International Journal on Advances in Intelligent Systems

2019 vol. 12 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 12, Number 1 & 2, 2019

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
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
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
Sukalpa Chanda, Gjøvik University College, Norway
David Chen, University Bordeaux 1, France
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
Ryszard S. Choras, University of Technology & Life Sciences, Poland
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
Mirela Danubianu, "Stefan cel Mare" University of Suceava, Romania
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
Zhi-Hong Deng, Peking University, China
Stojan Denic, Toshiba Research Europe Limited, UK
Vivek S. Deshpande, MIT College of Engineering - Pune, India
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
Mihaela Dinsoreanu, Technical University of Cluj-Napoca, Romania
Ioanna Dionysiou, University of Nicosia, Cyprus
Roland Dodd, CQUniversity, Australia
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
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, Daimler TSS GmbH, 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, Concordia University - Chicago, USA
Gregor Grambow, AristaFlow GmbH, Germany
Fabio Grandi, University of Bologna, Italy
Andrina Granić, University of Split, Croatia
Carmine Gravino, Università degli Studi di Salerno, Italy
Michael Grottke, University of Erlangen-Nuremberg, Germany
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
Maki Habib, The American University in Cairo, Egypt

Till Halbach, Norwegian Computing Center, Norway
Jameleddine Hassine, King Fahd University of Petroleum & Mineral (KFUPM), Saudi Arabia
Ourania Hatzi, Harokopio University of Athens, Greece
Yulan He, Aston 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
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
Hermann Kaindl, Vienna University of Technology, Austria
Ahmed Kamel, Concordia College, Moorhead, Minnesota, USA
Rajkumar Kannan, Bishop Heber College(Autonomous), India
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
Nittaya Kerdprasop, Suranaree University of Technology, Thailand
Katia Kermanidis, Ionian University, Greece
Serge Kernbach, University of Stuttgart, Germany
Nhien An Le Khac, University College Dublin, Ireland
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
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
Philippe Le Parc, University of Brest, France
Gyu Myoung Lee, Liverpool John Moores University, UK

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, Queen Mary University of London, 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
Agostino Poggi, Università degli Studi di Parma, Italy
R. Ponnusamy, Madha Engineering College-Anna University, India
Wendy Powley, Queen's University, Canada
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, University of Applied Sciences Fulda, Germany
Andreas Riener, Johannes Kepler University Linz, Austria
Ivan Rodero, NSF Center for Autonomic Computing, Rutgers University - Piscataway, USA
Alejandro Rodríguez González, University Carlos III of Madrid, Spain
Paolo Romano, INESC-ID Lisbon, Portugal
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
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
Rainer Schmidt, Austrian Institute of Technology, Austria
Bruno Schulze, National Laboratory for Scientific Computing - LNCC, Brazil
Ingo Schwab, University of Applied Sciences Karlsruhe, Germany
Wieland Schwinger, Johannes Kepler University Linz, Austria
Hans-Werner Sehring, Namics AG, Germany
Paulo Jorge Sequeira Gonçalves, Polytechnic Institute of Castelo Branco, Portugal
Kewei Sha, Oklahoma City University, USA
Roman Y. Shtykh, Rakuten, Inc., Japan
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
Cristian Stanciu, University Politehnica of Bucharest, Romania
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
Dan Tamir, Texas State University, USA
Jinhui Tang, Nanjing University of Science and Technology, P.R. China
Yi Tang, Chinese Academy of Sciences, China
John Terzakis, Intel, USA
Sotirios Terzis, University of Strathclyde, UK
Vagan Terziyan, University of Jyvaskyla, Finland
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
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, University 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
Zhenzhen Ye, Systems & Technology Group, IBM, US A

International Journal on Advances in Intelligent Systems
Volume 12, Numbers 1 & 2, 2019

CONTENTS

"Smart" Participation: Confronting Theoretical and Operational Perspectives

Clémentine Schelings and Catherine Elsen
LUCID Lab for User Cognition and Innovative Design
University of Liège
Liège, Belgium
e-mail: clementine.schelings@uliege.be; catherine.elsen@uliege.be

Abstract—**This paper explores the relatively new phenomenon of citizen participation in the Smart City context. We present a case study comparative analysis of three participatory approaches implemented in three European Smart Cities. Each of those operational perspectives is studied in view of the theoretical concepts conveyed by the scientific state of the art, this way highlighting similarities and gaps between theory and practice. The results are focused on (i) the various existing interpretations of the "citizen participation" and the "Smart City" definitions, on (ii) the different selection processes applied in all three cases to recruit the participating citizens and on (iii) the benefits and drawbacks associated with the implementation of participative processes in a Smart City. The article closes with a discussion about key elements to keep in mind when implementing a bottom-up participative approach in the context of a Smart City. Eventually, the confrontation between theoretical and practical perspectives results in a revisited version of Arstein's ladder of citizen participation, adapted to the Smart City context.**

Keywords-Smart City; citizen participation; Smart City definitions; operational perspective; selection of participants.

I. INTRODUCTION

This paper is an extended version of a previous, shorter publication presented at the conference Smart 2018, the Seventh International Conference on Smart Cities, Systems, Devices and Technologies [1].

The first Smart Cities were essentially focused on technological deployment aiming at optimizing urban performances, for instance thanks to freely accessible internet access, sensors and other pervasive devices. After this first wave of completely top-down and techno-centric cities (such as Songdo in South Korea or Masdar in the United Arab Emirates), we are slowly entering the era of a more bottom-up and participative model of Smart Cities. The citizens are now given an increasingly important role in the making of their smart built environments, because their acceptability is essential to insure the sustainability of the global smart model [2]. If many researchers acknowledge the fact that smart citizens are indeed key to Smart Cities, few information is yet available about how to implement a renewed participative approach, built on 1970 participatory models, in the making of such smart urban environments.

This research is one of the first steps of a larger research project, which is mainly focused on the citizens' perspective regarding the Smart City and the participative approach. This paper aims at studying and comparing different participatory initiatives conducted in 3 European Smart Cities particularly known for their citizen engagement and their bottom-up dynamics. The goal here is to document actual participative approaches in order to extract some key elements regarding citizen participation in the Smart City.

Comparing scientific perspectives with day-to-day, operational implementations of Smart City initiatives, this paper is structured in four additional sections. In Section II, we present a short literature review about participation in the Smart City. Section III then describes the interview-based methodology used for the comparative analysis of participative processes implemented in three carefully selected Smart Cities (one in the United Kingdom, one in the Netherlands and one in Spain). Section IV describes the obtained results: Subsection A gives the participatory context, while Subsection B is focused on the practical vision of two key definitions (Smart City and citizen participation) compared to more theoretical ones coming from the literature review, Subsection C presents the participants' selection processes in the three chosen cases and Subsection D focuses on the benefits and drawbacks related to the introduction of citizen participation in the Smart City. Section V discusses the results and raises some questions in regard of what the three chosen Smart Cities consider as "best practices", given their specific contexts.

II. STATE OF THE ART

This state of the art is kept voluntary short and will only present major theoretical models underlying the concepts of Smart City and citizen participation. Our subsequent intention is indeed to further study literature review in regard of empirical results in order to establish a comparison between theoretical and operational perspectives.

Two main concepts are at the root of this research project, namely "Smart City" and "citizen participation". Both concepts carry a multitude of (sometimes confused) definitions as they designate multifaceted realities [3][4]. As far as the "Smart City" concept is concerned, there are indeed a multitude of definitions and no real consensus about the meaning of this "buzzword" [5]. First of all, one should consider the common misconception according to which every Smart City is built from scratch, exactly like Songdo or Masdar [6]. Contrary to those emblematic and idealized

cities, which "*are the exception rather than the rule*", the "*actually existing smart city*" is far more nuanced, context-related and under-construction [6]. Keeping that in mind, we start this literature review with Giffinger's definition, one of the most frequently referred to. This definition puts some emphasis on the urban performance, which is nurtured by both information and communication technologies (ICT) and the smart inhabitants [7]. Giffinger's model dissects the concept of Smart City into six axes: economy, environment, governance, living, mobility and people [7]. Especially because of this "people" component, the citizen participation has lately become more and more popular in the Smart City context [8][9], building on the realization that citizens' potential rejection of the Smart City concepts could entirely jeopardize the sustainability of the global smart model itself [5][10]. Examples include the deployment of smart meters in each private home, which was among the first techno-centric, top-down smart initiatives. Although the guiding idea was to positively impact both personal consumptions and energy sector sustainable goals, acceptability was way below expectations as smart meters received a very cold reception from the inhabitants, sometimes even complete rejection [11][12][13]. Among the reasons for failure, those solutions missed the end-users' actual priorities, needs and concerns [14][15] and neglected the potentialities offered by users' active involvement into the design and decision processes. Citizens are thus increasingly considered as key actors of the making of the Smart City, and their sensitization and participation are the first steps towards awareness and acceptability [3]. The original vision of passive [15] or even invisible citizens [16] grows weaker, considering the significant influence of users' behaviors and practices on the adoption of (technological) solutions [14]. Gradually, the techno-centric smart environments give way to more eco-systemic Smart Cities and a shift is observed from the triple helix to the quadruple-helix model [17][18]. Side by side with universities, governments and industries, citizens are henceforth recognized as the fourth main stakeholder of any smart innovation [19]. Their role is no longer limited to on-the-move urban sensors and data generators [20], but shall extend to ideas generators, co-creators and co-decision makers given their local knowledge and use expertise [15]. Even though many authors nowadays share this viewpoint and promote citizens' engagement and empowerment, few information is available about how, concretely speaking, one should apply citizen participation in the specific context of Smart Cities [16]. In that regard, Fehér's study of a corpus of governmental, business and academic documents revealed that "*the expected active participation of citizens in the smart cities*" is one of the least documented [21]. Moreover, we suggest that older models of citizen participation, such as Arstein's ladder or Glass' objectives of participation [22][23], should be re-interpreted and might differently take place in practice given the renewed context of Smart Cities and given the opportunities offered by new technologies.

It is therefore crucial to confront theoretical and practical realities and to explore what local actors have in mind when referring to citizen participation in the Smart City.

III. METHODOLOGY

The methodology used to conduct this research is a comparative analysis of three cases, nurtured by semi-structured interviews with several stakeholders linked to smart projects and participative initiatives in each of those cases. This paper focuses on three European Smart Cities, the first one in the United Kingdom, the second one in the Netherlands and the last one in Spain. In all three cities, one research lab was chosen because it meets the following criteria: it is localized in an internationally recognized Smart City; it works in collaboration with the city officials and its main research activities are linked to citizen participation in future urban environments. The selection of those Smart Cities was moreover based on the Smart City Index, an international ranking proposed by Cohen, which is one among the few to consider some participatory dimension, at least beyond the voter turnout. The three finally chosen Smart Cities rank well in regard of inclusion (especially number of civic engagement activities offered by the municipality and voter participation in municipal elections) and creativity (in particular, number of registered living labs) [24].

Beyond those similarities, the three research centers remain quite different in their approaches. The Dutch lab generally considers self-organized citizens' communities and bottom-up movements as essential triggers for any launched project, while the British lab rather tries to integrate a participative dimension to existing projects that would not make sense otherwise. The Spanish lab holds an intermediate position, conducting participative experiments essentially in the public space and starting as well from a living community or a given context. Thus, the Dutch and the Spanish labs are always involved in participatory initiatives, but the British lab also conducts some research projects without any citizen participation. Another difference between the labs lies in the end-use of the material produced through the participative process. The British lab seeks to develop a marketable product, while the Dutch lab rather promotes open-access material that can be freely reused after the end of each project. The Spanish lab, on the other hand, gets involved in upstream phases of the decision-making process and rather delivers information and recommendations for the benefit of the municipality. A last difference is linked to the various profiles and backgrounds of the members of the three labs that therefore develop different identities. The British lab is mainly composed of computer scientists using data for socio-technological purposes. The Dutch lab brings together researchers with data, design and digital humanities backgrounds. The Spanish lab, specialized in Arts and Science, includes experts in Physical, Chemical, Computer and Social Sciences.

In practice, each interview was expected to last about one hour, but the effective length varies between forty and eighty minutes. Several types of stakeholders were interviewed: directors of the research centers, labs' team members, Smart City managers, city officials and other experts from the fields of participation, technology and urban planning. Given this variety of interviewees' profiles, different sets of questions

were prepared, in line with the specific expertise of each actor. In addition, some essential issues were discussed with the complete sample of respondents, such as their own definitions of "Smart City" and "citizen participation".

As a first step of our comparative analysis, this paper will focus on eight essential interviews and more specifically on the results of meetings conducted with three lab directors and five team members. We decided to start our study with those stakeholders because they are very close to fields' realities: the team members are the day-to-day operational actors, while the directors are the spokespersons of each lab and therefore structure those labs' vision and attitude. The idea is to understand the global visions of those three labs and to compare their different interpretation of the participative approach, given their actual perception of the Smart City.

Globally, eight main themes are addressed through the interviews (see Table I). Additional questions regarding the presentation of the city (specificities, history, population) and the policy (objectives, priorities, citizens' input) are discussed with city officials and Smart City managers, but will not be presented in this paper.

IV. RESULTS

The results of the eight interviews are structured in four subsections. First, we will present the contexts in which citizens become active participants for each city. Then, we will present interviewees' definitions of the Smart City and the citizen participation, in comparison with the scientific state of the art. We will next compare the participants' selection processes as conducted in all three labs and we will study the impact such processes have on the recruited citizens' profiles. Eventually, we will detail the perceived benefits and drawbacks resulting from the implementation of citizen participation in concrete smart urban environments.

A. Participatory context

The citizen participation is a complex process that may tire the citizens if their input is repeatedly requested for each and every project related to the Smart City. Therefore, it is of crucial importance to wisely choose topics for which participants' contribution is considered essential. Each lab has a different strategy regarding this issue. The British lab focuses on "*the stress points in the city (…), priorities, which have been identified with the council*" and uses citizen participation mainly to get feedbacks about the solutions developed by the researchers in cooperation with the local authorities. The logic of the Dutch lab is quite different. Once again, they start from context-specific urban problems, but the chosen topics result from shared interests between the citizens' preoccupations and the local authorities' priorities. Thus citizens are always involved in projects that they feel concerned about, and that they wanted to integrate even prior to any involvement from the city itself. The Spanish lab, for its part, always initiates a participatory process when requested by a different stakeholder, be it municipality or community members or even sometimes a more complex group bringing together several profiles. Therefore, the proposed topic always results from a demand of some locally involved people. However, even though the lab does not choose the specific topic, its expertise in environmental health and air quality definitely fuels the participative processes. Another difference between the three approaches is the timing chosen for citizens' participation. British citizens often participate at the end of the process, while the Dutch citizens always participate from the beginning and generally during the whole project. Spanish citizens can be part of the project from the beginning or join later, especially in the case of broad public participation occurring in public spaces. A more continuous participation is also possible when considering co-design sessions for instance.

TABLE I. MAIN THEMES STRUCTURING THE INTERVIEWS WITH THE DIRECTORS AND THE TEAM MEMBERS OF THE LAB

Common themes	Directors
- Presentation of each actor (background and role) - Own definitions of the two main concepts (Smart City and citizen participation) - Presentation of concrete projects (context, success stories, possible improvements) - Participatory approach (benefits, drawbacks, challenges) - Technology (role, ethics, privacy)	- Contacts with other stakeholders of the ecosystem (city officials, citizens, industrial partners)
	Team members
	- Participatory methodology (phases, methods, objectives) - Participants (roles, selection criteria, profiles)

B. Definitions

The two following subsections aim to define the Smart City and the citizen participation on basis of the interpretations proposed by the eight interviewees. The results are examined with respect to the state of the art, highlighting the convergences and the divergences between theory and practice.

1) Smart City: We focus here on the definition of the Smart City, as perceived by the stakeholders interviewed on the field. On the basis of the most widespread definitions, we will compare the different visions hold by those experts (see Table II and Table III).

The first interesting observation is that there is a distinction between their current vision (see Table II) and their prospective vision (see Table III) of what the Smart City is. In other words, the interviewees are fully conscious that the Smart City is an ongoing process that can be described on the one hand on the basis of current initiatives, with their promising achievements and their manifest limitations, or, on the other hand, on the basis of the likely evolutions and hopes for the future. All eight interviewees are moreover fully conscious that their own definitions match their personal "*way of understanding a Smart City*" (Director of the Spanish lab) and rely both on their scientific background and their perception while experiencing their city becoming smarter. In the interviewees' discourses, we obviously find key elements that meet some definitions from the state of the art. The interviewees' propositions are identified by codes (see Table II and Table III), which are referenced in brackets hereafter.

TABLE II. INTERVIEWEES' CURRENT VISION OF THE SMART CITY

A Smart City is…		Interviewees	
		Directors of the labs (D)	*Team members (M)*
Smart City	***United-Kingdom (U)***	DU1 a technology-connoted word DU2 a city for one citizen category	MU1 a smartphone-adapted city MU2 a fuzzy concept
			MU3 the use of data science and artificial intelligence to better understand its needs
	Netherlands (N)	DN1 a set of fully autonomous systems DN2 a top-down controlled city DN3 an easily managed city DN4 a city of "*dumb citizens*"	MN1 a set of technological infrastructures MN2 a product of big technology companies MN3 a concept disconnected from citizens MN4 an optimized and efficient city
			MN5 a maybe more efficient city MN6 a city developed for the companies
	Spain (S)	DS1 a multi-meaning word	MS1 a responsive and reactive city regarding its citizens' needs

DU = Director of the lab in the United-Kingdom (UK); DN = Director of the lab in the Netherlands; DS = Director of the lab in Spain; MU = team Members of the lab in the UK; MN = team Members of the lab in the Netherlands; MS = team Member of the lab in Spain.

TABLE III. INTERVIEWEES' PROSPECTIVE VISION OF THE SMART CITY

A Smart City should be …		Interviewees	
		Directors of the labs (D)	*Team members (M)*
Smart City	***United-Kingdom (U)***	DU3 a technology-improved city DU4 an inclusive city	MU4 a set of facilitating technologies MU5 a support in daily life MU6 an assistance for everybody
			MU7 a system facilitating decision-making
	Netherlands (N)	DN5 a less obvious city management DN6 a city of creative citizens DN7 a city of "*smart citizens that are able to fulfill their own information needs*"	MN7 /
			MN8 a more citizen-centric city MN9 an improved living environment
	Spain (S)	DS2 a rethink or a redesign of the city DS3 a set of solutions defined thanks to citizen participation	MS2 a dynamic and flexible city MS3 an inclusive city

DU = Director of the lab in the United-Kingdom (UK); DN = Director of the lab in the Netherlands; DS = Director of the lab in Spain; MU = team Members of the lab in the UK; MN = team Members of the lab in the Netherlands; MS = team Member of the lab in Spain.

First of all, each expert mentions the technological aspect of the Smart City, be it considered as a positive or a negative element (DU1, DU3, MU1, MU3-4, DN1, MN1-2, MN6). Following some authors, new technologies are obviously part of the Smart City, in the sense that they support any other key aspect of the city such as wellbeing and quality of life [8][25]. This vision is shared by the interviewees, but perhaps in a more nuanced way as they feel that actual Smart Cities may misinterpret this use of technology, making it an end per se especially due to the market pressure. The Dutch team members even suggest that the Smart City, as currently configured, will only benefit big companies (MN2, MN6), such as those who originally introduced the concept [6]. However, the two British team members still believe that technological developments will evolve into daily-life facilitators, as much for the citizens as for the decision makers (MU4-5, MU7). The Dutch lab is more cautious and considers that the current practical message conveyed by the Smart City is not yet the perfect solution for our future urban ideal (MN5, MN7). Even though they recognize that technology should help to generate more efficient urban systems (MN4), they doubt those technical improvements will suffice to produce more livable urban spaces (MN5, MN9). The Spanish lab also remains prudent, since the introduction of smartness into the city is not only based on technology, but also on the people that will "*redesign or rethink a little bit the city*" (DS2). Actually, this nuance and moderate (mis)trust regarding the Smart City concept is also the consequence of an almost exclusively top-down governance of many smart projects (DN2). This approach, although neglecting citizens' input (MN3, MN8), provides the advantage of easily managing the city (DN3, DN5) and rather efficiently optimizing its day-to-day operation [7][26]. Ben Letaifa yet emphasizes the importance of a complementary bottom-up approach through citizen participation [5]. Furthermore, Giffinger insists on the fact that a city cannot be smart and efficient unless citizen's intelligence is valued and exploited [7]. According to the interviewees, citizens should indeed play a specific role in their smart urban environments, and should be empowered in order to actively participate (DN4, DN6-7, DS3). Citizens are indeed best placed to express the specific needs of the city, which should orient the solutions that ought to be developed (MS1). The Dutch director even specifies that citizens should themselves be able to respond to their information needs, i.e., to become "*self-decisive, independent and aware citizens*" [7]. This citizen autonomy is only possible in an inclusive Smart City (DU2, DU4, MU6, MS3) and one of the next big challenges is to limit obstacles to such inclusion, such as the digital divide [15]. Following one of the Spanish team members, this inclusivity is especially hard to reach while the "*Smart City discourse narrative*" focuses exclusively on technological aspects, and is therefore far too often "*restricted to a specific target group*". Finally, compared to the literature, one important aspect is missing from the interviewees' discourses: sustainability. Surprisingly, no participant refers to environmental and demographic issues while those are among the main reasons to promote smart initiatives, offering a long-term solution for our urban environments [20][27]. This demonstrates the extent to which the Smart City is a complex concept with many meanings and no

unanimous definition, especially in regard of specific, locally constrained situations (MU2, DS1). According to the participants, the Smart City should, as far as possible, remain dynamic and flexible, i.e., adaptive to every city particular context (MS2).

Giving a definition of such a complex notion is sometimes very difficult for the interviewees. Therefore, two of them formulate their answer on the basis of definitions coming from the state of the art. The researcher shows them five references (Table IV) and they can pick those that match or contradict their mind, while commenting and arguing their choice. The most appropriate definition is Giffinger's [7], while Dameri's [25], Toppeta's [28] and Hall's [26] are considered less convincing, probably because those three envision the citizen as a recipient, rather than a real actor of the Smart City. This idea of a passive citizen is obviously not in line with the participatory vision of the selected labs, but is clearly ever present in the literature. The fifth definition comes from the Smart City Institute [29] and is well received by the interviewees, since it reflects both technological and eco-systemic aspects of the Smart City, including citizens' equal involvement as the other smart actors.

TABLE IV. SMART CITY DEFINITIONS

Reference	Definition
GIFFINGER (2007)	A city well performing in a forward-looking way in economy, people, governance, mobility, environment, and living, built on the smart combination of endowments and activities of self-decisive, independent and aware citizens.
HALL (2000)	A city that monitors and integrates conditions of all of its critical infrastructures, including roads, bridges, tunnels, rails, subways, airports, seaports, communications, water, power, even major buildings, can better organize its resources, plan its preventive maintenance activities, and monitor security aspects while maximizing services to its citizens.
DAMERI (2013)	A smart city is a well defined geographical area, in which high technologies such as ICT, logistic, energy production, and so on, cooperate to create benefits for citizens in terms of well being, inclusion and participation, environmental quality, intelligent development; it is governed by a well defined pool of subjects, able to state the rules and policy for the city government and development.
TOPPETA (2010)	A city combining ICT and Web 2.0 technology with other organizational, design and planning efforts to de- materialize and speed up bureaucratic processes and help to identify new, innovative solutions to city management complexity, in order to improve sustainability and livability.
SMART CITY INSTITUTE (2015)	A "smart city" is a multi-stakeholders' ecosystem (composed with local governments, citizens' associations, multinational and local businesses, universities, international institutions...) engaged in a sustainability strategy using technologies (ICT, engineering, hybrid technologies) as enabler in order become more sustainable (economic prosperity, social well-being and conservation of our natural resources).

2) Citizen participation: Another notion difficult to grasp is the citizen participation, although this time it goes back to a nearly fifty-year-old concept [30]. Throughout the years, the participatory approach has evolved into new practices and its "smart" interpretation is certainly still another perspective to take into account. Based on the experts' interviews and the keywords they use, we identify four main axes around which we summarize their propositions in order to characterize participation in the age of Smart Cities: communication, citizen control, conditions and data manipulation (Figure 1).

The three labs generally tend to agree on some key aspects of citizen participation, but each of them insists on different axes. First of all, the British and the Spanish labs notice that participation is above all **communication**, and most preferably two-way communication. Information has to be exchanged between citizens and power holders, be they researchers or local authorities, because every actor's perspective is valuable and should at least be listened to. This continuous dialog between the different stakeholders is

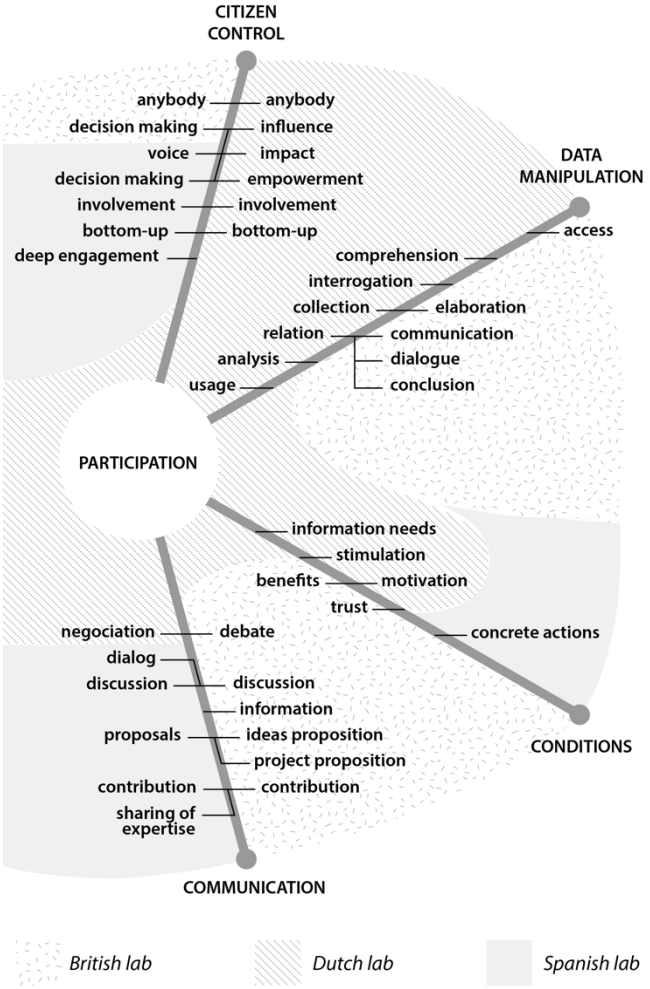


Figure 1. Axes of citizen participation on basis of interviewees' visions

an opportunity to explore everybody's perspective, to share personal experiences, to benefit from each individual expertise and to enrich them. There are several levels of communication depending on the contribution of the participants, who either just receive information, propose their own ideas or even negotiate with the power holders. British and Spanish actors put a certain emphasis on verbal exchanges, which do not yet suffice to qualify as participation according to some authors [31]. One step further, all three labs agree with Arnstein and consider that "*citizen participation is citizen power*", meaning that citizens should have a real impact on the decision-making of any participative process [22]. Citizens are not just informed, educated or consulted to ease tensions, but should have an actual voice translated into action [22][32]. The Dutch and the Spanish labs both consider that this **citizen control** goes hand in hand with involved and empowered citizens, which means that they are given the opportunity to actively and wisely participate. Furthermore, anybody should enjoy such opportunity, according to the British and the Dutch labs, irrespective of gender, social status or even technology acquaintance. Along with this empowerment, the citizens also have a responsibility since they need to engage themselves in the participatory process. Therefore, beyond being offered with the possibility to participate, all three labs are conscious that citizens' willingness to participate is crucial and that they are some **conditions** that can ease the participative process and impact its implementation. The Dutch lab, in accordance with Klandermans and Oegema, specifies that the participants have to be motivated in order to actually take part to the project [33]. More importantly, participation often arises from an information need, directly expressed by the participants or identified after a stimulation phase. Consequently, citizens should be present from the early phases of the project [34], in order to make sure their needs will nurture the project definition. Moreover, the British lab is convinced that participation cannot efficiently operate without trust and benefits. Citizens are indeed more prone to participate if they "*foresee the benefits in the long run*", such as time and money savings. Following the Spanish lab, processes that end up providing concrete actions and results also motivate participants. They indeed generally want to be agents for change, transforming and impacting their environment, their neighborhood, their community or even their own person. The Dutch lab adds that it is very important to tell people about the ins and outs of the project from its beginning, even if sometimes their participation can remain quite modest, rather than deluding and letting them believe that their individual thoughts will automatically be part of the final output. As documented in the literature, such tokenism will inevitably result in disappointment, mistrust and failure of the participative process [32]. Eventually, the fourth axis concerns **data manipulation**, which is intrinsically linked to the era of the Smart Cities. This axis has yet not been extensively documented in the literature review about citizen participation, maybe because there is a temporal gap between participatory theories introduced in the 70s and the first references to smart technologies appearing in the early 2000s. The "data manipulation" designates the way citizens interact with the data produced through the participative process. According to the Dutch and Spanish labs, citizen participation is not limited to data collection, but should extend to their understanding, appropriation (interrogation and relation), analysis and usage by the citizens in order to create new knowledge. Indeed, new technologies might impact participative processes and are seen as an empowering factor, since "*digital technology allows cities to engage with citizens in decision-making processes*" [9]. This new form of participation will enable participants to elaborate their own data, to communicate about them, to draw evidence-based conclusions and to propose relevant actions for their local environment. Learning to manipulate data will therefore empower the citizens and give more weight to their concerns and ideas, while their local expertise is sometimes questioned because considered as less legitimate by some professionals.

C. Selection of participants

Given their different approaches, the three labs also show some discrepancies regarding the participants' selection. This section will present which participant profiles are targeted when a participative process is implemented, according to each Smart City. One recurrent goal in participatory processes is to make everyone participate, but in practice it is considered as nearly impossible. To select the participants, all three labs therefore start from a local neighborhood, but their different interpretation of "local" has implications on the profiles of the sampled participants. Figure 2 summarizes the descriptions proposed by the three labs regarding recurrent citizen profiles taking part to their smart initiatives. The shaded zones in Figure 2 highlight the keywords discribing similar citizens' profiles in the three labs.

The Dutch lab "*select(s) (...) citizens basically by tapping into existing platforms or organizations that feed into the community*" while the British lab focuses on one specific geographical area. As a matter of fact, the Dutch interpretation is linked to existing communities that have already initiated some projects in order to solve local issues. In line with its research interests, the Dutch lab chooses to support and develop the ideas of the community, because it seems more relevant to tackle actual people's concerns and to meet a real need. The British perspective is quite different and rather aims at testing on pilot sites some technologies, which would in fine be deployed at scale, requiring to get more "*general users*". Therefore, the British researchers just select a neighborhood and consequently the whole group of people living there. Halfway of those two approaches, the Spanish lab proceeds on a case-by-case analysis, alternating the recruitment of "*given communities and neighbors in general*". This switch of strategy is explained by two main

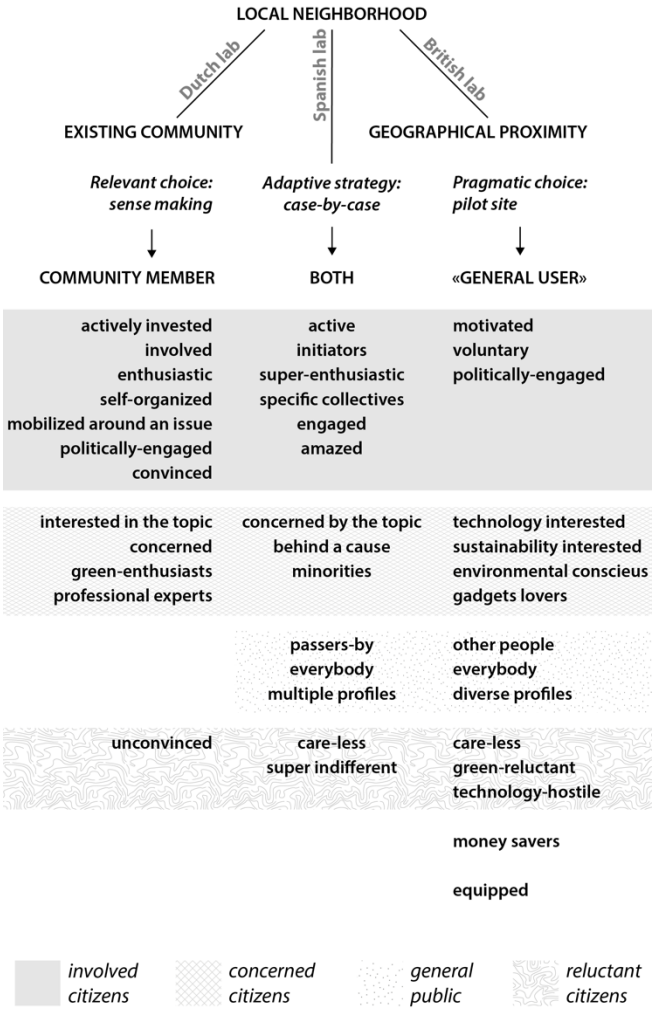


Figure 2. Participants' profiles on basis of interviewees' selection process

factors: (1) the initiator of the participative project and (2) the chosen participatory method. Actually, the person or the group of people who initiates the participative process can be either a municipality or a local community itself, which will then automatically feed the selected group of citizens. In the case of a more top-down initiative, decision-makers might face difficulties recruiting those local communities, which could claim for autonomy. Moreover, their position rather pushes public administrators to select as many people as possible, trying to reach some citizens' representativeness. In addition, the participants' profile will also vary depending on the participatory method. For instance, co-design workshops about very specific topics require a long-term commitment that is more easily achieved with organized communities of concerned citizens. Conversely, pop-up interventions deployed in public spaces in order to sensitize the citizens, explore their perceptions and/or test some solutions call "*every person passing by*" to participate.

