing.
- Communicate well with employees and stakeholders about the outsourcing plan; this may reduce the severity of resistance."

Even though these factors are useful to get an overview about common success factors, they are of limited applicability for the specific issue of switching ITO providers. A review of 191 ITO articles relevant to practice from the early 1990s until 2009 found that "the three major categories of determinants of ITO success are *ITO decisions, contractual governance, and relational governance*. These determinants are depicted as direct relationships to ITO success" [26] in Fig. 2.

Although organizational capabilities are also important as a success contributing factor, they are neither depicted in Fig. 2 nor are they described in the section about the determinants of success. Reference [26] recognises that "the most widely cited papers on this topic identify a mix of complementary capabilities that lead to ITO success". Reference [27] develops this further into a list of nine pertinent organizational capabilities shown in Table I.

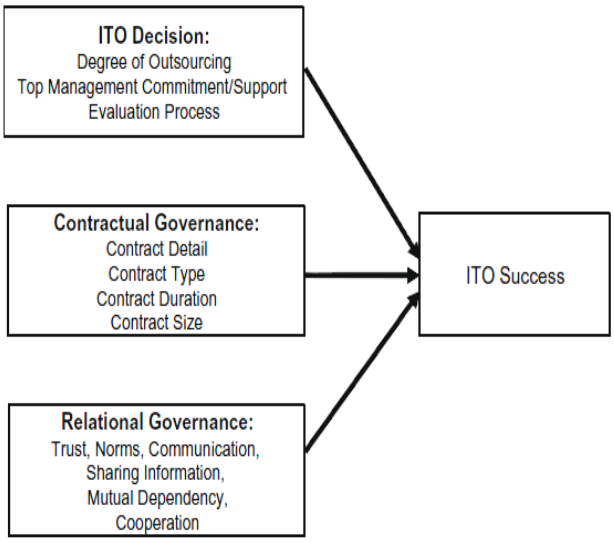


Figure 2. Three main categories of determinants of ITO success [26]

Reference [27] summarises research findings thus: "overall, we know *ITO decisions* that entailed selective use of outsourcing, the involvement of senior managers, and rigorous evaluation processes, were associated with higher levels of ITO success. *Contractual governance* also positively affected ITO success. In general, more contract detail, shorter-term contracts, and higher-dollar valued contracts were positively related to outsourcing success…. *Relational governance* positively affected ITO outcomes. Trust, norms, open communication, open sharing of information, mutual dependency and cooperation were always associated with higher levels of ITO success". The researchers found that top management commitment/support is the most critical success factor [26]. That trust plays a vital role in the success of ITOs is emphasized by Reference [11]. Reference [11] adds that "Sabherwal also suggests that a 'psychological contract' exists in outsourcing relationships. This contract, which consists of unwritten and often unspoken expectations, is supported by the level of trust between the parties, and plays a role in resolving unanticipated problems or changes in the accomplishment of outsourced activities".

Based on these findings, it seems clear that trust and the management of relationships between the client and the outsourcing provider are important factors contributing to success. However, given that significant amounts of capital are often invested in outsourcing deals, clients should probably not solely rely on relational governance factors such as trust and relationship. Reference [10] endorses this view in asserting that it is not advisable to completely rely on partnership factors and neglect contract negotiation – "a good contract is essential to outsourcing success because the
contract helps establish a balance of power between the client and the vendor".

Understanding the budget is of critical importance [24]. Reference [10] proposed the hiring of external experts as they know the hazards of outsourcing and how they can be managed. They argue that the additional costs may be justified in relation to the potential impact of the hidden costs. Other researchers found that "managing costs is less

TABLE I. ORGANISATIONAL CAPABILITIES RELEVANT TO ITO SUCCESS [27]

	Capability		Capability
1	IS/IT leadership	6	Informed buying
2	Business systems thinking	7	Contract facilitation
3	Relationship building	8	Contract monitoring
4	Architecture planning	9	Vendor development
5	Making technology work		

important than managing portfolio configuration, complexity and risk" [25]. This implies the importance of actively managing the outsourcing provider. Reference [10] emphasis this notion: "When an activity is outsourced, it is crucial to retain a small group of managers to handle the vendor. These managers must be able to develop the strategy of the outsourced activity and keep it in alignment with the overall corporate strategy."

Success itself can be considered an important factor contributing to success. "Specifically, ITO success fuelled higher levels of trust (relational governance, built stronger client and supplier capabilities, and determined the kinds of ITO decisions and ITO contracts clients made moving forward"[26] Reference [26] concludes that: "Conversely, ITO failure fuelled greater need for controls, monitoring mechanisms, tougher contracts, and determined the kinds of ITO decisions clients made".

It is advisable to view success factors in specific contexts. Reference [22] observes that: "For these reasons, the wide range of success advice and prescriptions appearing throughout the literature must be viewed as highly conditional – not only in terms of the success constructs the author/s have adopted, but also in terms of the contextual situation of each of the organisation." This indicates for example that an organization which has outsourced its IT services in just one country will need a different transition strategy than an organization which has outsourced its IT services in 5 countries.

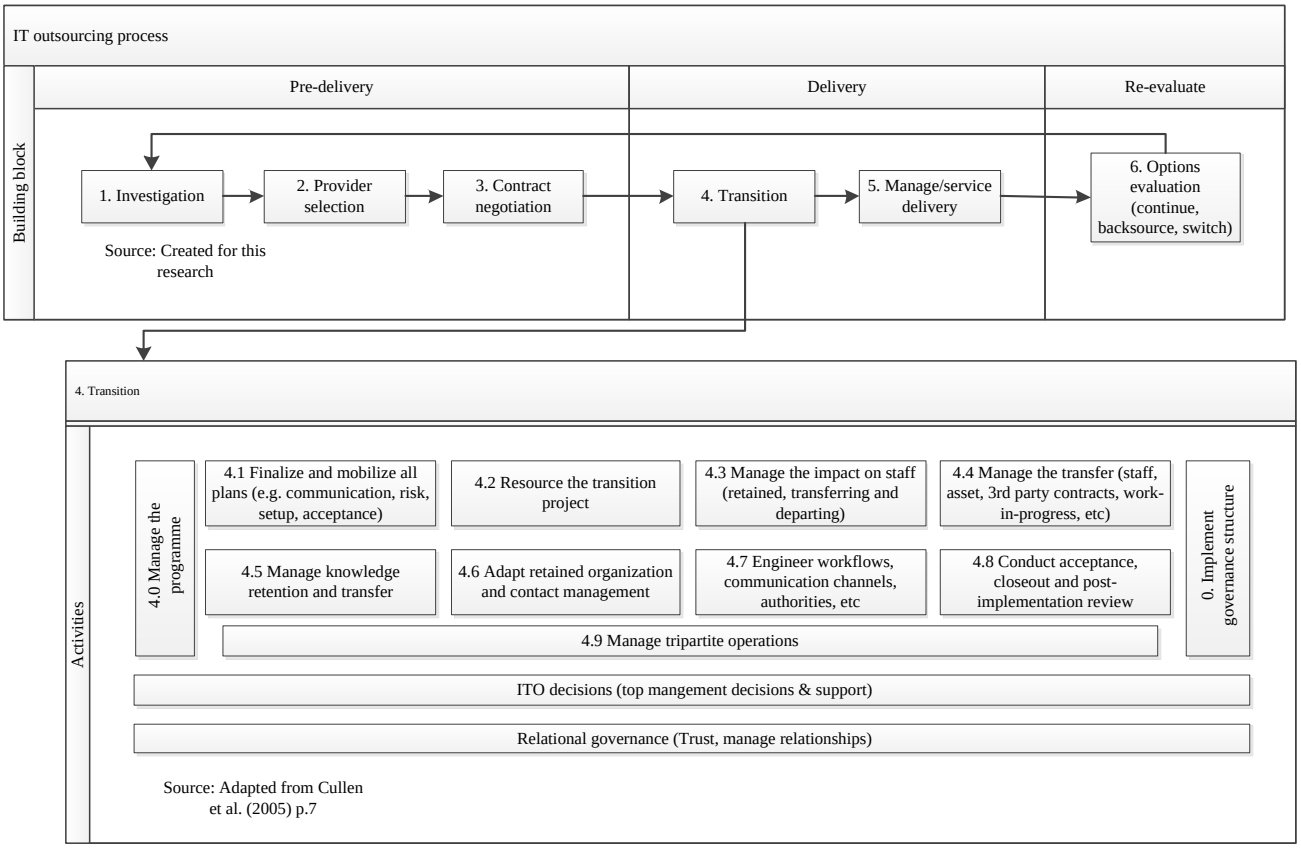


Figure 3. Conceptual framework - Switching providers with the focus on transition

E. ITO Methodologies

Reference [25] defines a detailed process model using nine building blocks with 54 activities. This model describes the complete ITO process lifecycle and appears to be the most comprehensive in the academic literature. Many ITO process models distinguish between activities before signing the contract (pre-delivery) and after signing the contract (delivery & re-evaluate) [4, 8, 25, 28]. The ITO process model for this research is depicted in Fig. 3. The six major building blocks are: investigation, provider selection, contract negotiation, transition, manage/service delivery, and options evaluation. The first three building blocks can be considered as pre-delivery phase, the next two can be considered as delivery-phase, and the last activity can be considered as the re-evaluation phase.

Transition "sets the tone for the entire relationship and involves handover of outsourced services from either the client's internal IT department or the incumbent service provider" [2]. Transition can be summarized as the seminal milestone for the successful implementation of an outsourcing contract [2]. Reference [29] defines the transition stage as "implementing the new way of operating" and states that it is the goal of transition to ensure that the new way of working is realized.