Given their divergent selection strategies, Dutch and British labs' participants present different profiles, which are all quite well represented in the Spanish samples. As far as the Dutch community members are concerned, they are of course very active and are described as "*involved*" and "*invested*" in the topic or even in concrete actions. This also means more environmental-conscious citizens that are generally interested in any initiative related to the smart city agenda. Although the Dutch sample mainly comprises pro-active citizens, all participants might not be convinced by the process, in particular when a change of habits is involved. For instance, some people could have strong interest in the environmental topic but at the same time believe that they already manage their own situation quite successfully, and that other people should improve their individual behaviors and practices first. Therefore, even if they seem less enthusiastic, those participants are still the engaged ones that always show up at this kind of participatory process, or that have already started their own initiative. Since the British recruitment is made on a voluntary basis, the same super-enthusiastic profiles are also present but this time they are not self-organized around common values. The only condition to participate to the British project is to be equipped, i.e., for instance in a project of garden watering the condition is to have a garden. Besides the always-involved people, other profiles show up such as careless people, technology- and green-reluctant citizens that may decide to participate in order to save time or money for instance. Contrary to the Dutch communities, the British participants therefore constitute a less homogeneous sample presenting a limited amount of shared values and interests, but rather a group of people motivated to participate for various reasons. The Spanish participants, for their part, are closer to the British profiles, in the sense that they are sometimes showing enthusiasm and sometimes indifference. However, those less motivated citizens are only present in the case of kiosks for instance or any other one-time opportunity to participate. In contrast, in the case of a more demanding and continuous participation approach, such as co-design sessions, the Spanish sample is mainly composed of community members, characterized by higher engagement and motivation.

D. Benefits and drawbacks of smart participation

This section focuses on the benefits and drawbacks of smart participation as they are reported by the interviewees. More particularly, our hypothesis is that the implementation of a participative process in a Smart City might lead to several consequences, as well positive or negative effects and externalities. Those (dis)advantages are often already documented in the state of the art about citizen participation in general, irrespective of its applicability in a Smart City. However, this specific digital context may reveal new repercussions, which deserve to be taken into account when introducing a participatory dynamic in a Smart City.

1) Benefits: Figure 3 highlights the main benefits following our eight respondents. Benefits are organized according to three levels of stakeholders: the individual level corresponds to the personal gains of one participant,

the participants' level refers to the collective advantages collected by the people who are involved in the participatory process, and the beneficiaries' level takes into account the more global benefits, i.e., for the participants, the neighborhood, the local community, the municipality or even the city professionals (engineers, architects, urban planners, designers). Furthermore, all those contributions from the citizen participation to the Smart City agenda are perceived at different time phases. Indeed, the pre-participation benefits are often associated with promises or expectations that might be realized in a post-participation phase and broaden the extent of benefits achieved. During the participative activities, other elements intervene and they often constitute essential premises of the final success of the whole participatory approach.

Considering the pre-participation benefits, each lab has a different but complementary vision. While the British lab is focused on the incentive to reward the citizens for their participation, the Dutch lab rather mentions the importance of participation to ensure the relevance and the sustainability of the project, and the Spanish lab envisions participation as a huge opportunity to take action for every potential participant. Once again, those three postures correspond to their philosophies, respectively starting from a community, a project or an alternation between both.

The three labs are more in line when they consider the direct benefits of participation, which occur during the process itself. They above all stress the awareness as the biggest contribution, advantageous for all three stakeholders' levels. Indeed, through their participation, the citizens become more conscious of the operational constraints, i.e., the economic, technical, normative, etc. aspects of the project that they may ignore if they are out of their personal or professional expertise. Participants also gain a clearer view and a better understanding of environmental issues and technological innovations, two major elements of the Smart City era. The Dutch lab even points out that citizens are more aware of their own living environment, which they now see with brand new eyes.

Similarly, the city officials and professionals become aware of the citizens' field perspective, i.e., their actual and local problems, needs and usages. Such a practical experience of the area is clearly an expertise that the so-called experts in particular may lack. The Spanish and the Dutch labs therefore insist on the necessity to share contextual information, whether be between participants or, at a larger level, with the professionals, the power holders and the communities. Thereby, participants will also develop new knowledge and capacities, especially regarding data and technologies. Those learning processes and awareness favor the citizens' empowerment, since their new capabilities allow them to make "*not better choices or different choices, but they at least are informed in which choices they can make, based on that data*" (Director of the Dutch lab), as far as their behaviors, lifestyles and habits are concerned.

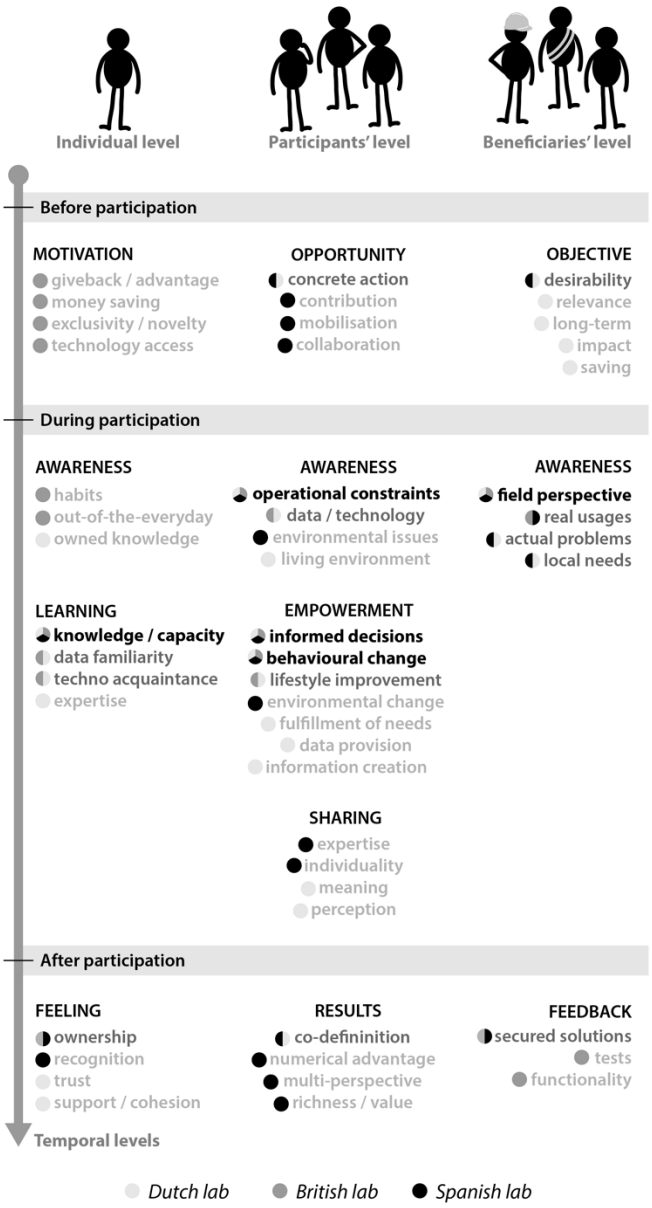


Figure 3. Benefits of the implementation of a participative process in a Smart City

At the end of the participatory process, the additional benefits naturally include the feedbacks, which lead to final solutions that are functional and adapted to the citizens' needs and concerns. Besides this likely optimized reception, the co-definition of the results makes them richer and more valuable. Moreover, participation enables to gather much more data, which was very useful when the Spanish lab collected air quality measurements through the installation of hundreds of chemical sensors for instance. Finally, the participation provides a sense of ownership to the participants, who feel that they have personally and collectively contributed to the project and are recognized for

the time and the efforts they invested. The community comes back with more cohesion and support, and participation might even build trust towards the municipality.

2) Drawbacks: Figure 4 emphasizes the main drawbacks of participation, even though the interviewees rather call them "*challenges*" or "*difficulties*". As a matter of fact, all the identified downsides can be organized into two categories: the threats and barriers that may accentuate one critical aspect of the participative process (e.g., representativeness), and the resulting consequences, i.e., the potential risks and disadvantages, which may slow down, compremise or completely derail the participative process.

In comparison to the state of the art, several drawbacks mentioned by the interviewees correspond to the well-known limits of the traditional 70's participatory theories. The tokenism, or pseudo-participation, is a recurrent problem, which results from a symbolic consideration of the citizen input, in order to complete the participatory obligation and/or to ease one's conscience [32]. The participants' contribution, limited and often punctual, is therefore not taken into account by the power holders [22]. Of course, citizens are conscious that their participation make few or no difference, so they feel disappointed and insignificant because "*they thought they were more important*" and "*do not want to be in the margins of whatever*" (Team member of the Dutch lab). Another inescapable issue is the lack of representativeness of the sample, which generally includes the most engaged and motivated citizens [34]. Following the Dutch and the Spanish labs, the main difficulty is to find a way to reach the whole citizenry, which is impossible given their various profiles, especially regarding language and culture. Moreover, some populations are even harder to contact, such as the poor and elderly for instance. Both representativeness and tokenism are not referred as inconvenient by the British lab, given its specific participatory strategy. First, the recruitment of citizens occurs in a determined geographical area, which eases the representativeness. Second, the participants' input occurs during the evaluation phase, which is the moment when citizen participation is popularly considered as the most valuable and legitimate.

There is a consensus among the three labs that "*time constraint is dramatic*" (Director of the Spanish lab), as much for the researchers or organizers of the participatory process than for the participants themselves. The first effect of timing is the difficulty to end up with concrete solutions, results or actions that will impact policy [34], while being committed to the budget and ensuring the continuation of the project by the beneficiaries on their own. In the literature, this time-consuming aspect of participation is also often raised by practitioners who are encouraged to integrate participation into their day-to-day work [35]. Timing is also critical for the citizens who have other concerns and

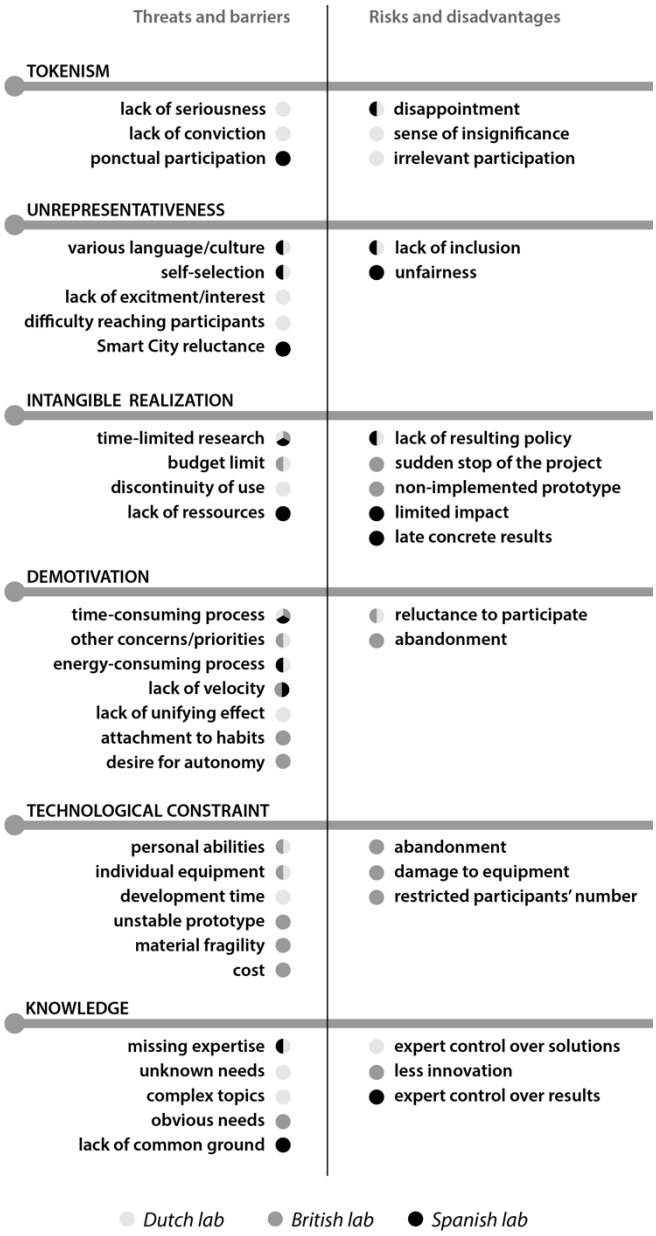


Figure 4. Drawbacks of the implementation of a participative process in a Smart City

priorities, which may dissuade them to invest energy for participating. Even when they are informed of the benefits, they might sometimes prefer to keep their current situation and even pay more money, rather than making additional efforts or changing their actual habits and behaviors [14]. Convincing them to get involved is therefore even more complicated when the foreseen advantages will be only perceived at the end of a long process, which makes them intangible and pushes participants to progressively lose interest. Another element that may sometimes lead to abandon the project is the technological issue, which reveals particularly present in the case of smart projects. The problem is not only related to the unfamiliarity of the

citizens with the technologies, but also to their lack of equipment to manipulate them, which are the two chore characteristics of the digital divide [15]. For instance, the Dutch lab found alternatives when they realized that some neighborhoods had no Wi-Fi, and the British lab had to deal with "*users (that) keep breaking the sensors*" (Team member of the British lab). Furthermore, balancing citizen participation and technological development is all the more difficult given that it requires a temporal synchronization and that the fragile prototypes are available in reduced number. Finally, the last limit reported by the interviewees is the lack of knowledge, if not naivety, of the citizens in certain complex domains [36]. For example, the Spanish lab would like to collectively analyze the data with the participants, but it remains a task reserved for specialists who will "*eventually look for certain results and not others*" (Team member of the Spanish lab) and orient the following discussions and decisions. In addition, the citizens sometimes face difficulties expressing their needs and sometimes propose ideas that are less innovative than already-existing solutions.

V. DISCUSSION

The participative approach is gaining more and more popularity in Smart City projects, but there is very little practical advice about how to conduct a participatory methodology in such specific context. Given the ground experience of the interviewed experts, we identify several questions emerging from their ongoing and completed projects in terms of concept definitions, selection of participants, benefits and drawbacks. Those key elements provide useful information both for scientific researchers and operational stakeholders.

First, the various existing interpretations of the Smart City concept definitely have an impact on its operational implementation. For instance, the concept of pervasive technology seems to play a major part in the current vision of the Smart City, but the citizen is expected to play a larger role in our future smart cities. The interviewees' prospective vision of the Smart City is generally closer to the definitions found in the scientific state of the art, while their current vision is less optimistic and is probably nurtured by the first failures encountered by Smart City projects around the world. Moreover, the interviewees' visions of the Smart City are affected by the Smart City discourses, such as the marketing literature conveyed by IBM, Cisco or Siemens. Undoubtedly, this approach is inappropriate to an "*actually existing smart city*" [6] such as our three European cases and unsatisfactory for our interviewees who therefore develop a prospective vision exceeding the techno-centric popular belief. Furthermore, this variety of interpretations is also linked to the fact that "*the smart city concept encompasses intangible aspects such as quality of life and well-being components, whose measurement is subjective and difficult to perform*" [37]. One team member of the Dutch lab even considers that the technology is just as difficult to grasp, since it "*is just very much an invisible world and a government program*". Given the plethora of interpretations and definitions, each ecosystem of actors working on smart initiatives should at least, and as a priority, agree on a shared vision, generating clear objectives and means to achieve them. The question to keep in mind is: how do we define the Smart City, and especially regarding the roles played by the technologies and by the citizens? Although the absence of a consensual definition may seem problematic, it represents at the same time an opportunity for the local key stakeholders to adjust and to contextualize their own definition, thus falling outside the preconceived notion of a technocratic city and finding a balance between technological and collective intelligences.

The second attention point concerns the definition of the citizen participation. Among the four axes previously identified (Figure 1), the communication, the citizen control and the conditions are explicitly discussed in the literature review, but the data manipulation is not yet part of the traditional scientific discourse. Citizen appropriation of the produced data is nonetheless a new form of participation and this technological dimension is even more crucial in the current smart context. This late integration of this data component as an additional facet of the citizen participation is clue that older concepts introduced in the 70s should evolve and that new participatory tools and methods are needed to complement the existing ones. Indeed, Arnstein's ladder is nowadays still a valid theory, but it may lack some new steps, indicative of the numeric participation. Therefore, one question to ask is: how can the new technologies support the participative process and the citizens' active, inclusive involvement? Based on the operational perspectives of the labs, Figure 5 below is an attempt to add this technological component to Arnstein's theory, considering new participatory modes such as data manipulation, online platforms, mobile applications and sensors. This supplemented version of the original ladder attests to the new alternatives and specificities of the numeric participation, which oscillates between rather low and rather high influence and decision power of the citizens. Nonetheless, contrary to Arnstein's willingness to reach the upper levels of citizen power [22], our perspective is that every step of the ladder is legitimate (or even complementary), except the therapy and the manipulation, if the citizens are conscious of their role and of the objective of their participation. As one member of the Dutch lab said, participation has to be taken seriously, but we believe that sometimes more modest participatory processes can fill a need, even if the participants remain passive informants for instance. Moreover, time constraints render impossible the ideal scenario, i.e., some kind of persistently, continuous and super-active participation of each participant at each step of the process and in each case. In order to avoid weariness and overload of participants, facilitators and city officials, we suggest to make compromises and choose carefully when a full citizen power is necessary and feasible, this choice becoming thus one of the biggest challenges when implementing a participative process in a Smart City. Following Glass, the chosen methodology (and therefore the associated citizen decision-power) depends on the objective of the Participation (e.g., information exchange,

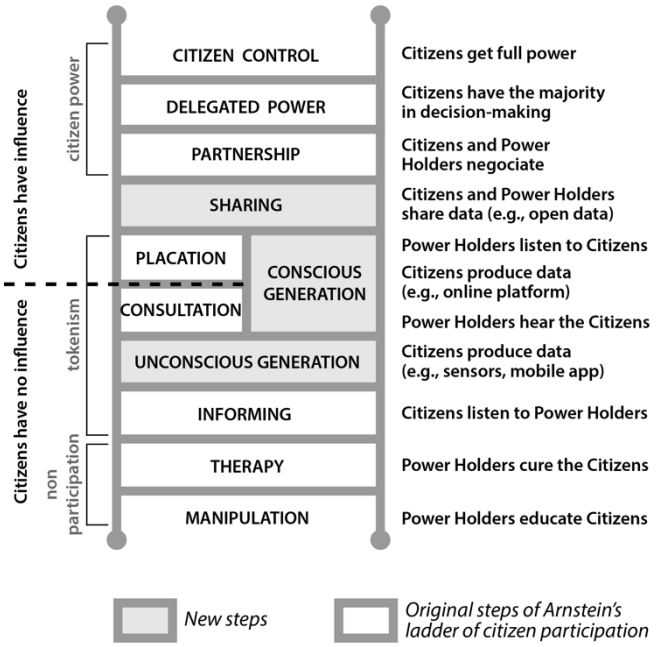


Figure 5. Arnstein's ladder of citizen participation adapted to the Smart City context

representational input, education or decision-making supplement) [23]. From our point of view, this decision should also particularly rely on the object of the participation, which can sometimes require more usage or professional expertise, more local or global perspective, more deep or "automatic" contribution, etc. The projects conducted by the three labs are the proof that several levels of participation deserve to exist and result in different benefits (and drawbacks). One last impact of the digital era on the participatory theories relates to the inclusive dimension of the participative process. While the literature review often envisions the Smart City as an exclusive concept, generating digital divide, the interviewees also mention that technology can increase the participation rate, because there are much more diffusion channels (e.g., social networks) and a better access to information.

The interviewees' interpretations about citizen participation introduce the notion of citizens' motivation, nurturing our third focus point. The results regarding the selection of the citizens show that participants can be characterized by different motivation spectrums: Dutch citizens share the same values while the British participants have more diverse interests. Following Deci, the participants' motivation may have intrinsic or extrinsic sources [38]. In other words, the citizens can respectively decide to participate "*because it is inherently interesting or enjoyable*" or "*because it leads to a separable outcome*" like for instance a reward [38]. In our case, the benefits promoted by the British lab, such as technology exclusivity, time or money savings, might be identified as extrinsic motivations. The Dutch and Spanish participants rather seem to be motivated by intrinsic factors, such as the personal willingness to take part to the life of their community or to collaborate around shared values and interests. According to Amabile's extensive research on the subject, this dichotomy between extrinsic and intrinsic motivations has consequences on the participants' creativity: extrinsic motivations could undermine the intrinsic motivation and the creative outputs, because the subject is not performing for its own sake anymore but rather for an external purpose [39]. Therefore, in our opinion, extrinsically motivated people will maybe more easily grow weary than intrinsically motivated citizens, who will probably commit themselves to participate in the long run. However, in the domain of technologies, participants' remuneration reveals quite decisive, not so much as the primary motivation to participate, but rather as a reinforcement of long-term commitment [40]. Consequently, our third question is: what are the citizens' motivations and what is the potential impact on the participants' long-term involvement within the project? Our point of view is that several sources of motivation are complementary and should be mobilized at different stages of the process. On the one hand, offering stipend from the beginning presents a high risk to participant's creativity [39]. On the other hand, stipends offer the advantage to reach more profiles of citizens and to value their engagement as a real job, which maintains commitment and reduces dropout [40]. Therefore, the recruitment of the citizens should, as far as possible, be based on intrinsic motivations, but some compensation must be considered during the process for long-term participation or when a more general, mixed public is needed.

Another important consequence regarding the selection of the participants is related to the representativeness of the sample. One recurrent wish of the interviewees is to reach everybody, but they agree that this dream scenario is too optimistic. Therefore, the three labs developed their own practical approach. On the one hand, the Dutch lab relies on existing communities, already active and probably prone to participate. On the other hand, the British lab recruits the most motivated citizens from a limited geographical area, based on some kind of "*first come, first served*" rule. Finally, the Spanish lab uses both strategies, depending on the initiator of the participative process and the chosen participatory method. The British lab hopes to get more "*general users*" in the sense that the researchers do not know anything about the selected citizens, nor about their diverse motivations, leaving the possibility to include participants who have reservations about some aspects of the project. Even if the British and the Spanish samples are generally more heterogeneous, none of the three labs insures a representative sample. We should then be aware that each approach provides different target audiences and ask ourselves: how will the participants be selected and what are the consequences on the variety of the citizen profiles and, as a result, on the project outcomes? If none of the extreme situations is optimal, maybe the Spanish adaptive strategy is a good alternative. Indeed, the potential bias of the British and the Dutch approaches, i.e., low citizen motivation versus only-motivated citizens, are reassessed for each project in order to choose the selection criterion that will best support this specific case.

Regarding the benefits of citizen participation, all three labs are truly convinced by the participation contribution to

the making of a Smart City. Their individual interpretations sometimes differ, but they all take root in the same vision of more aware, empowered and knowledgeable citizens. Moreover, they all agree on the importance to mobilize citizen's field perspective, which the professional and official stakeholders are definitely lacking. Contrary to the state of the art, the interviewees never mentioned the professional protectionism [41] or the political alibi [42] as major limits, while those are among the most frequent reasons a participative process might fail to achieve concrete results. Actually, our hypothesis is that our three cases faced their own sets of problems, but also represent three success stories, which would not have been the case if the municipality and the lab were not aware of the benefits of citizen participation. Consequently, before launching a participative process, every stakeholder should wonder: what knowledge or skill can I bring to the others and what can I learn from them? Indeed, the realization that collective intelligence and professional expertise are complementary [43] is the key to build trust and to implement an effective participative process. Following Glass, this efficiency also relies on the chosen participatory technique that has to fit the pursued objective [23]. In order to enhance the impact of the participation, we also believe that the technique has to match the temporal frame of the participation process. For instance, some exploratory methods should not be used too late in the design process, at the risk of generating frustration because the participants' proposals cannot be implemented in an advanced solution or a nearly-finished project. The Dutch and the Spanish labs therefore promote co-design sessions with a citizen engagement as soon as the early phases of the process, while the British lab invites the participants to test some technologies in the late evaluation stages of the process.

VI. Conclusion and Future Work

This paper considers the citizen participation in the Smart City from the operational perspective. Based on interviews with field actors, three Smart Cities' perceptions and participative approaches are compared and confronted with the literature review. The results show that the theoretical definitions of the "Smart City" rather correspond to the interviewees' prospective visions, while their current vision is not that optimistic, especially regarding the role citizens might play. The interviewees' interpretation of the "citizen participation" is close to the existing theoretical models, but enriched by a new dimension related to the technological era, which we call "data manipulation". Regarding the participants' selection, striving to reach every citizen is seen as an un-achievable ideal and all three labs develop their own alternative approach, tapping into existing communities, focusing on a specific geographical area or mixing the two strategies on a case-by-case basis. This choice has a direct impact on the participants' profiles, in terms of interests and motivations, or even creativity and commitment to the project. The perceived benefits of the implementation of citizen participation in a Smart City are not really different from the ones in the literature review, even though a particular emphasis on awareness, empowerment and learning suggests that citizens might gain new skills and knowledge, especially regarding smart technologies. Conversely, the drawbacks reveals that some technological constraints could jeopardize the smart participation in particular, compared to more traditional contexts. The nuanced interviewees' visions highlight key elements that should be kept in mind while implementing a participative approach in the Smart City. Moreover, confronting practical and theoretical perspectives helps us to revise the traditional Arnstein's ladder of citizen participation into an adapted version reflecting the Smart City context. Given the variety of interpretations, further research will explore other case studies nurturing our comparative analysis. Future work will also deepen the citizens' perspective regarding their participation in the Smart City (preferences, barriers and motivations).

Acknowledgment

This research is part of the "Wal-e-Cities" project, funded by the FEDER and the Walloon region. The authors would like to thank the funding authorities, as well as all who contributed to this research, in particular the interviewed lab directors and team members.

References

[1] C. Schelings and C. Elsen, "The 'bottom-up smart city': filling the gap between theory and practice," Proceedings of Smart 2018: The Seventh International Conference on Smart Cities, Systems, Devices and Technologies, IARIA, Barcelona, pp. 54-60, 2018.

[2] T. Monfaredzadeh and R. Krueger, "Investigating social factors of sustainability in a smart city," Procedia Engineering, vol. 118, pp. 1112-1118, 2015.

[3] M. Angelidou, "Smart city policies: a spatial approach," Cities, vol. 41, pp. S3-S11, 2014.

[4] K. Maier, "Citizen participation in planning: climbing a ladder?, " European Planning Studies, vol. 9, no. 6, pp. 707-719, 2001.

[5] S. Ben Letaifa, "How to strategize smart cities: revealing the SMART model," Journal of Business Research, vol. 68, pp. 1414-1419, 2015.

[6] T. Shelton, M. Zook, and A. Wiig, "The 'actually existing smart city'," Cambridge Journal of Regions, Economy and Society, vol. 8, no. 1, pp. 13–25, 2015.

[7] R. Giffinger et al., Smart cities: ranking of European medium-sized cities. Centre of Regional Science, Vienna University of Technology, 2007.

[8] R. G. Hollands, "Will the real smart city please stand up? Intelligent, progressive or entrepreneurial," City, vol. 12, no. 3, pp. 303-320, 2008.

[9] J. Willems, J. Van den Bergh, and S. Viaene, "Smart city projects and citizen participation: the case of London," in Public sector management in a globalized world, R. Andeßner, D. Greiling and R. Vogel, Eds. Wiesbaden: NPO-Management, Springer Gabler, pp. 249-266, 2017.

[10] A.-G. Paetz, E. Dütschke, and W. Fichtner, "Smart homes as a means to sustainable energy consumption: a study of consumer perceptions," Journal of Consumer Policy, vol. 35, pp. 23-41, 2012.

[11] N. Balta-Ozkan, R. Davidson, M. Bicket, and L. Whitmarsh, "Social barriers to the adoption of smart homes," Energy Policy, vol. 63, no. C, pp. 363–374, 2013.

[12] K. Buchana, N. Banks, I. Preston, and R. Russo, "The British public's perception of the UK smart metering initiative: threats and opportunities," Energy Policy, vol. 91, no. C, pp. 87-97, 2016.

[13] R. Bertoldo, M. Poumadère, and L. C. Rodrigues Jr., "When meters start to talk: the public's encounter with smart meters in France," Chemical Physics Letters, vol. 9, pp. 146-156, 2015.

[14] F. Bartiaux, "Does environmental information overcome practice comportmentalisation and change consumers' behaviours?," Journal of Cleaner Production, vol. 16, no. 11, pp. 1170-1180, 2008.

[15] D. Gooch, A. Wolff, G. Kortuem, and R. Brown, "Reimagining the role of citizens in smart city projects," Adjunct Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing, New York, 2015, pp. 1587-1594.

[16] V. Thomas, D. Wang, L. Mullagh, and N. Dunn, "Where's Wally? In search of citizen perspectives on the smart city," Sustainability, vol. 8, no. 3, pp. 207-219, 2016.

[17] L. Leydesdorff and M. Deakin, "The triple-helix model of smart cities: a neo-evolutionary perspective," Journal of Urban Technology, vol. 18, no. 2, pp. 53-63, 2011.

[18] R. P. Dameri, "Comparing smart and digital city: initiatives and strategies in Amsterdam and Genoa. Are they digital and/or smart?," in Smart City, R. P. Dameri and C. Rosenthal-Sabroux, Eds. Cham: Springer, pp. 45-88, 2014.

[19] D. Schuurman, B. Baccarne, L. De Marez, and P. Mechant, "Smart ideas for smart cities: investigating crowdsourcing for generating and selecting ideas for ICT innovation in a city context," Journal of Theoretical and Applied Electronic Commerce Research, vol. 7, no. 3, pp. 49-62, 2012.

[20] A. Vanolo, "Is there anybody out there? The place and role of citizens in tomorrow's smart cities," Futures, vol. 82, pp. 26-36, 2016.

[21] K. Fehér, "Contemporary smart citizs: key issues and best practices," Proceedings of Smart 2018: The Seventh International Conference on Smart Cities, Devices and Technologies, IARIA, Barcelone, pp. 5-10, 2018.

[22] S. R. Arnstein, "A ladder of citizen participation," Journal of the American Institute of Planners, vol. 35, no. 4, pp. 216-224, 1969.

[23] J. J. Glass, "Citizen participation in planning: the relationship between objectives and techniques," Journal of the American Institute of Planners, vol. 45, no. 2, pp. 180-189, 1979.

[24] B. Cohen. *The smartest cities in the world 2015: Methodology.* [Online]. Available from: http://www. fastcoexist. com/3038818/the-smartest-cities-in-the-world-2015-methodology 2019/06/10.

[25] R. P. Dameri, "Searching for smart city definition: a comprehensive proposal," International Journal of Computers & Technology, vol. 11, no. 5, pp. 2544-2551, 2013.

[26] S. P. Hall, "Creative cities and economic development," Urban Studies, vol. 37, no. 4, pp. 639-649, 2000.

[27] H. Chourabi et al., "Understanding smart cities: an integrative framework," Proceedings of the 45th Hawaii International Conference on System Sciences, HICSS, Maui, pp. 2289-2297, 2012.

[28] D. Toppeta, "The smart city vision: how innovation and ICT can build smart, 'livable', sustainable cities," The Innovation Knowledge Foundation, vol. 5, pp. 1-9, 2010.

[29] J. Desdemoustier and N. Crutzen, Smart cities en Belgique: analyse qualitative de 11 projets. Smart City Institute, University of Liège, 2015.

[30] M.-H. Bacqué and M. Gauthier, "Participation, urban planning and urban studies," Participations, vol. 1, no. 1, pp. 36-66, 2011.

[31] R. Luck, "Dialogue in participatory design," Design studies, vol. 24, no. 6, pp. 523-535, 2003.

[32] R. Luck, "Learning to talk to users in participatory design situations," Design Studies, vol. 28, no. 3, pp. 217-242, 2007.

[33] B. Klandermans and D. Oegema, "Potentials, networks, motivations, and barriers: steps towards participation in social movements," American Sociological Review, vol. 52, no. 4, pp. 519-531, 1987.