Transition includes the following activities: "conducting knowledge transfer, determining and implementing new governance structures, and applying the processes of the service provider" [2]. This demonstrates that many actions need to take place during transition before an outsourcing project can be actually implemented [24]. "The parties should have a clear understanding, typically set out in a detailed transition plan, as to how operations, assets, and employees will be transitioned to the vendor….The parties may want to consider including testing requirements in the agreement, as well as the operation of parallel operating environments for a specified period. In order to reduce customer dissatisfaction in the early phases of the outsourcing relationship, it is useful for the parties to have an understanding about the levels of service to be delivered to the customer during transition" [18].

Reference [25] has identified the main transitional activities as shown in Fig. 3, whilst reference [16] have defined the following 8 main activities during transition: "*Distribute the contract*", "*interpreting the contract*", "*establishing post contract management infrastructure and process*", "*implementing consolidation, rationalization, and standardization*", "*validating baseline service scope, costs, levels, and responsibilities*", "*managing additional service requests beyond baseline*", "*fostering realistic expectations of supplier performance*", and "*publicly promoting the IT contract*".

The cost for the transitional building block can take a significant portion of the overall costs [2]. It is assumed

that "over two-thirds of the problems in these unsuccessful engagements arise due to failed or poor transition" [2]. Due to the lack of statistical information regarding what percentage of switching ITO providers fail due to poor transition, it is assumed in this review that the percentage is at least as high as this.

III. Critical Issues in Switching Providers

When companies outsource their IT the first time it can be assumed that the majority of IT experts will transfer from the client company to ITO provider. Together with the IT experts, the client specific knowledge is transitioned to the provider. This reduces the negative performance impact. In contrast, when providers are switched it cannot be anticipated that the majority of IT experts (together with the client specific knowledge) will transition from the incumbent provider to the new provider. Reference [8] concludes that "a
long-term outsourcing relationship with a prior vendor means that much daily operational knowledge stays with the prior vendor. The client's knowledge loss exacerbates the problem of knowledge transfer as the client no longer possesses the information that the new vendor critically needs to service the client". The new provider requires close cooperation with the incumbent provider, who can pursue two different exit strategies. They can either actively co-operate with the new provider or "pursue a hostile strategy of being uncooperative" [7].

It can be assumed that the leaving provider has only marginal interest in actively supporting the incoming provider, for example with knowledge transition. This is particularly the case if the outgoing provider is not contractually obliged to support the incoming provider. This is confirmed by Reference [7] who find: "Being competitors, the transfer of resources between the outgoing (i.e., incumbent) and incoming (i.e., new) vendor presents a series of challenges not present in traditional outsourcing arrangements. Technologies, tools, business processes, intellectual properties and knowledge have to be transferred between vendors, not just between client and vendor. Pure monetary reward may encourage cooperation in traditional outsourcing; but in vendor transition, the outgoing vendor is reluctant to transfer assets to the incoming vendor. Such assets (e.g. source code) often provide the outgoing vendor with competitive advantage in other contracts." Reference [7] named source code as an example of this, but the findings apply to all client specific knowledge.

Intellectual property is already a complex topic in first generation outsourcing deals [30] and it becomes even more complex if the incumbent outsourcing provider is asked to transfer the intellectual property to its rival. In particular since IT outsourcing is based on sharing "business secrets" [30]. Managers often do not think about the termination of an outsourcing deal [10, 16] "…therefore, they often fail to plan an exit strategy…" [10] or draft only a contract which is too high-level for later execution [16].

With the risk of loss of knowledge comes the risk of degraded service quality. Reference [8] found that switching often leads to "temporary service disruptions of operations, lowered service levels and frustrations and dissatisfaction among the client employees". In addition this can lead to broken transition milestones, extended project duration and additional costs. Clients should take into consideration that once the contract of the incumbent provider has expired, the provider will leave regardless of whether the new provider is already prepared to deliver the service [7]. This can negatively impact service levels and even risk business continuity if the new provider is not completely ready. Alternatively, the client needs to be prepared to additionally pay the old provider for extending the contract until the new provider can adequately deliver the IT services.

"When contracts expire there is a need to have an exit strategy focusing not only on the economic success of the IT outsourcing, but also to question issues such as core competence management, access to resources, and the maturity of the relationship." [16]. Clients should make sure that the contract with the initial service provider contains a transition clause which regulates how and what the incumbent provider needs to transition to the new provider. Reference [30] suggests that: "The client may insist on having the right to purchase the assets and infrastructure that are being used to provide the services and employ the persons on the team that were providing such services". Reference [30] demands that: "The transition clauses also cover the effects of termination on various aspects such as payment of outstanding fees, escrow, IP and confidential information, and current work orders". Clients are well advised "to think exit" and plan accordingly right from the beginning even if this seems to be an unnecessary activity since the outsourcing deal has not been started yet [16].