[34] G. Rowe and L. J. Frewer, "Public participation methods: a framework for evaluation," Science, Technology, & Human Values, vol. 25, no. 1, pp. 3-29, 2000.

[35] P. Kristensson, A. Gustafsson, and T. Archer, "Harnessing the creative potential among users," The Journal of Product Innovation Management, vol. 21, pp. 4-14, 2004.

[36] E. Bjögvinsson, P. Ehn, and P.-A. Hillgren, "Design things and design thinking: contemporary participatory design challenges," DesignIssues, vol. 28, no. 3, pp. 1–16, 2012.

[37] R. Carli, M. Dotoli, R. Pellegrino, and L. Ranieri, "Measuring and managing the smartness of cities: a framework for classifying performance indicators," Proceedings of the 2013 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 1288-1293, 2013.

[38] R. M. Ryan and E. L. Deci, "Intrinsic and extrinsic motivations: classic definitions and new directions," Contemporary Educational Psychology, vol. 25, pp. 54-67, 2000.

[39] T. M. Amabile, "Social environments that kill creativity," in Readings in Innovation, S. S. Gryskiewiz and D. A. Hills, Eds. Greensboro, NC: Center for Creative Leadership, 1992.

[40] D. L. Kleinman, J. A. Delborne, and A. A. Anderson, "Engaging citizens: the high cost of citizen participation in high technology," Public Understanding of Science, vol. 20, no. 2, pp. 221-240, 2011.

[41] J. Hill, Actions of architecture: architects and creative users. Routledge, London, 2003.

[42] R. A. Irvin and J. Stansbury, "Citizen participation in decision making: is it worth the effort?," Public administration review, vol. 64, no. 1, pp. 55-65, 2004.

[43] E. B.-N. Sanders and P. J. Stappers, "Co-creation and the new landscapes of design," CoDesign, vol. 4, no. 1, pp. 5-18, 2008.

A Hybrid Approach for Personalized and Optimized IaaS Services Selection

Hamdi Gabsi*, Rim Drira†, Henda Hajjami Ben Ghezala‡
RIADI Laboratory. National School of Computer Sciences
University of Manouba,
la Manouba, Tunisia
*Email: *hamdi.gabsi@ensi-uma.tn, †rim.drira@ensi-uma.tn, ‡henda.benghezala@ensi-uma.tn*

***Abstract*—Cloud computing offers several service models that change the way applications are developed and deployed. In particular, Infrastructures as a Service (IaaS) has changed application deployment as apart from cost savings, it removes the confines of limited resources' physical locations and enables a faster time-to-market. Actually, a huge number of IaaS providers and services is becoming available with different configuration options including pricing policy, storage capacity, and computing performance. This fact makes the selection of the suitable IaaS provider and the appropriate service configuration time consuming and requiring a high level of expertise. For these reasons, we aim to assist beginner cloud users in making educated decisions and optimized selection with regard to their applications' requirements, their preferences, and their previous experiences. To do so, we propose a hybrid approach merging both Multi-Criteria Decision Making Methods and Recommender Systems for IaaS provider selection and services configuration. Moreover, we propose a service consolidation method to optimize the selection results by improving the resources' consumption and decreasing the total deployment cost. Our solution is implemented in a framework called IaaS Selection Assistant (ISA); its effectiveness is demonstrated through evaluation experiments.**

Keywords- IaaS services selection; Services Consolidation; Cost Optimization; Recommender Systems; Multi-Criteria Decision Making.

I. INTRODUCTION

In this research paper, we propose a hybrid approach for personalized and optimized IaaS services selection based on our previous work [1].

The total market value for public cloud infrastructure services, according to a report from the Analytical Research Cognizance [2], is forecast to reach 775 million dollars by 2019, up from 366 million dollars in 2015. One of the greatest benefits of IaaS platforms is the elasticity of a shared pool of configurable computing resources in response to the user's requirements. With the mature of the IaaS landscape, providers vary notably in terms of the services, features, and pricing models they offer. Due to this diversity, selecting the appropriate IaaS provider becomes a challenging task. In fact, each IaaS provider offers a wide range of services, which must be appropriately selected and correctly configured. This fact leaves users in the agony of choice and leads to a steep documentation curve to compare IaaS providers and their services. Thus, it is crucial to assist cloud users during the selection process.

In this context, several works such as [3]-[4] have shown an interest to address IaaS selection issue. However, these works focused mainly on assisting IaaS services selection based on functional application requirements and Quality of Services (QoS), which we call application profile. Few studies have highlighted the importance of involving the user in the selection process by considering his preferences and his previous experiences, which we call user profile. Consequently, there is a need for a selection process centered on both user and application profiles. Moreover, the lack of a standardized framework for the representation of user requirements and selection criteria makes it difficult to compare and evaluate the relevance of IaaS service configurations offered by different providers. Thus, it is important to define clearly relevant selection criteria that should be taken into consideration to evaluate IaaS services and select the most suitable services.

In our work, the selection process is defined as a two-step strategy. The first step consists in detecting automatically suitable IaaS provider meeting user requirements and preferences. The second step consists in retrieving the suitable IaaS service configuration (Virtual Machine (VM) instance) given a specific application requirement. To do so, we propose a hybrid approach based on Recommender Systems (RS) and Multi-Criteria Decision Making Methods (MCDM).

RS are programs, which provide relevant items (e.g., movies, music, books and products in general) to a given user by predicting his/her interest in items based on his/her profile and the ratings given by other similar profiles [5]-[6]. The first step of our approach is based on recommendation techniques.

Once the suitable IaaS provider is chosen regarding the user's profile, the user needs to be assisted to handle the services selection and configuration. For us, the cloud services selection is a MCDM problem [7]-[8]. MCDM can be defined as a process for identifying items that match the goals and constraints of decision makers with a finite number of decision criteria and alternatives [8]. In our work, we consider IaaS Service selection as a MCDM problem since

users have to select a service amongst several candidates' services with respect to different criteria. We study and choose the adequate MCDM technique to assist IaaS services selection.

After identifying suitable IaaS services, we aim to optimize the application deployment cost and improve the IaaS services consumption. To do so, we propose a service consolidation method using the knapsack algorithm [9].

Therefore, this work aims to assist and optimize IaaS services selection by involving the user in the selection process and by combining RS and MCDM techniques.

The contributions of this paper can be summarized as follows:

- Defining a classification for relevant criteria that should be used during the selection process. These criteria consider both applications profiles including functional and non-functional requirements and user's profile including personal preferences, previous experiences and even lessons learned from experiences of other users.
- Presenting a new hybrid approach based on MCDM and RS techniques for IaaS provider and services selection.
- Proposing a consolidation method to increase the selected services consumption and optimize the application deployment cost.
- Implementing this approach in a framework, which we term ISA for IaaS providers and services selection.

The present work is a comprehensive extension to our previous work [1]. We present three extensions to our initial approach and demonstrate the improved framework ISA that encompasses the enhancements of our IaaS service selection process.

First, we generalize our approach to cover medium and large application profiles, which cannot be satisfied by a single IaaS service. We consider, in this context, a cloud application as a set of deployment entities, each deployment entity presents a particular functional requirement characterized by a specific configuration in terms of CPU, storage, memory and networking capacity and defines several non-functional requirements. In that respect, the user's application can be deployed on several VMs with different configurations. Each VM will be assigned to a particular deployment entity.

Second, handling medium and large applications requires improvements of our proposed selection process in order to reduce the search space and improve the overall response time of our approach while maintaining high precision. For this purpose, we propose a mapping strategy based on the workload type of each deployment entity composing the user's application and the VM configuration families proposed by IaaS providers.

Third, we optimize our selection approach to take into account the scenario where the proposed services (VMs) may be not entirely used. We need to increase the IaaS service consumption while maintaining or decreasing the total application deployment cost. Therefore, we propose a method for service consolidation using the knapsack algorithm to reach this purpose.

The remainder of this paper is organized as follows: Section II presents a motivating scenario. Section III summarizes existing IaaS service selection techniques. Section IV illustrates the proposed cloud services selection criteria. Section V details our hybrid selection approach. Section VI presents and evaluates the framework ISA. Section VII provides concluding remarks and outlines our ongoing works.

II. Motivating scenario

Let us suppose the following scenario where a recently launched company named "A" is planning to develop flexible and innovative customer-centric services to attract new customers and improve its efficiency. In order to provide these services with high efficiency and low maintenance cost, "A" plans to use IaaS services, considering the following reasons:

- Cost reduction: The maintenance cost of dedicated hardware, software, and related manpower in "A" will be highly reduced by using cloud services.
- Improvement in flexibility and scalability: IaaS services enable "A" to respond faster to changing market conditions by dynamically scaling up and down on demand.
- Faster time to market: IaaS services enable "A" to expeditiously dispose its developed services to the market.

To deploy its services, "A" looks for IaaS services. However, most of A's engineers lack expertise in cloud services field to be able to select easily and efficiently appropriate IaaS provider and services. In today's market, there are many IaaS providers. Each provider offers several services varying in QoS attributes with possibly different functional configuration such as numbers of virtual cores and memory size. In order to select appropriate IaaS services among a growing number of available services, "A" tries to compare its applications profiles (functional & non functional requirements) to IaaS providers offers. To do so, the company needs to peruse the content of each provider website and compare service offerings to decide the most suitable IaaS service with regard to its needs. This type of selection process can be more complicated as the company's requirements evolve and diversify.

Therefore, automatic IaaS services selection becomes a highly required necessity in order to entirely take advantages of cloud computing services and improve the efficiency of many companies.

III. Related Work

Several studies have addressed the selection of IaaS services. We present a classification of the recent research approaches.

A. Recommender systems

RS can be defined as programs, which attempt to recommend suitable items to particular users by predicting a user's interest in items based on related information about the users, the items and the interactions between them [5]. Generally, RS use data mining techniques to generate meaningful suggestions taking into account user's preferences. Many different approaches using RS have been developed to deal with the problem of cloud services selection.

Zhang et al. [10] have offered a cloud recommender system for selecting IaaS services. Based on the user's technical requirements, the system recommends suitable cloud services. The matching between technical requirements and cloud services features is based on a cloud ontology. The proposed system uses a visual programming language (widgets) to enable cloud service selection.

Zain et al. [6] propose an unsupervised machine learning technique in order to discover cloud services. The authors classify cloud services into different clusters based on their QoS. The main focus of this study is to offer users the option of choosing a cloud service based on their QoS requirements.

B. MCDM-based approaches for cloud service selection

The MCDM approach is defined as a process for specifying items that best fit the goals and constraints of decision makers with a finite number of decision criteria and alternatives [11]. Several MCDM methods are used for cloud service selection such as the analytic hierarchy process/analytic network process (AHP/ANP) [12], Multi-Attribute Utility Theory (MAUT) [13], and Simple Additive Weighting (SAW) [11].

Chung et al. [14] used the ANP for service selection. They suggest a set of high level criteria for cloud service selection and use a survey of CIO, CEO, and ICT experts to determine the importance of each criterion.

Lee et al. [15] proposed a hybrid MCDM model focused on IaaS service selection for firms' users that are based on balanced scorecard (BSC), fuzzy Delphi method (FDM) and fuzzy AHP. BSC is used to prepare a list of decision making factors. FDM is used to select the list of an important decision-making factors based on the decision makers' opinion (using a questionnaire) and FAHP is used to rank and select the best cloud service. This work's focus is on the migration of the whole company ICT to cloud based on a set of general cloud service features.

Zia et al. [8] propose a methodology for multi-criteria cloud service selection based on cost and performance criteria. The authors present this selection problem in a generalized and abstract mathematical form. Table I illustrates the mathematical form. The service selection process is fundamentally a comparison between the vector service descriptor D against all rows of the decision matrix followed by the selection of the services whose description vector best matches with the user's requirement vector.

Table I. Problem formalization [8]

Mathematical form	Description
Services set	S_1, S_2, ..., S_n A set of services contains all the service offerings from, which the user (decision maker) will select the suitable service with regard to his requirements. a service is to be selected by the user (decision maker).
Performance criteria set	C_1, C_2,..., C_n A set of values where C_i represents a criterion that may be a useful parameter for service selection.
Performance measurement functions set	To each criteria C_i there corresponds a unique function f_i, which when applied to a particular service, returns a value p_i that is an assessment of its performance on a predefined scale.
Service descriptor (vector)	A row vector D_i that describes a service S_i, where each element d_j of D_i represents the performance or assessment of service S_i under criteria C_j. Performance criteria must be normalized to eliminate computational problems resulting from dissimilarity in measurement units. The normalization procedure is used to obtain dimensionless units that are comparable.
Decision matrix	The service descriptor vectors D_i can be combined to form the decision matrix where each value is the evaluation of the service s_i against the criteria c_j.
User requirement criteria vector	A vector R where each value r_i is the user's minimal requirement against a criteria c_j. These values must be normalized as the vector service descriptor.
User priority weights vector	A vector W where each value w_i is the weight assigned by a user to criteria. c_i

Table II summarizes the most used approaches by identifying the approach's input, the approach's output and the application areas.

The above-mentioned research studies did not fail to take into consideration the application's functional requirements. However, they present two main shortcomings; (i) they do not accommodate the user' preferences in the decision making and (ii) they handle every application deployment as a new case without taking into account the results of similar previous experiences.

TABLE II. SELECTION APPROACHES

Domain	Method	Input	Output	Application	Literature
Multi-criteria decision-making (MCDM)	SAW	Subjective assessment of relative importance of criteria.	Evaluation value of alternatives.	Applied when requiring low decision accuracy.	[11][8][16]
Multi-criteria optimization	Matrix factorization	Different types of data of interest to users and represented by matrix .	QoS estimation and a set of recommended services.	Applied to a problem that involves different types of data and has missing entries.	[17][18]
Logic based matching approach	First-order logic	Service description and user requirements.	Matched services	Applied to filter out unmatched services to reduce computation complexity.	[11][19]
Recommender System	Collaborative filtering	User's profile	Recommended items	Applied to find personalized recommendations according to user's profile.	[4][10][20]

To the best of our knowledge, no specific research study has taken into account both the user's profile and the application's requirements. Consequently, there is a need for a structured selection process where clearly both selection criteria are defined and used.

IV. CLOUD SERVICES SELECTION CRITERIA

Specifying clear selection criteria presents crucial importance in order to recommend the relevant IaaS services. Our purpose is to clearly identify these criteria and take them into account to personalize the selection process according to the user's profile and respond to his application requirements. Thus, we classify selection criteria into three categories. The first category is the application's profile, which includes functional and non- functional requirements. The second category is the user's profile, which represents user's personal preferences and previous experiences. The third category is the previous experiences of other users with their ratings. Figure 1 illustrates our proposed selection criteria.

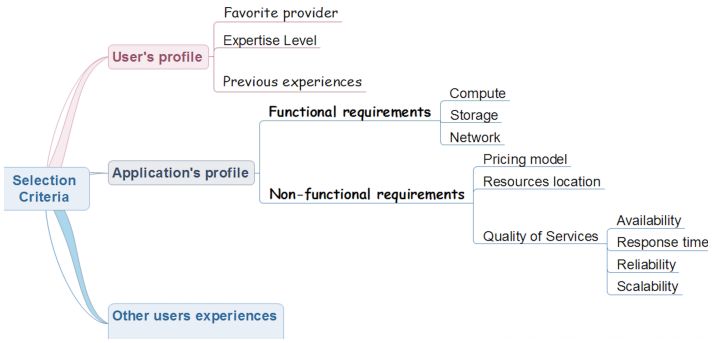


Figure 1. Selection Criteria

As shown in Figure 1, the selection criteria are classified as the following:

- **Application's profile:** the application's profile defines the functional and non-functional application requirements.
  In our context, we consider that a cloud application is a set of deployment entities each deployment entity has specific functional and non-functional requirements. We define the application profile as a set of all deployment entities' requirements. The functional requirements contain the following specifications:
  - Storage: represents storage needs in terms of memory space.
  - Network: represents connection needs and network usage.
  - Compute: gathers calculation needs and the virtual machine's capacity.

  Non-functional requirements include pricing models, the quality of services (QoS) and the resources location.
  - The pricing model: depends on the user's estimated budget. The pricing model can be evaluated per hour or per month. Also, it can be on demand, reserved or bidding.
  - QoS: we focus on the response time and availability. The availability is the time ratio when the service is functional to the total time it is required or expected to function in.
  - Resources location: The user can precise his nearest resources location because it is important to take into account the proximity when selecting the cloud infrastructure services. According to [19], during the interaction between the users and servers, there is a strong inverse correlation between network distance and bandwidth. Thus, factoring the proximity into the selection of IaaS services can significantly reduce the client's response time and increase the network bandwidth.
- **User's profile:** it includes user's favorite providers, expertise level in cloud and previous experiences. A favorite provider can be chosen based on previous successful experiences using this provider. We take this choice into consideration while identifying the appropriate cloud provider meeting user's requirements. In our case, the user can specify one or multiple favorite providers. The user's expertise level can be: beginner, intermediate or expert. The weight of a user's previous

experience in our knowledge base increase with his level of expertise and experience in order to enhance our recommendations relevance. A previous experience contains the selected IaaS provider, the deployed application profile and a rating out of 5 presenting feedback and an evaluation of this experience. We suppose that evaluating ratings are trustworthy and objective.

- **Previous users experiences:** The more the knowledge base of our recommender system is rich, the more recommendations will be relevant. Therefore, previous users experiences, which include the deployed application's profile, the selected IaaS provider and the evaluating rating will improve the accuracy of our recommendations.

Based on the selection criteria, more precisely the application profile, we propose to optimize the search space. Indeed, we suppose that each deployment entity is characterized by a specific workload type. According to Singh et al. [21], cloud workloads can be defined based on four main low-level measurable metrics that when adjusted can affect the workloads' performance. These metrics are the CPU cores, the memory size (RAM), the networking capacity and the storage size, which present respectively the compute, the network and the storage requirements.

We propose a mapping between the workload type of the application deployment entities and the VM configuration families proposed by the IaaS providers. The above-mentioned metrics will be used as a high-level interface that maps the workload type onto a set of candidate IaaS services. Indeed, the workload type can be automatically extracted based on the weight assigned to each metric, for instance, computation-intensive workload is characterized by a higher weight for the CPU metric. In the case that the weights given by the users are equal or insignificantly different, the workload type is defined as general. Thus, workload types are easily identifiable. If we can manage to map these workloads type onto specific categories of IaaS services, then the service selection will become more efficient by decreasing the search space and improving the overall response time. For this purpose, our mapping strategy is based on identifying IaaS service categories disposed by cloud provider, then, establishing the relation between the service categories and the workload type.

Cloud providers dispose IaaS services in different categories with various configurations in terms of CPU, storage, memory and networking capacity. We conduct that most cloud providers classify their services into the following categories based on VM configurations: compute optimized, memory optimized, storage optimized and general purpose.

These categories are identified to offer better performance with respect to a specific workload types (such as computation-intensive or memory-intensive). Thus, It is obvious that the relation between the workload type and the IaaS services configurations are based on the service category. More precisely, the computation-intensive workload type is mapped to IaaS services of the compute optimized category, the memory-intensive workload type is mapped to the memory optimized category, the storage-intensive workload type is mapped to the storage optimized category, and general workload type is mapped to general purpose category.

V. Hybrid approach for IaaS services selection based on RS & MCDM

The selection of IaaS provider and services configuration is a complex issue. To tackle this issue, we propose a two steps selection process. The first step focuses on selecting the IaaS provider based on RS approach, which is the collaborative filtering. The purpose of this step is to reduce the number of inappropriate IaaS provider, which may not interest the user. The second step concerns the configuration of services within the selected provider from the first step. It's based on the SAW algorithm, which is a MCDM method. Our proposed approach shows how MCDM techniques and RS are complementary in order to involve both technical and personal aspects in the selection process.

Figure 2 illustrates our proposed approach.

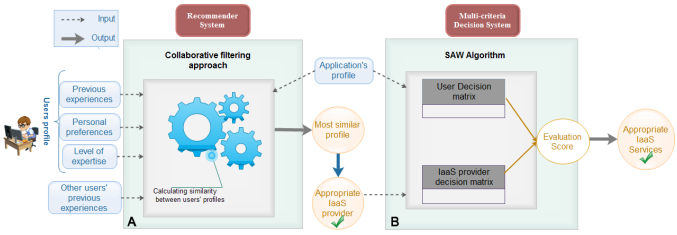


Figure 2. Hybrid approach for IaaS services selection

A. Recommender System

The first step aims to take into consideration the user's preferences, previous experiences and expertise level during the selection process. In our approach, we use the collaborative filtering algorithm also known as k-NN collaborative filtering. This recommendation algorithm bases its predictions on previous users experiences and their profiles. The main assumption behind this method is that other users ratings can be selected and aggregated so that a reasonable prediction of the active user's preferences is deduced.

To recommend the IaaS provider meeting the user's profile, first, we select the users profiles, which have the same or higher expertise level than the active user "A". For instance, if "A" has the expertise level intermediate, then, from our knowledge base, we select a first list named "list 1" of users profiles, which are intermediate or expert and their rated experiences.

Second, among the high rated previous experiences of "list 1", we select those, which are based on the favorite providers of "A" in order to create a second list named "list 2".

Third, among these experiences, "A" can refine "list 2" by identifying experiences that have similar workload types to his application's profile workload. We obtain "list 3". Indeed, we aim by these three steps verifying if "A" favorite providers can be suitable for "A" application profile. Otherwise, we skip the second step to apply the third step on "list 1".

Then, a rating $R_{(A,f_i)}$ is calculated for each one of candidate providers f_i of list3. $R_{(A,f_i)}$ is calculated as below:

$$R_{(A,f_i)} = \frac{\sum_{j=1}^{n} w_{(A,j)}(v_{j,f_i} - \overline{v_j})}{\sum_{j=1}^{n} |w_{(A,j)}|}$$

where n is the number of identified users' profiles of "list 3", $w_{(A,j)}$ is the similarity between the profile of "A" and the identified users profiles j of "list 3", v_{j,f_i} is the rate given by the user j to the provider f_i, $\overline{v_j}$ is the rating's average given by the user j to the favorites providers of "A". We calculate similarity between "A" and the identified users using cosine similarity.

$$w_{(A,j)} = \frac{\sum_{k=1}^{n} v_{A,k} * v_{j,k}}{\sqrt{\sum_{k=1}^{n} v_{A,k}^2 \sum_{k=1}^{n} v_{j,k}^2}},$$

where the sum on k is the set of providers for which "A" and the selected users in" list 3" both assigned a rating, $v_{j,k}$ is the rate given by the user j to the provider k.

Finally, we propose to "A", the set of providers sorted according to the rate calculated, thus the active user can select one provider.

B. Multi-Criteria Decision Making Selecting the Cloud Instances

Once the IaaS provider is selected, the second step consists in determining the suitable IaaS service for each deployment entity.

Several and conflicting criteria have to be taken into account when making a service selection decision. No single service exceeds all other services in all criteria but each service may be better in terms of some of the criteria. Since users have to decide which service to select amongst several candidates services with respect to different criteria, we consider IaaS Service selection as a MCDM problem.

Among MCDM methods, we use the SAW method also known as weighted linear combination or scoring methods. It is based on the weighted average of different criteria.

In our case, the number of service configuration components such as CPU cores and memory size scale linearly in most services configurations. Hence, a linear model is suitable for this kind of problem. The basic assumption being that there is a correlation of identity between real-world cloud instance performance and the underlying low-level specification of the hardware, which is specified on the cloud providers websites. Hence, we want to map the performance of the IaaS service to the right deployment entity using a simple linear model. The purpose of using SAW method in our approach is to respond exactly to the application's profile.

To do so, first, the user introduces functional requirements for each deployment entity; compute requirements (e.g., virtual Central Processing Unit (vCPU)), memory requirements (e.g., RAM size) storage requirements (e.g., hard drive's size), network requirements (e.g., throughput and bandwidth).

Second, for each specified requirement the user assignes a particular weight presenting its importance.

Third, based on the weight assigned to each requirement the workload type of the deployment entity is deducted and a set of candidate services is identified. The user inserts the QoS required (e.g., response time and availability) and the pricing model.

To be able to apply the SAW algorithm, we need to formalize our decision problem. For that, we define a decision matrix related to the user. In parallel an analogous decision matrix is defined for the IaaS provider selected in the first step. The decision matrix is a combination of service descriptor vectors. Each service descriptor vector represents the performance of a service under a particular criterion. These criteria represent functional and non-functional requirements for the user. Table III demonstrates an extract form of the decision matrix related to Azure Microsoft [22].

TABLE III. EXTRACT OF DECISION MATRIX FOR MICROSOFT AZURE (VIRTUAL MACHINE)

Service	VCPU	RAM	Hard Drive's size	Cost
A0	1	0.75 GB	19 GB	$0.02/h
A1	1	1.75 GB	224 GB	$0.08/h
A2	2	3.5 GB	489 GB	$0.16/h
A3	4	7 GB	999 GB	$0.32/h
A4	8	14 GB	2039 GB	$0.64/h
A5	2	14 GB	489 GB	$0.35/h
A6	4	28 GB	999 GB	$0.71/h

The SAW algorithm is based on the calculation of one score to each alternative (an alternative in our case is an IaaS service offered by the selected IaaS provider). According to the following SAW formula, the alternative score is calculated as (A_i)=$\sum w_j v_{ij}$, where w_j is the alternative's weight i according to criterion j and v_{ij} its performance. The alternative with the highest score will be suggested. By applying this formula, the recommended IaaS service will automatically be the most performing service, because

it has the highest performing values in the decision matrix (highest number of vCPU, largest hard drive's size, highest cost, etc.). However, this does not entirely meet the user's requirements, because, he/she must not necessarily select the most performing IaaS service, which will evidently have the highest cost. Whereas, he/she should select the service, which meets exactly his/her requirements in order to pay the minimum possible cost. To solve this, we proceed as follows:

- First, we create a decision matrix representing each deployment entity's functional and non-functional requirements. Then, we determine for each service descriptor vector, the absolute value of the difference between its criteria performance and those of the service descriptor vector related to the IaaS provider. In this way, we will have significant values. In fact, low criteria values mean that they accurately match the user's requirements.
- Second, we calculate the score for each alternative using SAW algorithm. Yet, to be able to do so, we need to modify each criterion's weight to get significant results. Indeed, we have previously mentioned that a low criterion's value means that it may interest the user, if this criterion has a high weight, the multiplication of its weight by its value gives a low score. Therefore, this alternative will be considered as unimportant, yet this is not the case. To solve this problem we take the dual of each weight, meaning that, the subtraction of 1 by the weight's value given by the user. Then we normalize each weigh by dividing on the sum of the weights. Thus, we ensure that the weight values are between 0 and 1 and the sum is always equal to 1. Consequently, one low weight value indicates major importance of a given criterion. Therefore, we can calculate the score for each alternative using the SAW algorithm. The most relevant alternative (IaaS service) will incontrovertibly have the lowest score.

To illustrate this, we propose our personalized SAW algorithm 1. We suppose that the user has introduced his/her decision matrix $UserMat[i][j]$ as well as the weights of each criterion $Weight[j]$. In addition, we suppose that we have the decision matrix $ProvMat[i][j]$ containing IaaS services offered by the IaaS provider. In the decision matrix $UserMat$, $UserMat[i][j]$ represents the IaaS service i under the criterion j.

$$UserMat = \begin{bmatrix} u_{00} & \dots & u_{0n} \\ \vdots & \ddots & \vdots \\ u_{n0} & \dots & u_{nm} \end{bmatrix}$$

The personalized SAW algorithm gives as output, the index i representing the adequate cloud service i in the decision matrix.

Algorithm 1 Personalized SAW Algorithm

Require: $Weight[i] \neq 0$
  $Min = 0$
  for $int\ i\ from\ 0\ to\ n$ **do**
    for $int\ j\ from\ 0\ to\ n$ **do**
      $Sub[i][j] = abs(ProvMat[i][j] - UserMat[i][j])$
    end for
  end for
  for $int\ j\ from\ 0\ to\ m$ **do**
    $DualWeight[j] = 1 - Weight[j]$
    $Normalize(DualWeight[j])$
  end for
  for $int\ i\ from\ 0\ to\ n$ **do**
    $Score[i] = 0$
    for $int\ j\ from\ 0\ to\ m$ **do**
      $Score[i] = Score[i] + Sub[i][j] * DualWeight[j]$
    end for
  end for
  for $int\ i\ from\ 0\ to\ n$ **do**
    if $Score[i] < Min$ **then**
      $Min \leftarrow Score[i]$
      $Index \leftarrow i$
    end if
  end for
  return i

C. Service consolidation

Identifying suitable IaaS services (i.e., VMs in our case) for each deployment entity does not ensure that the VM will be entirely used. In a typical scenario, the selected VM is underutilized [23]. To increase the resource utilization, we aim to integrate as many deployment entities as possible to be assigned to each selected service, thus decreasing the number of required services for application deployment. The final configuration must support all the requirements of the application and the preferences of the user with respect to the service performance and price.

To do so, we proceed as follows; first, we start with the largest service S_L, which has the highest performance in the list of proposed services. We use the price as an indicator of service capacity. Second, we accommodate as many deployment entities as possible in this service with respect to its performance (i.e., the service performance can respond to the added deployment entities). Third, we upgrade the service by choosing the next higher performance of the VM instance of the same family as the service S_L, then we consolidate more deployment entities in the service. If the new service's configuration (i.e., the upgraded service) has an equal or lower price than the earlier configuration of all consolidated services, the upgrade is positive and acceptable. We continue the same process for the remaining deployment entities of the application.

To consolidate deployment entities in a service, we cast consolidation into the optimization knapsack problem [9]. Indeed, the knapsack problem is a combinatorial optimization problem. Given a set of items, each item has a weight and a value, the knapsack problem consists in identifying the number of each item to include in a collection so that the total weight is less than or equal to a given limit and the total value is as large as possible.

First, let us formalize the knapsack's problem in our context:

- The knapsack is the largest service, which is not entirely used
- The items are the deployment entities
- The weight is the cost of each single service assigned to a deployment entity

Second, to solve the knapsack problem and handle the challenge of consolidating multiple deployment entities into services, a greedy approximation algorithm [9] is used. The greedy algorithm is an algorithmic paradigm that follows the problem solving heuristic of making the locally optimal choice at each stage with the intent of finding a global optimum. It iteratively makes one greedy choice after another, which reduce each given problem into a smaller one and approximate a globally optimal solution in a reasonable amount of time. In our case, the greedy choice consiste in selecting in each iteration the largest deployment entity among the non-integrated entities.

We detailed our consolidation approach in Algorithm 2. Service consolidation has advantages and disadvantages. Consolidating deployment entities can reduce the network overhead and increases the application's performance. However, service consolidation can cause several challenges related to fault tolerance.

VI. ISA: A Framework for IaaS Selection Assistant

We conduct a set of experiments to evaluate the efficiency of our proposed approach. To do so, we develop the framework ISA by extending our previous framework. In our previous work [1], we suppose that the application profile can be satisfied by just one VM. In this evaluation, we assume that an application profile may require more than one VM. The main purpose through this evaluation is, firstly, to demonstrate that the idea of merging RS and MCDM techniques in a structured approach based on two well defined steps as explained in Section V, provides satisfactory results for several application types (i.e., medium and large applications). Secondly, we aim to validate that our approach proves to be efficient rather than using RS and MCDM techniques each independently.

The framework ISA has been designed to support different IaaS providers such as Amazon, Google and Azure

Algorithm 2 Services Consolidation Algorithm

Input: DT Set of application deployment entities
SD Set of single services assigned to each deployment entity
Initial_Application_Price
Output: Updated_deployment_entities (After consolidation)
Updated_services (After consolidation)
Begin
$Consolidation\_cost \leftarrow$ Initial_Deployment _cost
$i, j \leftarrow 0$
$S_L \leftarrow S_k$, where S_k is the largest in SD)
Update (SD) : $SD \leftarrow SD$ - $\{S_L\}$
Update (DT) : $DT \leftarrow DT$ - $\{Entity_i\}$, where $Entity_i$ is the deployment entity performed by the service S_L
while $(\neg Empty(SD)) \vee (i \leq nb\_services)$ **do**
  while $(\neg Empty(DT)) \vee (j \leq nb\_entities)$ **do**
    Select the largest entity $Entity_L$
    if S_L performance respond to $Entity_L$ **then**
      $Consolidate$ ($Entity_L$, S_L)
      $Update$ (SD)
      $Update$ (DT)
    else
      $S'_L \leftarrow$ Upgrade (S_L)
      Calculate_New_cost
      if $New\_cost \leq Consolidation\_cost$ **then**
        $Consolidation\_cost \leftarrow New\_cost$
        $Consolidate$ ($Entity_L$, S'_L)
        $Update$ (SD)
        $Update$ (DT)
      end if
    end if
    $Entity_L \leftarrow Entity_{L+1}$ {Next_ Largest_entity $\in DT$}
    $j \leftarrow j + 1$
  end while
  $S_L \leftarrow S_{L+1}$ {Next_ Largest_service $\in SD$}
  $Update$ (SD)
  $Update$ (DT)
  $i \leftarrow i + 1$
end while
return SD, DT
End

Microsoft. It aims to guide users step by step in the selection process and propose relevant services.

For this evaluation, we have used Eclipse Modeling Framework, Java Platform Enterprise Edition (JEE) and Mahout eclipse framework [24]. We conduct experiments on 20 real users (PhD students).

We define the experiments' conditions as follows:

- Supported IaaS provider: Amazon, Google, Microsoft Azure
- Number of users: 20
- Number of items (IaaS services): 45
- Active user's profile:
  - Favorite provider: Amazon
  - Expertise level: Beginner
  - Previous experiences: 0
- Active user's application profile: It is defined in Table IV
- The non-functional requirements are defined as follows:
  - QoS: QoS is defined in Table IV
  - Pricing model: Per hour
  - Resource Location: US regions (e.g., US-West, US-East, etc.)

According to the weights given by the user, we assign for each deployment entity the appropriate workload type. Table V illustrates the assigned workload types.

TABLE V. WORKLOAD MAPPING

Deployment Entities	Workload Type
E1	General Purpose
E2	Storage Optimized
E3	General Purpose
E4	Compute Optimized

We define in Table VI the decision matrix "$ProvMat[][]$" used by the personalized SAW algorithm of our approach. For the sake of brevity, we present in Table VI six configuration models of Virtual Machines instances provided by Amazon [25]. Each value in Table VI is verified and identified from cloud provider's official web site. We carry out simulations and evaluations from two steps.

The first step consists on evaluating the effectiveness of ISA using the recall (R), the precision (P), the Top-k precision (P_k) and the R- precision (P_r) metrics. In this context, the precision evaluates the capability of the our framework to retrieve top-ranked IaaS services that are most relevant to the user need, and it is defined to be the percentage of the retrieved IaaS services that are truly relevant to the users requirements. The recall evaluates capability of the system to get all the relevant services. It is defined as the percentage of the services that are relevant to the user requirements.

Formally, we have;

$$P = \frac{|S_{Rel}|}{|S_{Ret}|} \qquad R = \frac{|S_{Rel}|}{|Rel|}$$

$$P_k = \frac{|S_{Rel,k}|}{k} \qquad P_r = P_{|Rel|} = \frac{|S_{Rel,Rel}|}{|Rel|}$$

where Rel denotes the set of relevant IaaS services, S_{Ret} is the set of retuned services, S_{Rel} is the set of of returned relevant services and $S_{Rel,k}$ is the set of relevant services in the top k returned services. Among the above metrics, P_r is considered to most precisely capture the precision and ranking quality of the framework. We also plotted the recall/precision curve (R-P curve). An ideal selection framework has a horizontal curve with a high precision value; an inappropriate framework has a horizontal curve with a low precision value. The R-P curve is considered by the (Information Retrieval) IR community as the most informative graph showing the effectiveness of a selection framework [26].

We evaluated the precision of the retrieved services for each deployment entity, and report the average Top-2 and Top-5 precision. To ensure the top-5 precision is meaningful, we ensure that ISA returns a total of 20 services per application profile. The Figure 3 illustrates the results. The top-2 and top-5 of ISA for the deployment entities E1, E2, E3 and E4 are respectively 98% for the Top-2 retrieved services and 80%, 60%, 80%, 80% for the Top-5 retrieved services. In order to interpret our results and illustrate the overall performance of ISA, we plot the average R-P curves for different applications profiles. As mentioned previously, a good selection framework has a horizontal curve with a high precision value. Typically, precision and recall are inversely related, ie. as precision increases, recall falls and vice-versa. A balance between these two needs to be achieved by a selection framework. As illustrated by the Figure 4, for a recall average equals to 0.68 we have 0.87 as precision average value. In fact, as an example, for the active user's application profile defined in Table IV, ISA returns a total of 20 services i.e., $|S_{Ret}| = 20$, for each deployment entity, we have the following precision values; $\frac{18}{20}, \frac{17}{20}, \frac{17}{20}, \frac{18}{20}$. We obtain a precision average $P = 0.87$. As a recall value, we have, for each deployment entity, $\frac{18}{25}, \frac{17}{26}, \frac{17}{25}, \frac{18}{26}$, we obtain a recall average $R = 0.68$.

It is worth pointing out that in some cases, depending on particular requirements, a high precision at the cost of recall or high recall with lower precision can be chosen. Thus evaluating a selection framework must be related to the purpose of the selection and the search process. In our case a compromise between the recall and the precision values is necessary. Therefore, we can announce that ISA provides accurate results for IaaS services selection.

TABLE IV. APPLICATIONS PROFILES

Application profile								
Functional requirements							QoS	
Deployment Entities	Compute			Storage	Network		Response time	Availability
	vCPU	CPU events/s	RAM	Hard drive's size	Bandwidth	Throughput		
E 1 Weights	0.25		0.3	0.25	0.2		0.5	0.5
E 1 Values	2	$6< v \leq 12$	8	60	4	-	$v \leq 900$	90%
E 2 Weights	0.2		0.2	0.5	0.1		0.7	0.3
E 2 Values	2		10	400	2	42	≤ 900	95%
E 3 Weights	0.25		0.25	0.25	0.25		0.5	0.5
E 3 Values	2	$10 < v \leq 20$	8	30	6	-	≤ 900	95%
E 4 Weights	0.5		0.3	0.1	0.1		0.8	0.2
E 4 Values	16	$50 < v \leq 80$	32	300	10	60	700	95%

TABLE VI. AMAZON DECISION MATRIX [25]

Model	Family	vCPU	CPU Cred-its/hr	RAM GB	Hard drive GB	Bandwidth Gbit s^{-1}	Throughput Mbit s^{-1}	Price h^{-1}	Response time ms	Availability
t2.nano	General purpose	1	3	0.5	30	-	4	$0.0058	63	99%
c5d.4xlarge	Compute optimized	16	81	32	400	5.5	435.7	$0.768	22	99%
m5a.large	General purpose	2	36	8	30	3.12	256	$0.086	53	99%
t2.large	General purpose	2	36	8	30	-	42	$0.0928	50	99%
i3.large	Storage optimized	2	54	19.25	475	-	53.13	$0.156	42	99%
c5d.xlarge	Compute optimized	4	54	8	100	3.5	437.5	$0.192	31	99%

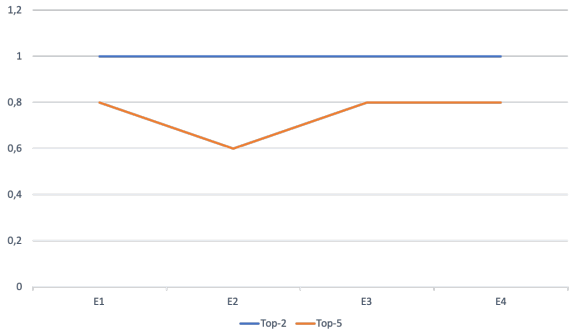


Figure 3. Top-k precision for retrieved services

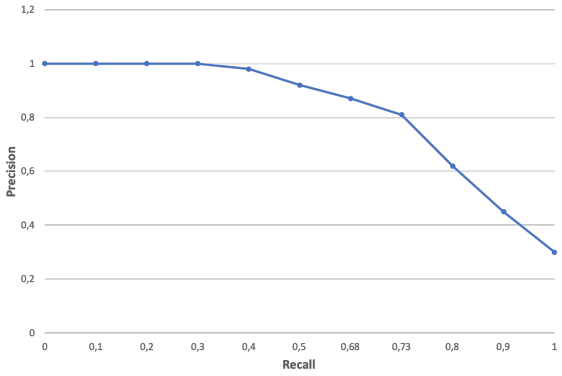


Figure 4. R-P Curves of ISA

The second step of our evaluation consist on comparing our framework to classic RS based on CF technique. Although the number of users and items is relatively small compared to commercial RS, it proves to be sufficient for the purpose of these experiments. For each deployment entity, we present the predicted ratings for each deployment entity described in Table IV.

As illustrated in Figures 5,6, 7, and 8 the highest predicted ratings given by our approach to the deployment entities E_1, E_2, E_3 and E_4 are, respectively, 0.8379, 0.8979, 0.9039, 0.9798. The recommended IaaS services are, respectively, m5d.large i3.large, m5a.large and c5d.2xlarge. For clarity and visibility purposes, we did not display all instances' predicted ratings of Tables III and VI.

The metrics used to evaluate our approach are the Root-Mean Square Error (RMSE) and The Normalized Discounted Cumulative Gain (NDCG).

The RMSE is a metric widely used to evaluate predicted ratings [27]. It represents the sample standard deviation of the differences between predicted values and expected values. RMSE is the square root of the average of squared

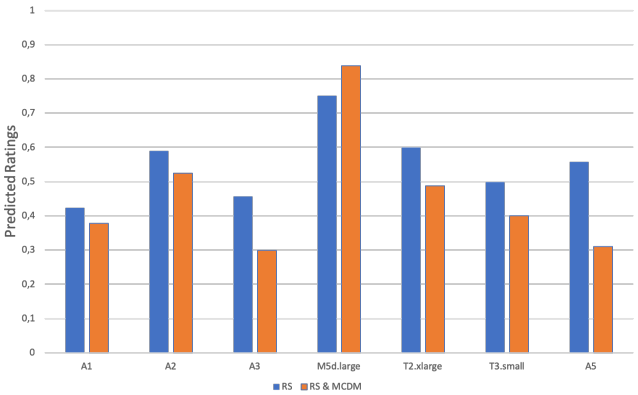


Figure 5. Predicted ratings for the deployment entity E_1

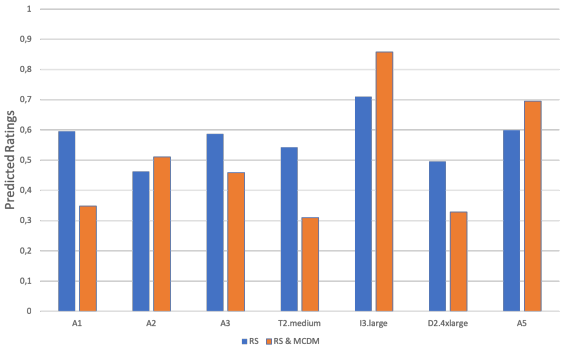


Figure 6. Predicted ratings for the deployment entity E_2

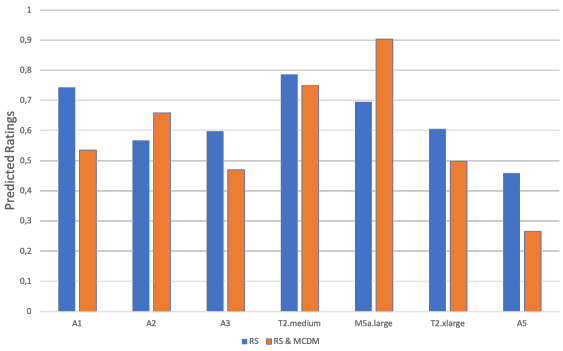


Figure 7. Predicted ratings for the deployment entity E_3

errors.

$$RMSE = \frac{\sqrt{\sum_{i=1}^{n}(p_{A,i} - \widehat{p}_{A,i})^2}}{N}$$

where $p_(A, i)$ is a predicted value by user "A" for item i, $\widehat{p}_{A,i}$ is the expected value of user "A" for item i, and N is the number of predicted values. In order to be able to calculate RMSE values, we assume that users introduce their expected rating values.

The Normalized Discounted Cumulative Gain (NDCG) is

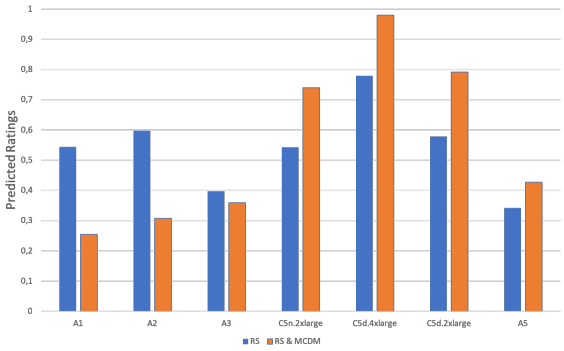


Figure 8. Predicted ratings for the deployment entity E_4

a measure of ranking quality. NDCG is defined as

$$NDCG_N = \frac{DCG_N}{IDCG_N}$$

where DCG_N and $IDCG_N$ are the Discounted Cumulative Gain (DCG) of top-N items of a predicted ranking and the ideal ranking, respectively. DCG_N is calculated by

$$DCG_N = \sum_{i=1}^{N} \frac{2^{(rel_i)} - 1}{log_2(i+1)}$$

where rel_i is the value of the item at position i of a ranking and $IDCG_N$ is calculated by

$$IDCG_N = \sum_{i=1}^{REL} \frac{2^{(rel_i)} - 1}{log_2(i+1)}$$

where REL represents the list of relevant items (ratings ≥ 0.5). The value of NDCG is between 0 and 1. where a larger value means a better ranking, and 1 implies the ideal ranking.

We illustrate the result of comparing the CF technique to our work in Table VII.

TABLE VII. RMSE & NDCG AVERAGE

Deployment Entities	RS		RS & MCDM	
	RMSE	NDCG	RMSE	NDCG
E1	0.041	0.571	0.032	0.71
E2	0.052	0.43	0.034	0.81
E3	0.033	0.62	0.038	0.76
E4	0.045	0.65	0.031	0.79
Average	0.04275	0.567	0.033	0.767

When conducting the CF approach, we obtained respectively 0.04275 and 0.567 as RMSE and NDCG average. However, the RS & MCDM approach gave us 0.033 and 0.767 as RMSE and NDCG average as illustrated in Figure 9. So, in terms of RMSE (i.e., 0.04275 vs. 0.033), the merging of MCDM & RS performs better than RS only. In terms of NDCG (i.e., 0.567 vs. 0.767), RS & MCDM present better result than the CF approach.

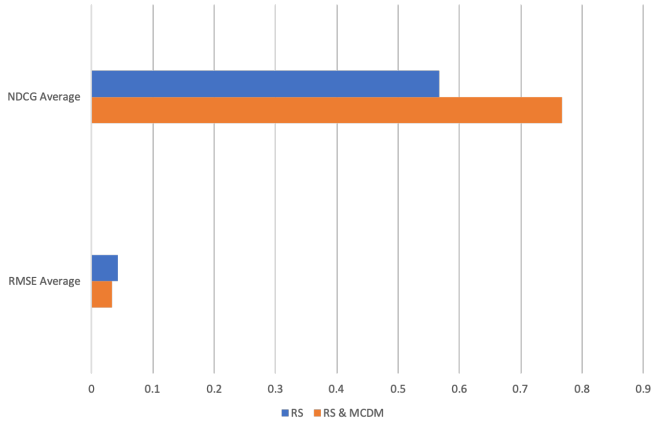


Figure 9. RMSE & NDCG Average

It is worth pointing that the use of CF algorithm only conducts to calculate predicted ratings for all items in our knowledge base, which can be time consuming. However, by applying the step one of our approach we can reduce the number of candidate services by providing only services related to the selected IaaS provider. In addition, the selection of IaaS services using CF algorithm will be associated with previous users experiences in our knowledge base. Although we identify the most similar users, their application profiles must be more or less different to the active user application profile. Consequently, the predicted IaaS services are less accurate. In conclusion, these experiments show that our approach performs better than using RS only.

After identifying suitable IaaS services, we aim to optimize the application deployment cost. To do so, we apply our consolidation algorithm to integrate potential services. It is worth pointing that the cost of the recommended services is estimated to 2.123 $ per hour ($0.113\$ + 0.156\$ + 0.086\$ + 1.768\$$). We consider the result of the consolidation algorithm is acceptable if it provides a cost $\leq 1.123\$$.

As described in Section V, the first step of the consolidation algorithm is identifying the largest service recommended by our framework, which is c5d.4xlarge. Second, we verify if this service can perform the largest deployment entities (E_2) added to its assigned deployment entity (E_4), which is not the case (the performance evaluation is based on parallel computing [28]). We continue applying the steps of our Algorithm 2 to conclude that the deployment entities E_1 and E_3 can be consolidated and performed by the upgraded service c5d.xlarge. The total cost for the application dropped to $1.116\$/h$ (compared to the nonconsolidated services). Thus, we consider that the consolidation algorithm provides acceptable results that optimized the application deployment cost.

Following the process of service selection using our proposed framework shows the feasibility and the effectiveness of our approach in IaaS service selection.

VII. Conclusion

The motivation of our research stems from the need to assist users in selecting appropriate cloud infrastructure services. Although the market growth provides economic benefits to the users due to increased competition between IaaS providers, the lack of similarity with respect to how IaaS services are described and priced by different providers makes the decision on the best option challenging. The decision also needs to consider the user's preferences over different features. To raise this challenge, we proposed a new hybrid approach based on MCDM and RS techniques that transform the IaaS services selection from an ad-hoc task that involves manually reading the provider documentation to a structured and guided process. By generalizing our previous work [1], we take into consideration medium and large application profiles, which cannot be fulfilled by a single IaaS service. Thus, several services are recommended to satisfy the user requirements. In order to to improve the selected services' utilization and optimize deployment costs we introduce a consolidation method inspired from the knapsack algorithm.

Although we believe that our approach leaves scope for a range of enhancements, yet it provides suitable results. The experimental evaluation conducted against typical RS technique highlights the main benefits of the proposed approach.

For our ongoing works, we are focusing on studying the relation between the deployment entities of the user's application. In fact, the deployment of an application's component as independent deployment entities entails communications between these entities. This communication may introduce new networks requirements and add several constraints such as data flow management.

References

[1] H. Gabsi, R. Drira, and H. H. B. Ghezala, "Personalized iaas services selection based on multi-criteria decision making approach and recommender systems," *International Conference on Internet and Web Applications and Services (ICIW 2018), IARIA, Barcelona, Spain*, pp. 5–12, 2018, ISBN: 978-1-61208-651-4 ISSN: 2308-3972.

[2] "Analytical Research Cognizance," 2019, URL: http://www.arcognizance.com [accessed: 2019-01-23].

[3] M. Eisa, M. Younas, K. Basu, and H. Zhu, "Trends and directions in cloud service selection," *IEEE Symposium on Service-Oriented System Engineering*, 2016, ISBN: 978-1-5090-2253-3.

[4] S. Soltani, K. Elgazzar, and P. Martin, "Quaram service recommender: a platform for iaas service selection," *International Conference on Utility and Cloud Computing*, pp. 422–425, 2016, ISBN: 978-1-4503-4616-0.

[5] J. Lu, D. Wu, and G. Zhang, "Recommender system application developments: A survey," *Decision Support Systems*, vol. 74, pp. 12–32, 2015.

[6] T. Zain, M. Aslam, M. Imran, and Martinez-Enriquez, "Cloud service recommender system using clustering," *Electrical Engineering, Computing Science and Automatic Control (CCE)*, vol. 47, pp. 777–780, 2014.

[7] A. J. Ruby, B. W. Aisha, and C. P. Subash, "Comparison of multi criteria decision making algorithms for ranking cloud renderfarm services," *Indian Journal of Science and Technology*, vol. 9, p. 31, 2016.

[8] Z. Rehman, F. Hussain, and O. Hussain, "Towards multi-criteria cloud service selection," *International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, 2013, ISBN: 978-1-61284-733-7.

[9] J. Lv, X. Wang, M. Huang, H. Cheng, and F. Li, "Solving 0-1 knapsack problem by greedy degree and expectation efficiency," *Applied Soft Computing*, vol. 41, pp. 94–103, 2016.

[10] M. Zhang, R. Ranjan, S. Nepal, M. Menzel, and A. Haller, "A declarative recommender system for cloud infrastructure services selection," *GECON*, vol. 7714, pp. 102–113, 2012.

[11] L. Sun, H. Dong, F. Khadeer, Hussain, O. K. Hussain, and E. Chang, "Cloud service selection: State-of-the-art and future research directions," *Journal of Network and Computer Applications*, vol. 45, pp. 134–150, 2014.

[12] C. JatothG and U. Fiore, "Evaluating the efficiency of cloud services using modified data envelopment analysis and modified super-efficiency data envelopment analysis," *Soft Computing, Springer-Verlag Berlin Heidelberg*, vol. 7221-7234, p. 21, 2017.

[13] F. Aqlan, A. Ahmed, O. Ashour, A. Shamsan, and M. M. Hamasha, "An approach for rush order acceptance decisions using simulation and multi-attribute utility theory," *European Journal of Industrial Engineering,*, 2017, ISSN: 1751-5254.

[14] C. B. Do and S. K. Kyu, "A cloud service selection model based on analytic network process," *Indian J Sci Technol*, vol. 8, no. 18, 2016.

[15] ——, "A hybrid multi-criteria decision-making model for a cloud service selection problem using bsc, fuzzy delphi method and fuzzy ahp," *Indian J Sci Technol*, vol. 86, pp. 57–75, 2016.

[16] M. Whaiduzzaman, A. Gani, N. B. Anuar, M. Shiraz, M. N. Haque, and I. T. Haque, "Cloud service selection using multicriteria decision analysis," *The Scientific World Journal*, vol. 2014, p. 10, 2014.

[17] L. D. Ngan and R. Kanagasabai, "Owl-s based semantic cloud service broker," *International conference on web services (ICWS)*, pp. 560–567, 2013, ISBN: 978-1-4673-2131-0.

[18] F. K. Hussain, Z. ur Rehman, and O. K. Hussain, "Multi-criteria iaas service selection based on qos history," *International Conference on Advanced Information Networking and Applications*, 2014, ISSN: 1550-445X.

[19] Z. Li, L. OBrien, H. Zhang, and R. Cai, "On the conceptualization of performance evaluation of iaas services," *IEEE Transactions on Services Computing*, vol. 7, pp. 628 – 641, 2014.

[20] Q. Yu, "Cloudrec: a framework for personalized service recommendation in the cloud," *Knowledge and Information Systems*, vol. 43, pp. 417–443, 2014.

[21] S. Sukhpal and I. Chana, "A survey on resource scheduling in cloud computing: Issues and challenges," *Journal of grid computing*, vol. 14, pp. 217–264, 2016.

[22] "Microsoft Azure," 2019, URL: https://azure.microsoft.com/en-us/pricing/details/cloud-services [accessed: 2019-02-28].

[23] S. Soltani, P. Martin, and K. Elgazzar, "A hybrid approach to automatic iaas service selection," *Journal of Cloud Computing: Advances, Systems and Applications*, vol. 7, pp. 1–18, 2018.

[24] "Mahout Apache," 2019, URL: http://mahout.apache.org/ [accessed: 2019-02-28].

[25] "Amazon Instance Types," 2019, URL: https://aws.amazon.com/ec2/instance-types/?nc1=h_ls [accessed: 2019-02-28].

[26] J. Davis and M. Goadrich, "The relationship between precision-recall and roc curves," *International conference on Machine learning*, pp. 233–240, 2006.

[27] Z. Zheng, X. Wu, Y. Zhang, M. R.lyu, and J. Wang, "Qos ranking prediction for cloud services," *European Journal of Industrial Engineering,*, vol. 24, pp. 1213–1222, 2013.

[28] X. Liu, C. Wang, B. B. Zhou, J. Chen, T. Yang, and A. Y. Zomaya, "Priority-based consolidation of parallel workloads in the cloud," *IEEE Transactions on parallel and distributed systems*, vol. 24, pp. 1874–1883, 2013.

A Survey on Smart Cities, Big Data, Analytics, and Smart Decision-making

Towards an analytical framework for decision-making in smart cities

Marius Rohde Johannessen, Lasse Berntzen,
Department of Business, History and Social Sciences
University of South-Eastern Norway
Kongsberg, Norway
e-mail:{lasse.berntzen; marius.johannessen}@usn.no

Rania El-Gazzar
Department of Business and Law
University of South-Eastern Norway
Kongsberg, Norway
e-mail: rania.el-gazzar@usn.no

***Abstract*—Smart decision making is based on data combined with analytics to improve decision-making. This paper examines several application areas of smart cities, and related data sources used for decision-making. Further, we present a review of analytical techniques used in earlier studies. In many cases, systems may make decisions on their own. Such autonomous systems may play an essential role in the development of smart cities. In other cases, the data can be combined with historical data or other open data sources to play a role as the foundation for decision-making. Our findings are presented as an analytical framework, which will be used for further empirical studies into this domain.**

Keywords-smart decision-making; smart cities; big data; sensors; analytics; autonomous systems.

I. INTRODUCTION

This article is an expanded version of an earlier conference paper presented at ICDS 2018 [1], and offers an analytical framework for smart or (intelligent) decision-making in the context of smart cities. The framework is based on a review of literature, white papers and news sources covering the topic, as well as empirical data from a study on air quality monitoring. The analytical framework shows areas in need of further study and forms the basis for future research projects.

The analytical framework shows areas in need of further study and as such forms a research agenda for (big) data analysis in a smart city context. The target audience for this work is mainly Information Systems (IS) researchers and practitioners.

Smart decision-making uses a systematic approach to data collection and applies logical decision-making techniques instead of using intuition, generalizing from experience, or trial and error.

"Smart cities" is a multifaceted concept and has been defined in many different ways; more than 100 definitions of smart cities have been analyzed by the International Telecommunication Union (ITU)'s focus group on smart sustainable cities [2][3]. The mandatory requirement for smart cities is to improve quality of life and achieve sustainable development (economic, social, and environmental) through the use of Information and Communications Technology (ICT) and intelligence [4]. Definitions emphasized the technological aspect of a smart city as being "a technologically interconnected city" or Internet of Things (IoT) using big data is promoted to achieve the efficiency and intelligence in managing cities' resources [5][6].

A smart city is a city that is characterized as "instrumented, interconnected, and intelligent" [7][8][9]. This can be conceptualized as three layers, as shown in Figure 1.

These characteristics are enabled by the use of ICT, which constitute the heart of a smart city [10]. The "instrumentation" layer does data acquisition through sensor-based systems that provide real-time data through sensors, meters, and cameras, but also from social media and open data sources. The instrumentation layer enables capturing and filtering data from various sources for timely response. The inputs from the instrumentation layer are integrated and transformed into event-related information at the "interconnection" layer to provide rich insights for decision-making. The interconnection layer provides all forms of collaboration among people, processes, and systems to enable a holistic view supporting decision-making. At the "intelligence" layer, business intelligence and analytics are applied to the information provided by the interconnection layer and other city-relevant data and, then, the analyzed information is visualized to understand the city requirements and city policies, hence, make informed decisions and take actions. The intelligence layer is focused on deep discovery, analyses, and forecasting. These three layers that build up the smartness in a smart city are constructed by smart technology solutions and ICT infrastructure, such as IoT, big data, and the Internet.

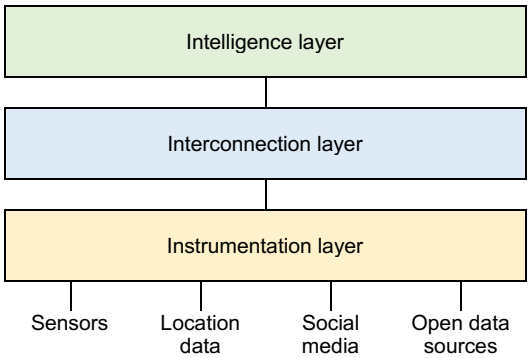


Figure 1. Three-layer model.

Regarding the intelligence layer that is concerned with decision-making, a review of studies on smart city and decision-making resulted in nine articles. This indicates that smart city and decision-making is an area that deserves further investigation on how to make a significant impact from big data [11].

In this article, we elaborate on smart or intelligent decision-making in the context of smart cities. Smart decision-making relies on data and analytics to make better decisions. By using autonomous systems, the decisions can be implemented in real time. Human intervention can be reduced to oversee the decisions and take over if the system is malfunctioning.

The primary focus of this article is on the instrumentation and intelligence layers, and the data sources and analytical techniques used for decision-making. The data is refined through the interconnection layer and processed by the intelligence layer to enable decision-making. The three-level model provides a systematic approach to collecting facts and applying logical decision-making techniques, instead of generalizing from experience, intuition (guessing), or trial and error.

The rest of the article is structured as follows: Section II discusses methodology. Section III focuses on the instrumentation layer, including identification of common data sources. Section IV describes some selected smart city application areas. Section V presents an overview of relevant analytical techniques. Section VI presents our analytical framework. Section VII contains our conclusion, some limitations, and ideas for future work.

II. METHODOLOGY

The purpose of this article is to begin exploring how common application areas of smart cities use, analyze and visualize data. Data analysis and visualization are essential for decision-making and intelligence in smart cities [7]-[9]. However, our literature review reveals little research in this area.

Figure 2 shows how data is analyzed and visualized. The analytics typically stores data for future use, e.g., for predictions. The visualization is used for human decision-making.

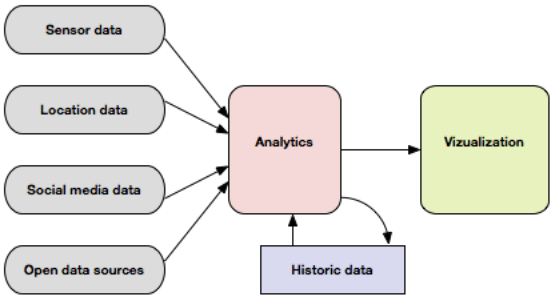


Figure 2. From data to decisions.

Thus, an analytical framework outlining the possible data sources, analytical and visualization techniques could be a valuable contribution to decision-making, as well as for future studies in this domain. Our research question for this study is "*Which data sources are applicable to the different application areas of smart cities*?" and "*which analytical techniques are available for analysis?"*

Data collection was done through several iterations. Initially, we planned on conducting this study as a pure literature review. However, there are few studies in this area so far. Using the Norwegian research library Oria (providing access to EBSCO, IEEE, JSTOR, PROQUEST and SAGE), we were only able to identify nine research papers (referenced in Table I) using the search phrases "smart city" and "decision-making" in the title. Thus, we had to rely on additional data sources and conduct a document analysis of industry white papers, as well as industry, technology and regular news sources. In addition, we applied existing empirical data from a previous study on air quality monitoring.

This exploratory approach led us to three themes, which we summarize in Section III, Table I. Further, the examined news sources and white papers identified nine application areas of data analysis in smart cities; parking, speed monitoring, public transport, traffic, environmental monitoring, energy management, waste handling, crime prevention, and home healthcare.

We conducted a second literature review round where we examined analytical techniques (intelligence layer). There were few, if any, studies explicitly combining smart cities and in-depth description of analytical technique, so we expanded our review and came up with 26 articles describing analytical techniques and methods relevant for smart cities. There were a lot more articles available, but the 26 we selected provides an overview of the most common methods and techniques. Snowballing from the reference lists of the articles revealed additional relevant references. We applied combinations of the following search phrases and keywords for the second round: "Big data analysis, tools, "research methods", statistics, "data analytics", "spatial data". In addition to the research papers, we have also examined additional web sources (digital methods initiative, Github). In both rounds of the literature review, we read the abstract and conclusions of the papers in order to identify which papers were relevant. The number of papers reported is what we were left with after this process.

For analysis, we have applied literature, findings from the air quality monitoring study, as well as data from industry to map potential data sources for each of the nine categories. This allowed us to create an initial framework of data sources for the nine identified categories.

III. DATA FOR DECISION-MAKING

At the instrumentation layer, data for decision-making may originate from many different sources. Laney [12] defines big data as data having high volume, high velocity and/or high variety. High volume refers to large amounts of data-demanding both specialized storage and processing. High velocity refers to streams of real-time data, e.g., from sensor networks or large-scale transaction systems. Finally,

high variety is about dealing with data from different sources having different formats.

Big data may originate from sensors. Another important source for big data is the world-wide-web. Web mining can be used to retrieve unstructured data (text) related to everyday events happening in a city. In this context social media, such as Facebook and Twitter can provide information about problems and citizen sentiments. Many government organization and private companies offer open data sets online that can be used for analysis and decision- making.

Marr [13] argues that the real value of big data is not in the large volumes of data itself, but in the ability to analyze vast and complex data sets beyond anything we could ever do before. Due to recent advances in data analysis methods and cloud computing, the threshold for using big data has diminished.

A. Sensors

Sensors and sensor networks are essential for smart decision-making. Sensors provide real-time information on a wide range of areas, such as weather, traffic, air quality, energy consumption, water consumption, and waste. Data from sensor networks are structured and easy to process, although different vendors and makers can introduce some difficulty. According to Cambridge dictionary, the word "sensor" means a device that is used to record that something is present or that there are changes in something. IoT is an infrastructure with interconnected units that may among other things act as sensor platforms. Botterman [14] defines IoT as:

"A global network infrastructure, linking physical and virtual objects, through the exploitation of data capture and communication capabilities. This infrastructure includes existing and evolving Internet and network developments. It will offer specific object-identification, sensor and connection capability as the basis for the development of independent federated services and applications. These will be characterized by a high degree of autonomous data capture, event transfer, network connectivity and interoperability". (p.12).