When providers are switched transitional activities can be extensively resource draining for client, who needs to manage (monitor and correct) the operations of both the incumbent and new provider and additionally the transition between the two. Even relatively simple transitions where the IT can be transferred directly from the client to the outsourcing provider can be a costly phase and "in some cases, they (the transition activities) halved or even cancelled out the company's potential savings from outsourcing" [31]. It can be assumed that the transitional activities for switching providers are even more costly. As a general rule it can be stated that the more idiosyncratic the IT service to be outsourced, the more complex and costly the transition. Most clients are not able to calculate the transition costs [31].

If the perception is that ITO can be handled as a commodity, there is a risk that companies which have chosen to switch outsourcing providers underestimate the effort, complexities and risks involved. Reference [4] has disputed the common perception that "once part of a business process has been outsourced, it can, if necessary, easily be 'un-plugged' from one supplier and 're-plugged' into another".

A. Pre-delivery Phase – Factors Contributing to Switching Success

The client should ensure that the new potential ITO provider conducts an extensive *due diligence* review. "Before the service providers make a final offer during contractual negotiations, a thorough due diligence activity is required to closely understand the actual outsourced work and its related dependencies." [2]. Due diligence is even more important when providers are switched to ensure that the interdependencies between client and leaving provider are fully understood. Due diligence lays the baseline for the overall project management of the outsourcing transition, encompassing scope, time and quality definition. Reference [4] has noted the importance of identifying essential specific knowledge before the actual transition phase to avoid disruptions during transition. Reference [4] suggests that: "Alternatively, organizations should systematically ensure that new and changed process knowledge is acquired, transferred, and retained. Actions to achieve this include auditing the quality of documentation periodically, co-locating or seconding internal staff with the supplier, appointing internal 'knowledge owners' for specific subject matter, and even occasionally negotiating to recruit key supplier staff as internal employees."

Aron and Singh [20] recommends that the clients should plan to have sufficient expertise in-house so that the client is able to train the new provider. Alternatively the incumbent provider needs to train the new provider, which in the experience of Aron and Singh [20] is suboptimal since providers are often competitors. Although this is a good recommendation the client will often have not the resources and the expertise to train the new provider. As a rule of thumb it can be said that the bigger and the more complex the outsourcing deal the less likely it is that the client has sufficient in-house resources.

Identifying knowledge gaps before the transition is likely to be only partly successful. Reference [8] noted that "at the time of the contract negotiations, both parties (client & new provider) were still largely unaware of the gaps in the knowledge that would trouble the change-over from the prior provider to the new provider". Much of the operational knowledge is only visible to the people involved in everyday operations [8]. This means that the client and the new provider can possibly face unexpected knowledge gaps during transition.

B. Building Block Transition - Factors Contributing to Switching Success

Good project management and realistic time schedules are critical. "Unrealistic transition timetables are a frequent source of trouble. Both buyers and providers should look with a sceptical eye at the viability of their transition timeframes" [3].

Various researchers [32, 33] estimate that the transition takes two to three months. Reference [31] finds that the average transition time for initial outsourcing deals is 12 months while reference [16] estimates that the transition for large outsourcing deals can take between "18 month and more than two years". Generally can be said that the more complex the outsourcing endeavor the longer the duration of the transition. The literature review has not revealed any figures for outsourcing deals where providers are switched. As an indication it can be expected that the transition to the new provider will take as long as the transition to the incumbent provider [16].

It is also important to incorporate project buffers or contingencies into the project plan. "Any organization that explores a new sourcing option in terms of suppliers, new services, or new engagement models…must plan on false starts. Executives often manage learning by pilot testing new sourcing options" [26]. Although this is a good method of learning and getting the experience for some sourcing options in principle, it is not easy to pilot test switching ITO providers in practice.

To effectively manage the transition the client needs to set up an overall transition governance structure. Reference [2] asserts that "both client and service providers need to develop and implement an appropriate governance model for efficiently conducting day-to-day activities and for monitoring it at a higher level". The governance structure should define project roles and responsibilities such as the project joint steering committee. All parties (client, new provider and old provider) should be part of the joint steering committee. Part of the responsibilities of the joint steering committee is it to manage conflicts and to implement a joint transition program to plan, monitor, execute, and report on all transition switch deliverables and milestones.

Managing the complex tripartite relationship is resource intensive. Reference [4] emphasizes the importance of sufficient resources from the client to manage the transition and materializing risks. The authors call for the active involvement of the client management to ensure that the old provider supports the new provider as needed and therefore minimize service disruptions.