B. Location data

Location data places an object in a specific position. Location is important both for stationary and mobile objects. For mobile objects, location data comes from the Global Positioning System (GPS) or from triangulation of radio signals, e.g., belonging to a mobile network.

C. Social media

Another possible data source for smart decision-making is social media. Social media has been defined differently among scholars [15]. However, we adopt the definition by Kaplan and Haenlein [16]: "Internet-based applications that build on the ideological and technological foundations of Web 2.0, and that allow the creation and exchange of User Generated Content" (p.62).

Data retrieved from social media will mostly be unstructured (text, images, video), but also structured meta-data providing additional information, e.g., tags containing author, content type, title, date/time and location. Unstructured data from social media may provide insight into the perceptions and sentiments of smart city citizens.

D. Open Data Sources

Open data is data that can be freely used, reused and redistributed by anyone - subject only, at most, to the requirement to attribute and share alike. Open data has the following characteristics [17]:

- Availability and access: The data must be available as a whole and at no more than a reasonable reproduction cost, preferably by downloading from the Internet. The data must also be available in a convenient and modifiable form.
- Reuse and redistribution: The data must be provided under terms that permit reuse and redistribution.
- Universal participation: Everyone must be able to use, re-use and redistribute - there should be no discrimination against fields of endeavor or against persons or groups.
- Interoperability: The ability to interoperate - or intermix - different datasets (i.e., one piece of open material contained therein can be freely intermixed with other open materials).

E. Decision-making in Smart Cities

In the context of smart city, decision-making has been given less attention in the literature; Google Scholar found nine articles discussing decision-making in smart cities (See Table I). The nine articles investigated various aspects of the three layers described earlier.

Studies related to the interconnection layer have highlighted various collaboration aspects that are important for smart cities. Ojasalo and Tähtinen [18] proposed a model of an open innovation platform for public sector decision-making in a city. The authors identified three different kinds of relationships that are present and partly interwoven in open innovation platforms (i.e., governing, sparring, and collaboration). The proposed model helps in organizing the three types of relationships of an innovation platform with the city's decision-making and external actors, by combining different decision-making cultures between the public and private sector.

TABLE I. MAPPING LITERATURE TO SMART CITY LAYERS

Ref.	Instrumentation layer	Interconnection layer	Intelligence layer	Others
[18]		X		
[19]			X	
[20]			X	
[21]			X	
[22]	X			
[23]	X			
[24]	X	X	X	
[25]				X
[26]				X

At the intelligence layer, Eräranta and Staffans [19] discussed knowledge creation and situation awareness in collaborative urban planning practice, and how digitalization changes it. The authors argued that smart city planning is not

only a data-driven superlinear scaling practice, but an integrative and collaborative learning process facilitated by face-to-face interaction, advanced analyses and visualizations of available data, ongoing processes, and local history and stories. The authors brought in collaboration at the intelligence layer.

At the intelligence layer, Passe et al. [20] attempted to understand human behavior and decision-making about the built environment within an expanding range of spatial, political, and cultural contexts. The authors emphasized the importance of participation by a broad range of stakeholders in making decisions for the future of smart cities. The authors argued for the need to consider social dynamics in addition to building-occupant interactions, which requires investigating multiple scales and types of data to create new methodologies for design and decision-making processes. This approach moves data collection, analysis, design, and decision-making away from hierarchical relationships and utilizes the expertise of all stakeholders.

Also at the intelligence layer, Honarvar and Sami [21] talked about the various sensors embedded in different places of smart cities to monitor and collect data about the status of cities. Mining such data to extract valuable knowledge creates a challenge because various sources of data in smart cities are big, independent, heterogeneous and no semantic is integrated and annotated to them. The authors proposed an approach to leverage linked open data and semantic web technologies, data mining mechanisms, and big data processing platforms.

At the instrumentation layer, Khan et al. [22] emphasized the role of citizen participation as an important data source for social innovation and co-creating urban regeneration proposals through innovative IT systems. Those IT systems can use open government data, visualize urban proposals in 3D models and provide automated feedback on the feasibility of the proposals. Using those IT systems as a communication platform between citizens and city administrations offers an integrated top-down and bottom-up urban planning and decision-making approach to smart cities. In the same line, Foucault and Moulier-Boutang [23] proposed a governance model called "Smart City – organological". The model consists of an adaptive device built around differentiation of smart sensors and tags to improve human decision-making. The device is taking into account both "physical sensors" and "economic and social sensors" to capture the explicit or implicit needs.

At the level of the three layers, Nathali Silva et al. [24] expressed concerns about the continuous growth of the complex urban networks that is challenged by real-time data processing and intelligent decision-making capabilities. The authors proposed a smart city framework based on big data analytics. The framework operates on three levels: instrumentation layer (data generation and acquisition level), interconnection layer (collecting heterogeneous data related to city operations, data management and processing level), and intelligence layer (filtering, analyzing, and storing data to make decisions and events autonomously, and initiating execution of the events corresponding to the received decisions).

Some other topics were studied in the literature, e.g., Gang and Yang [25] studied design issues to improve the intelligence layer of city emergency management. Kurniawan et al. [26] investigated the development and optimization possibilities of Makassar City smart operation room. The authors used fuzzy multi-criteria decision-making to illustrate the project priority rank and further to determine the alternative optimal option in conducting the project.

IV. Application Areas

In order to understand more about data sources and decision-making techniques, we have examined some common application areas connected to smart cities which were identified in the literature review (See Table II). The first four areas are connected to transport:

- Smart parking
- Speed monitoring
- Smart public transport
- Smart traffic

The rest of the application areas represent the broadness of the smart city concept:

- Environmental monitoring
- Energy management
- Waste handling
- Crime prevention
- Home healthcare

A. Smart Parking

Smart parking assists drivers to find a nearby parking spot. The information provided to the driver can have many different forms, from public displays placed next to roads to mobile apps directing the driver to a free parking spot [27][28][29].

Smart parking data is sensor based. Outdoor sensors may be magnetic sensors located in capsules embedded in the ground, detecting the presence of a car, or cameras detecting if a parking spot if free or not. Indoor parking spots may instead have infrared or ultrasound sensors to detect the presence of cars.

Smart parking may also include payment solutions based on mobile phone apps, use of SMS, or dedicated devices like SmartPark™ [30]. The payment solutions may give the user the opportunity to pay for time actually used instead for paying for a fixed time period.

Smart parking sensor data provides information to city planners and car park companies about the occupancy of parking spots over time. The collected information can be used for decision-making regarding the construction of new parking sites, and to decide on pricing.

B. Vehicle Speed Monitoring

Vehicle speed monitoring warns drivers about their driving speed. The idea is to make drivers slow down if they are driving at excess speed. Speed monitoring units may be stand-alone, but state-of-the-art units are connected to the Internet and provide real-time information on driving habits [31].

Several technologies have been demonstrated for vehicle speed monitoring including the use of cameras, RADAR, LIDAR, and underground sensors [32]. A measurement station is put in a fixed position, and excess speed is shown on a display device.

Another approach is to install mandatory units in all vehicles. The driver can then be alerted of excess speed directly by the unit. Such units can also upload speed data through some kind of network [32].

(Some GPS devices warn the driver about excess speed, but such data are not relevant, since data are not uploaded for use by traffic authorities.)

Vehicle speed monitoring data can be used by traffic authorities and police to decide on traffic control locations. Such data can also be used to implement speed reducing measures, such as speed bumps or traffic lights, and even control such measures in day-to-day operations.

C. Smart Public Transport

One essential measure to reduce environmental footprint is to reduce car traffic, in particular the use of private cars. Well-developed public transport infrastructure can be an incentive to reduce traffic load. Car owners may also be discouraged by the toll charges or congestion charges implemented in many cities.

Smart public transport uses technology to provide public transport users with a better user experience [33]. Use of sensors and GPS technology can provide real-time data on arrivals and departures of public transport vehicles.

Smart ticketing solutions may use smart cards or mobile phones equipped with Near Field Communication (NFC) to make ticketing more efficient from a user point of view [34].

Online route planners may help users choose the most efficient route from one location to another location.

The data collected from smart public transport can be used for real-time situation reports and may also be used by public transport planners to adjust timetables, change routes, create new routes, and adjust fares.

Social media may be mined to find citizen perceptions of the public transport system.

D. Smart Traffic

Smart traffic is about using technology to ensure more efficient traffic management. Traffic management may use road lights and signs to optimize traffic flow in real time [35]. Commercial car navigation systems provide information on fastest and shortest routes. Some navigation systems collect information from other cars real time to detect bottlenecks and provide alternative routes.

Data may come from sensors embedded in the roads. The most common technique is to detect traffic density by embedding coils under the road surface to pick up passing cars. Alternatives are to use cameras or radar technology to detect traffic.

Data may also come from the vehicles themselves, by using radio transmissions or a cellular network [36].

The data collected may be used by the city-administration for road-planning, adjusting intervals of traffic lights. Data can also be used by transport companies to decide on best schedules for pick-ups and deliveries.

Mining social media may provide some information on how citizens experience traffic situation.

E. Air Quality Monitoring

Monitoring air quality and other environmental parameters are the important for decision-making. Some cities are enforcing restrictions on traffic when pollution levels reach a certain threshold [37].

In most cases, the air quality monitoring is done by fixed monitoring stations located throughout the city, but may also be done by mobile handheld units, or units installed in cars.

Measurements include gases: CO, CO_2, NO_X, and dust particles, normally 2,5 PM and 10 PM.

Collected data can be combined with other data sources, e.g., meteorological data, to provide real-time situation reports and make forecasts for future pollution levels. Data can be visualized and be made available to the public. Such data is particularly valuable for citizens with respiratory problems.

Social media may be mined to find citizen perceptions of air quality.

F. Energy Management

Smart power grids contribute to better energy management and reduced environmental footprint. An essential part of the smart grid is smart meters. Smart meters are devices that continuously measure the power consumption of households and buildings. Household appliances can communicate with the smart meter to schedule activities when the load on the power grid is low. The smart meters also communicate with energy management systems to optimize energy consumption [38]. Buildings can also take part in energy production through the use of solar panels and other alternative energy sources.

Sensor data may be combined with location data and open data sources to make forecasts. Social media data plays a minor role in the context of energy management.

G. Waste Handling

Sorting waste materials for recycling has become common practice. Garbage collection can be improved by only collecting waste when necessary. “Intelligent” waste containers can report their state of becoming full and get included in the schedule of trucks collecting the waste [39][40][41].

The recycling process itself can provide valuable data on types and amounts.

Data from the waste collection process can be used to decide on container size and pick-up patterns. Data may also be made public to show timeliness and efficiency of the waste handling, from garbage collection through recycling.

Social media data mining can be used to detect sentiments about garbage collection.

H. Crime Prevention

Crime prevention is about allocating police resources to areas most likely to get victims of crime, but also to find out where to establish surveillance by video cameras and other

means. Home or business security systems may discourage criminals and prevent crimes from being committed.

Data used for crime prevention will mostly be formerly reported crimes combined with open data sources, e.g., demographic data, property values, income levels of citizens, street light coverage, etc. [42][43].

Social media may be mined to find indications of unreported crimes.

I. Home Healthcare

Home health care is an important measure to enable healthcare patients to live in their homes as an alternative to nursing homes, and thereby reducing the burden on the healthcare system. Technology is important for patients to feel safe and to manage their health conditions. Safety alarms alert healthcare personnel about emergencies including fall detection. The safety alarm may have a built-in GPS device to provide healthcare personnel with the current location of the patient. Vanus et al. [44] describe how different sensors can be used to detect daily living activities in smart home care. Mshali, Lemlouma, Moloney, and Magoni [45] made a survey on health monitoring systems for use in homes. Smart medicine dispensers can alert the patient to take medication, and also notify healthcare personnel that medicine has been retrieved from the dispenser [46]. Sensor platforms are also used to monitor chronic diseases to make sure patients receive proper care [47]. Sensors may detect changes in medical conditions before the patients become aware of the change themselves [48]. Such data are important to make home healthcare smarter.

Open data may be used for planning purposes, e.g., statistics about the demography and increase of certain medical conditions. Social media does not play any significant role in home healthcare.

Table II summarizes the data sources applied in the different studies mentioned above.

TABLE II. MAPPING APPLICATIONS TO DATA SOURCES

Application areas	Data sources			
	Sensor data	Location data	Open data	Social media data
Smart parking	X	X	-	-
Speed monitoring	X	x	-	-
Smart public transport	X	X	-	x
Smart traffic	X	x	-	x
Air quality monitoring	X	x	X	x
Energy management	x	x	x	-
Waste handling	X	X	-	x
Crime prevention	-	X	X	x
Home health care	X	x	x	-

X major data source
x minor data source
- not applicable

V. ANALYTICAL TECHNOLOGIES, METHODS AND TECHNIQUES

There are few research articles on big data in smart cities that are explicit about the actual methods, technologies and techniques being used for data analysis, so we had to rely on more general themed big data analytics articles. In this section, we provide an overview of these and attempt to link methods with application areas. In an overview section such as this, there is only room for a brief overview of each individual technology and method. For in-depth descriptions, we refer to the individual articles and papers referenced.

A. Technologies for storage and retrieval

Our literature review of big data analytical tools returned a lot of hits covering not so much methods as technologies. The reason for this is that big data requires somewhat different approaches in terms of storage and retrieval. Large amounts of data and the need for effective and selective filtering and retrieval are some of the challenges these technologies attempt to address [49]. For smaller data sets traditional techniques, or combinations of new and old, are just as appropriate. For example, software such as MS Excel can handle sheets of up to 1 million rows [50], and SPSS remains a powerful tool for statistical analysis.

Key technologies for big data include cloud computing, distributed file systems and distributed programming environments, as well as new database technologies such as NoSQL and NewSQL [51]. Cloud computing and associated technologies have the advantage of being scalable and flexible, and No/NewSQL databases have more flexible data structures and rules compared to traditional SQL databases, allowing for easier filtering, storage and retrieval of data [52]. As big data is often un- or semi-structured, there is a need for technologies that allow for redundancy and novel combinations of data for exploration and analytical purposes [53].

A typical package for big data analysis would include Apache Hadoop, using a file system such as HDFS for distributed storage, combined with a NoSQL database [54]; [55][56]. NoSQL databases can further be divided into (at least) four categories [57]:

- *Key-value*, where each value corresponds to a primary key, offering a simple and powerful structure for handling big data.
- *Bigtable* or *wide-column*, a more structured approach capable of handling large and complex data sets.
- *Document,* where entries as stored as documents rather than relations, and
- *Graph,* which stores information about nodes and their relations.

Each category has different use cases, which are beyond the scope of this section. Details can be found in the following referenced papers: Apache HBase and Cassandra are a few examples of popular Wide-column databases [58][59]. Oracle Berkeley [60] is a popular key-value database, where a previous iteration was involved in the first Bitcoin implementation. Mongo DB is a popular document database, where data is stored in documents rather than records,

allowing for easily changing the data structure and varying the fields recorded for each piece of data.. Users include Amazon and Adobe [61]. Finally, Neo4J is an example of a graph database, used for areas such as bioinformatics, recommender systems and network graphs [62].

Distributed storage of data introduces some challenges related to processing and retrieval of data, which is where the map/reduce paradigm comes into play [63]. In short, map/reduce provides a programming model which creates a set of key-value pairs between records, regardless of where they are located in the distributed file system.

When the core technologies (file system, storage, database) are in place, the next layer is the analytical methods and techniques applied to analyze data.

B. Analytical methods and techniques

The literature review shows that there are many analytical methods in use, ranging from traditional statistical analysis to a plethora of machine learning methods and algorithms. There are also a few papers with a more critical perspective, warning researchers to not become deterministic and blind for the social construction of algorithms, but rather pay close attention to ethics and approach data-driven methods with a critical attitude and a thorough understanding of both domain and context [64]. An on-going survey from the Universities in Nottingham, Oxford and Edinburgh [65] is currently examining this, presenting respondents with case studies showing the outcome of different algorithms, and asking them to reflect on these different outcomes. We suggest that researchers from both academia and the private sector apply the same ethical and critical perspective to big data analysis as they would any other research project, and critically examine the fit between research question/hypothesis and the method being applied.

1) Statistical methods

Several papers combine Big data platforms with traditional statistical analysis. A study of terrorist ideology and attack type used a Hadoop platform to combine the global terrorism big data set with Google news data about terrorist attacks. The data was analyzed in SPSS using descriptive statistics and correspondence analysis [66]. A similar approach was applied to a study of electric vehicle customers [67]. However, for data sets that are "true" big data, with millions of cases, traditional statistical methods such as correlation will often show significant results between variables even where there is no real-life correlation, due to the sheer volume of data [68].

While traditional statistical methods have been successfully applied, several scholars point out that big data is often recognized by being unstructured and difficult to organize in variables such as we are used to from traditional statistical methods. Thus, new methods have emerged, such as sentiment, network, various link and content-based analytical techniques, and of course machine learning [69]. The remained of this section will provide an overview of some of the most commonly used methods. An excellent place to start could be the wiki of the *Digital methods initiative* [70], a collection of methods for digital research run by a consortium led by the University of Amsterdam.

2) Text analytics and Sentiment analysis

Text analytics, or text mining, refers to a set of methods for extracting and analyzing text-based content from news media, social networks, e-mails, blogs etc. [69]. One example using a set of text analytics methods is a study of hotel guest experiences, where guest ratings were analyzed to extract factors that were of particular importance to hotel guests [71]. Methods for text analytics include traditional content analysis [72] or various forms of discourse analysis [73], counting techniques such as word frequency, word count, word clouds [71]. Sentiment analysis has become a popular method in many areas such as politics [74][75], business and marketing [76]. Sentiment analysis is a classification process where words and phrases are classified as positive or negative in order to analyze public opinion on various things such as brands, political parties or current issues. Sentiment can be classified using pre-coded lists of words and phrases or as supervised machine learning [77].

3) Social network analysis

It is said that the Internet and social media has contributed to our current age being called the network society, as more and more of our lives can be seen as parts of a network [78]. Our Facebook and Twitter friends, LinkedIn contacts and the websites we follow, connect ideas and opinions. Social network analysis examines how information flows through a complex network and allows us to visualize and analyze networks by examining the connections and attributes of connections between nodes [79]. The basic use of network analysis is to identify patterns of interaction among the participants in a network. Typical variables measured are:
Degree: The number of participants a given participant interacts with, can be split into receiving (in-degree) and sending (out-degree) messages. High degree levels indicate strong networks and community. *Centrality*: How important a participant is to the network. Measured as closeness (the number of nodes between two participants), betweenness (how each participant helps connect other participants), and eigenvector (how well a participant is connected to other active participants). *Clustering*: The degree to which a set of participants form a group within the network. *Density*: The proportion of actual vs. potential connections within the network [80].

Social network analysis can for example be applied to understand how information flows between actors, such as in the study of disaster management after the Louisiana floods [81], or combined with graph databases when creating advanced recommender systems [82].

4) Spatial analysis

Combining data and location is a powerful analytical tool that has been around for a long time. In Utah, data from health records and environmental records were combined and used to predict areas likely to see more cases of cancer [83]. A similar study from Scotland combined geographical and demographic data to examine mortality rates in different parts of the country [84].

These and a range of similar studies rely on traditional compiled registry data, which by itself is a powerful tool. However, adding data sources such as sensors, data mining of the Internet etc can improve the predictive power of these

models, for example in fields such as epidemiology, transportation, flooding and environment/climate studies. Using sensors from cars to collect data along with traditional registry data such as traffic congestion statistics, researchers were able to create a detailed spatial distribution of Carbon emissions in China [85]. Another study combined geographical data with accident statistics, data from taxis, public transport, and social media to create a predictive model of traffic accident hotspots [86].

5) Machine learning

Artificial intelligence and machine learning are some of the most talked about issues in current science. At its core, machine learning involves creating software that improves through experience, by being shown some examples which are run through various algorithms, allowing the software to learn "by itself" [87]. Machine learning is the preferred method for development of software for computer vision, speech recognition, natural language processing, robot control and more, and has also become prevalent in sciences ranging from biology to the social sciences [88], as big data is making traditional methods more difficult.

At its core, machine learning consists of feeding data to a piece of software and applying various algorithms (step-by-step instructions) to make sense of the data [89]. There are hundreds of techniques and even more algorithms for machine learning, but all the different techniques can be categorized into two categories: Supervised and unsupervised [89]. A review of machine learning literature found 156 supervised and 46 unsupervised algorithms tested in 121 different studies [89].

Supervised machine learning involves feeding data to an algorithm along with a set of labeled "training data" [90]. For example, sentiment analysis could be conducted by manually coding a data set and feeding this to an algorithm which would then use the example data to continue coding more data. In order to verify the model, several iterations of evaluation are needed to calculate accuracy [90].

Supervised learning techniques include the following:

- *Regression*: for calculating the relationships between variables (several algorithms for big data regression).
- *Classification trees:* Where variables are split into multiple dimensions to form a tree structure.
- *Ensemble learning/aggregation*: Additional training techniques to ensure the accuracy of tree models.
- *Support vector machines*: Another method of classification.
- *Neural networks*: Often used for learning purposes such as speech recognition or social networks, Neural networks are made up of nodes and connections receiving and sending signals. Originally modeled on the human brain, but has since branched out in many directions.
- *Nearest neighbors*: Used in pattern recognition, regression and classification. For example, given the length of petals for a set of flowers, the algorithm can identify the different types of flower [89].

Unsupervised machine learning on the other hand, does not involve any manual coding. Data is fed unlabeled to the algorithm in order to make sense of it [89]. Some of the popular techniques include *Clustering*, where the purpose is to find connections between variables. Clustering is a popular technique in marketing and recommender systems, as it can discover what people with a certain set of characteristics typically buy or are interested in. Combining demographic data, purchase data, interests, or text from the news articles we read are typical data sources [91]. Another popular technique is *dimensionality reduction*, where the objective is to reduce the number of variables under consideration. For example, sensor-based location data can come from a number of different sources, with each source providing overlapping information. Dimensionality reduction can help reduce the number of variables into one "position" variable [92].

The methods reviewed in this section are summarized in Table III.

TABLE III. ANALYTICAL METHODS AND APPLICATIONS

Method	**Techniques/application**	
Social network analysis	Identify attributes, relations and networks (of networks)	
Sentiment analysis	Identify negative/positive opinions related to a topic/case/issue	
Spatial data analysis	Combine data and location to visualize where something happens/could happen	
Statistical analysis	Find relations between variables	
Machine learning	Unsupervised techniques	Find connected variables. Reduce dimensions (variables) providing data on the same thing.
	Supervised techniques	Classification and regression (relations) between different variables. Pattern recognition.

VI. ANALYTICAL FRAMEWORK

The purpose of our case studies is to examine data sources and methods used to analyze data. Seven examples of smart city applications show the importance of sensor data, but also the opportunities for using open data sets combined with sensor data to improve analysis and enable forecasting. Web mining and social media have limited use in these cases, but can be used to alert city administration about potential problems and sentiments.

The crime prevention case does not rely on sensor data, but on reports of crimes. Combining different open data sets can provide better insight related to crime prevention. The reported crimes can provide patterns, but combining data sets may shed light on underlying factors, like property values, incomes, absence of street lights and other factors.

In this study, we have examined mainly the instrumentation and interconnection layers, finding a set of data sources used in different smart city application areas, as shown in Table II.

When we map these findings to the three layers in Table I, we have the outline of an analytical framework as shown in Figure 3. The resulting analytical framework may guide future research efforts in the field.

Existing research and white papers provide examples of how big data can be applied for decision making, but as our framework shows, there is a need for both synthesizing existing studies as well as conducting new empirical studies to create a roadmap for decision-makers.

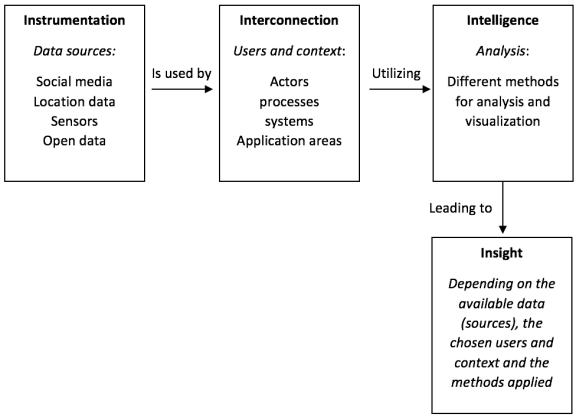


Figure 3. Analytical framework.

This roadmap would list relevant data sources and analytical techniques for different users and contexts. The framework forms a possible foundation for future studies in this area.

VII. Conclusion, Limitations and Future Work

In this article, we used nine common application areas of smart cities to explore their use of data. We examined relevant data sources, and their use. Data collected from sensors are very important for seven of the chosen application areas. Open data is often a valuable supplement to collected data. In some cases, location data are combined with other types of data. Social media data mining may play a role to show user perceptions and sentiments.

The collected data need to be processed and analyzed to be useful for decision-making.

Data will often be used for automatic decision-making. In eight of the chosen application areas, we found examples of data used for automatic decision-making:

- Smart parking: Automatic update of displays directing drivers to available parking spots.
- Speed monitoring: Automatic regulation of traffic lights, or even photographing speeding vehicle to issue a speed ticket.
- Smart public transport: Automatic updates of screens showing arrival and departure times.
- Smart traffic: Automatic control of signs and traffic lights to redirect traffic.
- Air quality monitoring: Automatic alerts to citizens in the areas, through signs or SMS service.
- Energy management: Automatic start of household appliances, based on grid load.
- Waste handling: Automatic updates of garbage truck schedules based on amount of garbage in each container.
- Home healthcare: Automatic requests for health care personnel to look into changing medical situation for patients living in their own homes.

For strategic and long-term decisions done by humans, the results of the processing and analysis need to be visualized in a meaningful way, e.g., through graphs, bar charts, pie charts often combined with a map or even embedded in a Geographic Information System (GIS) front-end.

However, researchers analysts should be aware that algorithms and methods are social constructions, and that the choice of algorithm and method in some cases will influence the outcome of research. Careful ethical and methodological considerations should therefore be considered when planning and implementing an analytics project.

This article studied application areas of smart cities and methods for data analysis to examine use of (big) data. The study is not exhaustive. We used example of application areas from literature, but as "smart cities" have ambiguous definitions, we may have overlooked some areas. Further, as we had to start examining white papers from industry it is likely we have missed interesting data from relevant sources even after our rigorous search in the most well-known big data/analytics companies.

A. Future research challenges

Finally, we would like to present some future research challenges derived from this literature review of smart city and data analytics.

Consolidation: Data analytics and smart cities are both new and intertwined fields, with research coming from both highly technical and (some) organizational fields. There are few studies that combine a technical solution with a thorough field test and case study. Thus, we argue there is a need for consolidation of the field, in order to move it towards the mainstream. City managers need off the shelf tools and simple processes in order to make use of analytics.

Analytical methods: We have attempted to summarize the analytical methods identified in literature in Table III. However, further work is needed in order to classify these methods for managers and the organizational level. Most of the literature on these methods were written for a technical audience, requiring skills that decision-makers may not have.

Application areas: The list of smart city application areas was selected based on the literature review. There are many other application areas, but the selected areas were considered to provide good examples on both data sources and analytical methods. Future work may include even more application areas.

Actors and process: In our framework (Figure 3) we include actors and process (including context) as factors. Few of the studies included in this review address these in detail, except brief mentions of the actors involved. As context is important in technological implementations, there is a likely need for more in-depth studies of how process and actors/stakeholders influence/are influenced by, the analytical process.

Finally, we intend to investigate further the use cases for smart cities' use of analytics and big data, so that we can present a comprehensive model of possible combinations of data sources, actors and contexts, and analytical techniques.

For practitioners and researchers with little technical background, a handbook of methods, techniques and use cases would be a valuable tool that could potentially improve and simplify data analysis strategies and outcomes.

REFERENCES

[1] L. Berntzen, M. Rohde Johannessen and R. El-Gazzar, "Smart Cities, Big Data and Smart Decision-making - Understanding ´Big Data´ in Smart City Applications", Proceedings of ICDS 2018.

[2] ITU-T. *Focus Group on Smart Sustainable Cities.* [Online]. Available from: https://www.itu.int/en/ITU-T/focusgroups/ssc/Pages/default.aspx [retrieved: 2018.02.04].

[3] ITU-T Focus Group on Smart Sustainable Cities, "*Smart sustainable cities: An analysis of definitions*. [Online]. Available from: https://www.itu.int/en/ITU-T/focusgroups/ssc/Documents/website/web-fg-ssc-0100-r9-definitions_technical_report.docx, 2014. [retrieved: 2018.02.04].

[4] A. Vesco and F. Ferrero, Eds., Handbook of Research on Social, Economic, and Environmental Sustainability in the Development of Smart Cities. IGI Global, pp. xxv-xxxi, 2015.

[5] Deloitte. *Smart Cities: Big Data*. [Online]. Available from: https://www2.deloitte.com/content/dam/Deloitte/fpc/Documents/services/systemes-dinformation-et-technologie/deloitte_smart-cities-big-data_en_0115.pdf, 2015. [retrieved: 2018.02.04].

[6] F. Bonomi, R. Milito, P. Natarajan, and J. Zhu, "Fog Computing; A Platform for Internet of Things and Analytics," in Big Data and Internet of Things: A Roadmap for Smart Environments, N. Bessis and C. Dobre, Eds. Studies in Computational Intelligence, 546, Springer, pp. 169–186, 2014.

[7] V. Albino, U. Berardi, and R. M. Dangelico,. "Smart Cities: Definitions, Dimensions, Performance, and Initiatives," Journal of Urban Technology, 22(1), pp. 3–21, 2015.

[8] IBM. *A vision of smarter cities*. [Online]. Available from: https://www-03.ibm.com/press/attachments/IBV_Smarter_Cities_-_Final.pdf, 2010. [retrieved: 2018.02.04].

[9] M. Kehoe et al., Smarter Cities Series: A Foundation for Understanding the IBM Approach to Smarter Cities, IBM Redguides for Business Leaders, pp. 1–30, 2011.

[10] E. Negre and C. Rosenthal-Sabroux, "Smart Cities: A Salad Bowl of Citizens, ICT, and Environment," in Handbook of Research on Social, Economic, and Environmental Sustainability in the Development of Smart Cities, A. Vesco and F. Ferrero, Eds., IGI Global, pp. 61-78, 2015.

[11] H. Chen, R. H. L. Chiang, and V. C. Storey, "Business Intelligence and Analytics: From Big Data to Big Impact," MIS Quarterly, 36(4), pp. 1165–1188, 2012.

[12] D. Laney, 3D Data Management: Controlling data, volume, velocity, and variety. Technical Report. META Group, 2001.

[13] B. Marr, Big Data – Using Smart Big Data Analytics and Metrics to Make Better Decisions and Improve Performance. John Wiley & Sons Ltd, 2015.

[14] M. Botterman, Internet of Things: An Early Reality of the Future Internet, Workshop Report, European Commission, Information Society and Media Directorate, 2009.

[15] J. W. Treem and P. M. Leonardi, 7 Social Media Use in Organizations Exploring the Affordances of Visibility, Editability, Persistence, and Association. Communication Yearbook, 36, pp. 143–189, 2012.

[16] A. M. Kaplan and M. Haenlein, "Users of the world, unite! The challenges and opportunities of Social Media," Business Horizons, 53(1), pp. 59–68, 2010.

[17] Open Knowledge International. *Open Data Handbook*. [Online] http://opendatahandbook.org/guide/en/what-is-open-data/ [retrieved: 2018.02.04].

[18] J. Ojasalo and L. Tähtinen, "Integrating Open Innovation Platforms in Public Sector Decision Making: Empirical Results from Smart City Research," Technology Innovation Management Review, 6(12), pp. 38–48, 2016.

[19] S. Eräranta and A. Staffans, "From Situation Awareness to Smart City Planning and Decision Making," Proceedings of the 14th International Conference on Computers in Urban Planning and Urban Management (CUPUM 2015), J. Ferreira and R. Goodspeed, Eds., Paper 197, pp. 1-17, 2015.

[20] U. Passe et al., "Methodologies for Studying Human-Microclimate Interactions for Resilient, Smart City Decision-Making, " Proceedings of the 32nd International Conference on Passive and Low Energy Architecture, P. La Roche and M. Schiler, Eds., pp. 1735-1742, 2016.

[21] A. R. Honarvar and A. Sami, "A Multi-source Big Data Analytic System in Smart City for Urban Planning and Decision Making," International Conference on Internet of Things and Big Data, Doctoral Consortium (DCIT), pp. 32-36, 2016.

[22] Z. Khan et al., "Developing Knowledge-Based Citizen Participation Platform to Support Smart City Decision Making: The Smarticipate Case Study," Information, 8, 47, pp. 1-24, 2017.

[23] J.-P. Foucault and Y. Moulier-Boutang, "Towards economic and social 'sensors': Condition and model of governance and decision-making for an organological Smart City," International Conference on Smart and Sustainable City and Big Data (ICSSC), pp. 106-112, 2015.