Reference [8] found that: "switching required close collaboration and mutual adjustment among all parties". Although the motivation of the old ITO provider to support the new provider might be low, it is a critical success factor for the overall transition success. "An uncooperative old supplier or an insensitive new supplier increases the risk of transition problems. Organisations must therefore carefully manage the delicate tripartite relationship tensions"[4]. Reference [8] also found that the old supplier is often needed to develop joint knowledge together with the new supplier to ensure that all parties meet their responsibilities - "critical to the success is the transfer of the knowledge of the client's environment and processes. Poor knowledge transfer may result in disruptions of operations, lowered service levels, and frustrations and dissatisfaction among the client's and the new vendor's employees".

Reference [10] emphasizes the importance of "commitment of employees transferred" to the provider and that the outsourcing success is related to it. "First, key employees must be retained and motivated. For most

activities, outsourcing does not mean transferring all the employees to the vendor. When an activity has been

performed in-house for a long period of time, firm-specific knowledge about how to run the activity smoothly has accumulated. Employees who possess this firm-specific knowledge must be identified".

What does this mean for switching providers? Clients need to identify employees from the incumbent provider who possess important firm specific knowledge and try either to reintegrate them into the client company or make sure that they move over to the new client or ensure adequate knowledge transfer. However, it is likely that the leaving provider will block the transfer of personal to stay competitive [7]. Transferring key employees early to the new provider could negatively impact the production capability of the incumbent provider. "Any transition in the key personnel should take place in a phased manner approved the client. This is critical for ensuring stability and consistency in the management of the project" [30]. Beulen, Tiwari, and van Heck [34] identify the following four major categories in an extensive literature review which fundamentally impact the performance of transition: transition planning, knowledge transfer, transition governance, and retained organisation. The four major categories are shown in Fig. 4.

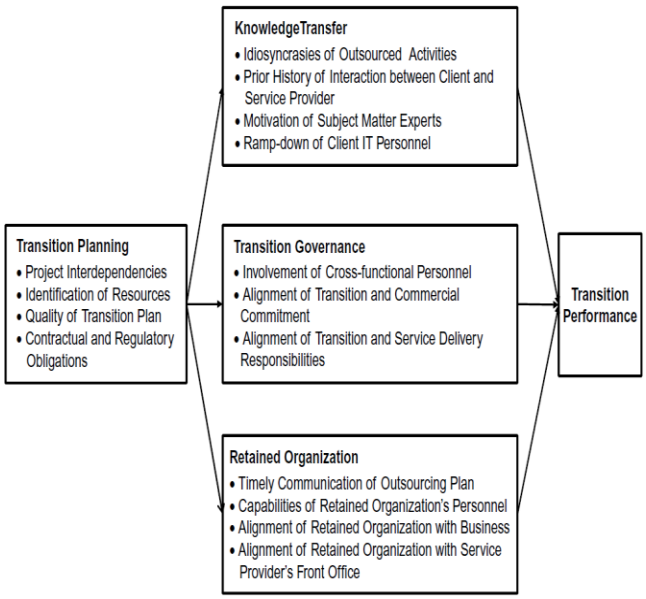


Figure 4. Theoretical framework of transitional performance [34]

In their longitudinal in-depth case study which was based on the 4 identified performance factors, the researchers found that knowledge transfer and transition governance had the strongest impact on transition performance [34].

IV. Concluding Remarks: Towards a Checklist for Successful ITO Transition

Even though the modern form of ITO practice effectively started in the late 1980s, it still cannot be considered a standardized routine management practice. Companies outsource their IT for different reasons though the primary objective is cost reduction. Several studies indicate a further growth of the ITO market of 5-8% per year [15]. The typical length of ITO contracts is 5-10 years [11] - a time span over which it is neither possible to foresee the clients' IT requests nor to estimate the impact of the overall economic environment. Various factors have led a number of clients to cancel their contracts prematurely.

The options for clients are to continue with the incumbent provider, switch the provider, or IT backsource (i.e., in-source again). It is estimated that between 25% [13] and 50% [5] of clients do not continue the relationship with the same provider. Miscellaneous factors influence these three sourcing options, most importantly the anticipated switching costs, the relationship between client and provider, and the fear of losing knowledge.

ITO success has not been extensively researched and there are contrasting conclusions regarding the contributing success factors [22]. Research has found that success needs to be considered in the context of the specific outsourcing arrangement. Several academics agree that the desirable outcomes need to be defined before the ITO starts, and that outcomes should be systematically assessed after it has been finalized and is underway.

General ITO factors contributing to success can be grouped into the major categories of ITO decisions, contractual governance, relational governance, and organizational capabilities [26]. In the category of ITO decisions, top management commitment and support is the most important factor [26]. In the relational governance category, trust and relationship management play a vital role [26]. However, given that significant amounts of capital are often invested in ITO deals, clients should not completely rely on relational governance factors such as trust and relationship. Important capabilities are required for success such as cost control and provider management. In addition, success itself can be considered as an important factor contributing to success.