[24] B. Nathali Silva, M. Khan, and K. Han, Big Data Analytics Embedded Smart City Architecture for Performance Enhancement through Real-Time Data Processing and Decision-Making. Wireless Communications and Mobile Computing, pp. 1–12, 2017.

[25] L. I. Gang and L. I. Yang, "Construction of Emergency Decision-making Intelligence System Against the Background of Smart City," Journal of Library Science in China, 3(4), 2016.

[26] F. Kurniawan, A. P. Wibawa, Munir, S. M. S. Nugroho, and M. Hariadi, "Makassar Smart City Operation Center Priority Optimization using Fuzzy Multi-criteria Decision-making," 4th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI), pp. 1-5, 2017.

[27] Smart Parking Ltd. *Company website*. [Online]. http://www.smartparking.com [retrieved: 2018.02.04].

[28] Fybr. *Company website*. [Online]. http://www.fybr.com [retrieved: 2018.02.04].

[29] WorldSensing. *Company website*. [Online]. https://www.worldsensing.com/industries/parking-operators/ [retrieved: 2018.02.04].

[30] SmartPark. *Company website*. [Online] https://smartpark.co.nz [retrieved: 2018.02.04].

[31] C. H. Schaffer, Customer Success Is Key – How a small manufacturer transformed an Internet of Things (IoT) solutions provider and unlocked $2 million is SaaS revenue. (Kindle edition) amazon.com, 2015. [retrieved: 2017.12.01].

[32] G. Kirankumar, J. Samsuresh, and G. Balaji, "Vehicle Speed Monitoring System [VSM] (Using RuBee Protocol)," IACSIT International Journal of Engineering and Technology, Vol. 4, No. 1, pp. 107-110, 2012.

[33] R. M. John et al., "Smart public transport system," International Conference on Embedded Systems (ICES), pp. 166-170, 2014.

[34] P. Chowdhury, P. Bala, and D. Addy, "RFID and Android based smart ticketing and destination announcement system," Advances in Computing Communications and Informatics (ICACCI), pp. 1-5, November 2016.

[35] R. Hawi, G. Okeyo, M. Kimwele, "Smart Traffic Light Control using Fuzzy Logic and Wireless Sensor Network," Computing Conference, London, pp. 450-460, 2017.

[36] K. Kumarmanas, S. Praveen, V. Neema, and S. Devendra, "An Innovative Device for Monitoring and Controlling Vehicular Movement in a Smart City," Symposium on Colossal Data Analysis and Networking (CDAN), pp. 1-3, 2016.

[37] A. Florea et al., "Low cost mobile embedded system for air quality monitoring - air quality real-time monitoring in order to preserve citizens' health," Sixth International Conference on Smart Cities, Systems, Devices and Technologies, (SMART), IARIA, pp. 5-12, 2017.

[38] C. Meinecke, Potentiale und Grenzen von Smart Metering. Springer. 2015.

[39] F. Folianto, Y. S. Low, W. L. Yeow, "Smartbin: Smart Waste Management System," IEEE Tenth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), pp. 1-2, 2015.

[40] A. S. Wiaya, Z. Zainuddin, and M. Niswar, "Design a smart waste bin for waste management," 5th International Conference on Instrumentation, Control, and Automation (ICA), pp. 62-66, 2017.

[41] H. Poddar, R. Paul, S. Mukherjee, B. Bhattacharyya, "Design Of Smart Bin For Smarter Cities," International Conference on Innovations in Power and Advanced Computer Technologies [i-PACT2017], IEEE Press, pp. 1-6, 2017.

[42] F. Wang, Ed., Geographic Information Systems and Crime Analysis, Idea Group Publishing, 2005.

[43] S. Chainey and J. Ratcliffe, GIS and Crime Mapping, Wiley, 2005.

[44] J. Vanus, J. Belesova, R. Martinek, J. Nedoma, M. Fajkus, P. Bilik and J. Zidek, "Monitoring of the daily living activities in smart home care," Human-centric Computing and Information Sciences, (7)30), pp. 1-34, 2017.

[45] H. Mshali, T. Lemlouma, M. Moloney and D. Magoni, "A survey on Health Monitoring Systems for Health Smart Homes," International Journal of Industrial Ergonomics, Elsevier (66), pp. 26-56, 2018.

[46] B. Kon, A. Lam and J. Chen, "Evolution of Smart Homes for the Elderly," Proceedings of the 2017 International World Web Conference, pp. 1095-1101, 2017.

[47] P. Leijdekkers, V. Gay and E. Lawrence, "Smart Homecare for Health Tele-monitoring," Proceedings of the First International Conference on the Digital Society, IARIA, 2007.

[48] S. Majumder, E. Aghayi, M. Noferesti, H. Memarzadeh-Tehran, T. Mondal, Z. Pang and M. J- Deen, "Smart Homes for Elderly Healthcare – Recent Advances and Research Challenges, Sensors, 17, 2496, 2017

[49] A. Katal, M. Wazid and R. Goudar, "Big data: issues, challenges, tools and good practices," Sixth International Conference on Contemporary Computing (IC3), pp. 404-409, 2013.

[50] P. Louridas and C. Ebert, "Embedded Analytics and Statistics for Big Data," IEEE Software, 30(6), pp. 33-39, 2013.

[51] J. Romero, S. Hallett, and S. Jude, "Leveraging Big Data Tools and Technologies: Addressing the Challenges of the Water Quality Sector," Sustainability, 9(12), 2160, 2017.

[52] S. Stieglitz, M. Mirbabaie, B. Ross, and C. Neuberger, "Social media analytics – Challenges in topic discovery, data collection, and data preparation," International Journal of Information Management," 39, pp. 156-168, 2018.

[53] Nosql-database.org, [Online] http://nosql-database.org/ [Retrieved 2018.09.01].

[54] A.S. Hashmi and T. Ahmad, "Big Data Mining: Tools & Algorithms. International Journal of Recent Contributions from Engineering, 4(1), pp. 36-40, 2016.

[55] F. Nasution, N. E. B. Bazin and A. Z. Daliyusmanto, "Big Data's Tools for Internet Data Analytics: Modelling of System Dynamics," International Journal on Advanced Science, Engineering and Information Technology 7(3), pp. 745-753, 2017.

[56] C. J. M. Tauro, S. Aravindh and A. B. Shreeharsha, "Comparative study of the new generation, agile, scalable, high performance NOSQL databases," International Journal of Computer Applications, 48(20), pp. 1-4, 2012.

[57] K. M. Anderson, A. A. Aydin, M. Barrenechea, A. Cardenas, M. Hakeem and S. Jambi, "Design Challenges/Solutions for Environments Supporting the Analysis of Social Media Data in Crisis Informatics Research," 48th Hawaii International Conference on System Sciences, pp. 163-172, 2015.

[58] Y. Huang, M. Lin and F. Yu, "Analysis on financial development correlativity between Zhejiang, Jiangsu, and Shanghai," WIT Transactions on Information and Communication Technologies 49, 2014.

[59] Oracle, "Oracle Berkeley," [Online] https://www.oracle.com/database/berkeley-db/db.html, [retrieved: 2018.09.01], 2018.

[60] K. Chodorow, "MongoDB: The Definitive Guide: Powerful and Scalable Data Storage, " O'Reilly Media Inc.", 2013.

[61] J. J. Miller, "Graph database applications and concepts with Neo4j," Proceedings of the Southern Association for Information Systems Conference, Atlanta, GA, USA, 2013

[62] A. Katal, M. Wazid, and R. H. Goudar. "Big data: issues, challenges, tools and good practices," Sixth International Conference on Contemporary Computing (IC3), IEEE, pp. 404–409, 2013.

[63] C. Fuchs, "From digital positivism and administrative big data analytics towards critical digital and social media research," European Journal of Communication, 32(1), pp. 37-49, 2017.

[64] Unbias Research Team, "Algorithmic Preference Survey," [Online] https://unbias.wp.horizon.ac.uk/2018/07/20/unbias-algorithmic-preference-survey/ [Retrieved 2018.09.01]

[65] K. D. Strang and Z. Sun, "Analyzing Relationships in Terrorism Big Data Using Hadoop and Statistics," Journal of Computer Information Systems, 57(1), pp. 67-75, 2017.

[66] R. G. Qiu, K. Wang, S. Li, J. Dong and M. Xie, "Big data technologies in support of real time capturing and understanding of electric vehicle customers dynamics," 5th IEEE International Conference on Software Engineering and Service Science (ICSESS), 2014.

[67] S. T. McAbee, R. S. Landis and M. I. Burke, "Inductive reasoning: The promise of big data," Human Resource Management Review, 27(2), pp. 277-290, 2017.

[68] A. Gandomi and M. Haider, "Beyond the hype: Big data concepts, methods, and analytics," International Journal of Information Management. 35(2), pp. 137-144., 2015.

[69] Digital Methods Initiative, Wiki [Online] https://wiki.digitalmethods.net/ [Retrieved 2018.09.01]

[70] Z. Xiang, Z. Schwartz, J. H. Gerdes Jr and M. Uysal, "What can big data and text analytics tell us about hotel guest experience and satisfaction?" International Journal of Hospitality Management, 44, 120-130, 2015.

[71] Ø. Sæbø, "Understanding Twitter Use among Parliament Representatives: A Genre Analysis," In E. Tambouris, A. Macintosh, & H. de Bruijn (Eds.), Electronic Participation (Vol. 6847, pp. 1-12): Springer Berlin / Heidelberg, 2011.

[72] M. Granath, "The Smart City – how smart can 'IT' be? : Discourses on digitalisation in policy and planning of urban development," Linköping University, [Online] http://liu.diva-portal.org/smash/record.jsf?pid=diva2%3A956501&dswid=-3723 [Retrieved 2018.09.01]

[73] A. Al-Rawi, "Online political activism in Syria: Sentiment analysis of social media," SAGE Research Methods Cases, 2017.

[74] J. O. Øye, "Sentiment Analysis of Norwegian Twitter Messages. (Master Thesis)," NTNU, Trondheim, 2014.

[75] H. Shirdastian, M. Laroche and M. O. Richard, "Using big data analytics to study brand authenticity sentiments: The case of Starbucks on Twitter," International Journal of Information Management, 2017.

[76] W. Medhat, A. Hassan and H. Korashy, "Sentiment analysis algorithms and applications: A survey," Ain Shams Engineering Journal, 5(4), pp. 1093-1113, 2014.

[77] M. Castells, "The New Public Sphere: Global Civil Society, Communication Networks, and Global Governance. The Annals of the American Academy of Political and Social Science, 616(1), pp. 78-93, 2008.

[78] J. Scott, "Social network analysis," Sage, 2017.

[79] A. Mazur, C. Doran and P. R. Doran, " The use of social network analysis software to analyze communication patterns and interaction in online collaborative environments," International Conference on Education, Training and Informatics, ICETI," Orlando, FL, 2010.

[80] J. Kim and M. Hastak, "Social network analysis: Characteristics of online social networks after a disaster," International Journal of Information Management, 38(1), pp. 86-96, 2018.

[81] Y. Wang, L. Kung, W. Y. C. Wang and C. G. Cegielski "An integrated big data analytics-enabled transformation model: Application to health care," Information & Management, 55(1), 2017.

[82] W. Ball, S. LeFevre, L. Jarup and L. Beale, " Comparison of different methods for spatial analysis of cancer data in Utah," Environmental Health Perspectives, 116(8), pp. 1120-1124, 2008.

[83] S. Sridharan, H. Tunstall, R. Lawder, and R. Mitchell, "An exploratory spatial data analysis approach to understanding the relationship between deprivation and mortality in Scotland," Social Science & Medicine, 65(9), pp. 1942-1952, 2007.

[84] Z. Huang, F. Cao, C. Jin, Z. Yu and R. Huang, "Carbon emission flow from self-driving tours and its spatial relationship with scenic spots – A traffic-related big data method," Journal of Cleaner Production, 142, 946-955, 2017.

[85] K. Xie, K. Ozbay, A. Kurkcu and H. Yang, "Analysis of Traffic Crashes Involving Pedestrians Using Big Data: Investigation of Contributing Factors and Identification of Hotspots," Risk Analysis, 37(8), pp. 1459-1476, 2017

[86] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," Science, 349(6245), pp. 255-260, 2015.

[87] P. Domingos, "A few useful things to know about machine learning," Communications of the ACM, 55(10), pp. 78-87, 2012.

[88] I. Portugal, P. Alencar and D. Cowan, "The use of machine learning algorithms in recommender systems: A systematic review," Expert Systems with Applications, 97, pp. 205-227, 2018

[89] C. Crisci, B. Ghattas and G. Perera, "A review of supervised machine learning algorithms and their applications to ecological data. (Report). Ecological Modelling, 240, 113, 2012.

[90] Z. Fan, S. Chen, L. Zha, L. and J. Yang, "A Text Clustering Approach of Chinese News Based on Neural Network Language Model," International Journal of Parallel Programming, 44(1), 198-206, 2016.

[91] J. P. Cunningham and Z. Ghahramani, "Linear dimensionality reduction: Survey, insights, and generalizations," The Journal of Machine Learning Research, 16(1), pp. 2859-2900, 2015

Distributed Situation Recognition in Industry 4.0

Mathias Mormul, Pascal Hirmer, Matthias Wieland, and Bernhard Mitschang

Institute of Parallel and Distributed Systems
University of Stuttgart, Universitätsstr. 38, D-70569, Germany
email: firstname.lastname@ipvs.uni-stuttgart.de

***Abstract*—In recent years, advances in the Internet of Things led to new approaches and applications, for example, in the domains Smart Factories or Smart Cities. However, with the advantages such applications bring, also new challenges arise. One of these challenges is the recognition of situations, e.g., machine failures in Smart Factories. Especially in the domain of industrial manufacturing, several requirements have to be met in order to deliver a reliable and efficient situation recognition. One of these requirements is distribution in order to achieve high efficiency. In this article, we present a layered modeling approach to enable distributed situation recognition. These layers include the modeling, the deployment, and the execution of the situation recognition. Furthermore, we enable tool support to decrease the complexity for domain users.**

Keywords–Industry 4.0; Edge Computing; Situation Recognition; Distribution Pattern.

I. Introduction

This article is a revised and extended version of the SMART 2018 paper "Layered Modeling Approach for Distributed Situation Recognition in Smart Environments" [1]. The emerging paradigm Industry 4.0 (I4.0), describing the digitization of the manufacturing industry, leads to the realization of so-called Smart Factories [2]. In I4.0 and Internet of Things in general, devices equipped with sensors and actuators communicate with each other through uniform network addressing schemes to reach common goals [3][4]. One of these goals is situation recognition, which enables monitoring of I4.0 environments and, consequently, the timely reaction to occurring situations. For example, the occurrence of a machine failure in a Smart Factory, recognized by sensors of the machine, could lead to an automated notification of maintenance engineers.

Situations are recognized using context data that is usually provided by sensor measurements. In current approaches, such as the one we introduced in our previous work [5][6], situations are recognized in a monolithic IT infrastructure in the cloud. Consequently, involved context data needs to be shipped to the processing infrastructure in order to recognize situations. However, especially in domains where efficiency is of vital importance, e.g., Smart Factories, this approach is not feasible. In order to fulfill important requirements, such as low network latency and fast response times, the situation recognition needs to be conducted as close to the context data sources as possible and, therefore, in a distributed manner. Processing data close to the sources is commonly known as Edge Computing [7].

In this paper, we introduce an approach to enable a distributed situation recognition. By doing so, we introduce so-called *distribution patterns*. These patterns represent common ways to distribute the recognition of situations, i.e., exclusively in the *edge*, in on-premise or off-premise cloud infrastructures, or based on a hybrid approach. We provide a layered approach for modeling and executing the situation recognition based on these distribution patterns. Our approach builds on a set of requirements we derive from a use case scenario in the manufacturing domain. We validate the approach by applying it to our previous non-distributed situation recognition [5][6] that is based on the modeling and execution of so-called Situation Templates [8]. Furthermore, we introduce a modeling tool for Situation Templates as well as an automated distribution of the templates in the edge and backend cloud.

This article is a revised and extended version of the SMART 2018 paper "Layered Modeling Approach for Distributed Situation Recognition in Smart Environments" [1]. In addition to the previous paper, we describe how distributed situation recognition can be realized from the modeling of the situation using Situation Templates to the actual deployment. This is done by the introduced tool-based modeling support and the automated distribution of Situation Templates among the edge and backend cloud.

The remainder of this paper is structured as follows: Section II describes related work and foundational background. In Section III, we introduce a motivating scenario, which is used to derive requirements for our approach. In Section IV, we present the main contribution of our paper. Section V describes the process from modeling Situation Templates using a tool-based modeling support to the automated distribution of the situation recognition. Finally, Section VI concludes the paper and gives an outlook to future work.

II. Related Work and Background

In this section, we describe related work, as well as foundational concepts of our previous work that are necessary to comprehend our approach.

A. Related Work

In related work, approaches exist for distributed situation recognition using ontologies, e.g., by Fang et al. [9]. These approaches do not achieve the latency required in real-time critical scenarios, such as Industry 4.0 [2], due to time-consuming reasoning. The goal of our approach is to achieve low latency for distributed situation recognition in the range of milliseconds. Many approaches using ontologies are in the range of seconds to minutes, even without distribution [10], [11]. Using machine learning leads to similar limitations regarding latency [12].

In the area of distributed Complex Event Processing (CEP), Schilling et al. [13] aim at integrating different CEP systems

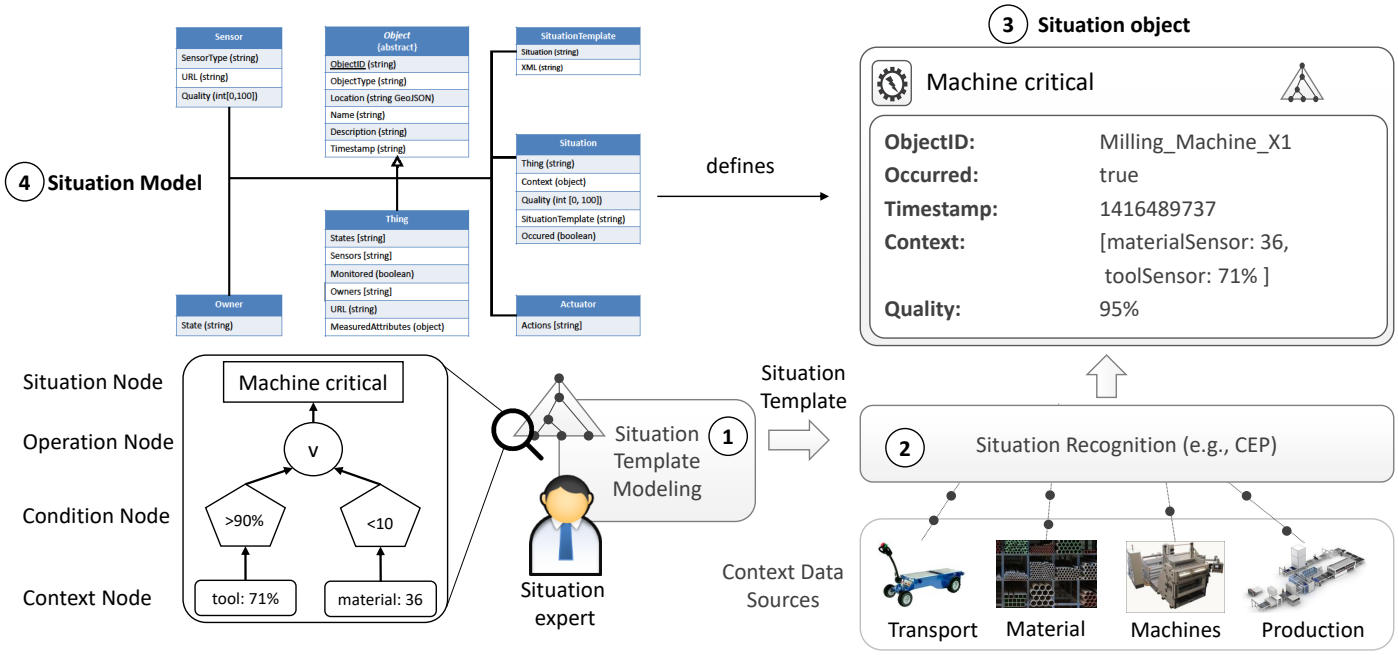


Figure 1. Previous approach for situation recognition [1]

using a common meta language. This allows to use different CEP systems and integrate the results. This could be beneficial for our distribution because we would not be limited to one execution environment. However, in [13], the queries have to be hand-written and distributed. This is difficult, especially for domain experts, e.g., in Industry 4.0, who do not have the necessary skillset. In our approach, we provide an abstraction by Situation Templates that can be modeled using a graphical user interface. Furthermore, the users are supported in splitting up these template as well as in the distribution decision.

Other approaches in distributed CEP, e.g., by Schultz-Moller et al. [14], follow the concept of automatic query rewriting. Here, CEP queries are split up using automated rewriting and are distributed on different operators based on a cost model, which is mostly based on CPU usage in the different nodes. In our approach, we want to support the user to select the desired distribution type. Since there are many aspects, such as data protection or security, that play a role in distributing the CEP queries correctly, this only can be known by a responsible expert user.

Finally, approaches exist that enable a massive distribution of sensors, e.g., by Laerhoven and Gellersen [15] in cloths, to detect activities of the person wearing the cloth. This is similar to detecting the situation in the edge cloud, but there is no concept presented in [15] to integrate the activities with other activities from different edge clouds or create a global situation involving different locations.

B. Background

In this section, we describe our previous work. Our first approach for situation recognition, this paper builds on, is depicted in Figure 1. This approach is a result of the issues of related work, as discussed in the previous section.

An important fundamental concept are Situation Templates, introduced by Häussermann et al. [8]. We adapted the Situation Templates in [6] to model and recognize situations. Situation Templates (see Figure 1 on the buttom left) consist of *context, condition* and *operation* nodes, which are used to model specific situations. Context nodes describe the input for the situation recognition, i.e., the context data, based on the definition of Dey et al. [16]. Context nodes are connected to condition nodes, which define the conditions for a situation to be valid. Operation nodes combine condition and operation nodes and represent the logical operators *AND, OR*, or *XOR*. Operation nodes are used to aggregate all condition nodes of the Situation Template into a single node, the situation node.

After modeling a Situation Template (Figure 1, Step 1), it is transformed into an executable representation (not depicted), which is realized using CEP or light-weight execution languages, such as Node-RED. The advantage of this transformation is that it provides a flexible means to recognize situations. These transformations can be found in [17][18]. Consequently, we are not limited to specific engines or data formats. Once the transformation is done, the executable Situation Template is handed over to the corresponding execution engine.

On execution (Figure 1, Step 2), context data originating from the context sources is validated against the conditions defined by the Situation Template, for example, through pattern recognition in CEP. On each validation, we create a so-called situation object [19], defining whether the situation occurred and containing the involved context data (Figure 1, Step 3). We created a Situation Model [19] (Figure 1, Step 4) to define the attributes of those situation objects. This leads to a better understanding of how context data led to the situation.

This previous approach for situation recognition works well, however, there are still some limitations this paper aims to solve. First, the current approach was built to monitor single things (e.g., devices). However, as the complexity of nowadays IT infrastructure rises, means need to be enabled to monitor more than one thing using the introduced Situation Templates.

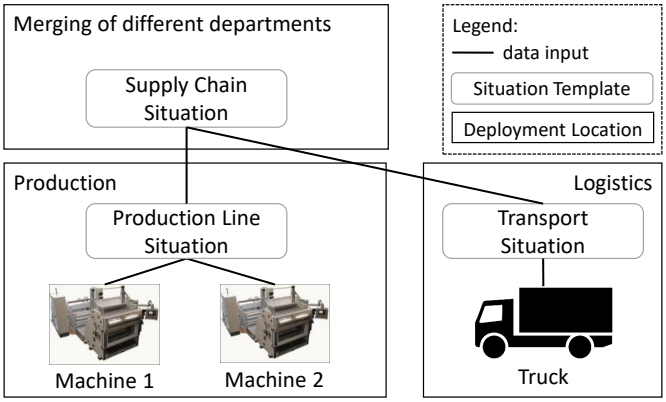


Figure 2. Motivating scenario for distributed situation recognition [1]

Furthermore, currently, the Situation Templates are executed in a monolithic manner because in former scenarios, distribution was not necessary. In current approaches, e.g., involving I4.0, however, this is necessary. Therefore, in this paper, we aim for enhancing our approach in order to be more fitting to recent scenarios.

III. Scenario and Requirements

In this section, we introduce a practical motivating scenario from the I4.0 domain, which is used throughout the paper to explain our approach. In the scenario, depicted in Figure 2, a specific part of the supply chain of a production company should be monitored. As depicted, there are several entities involved: (i) production machines, assembling products based on parts, and (ii) trucks, delivering the parts to be assembled. The monitoring should detect critical situations that could occur, for example, the failure of at least one of the machines, or a delivery delay of parts, e.g., caused by issues with trucks or with the supplier. Situations that could occur are: (i) *Production Line Situation*, indicating that one of the production machines is in an erroneous state, (ii) *Transport Situation*, indicating a problem with the truck, and (iii) *Supply Chain Situation*, indicating a problem with either the production line or the truck.

When applying our previous approach, described in Section II, to this scenario, new requirements arise that need to be coped with. We divide these requirements into ones that concern the modeling of Situation Templates and ones that concern the execution of the situation recognition. We derived eight requirements R_1 to R_8 for this scenario.

Modeling Requirements

Three requirements focus on the modeling of the situation recognition using Situation Templates.

- R_1 **- More powerful Situation Templates:** With our previous approach (cf. Section II-B), single machines can be monitored in an efficient way as evaluated in [6], which was sufficient for previous scenarios. However, in recent scenarios involving Industry 4.0, the requirements are increasing. In our motivating scenario, it is important to model dependencies between multiple entities within a single Situation Template, e.g., to recognize the *Production Line Situation*.
- R_2 **- Low modeling complexity:** In our previous approach, modeling Situation Templates involving a lot of context data has led to a cumbersome task and, consequently, to a high complexity and error-prone modeling. To cope with this issue, a new modeling approach is required that enables the reutilization of already existing Situation Templates to lower the modeling complexity of new Situation Templates.
- R_3 **- Domain-independence:** A consequence of the issue described in R_2 is domain-dependence. Large Situation Templates usually consist of a wide range of context data sources, e.g., Trucks or the Production Line of our scenario. However, these context data sources require domain experts of these specific areas. Consequently, Situation Templates need to be modeled by these experts together. This leads to high costs due to the communication overhead. Hence, our goal is to enable domain-independence for Situation Template modeling.

Execution Requirements

- R_4 **- Low latency:** In many domains, latency plays a crucial role. Especially in Smart Factory environments, the industrial automation layer has strong requirements regarding end-to-end latency up to 1 ms or even lower [20]. Therefore, the execution of the situation recognition needs to adapt to those requirements, so that critical situations like machine failures can be recognized in a timely manner.
- R_5 **- Low network traffic:** In modern scenarios, large amounts of data are produced that need to be stored and processed in order to recognize situations. For example, an autonomous car produces about 35 GB/hour of data [21]. In comparison, Budomo et al. [22] conducted a drive test and recorded a maximum and minimum upload speed of 30Mbps (13.5 GB/hour) and 3.5Mbps (1.58 GB/hour), respectively, using the current mobile communication standard LTE-A. Therefore, transferring all data of an autonomous car to the cloud is currently impossible. Consequently, reducing the network traffic is an important issue when recognizing situations.
- R_6 **- Reduced costs:** Costs are always an essential factor when it comes to data processing. Operating and maintaining an in-house IT infrastructure could lead to high costs. In contrast, using the pay-as-you-go approach of Cloud Computing, costs could be reduced. Enabling the lowest possible costs when recognizing situations is an important requirement for this paper.
- R_7 **- Data security & privacy:** Especially the processing of company data needs to be secure and, furthermore, privacy needs to be ensured. However, especially when processing data in the Public Cloud, companies need to trust the Cloud providers that they provide the security they require. Alternatively, companies can keep their data close, i.e., in a trusted environment. Additionally, in many countries and federations, data protection directives are in place to guarantee the protection and privacy of personal data that may not allow to send personal data to the cloud [23].

- R_8 **- Cross-company situation recognition:** Modern products and their components are rarely built completely by one company. Therefore, most actual scenarios are very complex, involve multiple companies, and require a cross-company situation recognition. Our motivating scenario in Figure 2 can be regarded as such an example, in which a manufacturing company cooperates with a logistics company. For example, a delayed delivery caused by a failure of the truck must be communicated to the manufacturing company. Consequently, our situation recognition approach needs to enable a cross-company situation recognition.

IV. Distribution of Situation Recognition

In our previous work, we already solved challenges regarding sensor registration and management [24], efficient solutions for a situation recognition [18], and the management of recognized situation [19]. Now, we concentrate on extending our previous approach by introducing a distribution of the situation recognition to fulfill the above-mentioned requirements R_1-R_8. For this, we first present (i) the modeling improvements for our approach to support the distribution we aim for. On this basis, we present (ii) the execution improvements to enable the distribution based on three distribution patterns including a decision support for each of those patterns.

The distributed situation recognition was implemented based on the existing prototype of our previous work, introduced in [18][19] by following adaptations: (i) the modeling for Situation Templates was extended, (ii) the transformation was enhanced to accept multiple things, and (iii) the communication between the distributed locations is enabled by messaging systems.

A. Modeling Improvements

In the following, we present the improvements regarding the modeling of Situation Templates to fulfill the requirements R_1-R_3. The extension of the Situation Templates, i.e. its schema, comprises (i) the modeling of multiple things within a single Situation Template, and (ii) a layered modeling by reutilizing already modeled Situation Templates. These extensions are depicted in Figure 3. Requirement R_1 describes the need for the modeling of more powerful situations, e.g., *Production Line critical*. However, a production line itself does not contain any sensors but rather describes the coherence and arrangement of multiple machines. Therefore, to model a situation describing the production line, we need to model all machines of the production line into a single Situation Template. By extending the Situation Template Schema to allow the modeling of multiple things, therefore, we fulfill requirement R_1.

It is obvious that from a certain amount of things in a single Situation Template and each thing having multiple sensors, the complexity of modeling such a Situation Template is becoming a problem. An excessive complexity restricts the usability of our modeling approach, hence, the reduction of the modeling complexity is required (cf. R_2). To cope with the increasing complexity of Situation Templates, we introduce the layered modeling approach. Instead of modeling everything within a single Situation Template, we use situations as context input for further situation recognition. Thereby, we implicitly reuse already modeled Situation Templates. Furthermore, we divide situations in two classes: *local situations* and *global situations* whereby local situations are recognized at the edge and global situations in the cloud. An equivalent modeling of the situation *Production Line critical* using the layered modeling approach is shown in Figure 3 (right side). Based on this comparison, we show the benefits of this approach:

- **Reusability:** By using situations as input, we reutilize existing Situation Templates. When modeling a global situation, we only need to model the relation between the already modeled local situations similar to putting together building blocks. A further advantage is that the local Situation Templates possibly were already used and tested for correctness, which lowers the error-proneness for modeling global situations.
- **Reduce complexity:** The reusability directly leads to less complex Situation Templates, since the modeling is based on the Divide and Conquer paradigm. By using the layered modeling approach, we fulfill the requirement R_2.
- **Distribution:** Since we do not have one single and complex Situation Template, but instead, multiple smaller ones, we already have a beneficial starting point for the distribution of the situation recognition as we can simply execute the different Situation Templates at different locations.
- **Support for specific domains:** Having multiple things within a single Situation Template could lead to the problem that knowledge from different domains is required. For example, our motivating scenario contains three domains - manufacturing, logistics, and their dependencies. Using the layered modeling approach, different domains can model Situation Templates independently. Consequently, the requirement R_3 is fulfilled as well.

As a result, by introducing an extended Situation Template Schema to enable the modeling of multiple things within a single Situation Template and the layered modeling approach, we fulfill all modeling requirements R_1-R_3.

B. Execution Improvements

The modeling improvements we presented in the last section serve as the foundation for the distribution of the situation recognition. As mentioned above, in our previous approach, the situation recognition was executed centralized in the cloud. Hence, all context data was sent to this cloud and was used as input for the situation recognition. However, lately, the term *Edge Computing* gains more and more attention. Shi et al. [7] define *the edge* as *"any computing and network resources along the path between data sources and cloud data centers"*. Therefore, in our context, Edge Computing refers to the processing of context data close to the data sources.

By introducing Edge Computing to our approach, a distribution of the situation recognition to the cloud and the edge can be performed. In the scenario of Figure 2, the distribution of the situation recognition seems obvious. Using the layered modeling approach, we can model the local situations *Production Line Situation* and *Transport Situation* and the global situation *Supply Chain Situation*. The situation recognition for the local situation is executed at the edge, i.e., locally in the factory or truck, respectively. The global situation is executed

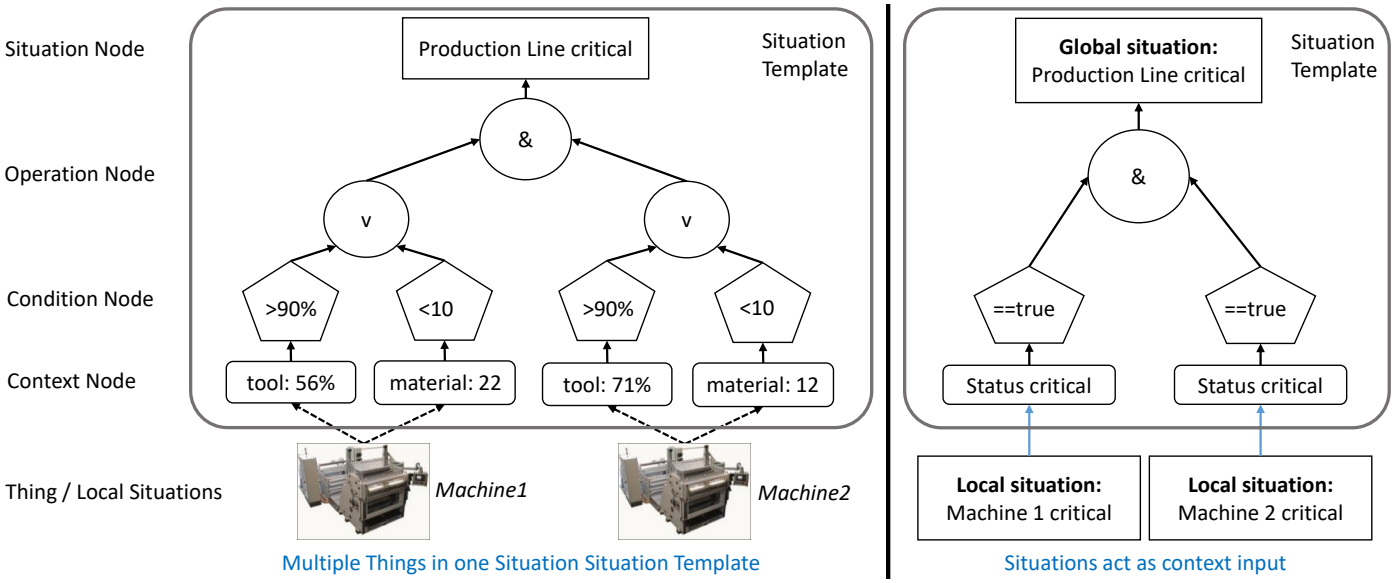


Figure 3. Modeling improvements for Situation Templates (legend see Figure 4) [1]

in the cloud and receives the local situations as input. However, based on the execution requirements R_4-R_8, this distribution might not always be ideal.

Therefore, in the following, we present the execution improvements resulting from the distribution of the situation recognition. First, we present the concept of context stripping and its benefits. Afterwards, we introduce three distribution patterns and a decision support for choosing the most suitable distribution pattern for a certain scenario.

1) Context Stripping: As presented in Section II-B, when a situation recognition is executed, situation objects are created that are defined by the Situation Model [19]. This situation object contains all context data that were used for the evaluation of this specific situation. In [19], we approximated the data volume of situation objects based on the amount of used context data. The results showed that the appended context data presents the majority of the data size of a situation object. Now, when using the layered modeling approach, we may use local situations that we recognized at the edge as input for the recognition of global situations in the cloud. That causes us to send all context data to the cloud again within the situation object. However, based on the scenario, we might not be interested in the context data of a situation object but only if the local situation occurred or not, so we can evaluate the global situation. Therefore, we introduce the concept of *context stripping*. By using context stripping, the context used for the situation recognition is not sent within the situation object. It only contains the most vital data for a further situation recognition in the cloud. Therefore, content-wise, a local situation only contains a boolean value, which describes if the local situation occurred or not and the required meta data for further processing.

This leads to a trade-off the user has to make based on his requirements. By using context stripping, the data size of a situation object can be strongly reduced. However, the context data that led to the evaluation of a specific situation object is discarded after processing. In our first approach, we explicitly wanted to store the context data within situation objects for a detailed historization of situations. This historization, for example, can be used afterwards for a root cause analysis of detected situations based on the involved context data. We are planning to conduct a performance evaluation regarding the degree of context stripping to be used in specific scenarios.

2) Distribution Patterns: As mentioned above, the distribution of the situation recognition is dependent on the execution requirements R_4-R_8. Therefore, a general solution for the distribution of the situation recognition is not possible. Instead, we introduce three different distribution patterns, depicted in Figure 4 based on the scenario shown in Figure 2. The *Type I* distribution pattern describes our previous approach. All context data, i.e., in this scenario, context data from a truck and two machines, are sent to the cloud. The situation recognition is executed in the cloud and all context data is available. In contrast, the *Type II* pattern describes the execution of the situation recognition at the edge, close to the data sources. In this case, it is often impossible to gather all context data from all sources, e.g., from the truck, since it is not part of the local network of the factory, where the machines are located. Therefore, only parts of the situation recognition may be executed at the edge. The *Type III* pattern is a hybrid solution based on both the *Type I* and *Type II* pattern and enables the execution of situation recognition at the edge, which results in local situations (i.e., *Production Line* and *Transport*) and the execution of situation recognition in the cloud, where the local situations are used to evaluate the global situation.

In the following, the different distribution patterns are described in more detail with regard to the execution requirements R_4-R_8. Each pattern comprises advantages for certain use cases and might not fulfill every execution requirement. Additionally, the presented distribution patterns are applicable to the distribution of data processing in general.

Type-I: Cloud-only (Figure 4, left)
Despite many advantages of Edge Computing, the Type-I distribution pattern still is a viable option. Introducing

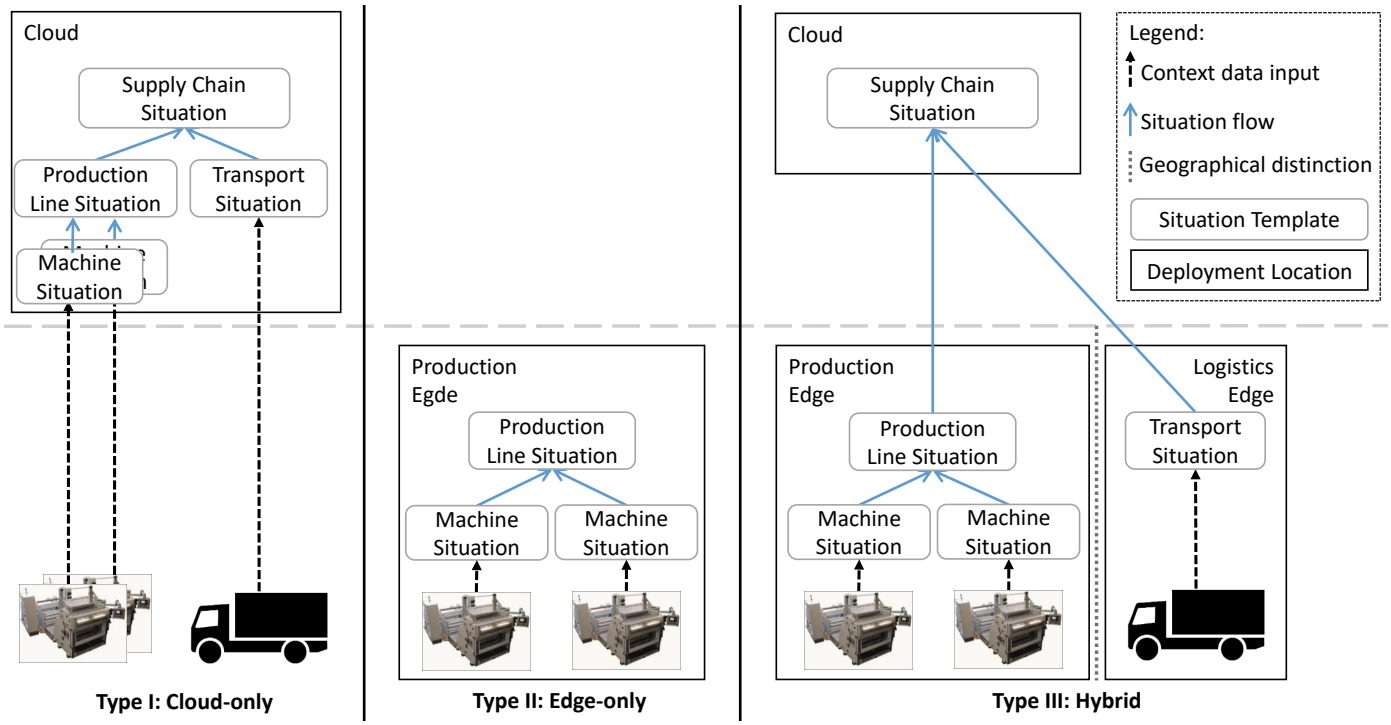


Figure 4. Distribution patterns [1]

Edge Computing is no trivial task and comprises multiple challenges [7]. Companies with low IT experience or no IT department benefit from outsourcing IT infrastructure and expertise to third-party cloud providers. This oftentimes is the case for SMEs, which then can solely focus on their products and the pay-as-you-go model provides a cost-effective and scalable infrastructure. Type I has the following implications regarding our requirements:

- R_4 **- Low latency:** Currently, when using an off-premise cloud, the requirement of 1 ms is already violated by the network latency itself. Therefore, requirement R_4 cannot be fulfilled.
- R_5 **- Low network traffic:** Since all context data must be sent to the cloud first, network traffic cannot be reduced. Requirement R_5 is not fulfilled.
- R_6 **- Reduced costs:** The calculation of costs is always very use case specific. If a company already outsourced its IT infrastructure to the cloud, then the introduction of Edge Computing results in new costs for hardware and IT staff. For scenarios, in which the data amount is relatively small or irregular, the gained advantages may not be worth the expenses. Therefore, the Type-I pattern can fulfill requirement R_6.
- R_7 **- Data security & privacy:** Since all context data is sent to the cloud, new security risks are introduced. Furthermore, company policies might prohibit sending sensitive or personal context data to the cloud. Therefore, requirement R_7 is not fulfilled.
- R_8 **- Cross company situation recognition:** Since all data is available in the cloud, companies can work together to execute a collaborative situation recognition. Requirement R_8 is fulfilled.

As shown, the Type-I pattern does not fulfill most requirements. Still, in non-critical scenarios where high latency is acceptable, the network traffic is low or fluctuating and the data is allowed to be sent to the cloud by the companies' policies or government regulations, the Type-I pattern is a sensible option. Especially for SMEs, the cost model of a public cloud is very attractive in comparison to self-managed data centers [25].

Type-II: Edge-only (Figure 4, middle)
In comparison, the Type-II distribution pattern describes the execution of the whole situation recognition at the edge. As already mentioned, this is only possible if all context data is available at the edge. Therefore, the situation recognition of local situations is best-suited for an edge-only execution. Type II has the following implications regarding our requirements:

- R_4 **- Low latency:** Yi et al. [26] show that latency can be reduced by 82% by moving an application to the edge of the network. As the situation recognition is executed as close as possible to the data sources, the requirement R_4 is fulfilled. With an execution time of 3ms for our situation recognition [18], the overall latency is kept comparably low.
- R_5 **- Low network traffic:** No context data is sent to the cloud, therefore, network traffic stays low and requirement R_5 is fulfilled.
- R_6 **- Reduced costs:** Floyer [27] presents a scenario of a wind-farm to project potential costs savings by additionally using Edge Computing with Cloud Computing instead of a cloud-only solution. An assumed 95% reduction in network traffic results in a cost reduction of about 64%, already including the on-site equipment for Edge Computing. For a cost-effective usage of Edge Computing, the data of the wind-farm had to be

reduced by at least 30% at the edge in order to reduce the costs for network traffic and thereby lower the overall costs. However, continuous IT staff and management of the on-site equipment as well as security measurements are not included. Therefore, again, costs are strongly use case dependent. Introducing Edge Computing does not increase the costs in general but they depend on the amount of saved network traffic. Therefore, requirement R_6 can be fulfilled.

- R_7 **- Data security & privacy:** One of the main concerns regarding the adoption of Cloud Computing still is security, especially in companies with few experience with Cloud Computing. Security and privacy of data is increased, since all context data and situations remain at the edge, i.e., a local network controlled by its company. Requirement R_7 is fulfilled.
- R_8 **- Cross company situation recognition:** In general, the data sources of different companies are geographically distributed and not in the same local network. Therefore, a cross company situation recognition is not possible. Requirement R_8 is not fulfilled.

Most requirements are fulfilled. However, more complex scenarios (cf. Figure 2) cannot be mapped to this pattern because of geographically distributed data sources. Therefore, the Type-II distribution pattern is best suited for company-internal situation recognition that fulfills critical requirements, such as latency and security. Especially in mobile environments, e.g., an autonomous truck, with high-volume data, the Type-II pattern is a good option.

Type-III: Hybrid (Figure 4, right)

Neither a Type-I nor a Type-II distribution pattern presents a viable option for our motivating scenario, since the truck produces too much data for a cloud-only solution and the geographical distribution of the data sources prevents an edge-only solution. Therefore, in the Type-III distribution pattern, the situation recognition is distributed to both the cloud and the edge. This leads to the recognition of local situations at the edge and global situations in the cloud and their advantages.

- R_4 **- Low latency:** The latency for local situations is reduced as described in Type-II. However, global situations are evaluated in the cloud and the latency is as described in Type-I. Therefore, the requirement R_4 is fulfilled only for local situations.
- R_5 **- Low network traffic:** As in Type-II, network traffic can be saved by shifting the situation recognition to the edge. The situation objects of the local situations must be sent to the cloud for the evaluation of global situations, thereby increasing network traffic. However, by using context stripping, the data size of situation objects can be massively reduced and still enable further processing of global situations. Therefore, requirement R_5 is fulfilled.
- R_6 **- Reduced costs:** The potential cost savings correspond to the cost savings of Type-II. However, by using context stripping for local situations, we reduce network traffic and thereby costs and still enable a situation recognition for complex scenarios. Therefore, requirement R_6 can be fulfilled.
- R_7 **- Data security & privacy:** Security and privacy of local situations match the Type-II pattern. Again, when using context stripping for local situations, we support complex scenarios and do not have to send sensitive context data within situation objects to the cloud. Therefore, R_7 is fulfilled.
- R_8 **- Cross-company situation recognition:** As in the Type-I distribution pattern, a collaborative situation recognition is possible. However, a big advantage is gained by using context stripping. Possibly sensitive context data of each company remains at their respective edge. Only context-stripped local situations are sent to the cloud for the collaborative evaluation of the global situation. Therefore, requirement R_8 is fulfilled.

TABLE I. FULFILLMENT OF EXECUTION REQUIREMENTS BY THE DISituation TemplateRIBUTION PATTERNS

	R_4	R_5	R_6	R_7	R_8
Type-I: Cloud-only	X	X	(✓)	X	✓
Type-II: Edge-only	✓	✓	(✓)	✓	X
Type-III: Hybrid	✓	✓	(✓)	✓	✓

Except reducing the latency for the evaluation of global situations, all requirements are fulfilled by this hybrid approach. Especially the usage of context stripping presents multiple advantages when transferring local situations to the cloud. The Type-III distribution pattern is best-suited for complex scenarios with multiple data sources that require a fast reaction to local situations and a centralized situation recognition of global situations without increasing the network traffic. Multiple companies can collaborate without sharing sensitive data or infringing government regulation.

Table I summarizes the analysis of the different distribution patterns. As shown, the Type-III hybrid approach fulfills all execution requirements. However, the potential costs are very use case specific and cannot be generalized (therefore, depicted in brackets). Consequently, if the fulfillment of all execution requirements is not mandatory, choosing a different pattern might be more cost-effective.

V. From Modeling to Deployment

In this section, we describe how distributed situation recognition can be realized from the modeling of the situation using Situation Templates to the actual deployment in order to recognize the modeled situation.

A. Tool-based modeling support

In Section IV-A, we present the layered modeling approach, resulting in modeling improvements to enable the reusability and distribution of Situation Templates. However, especially modeling complex Situation Templates is still a cumbersome and error-prone task without any tool support. To prevent this, we introduce the Situation Template Modeling Tool (STMT), a graphical web-based tool to support users with the modeling of Situation Templates. The previously introduced Situation Template Schema ensures the validity of a Situation Template and is integrated in the STMT to ensure the modeling of valid Situation Templates. Figure 5 depicts the modeling of the Situation Template *Production Line critical*. For illustration purposes, in contrast to the previously introduced Situation Template, one situation input was changed to a sensor input to

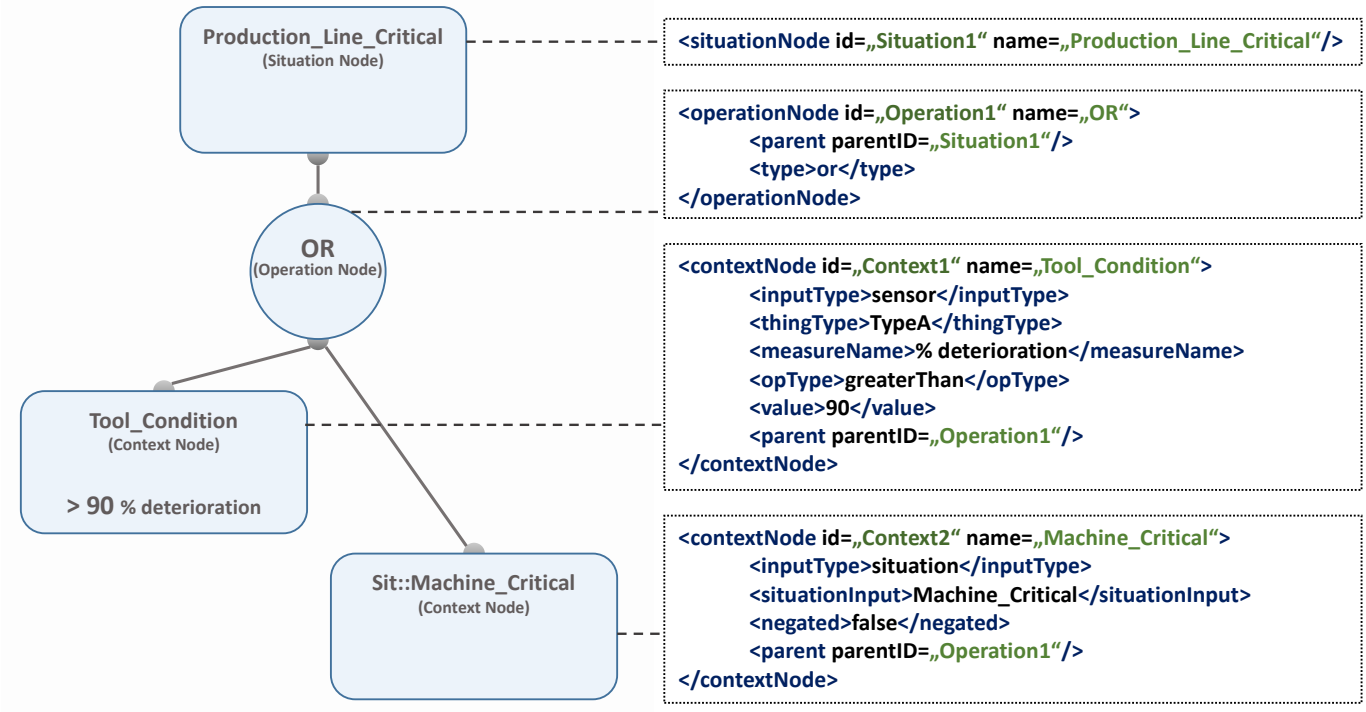


Figure 5. Modeling a Situation Template with a situation as context input and corresponding XML snippets

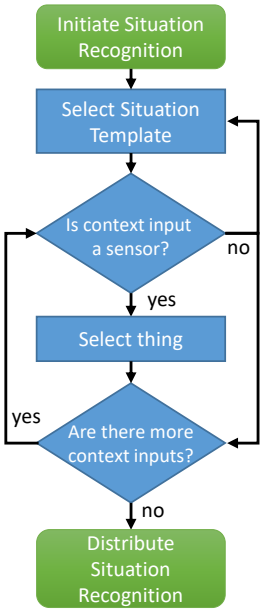


Figure 6. Flowchart for initializing the situation recognition

show the syntactical differences between a situation and a sensor input. As depicted, a Situation Template has a tree structure whereby the root node is the situation node. The only child of a situation node must be an operation node and represents one of the logical operators *AND, OR* or *XOR*. This operation node combines multiple children, which are either context nodes or additional operation nodes. The leaf nodes always constitute context nodes, either sensor input (*CPU_Load)* or situations (*Sit::Machine_Critical)*. Furthermore, on a conceptual layer, we divided the context nodes and condition nodes into separate nodes. However, for more clarity, we combined both nodes into a single one in the STMT.

On the right side of Figure 5, the corresponding XML snippets of the nodes are shown. Situation nodes and all other nodes contain an *id* and the *name* of the node. The id is used for the internal linking of nodes within a Situation Template. The name of the situation node is carried on as the name of the resulting situation object. The operation node further contains the element *parent* to enable the linking of nodes using the id. The element *type* describes the modeled logical operator. As mentioned, context nodes can have either sensor input or situations as input. This option is defined by the element *inputType*. Using a sensor input, the next element is *thingType*. Again, Situation Templates are generic and not defined for a specific thing. Therefore, we define *thing types*, i.e., a class of structurally identical things for which the same Situation Template can be used, since the things possess the same sensors. The condition that has to be met by the sensor input is defined by the elements *value*, i.e., the threshold, and *opType*, i.e., the operation type. The element *measureName* is for visualization purposes only. In comparison, a context node using situations as input constitutes the element *situationInput*. Since a situation is defined by a Situation Template, the value of this element refers to the name of a Situation Template. Using the element *negated*, we have the possibility to negate the Boolean value of a situation object. Additionally, to support the modeler, a database is connected to present possible thing types as well as previously modeled Situation Templates. Afterwards, Situation Templates can be stored in and loaded from a Situation Template repository.

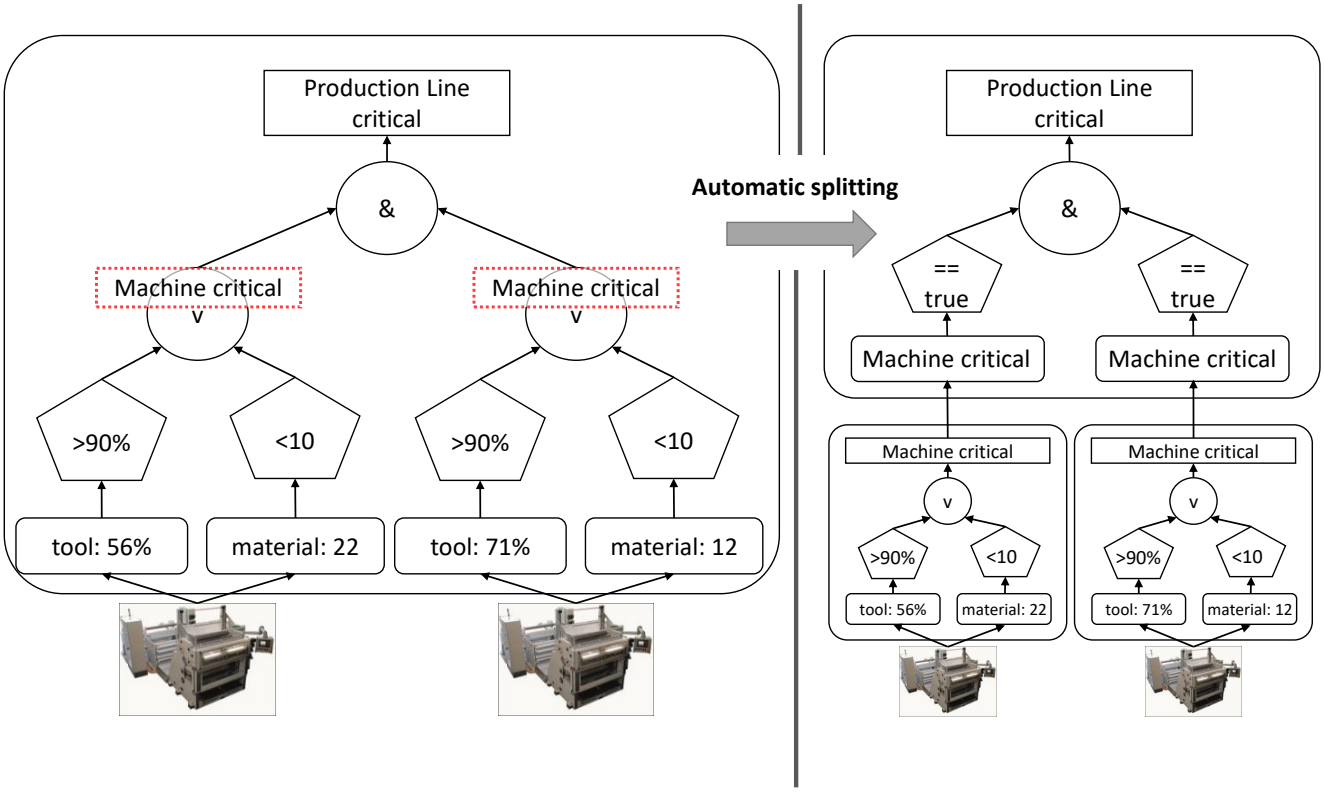


Figure 7. Automatic Splitting of Situation Templates

B. Initiating the Situation Recognition

With the STMT, it is possible to easily model complex Situation Templates. In the following, we present the procedure of initiating the situation recognition based on the modeled Situation Template. Figure 6 depicts the flowchart starting with the selection of a Situation Template. In our previous work, only one thing per Situation Template was allowed, therefore, the next step was the selection of a thing and the process was finished. Now, multiple things as well as situations can be used as input and the process must be changed accordingly. We traverse all context nodes and check their input type. If it is a sensor input, the corresponding thing must be selected that provides the sensor input. After that, the next context node is regarded. If the context node has a situation as input, the corresponding Situation Template is selected and the next context node of the former Situation Template is regarded. This newly selected Situation Template might contain situations as input as well, therefore, we pass through a recursive process. The process is finished when all things that are needed for the situation recognition are selected. This process describes a clean start scenario whereby no situation recognition is running and, therefore, the situation recognition for all Situation Templates has to be initiated.

C. Distributing to Edge and Cloud

In our extended approach, the Situation Templates are modeled using the STMT and the deployment can be initiated from the tool. The last step is the actual distribution to the edge and cloud. To support the user as much as possible, the distribution process can be divided into two steps: 1) automatic splitting, and 2) module distribution, which are introduced in the following.

1) Automatic Splitting: As shown in Figure 7, there are two ways to model a Situation Template. On the left side of the figure, the situation *Production Line critical* is modeled within one Situation Template. On the right side, the same situation is modeled by using the Layered Modeling Approach. By creating separate Situation Templates, we enable the distribution of those Situation Templates to the edge and the cloud. However, in simple scenarios like this one, the modeling on the left side of Figure 7 might be faster, easier and more intuitive. Still, to gain the advantages of a distributed situation recognition even when the user models a single Situation Template, we introduce *Automatic Splitting*. The splitting mechanism detects, if possible, *local* situations (i.e., *Machine critical* in Figure 7; left side) that can be extracted and splits the Situation Template into smaller ones (Figure 7; right side). This method can only be executed after the initialization, i.e., after selecting the things for the Situation Template. Since Situation Templates are modeled in a generic way, only then it is known, which context input belongs to which thing and, therefore, if context inputs originate from the same source, e.g., the same edge node. A prerequisite is that each thing contains meta data about its edge environment that is uniquely identifiable, e.g., by an *edgeID*. As result, each context node can be assigned to the same edge environment as the thing that acts as the context input for this specific context node. Therefore, the context node is annotated with the *edgeID* of the thing. The same applies to the condition nodes (as mentioned in Section V-A, in the implementation context nodes and condition nodes were combined). At operation nodes, multiple context nodes are

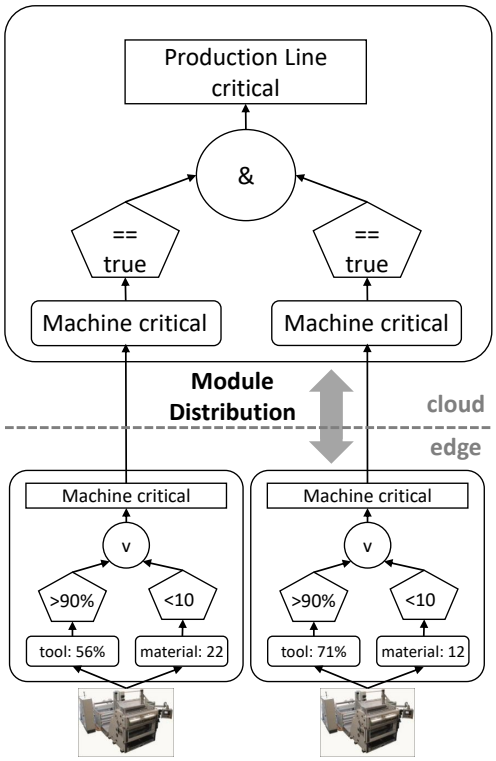


Figure 8. Modular distribution of Situation Templates to edge and cloud

combined and their annotated *edgeIDs* are compared. If they coincide, the operation node is annotated with the *edgeID*. Otherwise, the operation node cannot be distributed to a single edge node, since the different context inputs originate from different sources. This method is executed across all nodes of the Situation Template, starting from all context inputs up to the highest-level operation node. If the highest-level operation node is annotated with an *edgeID*, the whole Situation Template can be executed at the edge node. If only lower-level operation nodes are annotated, the Situation Template is split at this point and that operation node constitutes the highest level operation node in the newly created Situation Template.

2) Module Distribution: The last and final step is the module distribution. Now, we have different Situation Templates, which can be distributed to the edge and the cloud. If the nodes within the Situation Template are all annotated with the same *edgeID*, that Situation Template will be distributed to the corresponding edge environment. If no or multiple *edgeIDs* are present, the Situation Template is distributed to the cloud. Each environment, edge or cloud, contains a system for situation recognition (e.g., a CEP-based system, like Esper [28] and a messaging middleware (e.g., a MQTT-based system, like Mosquitto [29]). At runtime, machines periodically send their data to the message broker running at the edge node. Thereby, not only the situation recognition system can access the data but different applications as well. The situation recognition subscribes to the machine's data and its results, i.e., a situation object, are published to the message broker again, so that applications at the edge can access the situation objects directly. In parallel, all situation objects are mirrored to the message broker in the cloud. Therefore, all applications in the cloud can access the situation object as well and, more importantly, the situation recognition system in the cloud can use them as context inputs for the remaining situation recognition, which as well publishes the resulting situation objects to the message broker. In conclusion, this approach guarantees a flexible and scalable architecture for a distributed situation recognition using widely known and utilized technologies, such as CEP and messaging.

VI. Summary and Future Work

In this paper, we present an approach for distributed situation recognition. To support the distribution, we extend the Situation Template Schema so that multiple things and situations can be used for context input using a layered modeling approach. Furthermore, we present the concept of context stripping to reduce network traffic by removing the associated context of situation objects. We examine three distribution patterns based on execution requirements that are important for a situation recognition in complex environments. In addition, we describe how distributed situation recognition can be realized from the modeling of the situation using Situation Templates to the actual deployment. This is done by the introduced tool-based modeling support and an automated distribution of Situation Templates among the edge and backend cloud. This article is a revised and extended version of the SMART 2018 paper "Layered Modeling Approach for Distributed Situation Recognition in Smart Environments" [1].

In future work, we intend to create a sophisticated cost model, since choosing a suitable distribution pattern is very use-case dependent. Additionally, the management of the situation recognition after splitting, especially in the distribution pattern *Type III: Hybrid*, can become very complex and needs to be considered in future work. This way, users can receive a more detailed decision support based on their specific properties and requirements, which can lead to a faster adoption of new technologies like Edge Computing.

Acknowledgment This work is partially funded by the BMWi project IC4F (01MA17008G).

References

[1] M. Mormul, P. Hirmer, M. Wieland, and B. Mitschang, "Layered Modeling Approach for Distributed Situation Recognition in Smart Environments," in Tagungsband: SMART 2018, The Seventh International Conference on Smart Cities, Systems, Devices and Technologies. Xpert Publishing Services, Juli 2018, Konferenz-Beitrag, pp. 47–53. [Online]. Available: http://www2.informatik.uni-stuttgart.de/cgi-bin/NCSTRL/NCSTRL_view.pl?id=INPROC-2018-28&engl=

[2] D. Lucke, C. Constantinescu, and E. Westkämper, Manufacturing Systems and Technologies for the New Frontier: The 41st CIRP Conference on Manufacturing Systems May 26–28, 2008, Tokyo, Japan. London: Springer London, 2008, ch. Smart Factory - A Step towards the Next Generation of Manufacturing, pp. 115–118.