The outsourcing process may be conceptualized as six major building blocks - investigation, provider selection, contract negotiation, transition, manage/service delivery, and options evaluation. The first three building blocks can be considered as the pre-delivery phase, the next two can be considered as the delivery phase, and the last activity can be considered as the re-evaluation phase. The transition building block is a complex, risky, and challenging process of strategic importance which begins after the contract is signed and ends with service delivery. It is assumed that "over two-thirds of the problems in these unsuccessful engagements arise due to failed or poor transition" [2].

When providers are switched, it cannot be assumed that the accumulated IT expertise (both in terms of personnel and client specific knowledge) will transition from the incumbent provider to the new provider. This results in several major issues, which are significantly impacted by the strategy of the incumbent provider. Their reaction can be grouped into two categories – a cooperative strategy or hostile strategy. Clients are well advised to prepare for both scenarios. Switching providers can be extremely resource draining for clients, as clients need to manage (monitor and correct) the operation of the incumbent provider, the operations of the new provider and additionally the transition from the old to the new one. This means clients should budget and plan for extra resources and associated contingencies.

During the pre-delivery phase it is essential for a successful transition to identify specific knowledge that needs to be transferred. A strategy should be developed to establish how this knowledge will be transferred and key knowledge experts need to be identified. Clients may reckon that major knowledge gaps will only be recognised during the actual transition.

In the critical transition building block, several factors contributing to success have been identified. Employing a stringent project management methodology with focus on realistic time schedules and incorporated buffers is an important ingredient for success. Implementing an effective governance structure plays a vital role for a successful transition when providers are switched. Ensuring early knowledge transfer and the transfer of key knowledge experts from the incumbent provider are two of the most important factors for success. Finally, managing the complex tripartite relationship is resource intensive but an important factor for success. The conceptual framework depicted in Fig. 3 has been developed to guide further research.

In conclusion, the switching of ITO providers is a complex, risky and resource intensive endeavour with the transition stage being the major building block in a wider process. However, not much is known about methods, processes and strategies for switching ITO providers as most research has focused on the initial outsourcing. [7-9, 24]. We therefore list below a series of recommendations distilled from the analysis of existing literature and documented experience, which may be used as a framework for further research and practitioner guidance.

A checklist for a successful transition would include the following main items for consideration:

A. Planning and Strategy

- Establish an overall governance structure
- Establish a culture of trust – knowing that distrust has the potential to seriously disrupt the overall transition process
- Develop a transition strategy for
  - People
    - Identify employees from the incumbent provider who possess important firm specific knowledge
    - Try either to reintegrate key employees into the client company or make sure that they move over to the new client or ensure adequate knowledge transfer.
    - Identify which employees need to transition early
  - Processes
  - Knowledge
  - Assets
  - Intellectual property
  - Applications
- Develop a strategy for a mixed operation scenario (since often both providers need to work jointly for a defined time to ensure continuity of service for the client)
- Develop a strategy to deal with a hostile incumbent provider
- Jointly develop a detailed transition plan
- Ensure that the transition plan is realistically timed and agreed by all three parties (client, incumbent provider, and new provider)
- Ensure that sufficient time for knowledge transfer is incorporated into plans

B. Operational issues

- Define and agree detailed transition success criteria which are relevant for the customer organization
- Measure success and tie success to payment for the new provider
- Ensure that the transition manager from the new provider has a successful track record for similar transitions
- Ensure that senior management from all parties are actively involved in the process
- Establish clear escalation processes
- Implement a joint transition program to plan, monitor, execute, and report on all transition switch deliverables and milestones
- Establish joint teams (client, incumbent provider, and new provider) for all work packages
- Implement a change and communication program
- Consider hiring external consultants with a proven track record in switching ITO providers for transition support
- Expect and plan for degraded service levels during transition
- Adapt the retained organization to reflect future structures

C. Financial/budgetary management

- Estimate switching costs
- Add switching costs on top of the costs for the contract with the new provider
- Expect hidden costs (e.g. paying incumbent for transition of intellectual property)

In summary, this article has attempted to point up a number of key considerations for organisations considering the switching of IT outsourcing providers. This can provide significant business benefits but there are also many potential pitfalls. As reference [35] concludes ‘‘the successful leadership of an IT implementation will continue to be a subtle craft”, and this undoubtedly applies to the switching of outsourcing providers as much as it does to any major IT project. Trade-offs will have to be made – for example, between the long-term and short-term cost implications of switching providers; and success is often determined by making the right judgements at the right time, and implementing key decisions in the right manner - for example, in the phasing in of one provider, and the phasing out of another. It is hoped this analysis will help those practitioners involved in this quest to achieve a more successful outcome in what remains a difficult managerial and operational challenge.

REFERENCES

[1] M. Olzmann and M. Wynn, "Switching IT Outsourcing Providers–a Conceptual Framework and Initial Assessment of Critical Success Factors," in *The Third International Conferences on Advanced Service Computing* Rome, Italy, 2011, pp. 38-45.

[2] E. Beulen and V. Tiwari, "Parallel Transitions in IT Outsourcing: Making It Happen," in *Global Sourcing of Information Technology and Business Processes*. vol. 55, I. Oshri and J. Kotlarsky, Eds., ed: Springer Berlin Heidelberg, 2010, pp. 55-68.

[3] M. Robinson and P. Iannone. (2007, 30.01.2011). *9 Ways to Avoid Outsourcing Failure, A three-part approach to maximizing the value of an IT outsourcing deal*. Available: http://www.cio.com.au/article/205186/9_ways_avoid_outsourcing_failure/?pp=1&fp=4&fpid=15

[4] K. Sia Siew, K. Lim Wee, and K. P. Periasamy, "Switching IT Outsourcing Suppliers:Enhancing Transition Readiness," *MIS Quarterly Executive*, vol. 9, pp. 23-33, 2010.

[5] D. Whitten, S. Chakrabarty, and R. Wakefield, "The strategic choice to continue outsourcing, switch vendors, or backsource: Do switching costs matter?," *Information & Management*, vol. 47, pp. 167-175, 2010.

[6] D. Whitten and D. Leidner, "Bringing IT Back: An Analysis of the Decision to Backsource or Switch Vendors," *Decision Sciences*, vol. 37, pp. 605-621, 2006.

[7] C. Chua, W. Lim, S. Sia, and C. Soh, "Threat-Balancing in Vendor Transition," in *3rd International Research Workshop on Information Technology Project Management*, Paris, France, 2008, pp. 19-26.

[8] M. Alaranta and S. L. Jarvenpaa, "Changing IT Providers in Public Sector Outsourcing: Managing the Loss of Experiential Knowledge," in *43rd Hawaii International Conference on System Sciences*, Hawaii, 2010, pp. 1-10.

[9] D. Whitten, "Adaptability in IT Sourcing: The Impact of Switching Costs," in *Global Sourcing of Information Technology and Business Processes*. vol. 55, I. Oshri and J. Kotlarsky, Eds., ed: Springer Berlin Heidelberg, 2010, pp. 202-216.

[10] J. Barthélemy and D. Adsit, "The Seven Deadly Sins of Outsourcing [and Executive Commentary]," *The Academy of Management Executive (1993-2005)*, vol. 17, pp. 87-100, 2003.

[11] J. Dibbern, T. Goles, R. Hirschheim, and B. Jayatilaka, "Information systems outsourcing: A survey and analysis of the literature," *Data Base for Advances in Information Systems*, vol. 35, pp. 6-98, 2004.

[12] L. Willcocks, "Machiavelli, Management and Outsourcing: Still On The Learning Curve," *Strategic Outsourcing: An International Journal*, vol. 4, p. 13, 2011.

[13] M. C. Lacity, L. P. Willcocks, and J. W. Rottman, "Global outsourcing of back office services: lessons, trends, and enduring challenges," *Strategic Outsourcing: An International Journal*, vol. 1, pp. 13-34, 2008.

[14] M. Benaroch, Q. Dai, and R. Kauffman, "Should We Go Our Own Way? Backsourcing Flexibility in IT Services Contracts," *Journal of Management Information Systems*, vol. 26, pp. 317-358, 2010.

[15] M. C. Lacity, S. Khan, A. Yan, and L. P. Willcocks, "A review of the IT outsourcing empirical literature and future research directions," *Journal of Information Technology*, vol. 25, pp. 395-433, 2010.

[16] P. Gottschalk and H. Solli-Sæther, "Critical success factors from IT outsourcing theories: an empirical study," *Industrial Management & Data Systems*, vol. 105, pp. 685-702, 2005.

[17] R. Hirschheim and M. Lacity, "The myths and realities of information technology insourcing," *Communications of the ACM*, vol. 43, pp. 99-107, 2000.

[18] J. K. Halvey and B. M. Melby, *Business process outsourcing: Process, strategies, and contracts*: Wiley, 2007.

[19] D. Whitten and R. L. Wakefield, "Measuring switching costs in IT outsourcing services," *The Journal of Strategic Information Systems*, vol. 15, pp. 219-248, 2006.

[20] R. Aron and J. V. Singh, "Getting offshoring right," *Harvard business review*, vol. 83, pp. 135-43, 2005.

[21] N. F. Veltri and C. Saunders, "Antecedents of information systems backsourcing," in *Information Systems Outsourcing: Enduring Themes, New Perspectives and Global Challenges*, R. Hirschheim, A. Heinzl, and J. Dibbern, Eds., ed: Springer, 2006, pp. 83-102.

[22] S. Cullen, P. Seddon, and L. Willcocks, "IT outsourcing success: a multi-dimensional, contextual perspective of outsourcing outcomes," in *Second information Systems Workshop on global Sourcing: Service, Knowledge and Innovation*, Val d’Isere, France, 2008, pp. 1-38.

[23] J. H. Bracht van and H. R. Kaufmann, "The Evaluation of the Strategic Business Units of Derivatives within German Savings Banks against the Background of the current Economic Crisis: Systematic Literature Review & Initial Approach. ," in *Business Research Challenges in a Turbulent Era*, Elounda, Crete, Greece, 2011, pp. 1895-1913.

[24] D. C. Chou and A. Y. Chou, "Information systems outsourcing life cycle and risks analysis," *Computer Standards & Interfaces*, vol. 31, pp. 1036-1043, 2009.

[25] S. Cullen, P. Seddon, and L. Willcocks, "Managing outsourcing: the life cycle imperative," *MIS Quarterly Executive*, vol. 4, pp. 229-246, 2005.

[26] M. C. Lacity, S. A. Khan, and L. P. Willcocks, "A review of the IT outsourcing literature: Insights for practice," *The Journal of Strategic Information Systems*, vol. 18, pp. 130-146, 2009.

[27] D. F. Feeny and L. P. Willcocks, "Core IS Capabilities for Exploiting Information Technology," *Sloan Management Review*, vol. 39, pp. 9-21, Spring98 1998.

[28] V. Tiwari, "Transition During Offshore Outsourcing: A Process Model," *ICIS 2009 Proceedings*, p. 33, 2009.

[29] S. Cullen and L. Willcocks, *Intelligent IT outsourcing: eight building blocks to success*: Butterworth-Heinemann, 2003.

[30] M. A. Parikh and G. Gokhale, "Legal and Tax Considerations in Outsourcing," in *Information systems outsourcing: enduring themes, new perspectives, and global challenges*, R. Hirschheim, Heinzl, A., Dibbern, J.(Eds.), Ed., ed Berlin Heidelberg: Springer, 2006, pp. 137-160.

[31] J. Barthelemy, "The hidden costs of IT outsourcing," *Sloan Management Review*, vol. 42, pp. 60-69, 2001.

[32] R. Hirschheim, A. Heinzl, and J. Dibbern, *Information systems outsourcing: enduring themes, new perspectives, and global challenges*: Springer, 2006.

[33] E. Beulen, P. Ribbers, and J. Roos, *Managing IT outsourcing*: Taylor & Francis, 2011.

[34] E. Beulen, V. Tiwari, and E. van Heck, "Understanding Transition Performance During Offshore IT Outsourcing," *Strategic Outsourcing: An International Journal*, vol. 4, pp. 1-1, 2011.

[35] A. Mcafee 'When too much IT knowledge is a dangerous thing' *MIT Sloan Management Review*, Winter, pp 83-89, 2003

www.iariajournals.org

International Journal On Advances in Intelligent Systems

ICAS, ACHI, ICCGI, UBICOMM, ADVCOMP, CENTRIC, GEOProcessing, SEMAPRO, BIOSYSCOM, BIOINFO, BIOTECHNO, FUTURE COMPUTING, SERVICE COMPUTATION, COGNITIVE, ADAPTIVE, CONTENT, PATTERNS, CLOUD COMPUTING, COMPUTATION TOOLS, ENERGY, COLLA, IMMM, INTELLI, SMART, DATA ANALYTICS

issn: 1942-2679

International Journal On Advances in Internet Technology

ICDS, ICIW, CTRQ, UBICOMM, ICSNC, AFIN, INTERNET, AP2PS, EMERGING, MOBILITY, WEB

issn: 1942-2652

International Journal On Advances in Life Sciences

eTELEMED, eKNOW, eL&mL, BIODIV, BIOENVIRONMENT, BIOGREEN, BIOSYSCOM, BIOINFO, BIOTECHNO, SOTICS, GLOBAL HEALTH

issn: 1942-2660

International Journal On Advances in Networks and Services

ICN, ICNS, ICIW, ICWMC, SENSORCOMM, MESH, CENTRIC, MMEDIA, SERVICE COMPUTATION, VEHICULAR, INNOV

issn: 1942-2644

International Journal On Advances in Security

ICQNM, SECURWARE, MESH, DEPEND, INTERNET, CYBERLAWS

issn: 1942-2636

International Journal On Advances in Software

ICSEA, ICCGI, ADVCOMP, GEOProcessing, DBKDA, INTENSIVE, VALID, SIMUL, FUTURE COMPUTING, SERVICE COMPUTATION, COGNITIVE, ADAPTIVE, CONTENT, PATTERNS, CLOUD COMPUTING, COMPUTATION TOOLS, IMMM, MOBILITY, VEHICULAR, DATA ANALYTICS

issn: 1942-2628

International Journal On Advances in Systems and Measurements

ICQNM, ICONS, ICIMP, SENSORCOMM, CENICS, VALID, SIMUL, INFOCOMP

issn: 1942-261x

International Journal On Advances in Telecommunications

AICT, ICDT, ICWMC, ICSNC, CTRQ, SPACOMM, MMEDIA, COCORA, PESARO, INNOV

issn: 1942-2601