[3] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," Computer Networks, vol. 54, no. 15, 2010, pp. 2787 – 2805.

[4] J. S. He, S. Ji, and P. O. Bobbie, "Internet of things (iot)-based learning framework to facilitate stem undergraduate education," in Proceedings of the SouthEast Conference. ACM, 2017, pp. 88–94.

[5] M. Wieland, H. Schwarz, U. Breitenbücher, and F. Leymann, "Towards situation-aware adaptive workflows: SitOPT – A general purpose situation-aware workflow management system," in Pervasive Computing and Communication Workshops (PerCom Workshops). IEEE, 2015, pp. 32–37.

[6] P. Hirmer, M. Wieland, H. Schwarz, B. Mitschang, U. Breitenbücher, S. G. Sáez, and F. Leymann, "Situation recognition and handling based on executing situation templates and situation-aware workflows," Computing, 10 2016, pp. 1–19.

[7] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," IEEE Internet of Things Journal, vol. 3, no. 5, 2016, pp. 637–646.

[8] K. Häussermann, C. Hubig, P. Levi, F. Leymann, O. Simoneit, M. Wieland, and O. Zweigle, "Understanding and designing situation-aware mobile and ubiquitous computing systems," in Proc. of intern. Conf. on Mobile, Ubiquitous and Pervasive Computing. Citeseer, 2010, pp. 329–339.

[9] Q. Fang, Y. Zhao, G. Yang, and W. Zheng, Scalable Distributed Ontology Reasoning Using DHT-Based Partitioning. Springer Berlin Heidelberg, 2008, pp. 91–105.

[10] X. Wang, D. Q. Zhang, T. Gu, and H. Pung, "Ontology based context modeling and reasoning using OWL," in Pervasive Computing and Communications Workshops, 2004. Proceedings of the Second IEEE Annual Conference on, 2004.

[11] W. Dargie, J. Mendez, C. Mobius, K. Rybina, V. Thost, A.-Y. Turhan et al., "Situation recognition for service management systems using OWL 2 reasoners," in Pervasive Computing and Communications Workshops (PERCOM Workshops), 2013 IEEE International Conference on. IEEE, 2013, pp. 31–36.

[12] J. Attard, S. Scerri, I. Rivera, and S. Handschuh, "Ontology-based situation recognition for context-aware systems," in Proceedings of the 9th International Conference on Semantic Systems. ACM, 2013, pp. 113–120.

[13] B. Schilling, B. Koldehofe, U. Pletat, and K. Rothermel, "Distributed heterogeneous event processing: Enhancing scalability and interoperability of cep in an industrial context," in Proceedings of the Fourth ACM International Conference on Distributed Event-Based Systems. ACM, 2010, pp. 150–159.

[14] N. P. Schultz-Møller, M. Migliavacca, and P. Pietzuch, "Distributed complex event processing with query rewriting," in Proceedings of the Third ACM International Conference on Distributed Event-Based Systems, ser. DEBS '09. New York, NY, USA: ACM, 2009, pp. 4:1–4:12. [Online]. Available: http://doi.acm.org/10.1145/1619258.1619264

[15] K. V. Laerhoven and H. W. Gellersen, "Spine versus porcupine: a study in distributed wearable activity recognition," in Eighth International Symposium on Wearable Computers, vol. 1, Oct 2004, pp. 142–149.

[16] A. K. Dey, "Understanding and using context," Personal and ubiquitous computing, vol. 5, no. 1, 2001, pp. 4–7.

[17] P. Hirmer, M. Wieland, H. Schwarz, B. Mitschang, U. Breitenbücher, and F. Leymann, "SitRS - A Situation Recognition Service based on Modeling and Executing Situation Templates," in Proceedings of the 9th Symposium and Summer School On Service-Oriented Computing, 2015, Konferenz-Beitrag, pp. 113–127.

[18] A. C. Franco da Silva, P. Hirmer, M. Wieland, and B. Mitschang, "SitRS XT-Towards Near Real Time Situation Recognition," Journal of Information and Data Management, 2016.

[19] M. Mormul, P. Hirmer, M. Wieland, and B. Mitschang, "Situation model as interface between situation recognition and situation-aware applications," Computer Science - Research and Development, November 2016, pp. 1–12.

[20] O. N. Yilmaz, Y.-P. E. Wang, N. A. Johansson, N. Brahmi, S. A. Ashraf, and J. Sachs, "Analysis of ultra-reliable and low-latency 5g communication for a factory automation use case," in Communication Workshop (ICCW), 2015 IEEE International Conference on. IEEE, 2015, pp. 1190–1195.

[21] O. Moll, A. Zalewski, S. Pillai, S. Madden, M. Stonebraker, and V. Gadepally, "Exploring big volume sensor data with vroom," Proceedings of the VLDB Endowment, vol. 10, no. 12, 2017.

[22] J. Budomo, I. Ahmad, D. Habibi, and E. Dines, "4g lte-a systems at vehicular speeds: Performance evaluation," in Information Networking (ICOIN), 2017 International Conference on. IEEE, 2017, pp. 321–326.

[23] E. Directive, "95/46/ec of the european parliament and of the council of 24 october 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data," Official Journal of the EC, vol. 23, no. 6, 1995.

[24] P. Hirmer, M. Wieland, U. Breitenbücher, and B. Mitschang, "Automated Sensor Registration, Binding and Sensor Data Provisioning," in CAiSE Forum, 2016.

[25] S. Marston, Z. Li, S. Bandyopadhyay, J. Zhang, and A. Ghalsasi, "Cloud computingthe business perspective," Decision support systems, vol. 51, no. 1, 2011, pp. 176–189.

[26] S. Yi, Z. Hao, Z. Qin, and Q. Li, "Fog computing: Platform and applications," in Hot Topics in Web Systems and Technologies (HotWeb), 2015 Third IEEE Workshop on. IEEE, 2015, pp. 73–78.

[27] D. Floyer, "The vital role of edge computing in the internet of things," Oct. 2015. [Online]. Available: https://wikibon.com/the-vital-role-of-edge-computing-in-the-internet-of-things/

[28] "Complex event processing streaming analytics." [Online]. Available: http://www.espertech.com/

[29] "Eclipse mosquitto an open source mqtt broker." [Online]. Available: https://mosquitto.org/

All links were last followed on May 21, 2019.

Light-Fidelity (Li-Fi)

LED assisted navigation in large indoor environments

Manuela Vieira, Manuel Augusto Vieira, Paula Louro, Alessandro Fantoni
ADETC/ISEL/IPL,
R. Conselheiro Emídio Navarro, 1959-007
Lisboa, Portugal
CTS-UNINOVA
Quinta da Torre, Monte da Caparica, 2829-516,
Caparica, Portugal

e-mail: mv@isel.ipl.pt, mv@isel.pt, plouro@deetc.isel.pt, afantoni@deetc.isel.ipl.pt

Pedro Vieira
ADETC/ISEL/IPL,
R. Conselheiro Emídio Navarro, 1959-007
Lisboa, Portugal
Instituto das Telecomunicações
Instituto Superior Técnico, 1049-001,
Lisboa, Portugal
e-mail: pvieira@isel.pt

***Abstract*— In this work, a Light Emitting Diode (LED) assisted navigation system for large environments is presented. The LEDs are used both for room illumination purposes and as transmitters if modulated at high frequencies. The payload data together with the identifiers, IDs, assigned to the physical location of the transmitters are broadcast using an On-Off Keying (OOK) modulated scheme. The mobile receiver is a double p-i-n/pin SiC photodetector with light controlled filtering properties. Coded multiplexing techniques for supporting communications and navigation together on the same channel are analysed. A demonstration of fine-grained indoor localization is simulated. Different indoor layouts for the LEDs are considered. Square and hexagon mesh are tested, and a 2D localization design, demonstrated by a prototype implementation, is presented. The results showed that the LED-aided Visible Light Communication (VLC) navigation system makes possible not only to determine the position of a mobile target inside the unit cell but also in the network and concomitantly to infer the travel direction in time. Bidirectional communication was tested between the infrastructure and the mobile receiver.**

Keywords- Visible Light Communication; Indoor positioning; Square and hexagonal topologies; SiC technology; Optical sensor, Navigation system; Bidirectional communication.

I. INTRODUCTION

Light Fidelity (Li-Fi) is a technology for wireless communication between devices using light, to transmit data and position. VLC/LiFi can play a significant role by bringing redundancy to traditional RF solutions. It brings strong value-added features in terms of: complexity and cost, selectivity, quality of link, high precision positioning and security. Nevertheless, one of the limitations of LiFi existing solutions is that LiFi have been developed mainly for indoor applications [1]. With the rapid advancement of smart equipment, location based services that employ different kinds of communication and positioning technologies to localize the walker and to provide relevant services, start to develop. Although Global Positioning System (GPS) works extremely well in an open-air localization, it does not perform effectively in indoor environments, due to the disability of GPS signals to penetrate in in-building materials. Nowadays, indoor positioning methods are mainly based on Wi-Fi, Bluetooth, Radio-frequency identification (RFID), visible light communications (VLC) and inertia navigation [2, 3, 4, 5, 6]. Although many methods are available such as WiFi-based [7, 8] and visual indoor topological localization [9, 10], they require dense coverage of WiFi access points or expensive sensors, like high-performance cameras, to guarantee the localization accuracy.

VLC is a data transmission technology [11, 12, 13, 14]. VLC can easily be employed in indoor environments such as offices, homes, hospitals, airplanes/airports and conference rooms. Compared with other positioning methods, indoor VLC based positioning, has advantages, since it can use the existing LED lighting infrastructure with simple modifications. Due to the combination of illumination and communication, a lot of investigations have been attracted in VLC applications [15, 16, 17]. The system considers both positioning and lighting, thus it will save energy and realize high accuracy positioning with low cost at the same time. Besides that, due to the advantage of electromagnetic free, VLC based positioning is a green positioning method.

In the sequence, we propose to use modulated visible light, carried out by white low cost LEDs, to provide globally consistent signal-patterns and engage indoor localization. The LEDs are capable of switching to different light intensity levels at a very fast rate. The switching rate is fast enough to be imperceptible by a human eye. This functionality can be used for communication where the data is encoded in the emitting light in different ways [18]. A

photodetector can receive the modulated signals and decode the data. This means that the LEDs be twofold of providing illumination as well as communication.

The requirement of managing access to multiple devices in VLC is different from other types of networks. This is because the size of a cell can vary depending on how illumination is provided. The task of squaring the circle proposed by ancient geometers was proven to be impossible. The triangle of largest area of all those inscribed in a given circle is equilateral. The cells of a beehive honeycomb are hexagonal. So, research is necessary to design LED arrangements that can optimize communication performance while meeting the illumination constraints for a variety of indoor layouts, using four-code assignment for the LEDs in an extended equilateral triangle, square or diamond mesh. The main idea is to divide the space into spatial beams originating from the different light sources, and identify each beam with a unique timed sequence of light signals. The receiver, equipped with an a-SiC:H pinpin photodiode, determines its physical position by detecting and decoding the light signals. The overlap of different light beams at the receiver is used as an advantage to increase the positioning accuracy. Fine-grained indoor localization can enable several applications; in supermarkets and shopping malls, exact location of products can greatly improve the customer's shopping experience and enable customer analytics and marketing [19, 20].

Luminaires equipped with multi colored LEDs can provide further possibilities for signal modulation and detection in VLC systems [21]. The use of Red-Green-Blue (RGB) LEDs is a promising solution as they offer the possibility of Wavelength Division Multiplexing (WDM) which can enhance the transmission data rate. A WDM receiver have been developed [22, 23]. The device is based on tandem a-SiC:H/a-Si:H pin/pin light controlled filter. When different visible signals are encoded in the same optical transmission path, the device multiplexes the different optical channels, performs different filtering processes (amplification, switching, and wavelength conversion) and finally decodes the encoded signals recovering the transmitted information.

In this paper, a LED-assisted indoor positioning and navigation VLC system, for large indoor environment is proposed. The principle of the positioning scheme and the algorithm to decode the information are described and experimental results are presented. A 2D localization design, demonstrated by a prototype implementation is tested. Fine-grained indoor localization is demonstrated using square and hexagonal topologies. Finally, conclusions are addressed. The proposed, composed data transmission and indoor positioning, involves wireless communication, smart sensoring and optical sources network, building up a transdisciplinary approach framed in cyber-physical systems.

II. System Configuration

A VLC system consists on a VLC transmitter that modulates the light produced by LEDs and a VLC receiver based on a photosensitive element that receive and analyses the transmitted modulated signals. The transmitter and the receiver are physically separated, but connected through the VLC channel. For VLC systems, Line of Sight (LoS is a mandatory condition.

LED bulbs work as transmitters, broadcasting the information. An optical receiver extracts its location to perform positioning and, concomitantly, the transmitted data from each transmitter. Multiple LEDs can transmit simultaneously to the same receiver using joint transmission. To synchronize the signals from multiple LEDs, the transmitters use different ID´s, such that the signal is constructively at the receiver.

A. Shapes and topologies

Different shapes, for the unit cell, are proposed based on the joint transmission of four modulated LEDs: the square, the hexagonal. The unit cells, for the analyzed topologies, are displayed in Figure 1.

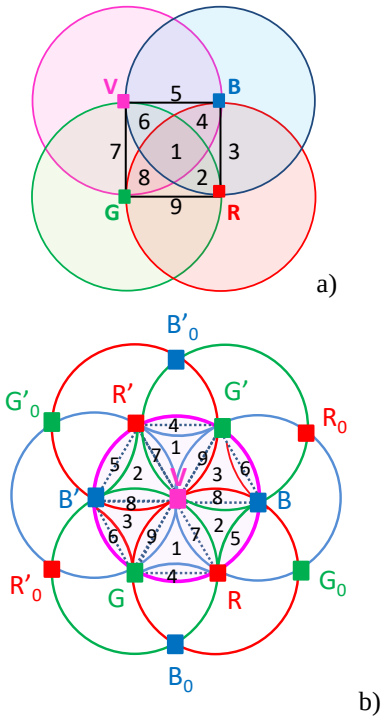


Figure 1. Top-down view of unit cells (LED array = RGBV color spots) having each one four modulated RGBV-LEDs located at the corners of the grid. a) Square cell. b) First hexagon ring.

In Figure 1a, the proposed LED arrangement employs four modulated LEDs placed at the corners of a square grid.

In Figure 1b, a cluster of three diamond cells sharing the violet node, fill the space with a hexagon, leading to the hexagonal topology. The estimated distance from the ceiling lamp to the receiver is used to generate a circle around each transmitter on which the device must be located in order to receive the transmitted information (generated location and coded data).

In all topologies, the grid sizes were chosen to avoid overlap in the receiver from adjacent grid points. To improve its practicality, the tested geometric scenario for the calculations, in both topologies, uses a grid in smaller size (2 cm between adjacent nodes). To receive the information from several transmitters, the receiver must be positioned where the circles from each transmitter overlap, producing at the receiver, a MUX signal that, after demultiplexing, acts twofold as a positioning system and a data transmitter. The generated regions, defined onwards as footprints, are pointed out in Figure 1 and reported in Table I.

TABLE I FINE-GRAINED CELL TOPOLOGY.

Footprints regions	Square topology	Hexgonal topology
P#1	RGBV	RGV R'G'V
P#2	RGB	RBV R'B'V
P#3	RG	G'BV GB'V
P#4	RBV	RGB_0V $R'G'B'_0V$
P#5	BV	RG_0BV $R' G'_0B'V$
P#6	GBV	$R_0G'BV$ $R'_0GB'V$
P#7	GV	RGBV R'G'B'V
P#8	RGV	RG'BV R'GB'V
P#9	RG	RGB'V R'GB'V

In the hexagonal topology, each node has six neighbors, so, eighteen footprints are possible. Twelve at the edges of the six equilateral triangles where four circles overlap (#P4 to #P9) and six at their centroids (#P1 to #P3), where only three channels are received. Taking into account the XY symmetry (Figure 1b), the R, G and B nodes and their symmetric (R'G'B') must be considered. When the received channels come from outside the hexagon edges (first ring), the nodes are label with *0* (see Figure 1b). When the signal comes only from one LED, the coordinates of the LED are assigned to the device's reference point. If the device receives multiple signals, *i.e.*, if it is in an overlapping region of two or more LEDs, it finds the centroid of the received coordinates and stores it as the reference point. This is the so called fine-graining of the unit cell.

For data transmission commercially available white RGB-LEDs and a violet (V: 400 nm) LED were used. The output spectrum of the white LED contains three peaks assigned to the colors red (R: 626 nm), green (G: 530 nm) and blue (B: 470 nm), that mixed together provide the white perception to the human eye. Each chip, in the trichromatic LED, can be switched *on* and *off* individually for a desired bit sequence. The luminous intensity is regulated by the driving current for white perception. They exhibit a wide divergence angle (2x60º), since they are also designed for general lighting and allow a wide delivery of the VLC signal around the surrounding area. The driving current of each emitter is controlled independently, suppling the respective coding sequence and frequency [14]. In both topologies, the driving current of the emitters having the same wavelength was always the same.

B. Cellular topologies for large environments

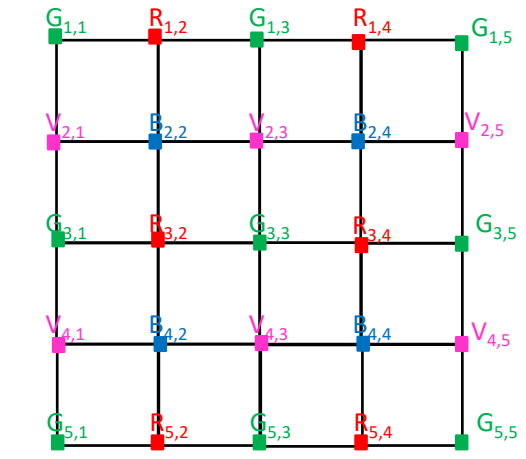
When the rooms are very large, such as in supermarkets or large shopping malls, the positioning technique (four-code assignment for the LEDs), is applied in the whole room.

Thinking about the design, the wall structure could be continued to form the tiles and a complete communication/illumination can be created. For cellular design, a regular shape in needed over the serving area. Squares, equilateral triangles, regular hexagons or a combination of this would be demanding, because the shape makes efficient use of space. They cover the entire area without any gaps decreasing the interference between the same code [24]. Since each cell is projected to use wavelengths only within its boundaries, the same wavelengths can be reused in other close cells without interference.

Like squares, regular hexagons also fit together without any gaps to tile the plane. Ceiling plan for the LED array layout is shown in Figure 1 (LED array = RGBV color spots).

Two topologies were set for the unit cell: the square, (Figure 2a) and the hexagonal (Figure 2b). In the first, the proposed LED arrangement employs four modulated LEDs placed at the corners of a square grid.

The unit cell $C_{i,j}$ is repeated in the horizontal and vertical directions in order to fill all the space. In the second topology (Figure 2b), the hexagonal, the same LED array was used, but in a no-orthogonal system. We select a pair inclined at 120 degrees to be the axes, labelled as X and Y. We have readdressed the nodes, in the oblique Cartesian system. Consequently, in both topologies, each node, $X_{i,j}$, carries its own color, X, (RGBV) as well as its ID position in the network (i,j).



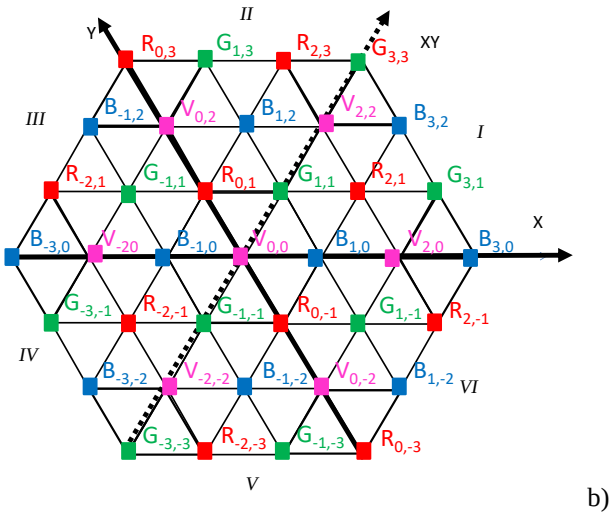


Figure. 2. Illustration of the proposed scenarios (LED array = RGBV color spots): a) Clusters of cells in orthogonal topology (square). b) Clusters of cell in hexagonal topology.

C. The OOK modulation scheme

A dedicated four channel LED driver, with multiple outputs, was developed to modulate the optical signals. The modulation of the emitted light was done through the modulation of the driving electrical current of the semiconductor chips of each LED. The modulator converts the coded message into a modulated driving current signal that actuates the emitters of each violet and tri-chromatic LEDs. A graphical user interface allows the control of the system, which includes the setting of the driving current, bit sequence and frequency of each emitter.

An on-off keying modulation scheme was used. The frame structures are illustrated in Figure 3, for both topologies.

For both, the frame is built based on three separate blocks; the synchronism (Sync), the ID address of the transmitter (ID) and the message to transmit (payload data).

A 32 bit codification was used. The first five bits are used for time synchronization. The same synchronization header [10101], in an ON-OFF pattern, is imposed simultaneously to all the emitters. Each color signal (RGBV) carries its own ID-BIT, so, the next bits give the coordinates of the emitter inside the array ($X_{i,j}$). Cell's IDs are encoded using a binary representation for the decimal number. In the square topology (Figure 3a), six bits are used: the first three for the binary code of the line and the other three for the column. In the hexagonal topology to code the positive and the negative coordinates "sign and magnitude" representation was used, setting bit to *0* is for a positive number, and setting it to *1* is for a negative number.

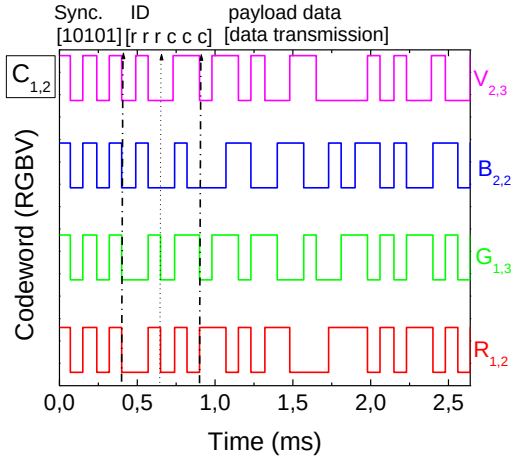
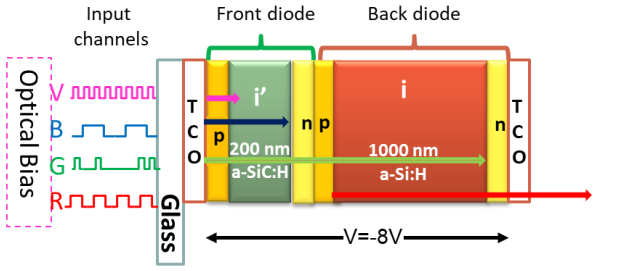


Figure 3. Frame structure. Representation of one original encoded message, in a time slot. a) Square topology; R_{12}; $G_{1,3}$; $B_{2,2}$ and $V_{2,3}$ are the transmitted node packet from the $C_{1,2}$ array in the network.. b) Hexagonal topology; $R_{0,-1}$; $G_{-1,-1}$; $B_{1,0}$ and $V_{0,0}$ are the transmitted node packet of the unit cell in the network.

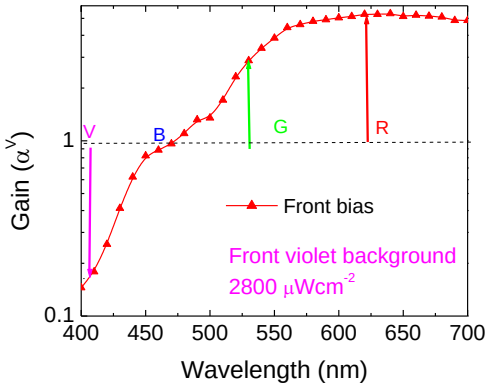
The remaining bits in the number indicate the absolute value. So, the next eight bits (ID) are assigned, respectively, to the *x* and *y* coordinate (*i*,*j*) of the emitter in the array (Figure 3b). For instance, R_{12} emitter sends a six ID_BIT [001 010] in the square topology while in the hexagonal one the eight ID bits are [0001 0010]. For both, the last bits in the frame are reserved for the message send by the X_{ij} node (payload data). A stop bit is used at the end of each frame.

D. The VLC receiver

The VLC receiver is an important component of a VLC system, as its performances are the ones that determine the overall systems performances. The VLC receiver, at the end of the communication link, is responsible to extract the data from the modulated light beam. It transforms the light signal into an electrical signal that is later decoded to extract the transmitted information.



a)



b)

Figure 4. a) Double pi´n/pin configuration and device operation. b) Spectral gain under violet front optical bias (α^V). The arrows point towards the optical gain at the analyzed R, G, B and V input channels.

The VLC receiver is a two terminal, p-i'(a-SiC:H)-n/p-i(a-Si:H)-n photodiode packed in two transparent conductive contacts (TCO).

The deposition conditions, optoelectronic characterization and device optimization are described elsewhere [13]. The configuration and operation is illustrated in Figure 4a. Due to the different absorption coefficient of the intrinsic absorption layers, both front and back diodes act as optical filters confining, respectively, the optical carrier produced by the blue and red light, while the optical carriers generated by the green light are absorbed across both (see arrow in Figure 4a).

The device operates within the visible range using for data transmission the modulated low power light supplied simultaneously by the RGBV LEDs located at the nodes of the array. A mix of R, G, B, and V pulsed communication channels (input channels; transmitted data), each one with a specific bit sequence, impinges on the device and are absorbed accordingly to their wavelengths. The combined optical signal (MUX signal; received data) is analyzed by reading out the generated photocurrent under negative applied voltage and violet background lighting, applied from the front side of the receiver [22, 25]. The optical bias enhances de long wavelength channels (R, G) and quenches the low ones (B,V). In Figure 4b, the spectral gain defined as the ratio between the photocurrent with and without applied optical bias, is displayed. The arrows point towards the gain at the analyzed R, G, B and V input wavelength.

The results show that the device acts as an active filter, under irradiation. It is interesting to notice that, as the wavelength increases, the signal strongly increases. This nonlinearity is the main idea for the decoding of the MUX signal at the receiver.

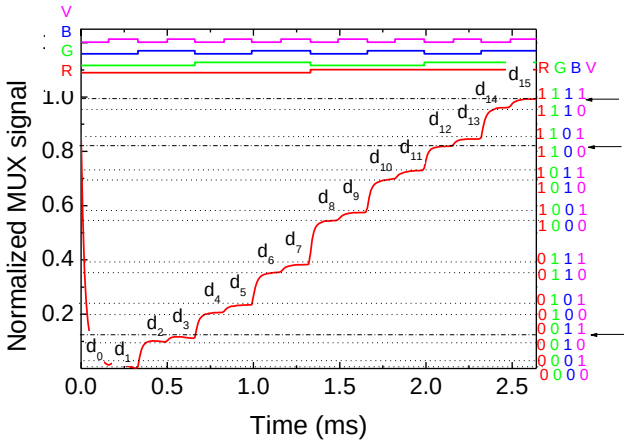
III. Experimental Results and Discussion

A. Coding/Decoding techniques

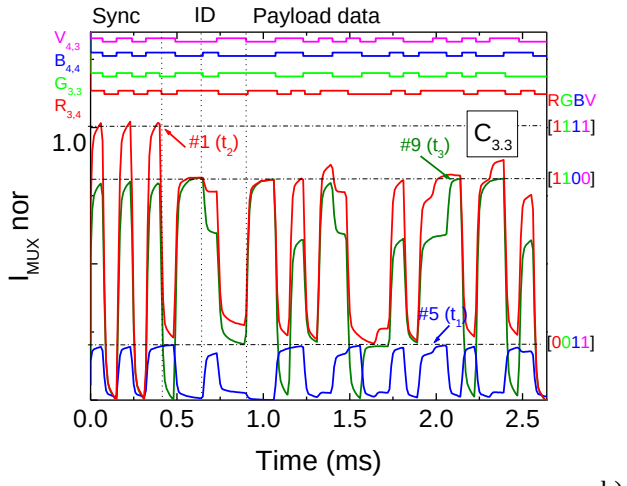
In Figure 5, the normalized received data due to the mixture of the four R, G, B, and V input channels, i.e., the MUX code signal in a stamp time, are displayed, for the square topology.

In Figure 5a, the bit sequence was chosen to allow all the on/off sixteen (2^4) possible combinations of the four input channels. For three times slots (t_1, t_2, t_3), in Figure 5b, the MUX signal acquired by a receiving positions P#9, P#1, and P#5 (see Table I), is displayed. The decoded packet of transmitted information is presented in the top of both figures. Results from Figure 5a show that the MUX signal presents as much separated levels as the *on/off* possible combinations of the input channels, allowing to decode the transmitted information [26]. All the ordered levels (d_0-d_{15}) are pointed out at the correspondent levels, and are displayed as horizontal dotted lines. In the right hand side of Figure 5a, the match between MUX levels and the 4 bits binary code assigned to each level is shown. Results show that each possible *on/off* state corresponds to a well-defined level. Hence, by assigning each output level to a 4-digit binary code, [X_R, X_G, X_B, X_V], with X=1 if the channel is *on* and X=0 if it is *off*, the signal can be decoded. The MUX signal presented in Figure 5a, is used for calibration purposes. Comparing the calibrated levels with the different generated levels (Figure 5b), in the same frame of time, a

simple algorithm [27] is used to decode the multiplex signals.



a)



b)

Figure 5. MUX/DEMUX signals under 390 nm front irradiation. On the top the transmitted channels packets [R, G, B, V] are decoded. a) Calibration cell. b) MUX signal in three successive instants (t_1, t_2, t_3).

In Figure 5b, the MUX signal, when the receiver is located in positions P#5, P#1 and P#9 confirms the decoding process. After decoding the MUX signals, the localisation of the receiver is straightforward. Taking into account, the frame structure (Figure 3), the position of the receiver inside the navigation cell and its ID in the network is revealed [28]. The ID position, in the unit cell, comes directly from the synchronism block, where all the received channels are, simultaneously, *on* or *off*. The 4-bit binary code ascribed to the higher level identifies the receiver position in the unit cell (Table I). Those binary codes are displayed in the right hand of the Figure 5b. For instance, the level [1100] corresponds to the level d_{12} where the red and the green channels are simultaneously *on*. The same happens to the other footprints (P#1 and P#9) (see arrows and dash-dot lines in Figure 5a). Each decoded message carries, also, the node address of the transmitter. So, the next block of six bits, in the square topology or eight in the hexagonal one, gives de ID of the received node. In P#5 the location of the transmitters, in the network, are $B_{4,4}$ and $V_{4,3}$ while in P#1 the assigned transmitters are $R_{3,4}$, $G_{3,3,}$ $B_{4,4}$ and $V_{4,3.}$ The last block is reserved for the transmission of the message (payload data). A stop bit (0) is used always at the end of each frame.

B. LED-aided navigation system

An interconnection network consists of set of nodes and communication links for the data transmissions. Synchronisation and communication between processing nodes is effectively implemented by message passing. A challenge in LED-based navigation system is the way to improve the data transmission rate while maintaining the capability for accurate navigation. The input of the aided navigation system is the MUX signal, and the output is the system state estimated at each time step (Δt). To compute the point-to-point exposure along a path, we need the data along the path.

As a proof of concept, the suitability of the proposed navigation data bit transition was tested, in the lab. The solution was to move the receiver along a known pattern path. For each transition, two code words are generated, the initial (i) and the final (f). If the receiver stays under the same region, they should be the same, if it moves away they are different. The signal acquisition on the different generated locations was performed and the transition actions were correlated by calculating the ID position codes in the successive instants.

In Figure 6a the simulated path is illustrated and in Figure 6b and Figure 6c MUX/DEMUX signals in six successive instants are displayed. Decoding, when the four channels overlap (P#1), is set on the top of the figures to direct into the packet sent by each node. In the right hand of the figures, the assigned 4-bit binary code is depicted at the successive instants.

Results show that, as the receiver moves between generated point regions, the received information pattern changes. Note that, between two consecutive data sets, there is one navigation data bit transition (a channel is missing or added). We observe that when the receiver initially moves from P#3 to P#1, the green, $G_{1,3}$ and the violet, $V_{2,3}$, channels are added and the 4-bynary bit code changes from [1010] to [1111]. In Figure 6c, the MUX signals acquired in three posterior instants (t_4, t_5, t_6) are displayed. At t_3 the assigned 4-bit [RGBV] code is [0011], changes to [1111] at t_5 and at t_6 it is [1101]. The wavelength of the received emitters, during the transmission, and so the different locations of the receiver inside the cell are assigned as: P#5 (BV), P#1 (RGBV) and P#8 (RGB) (see Table I).

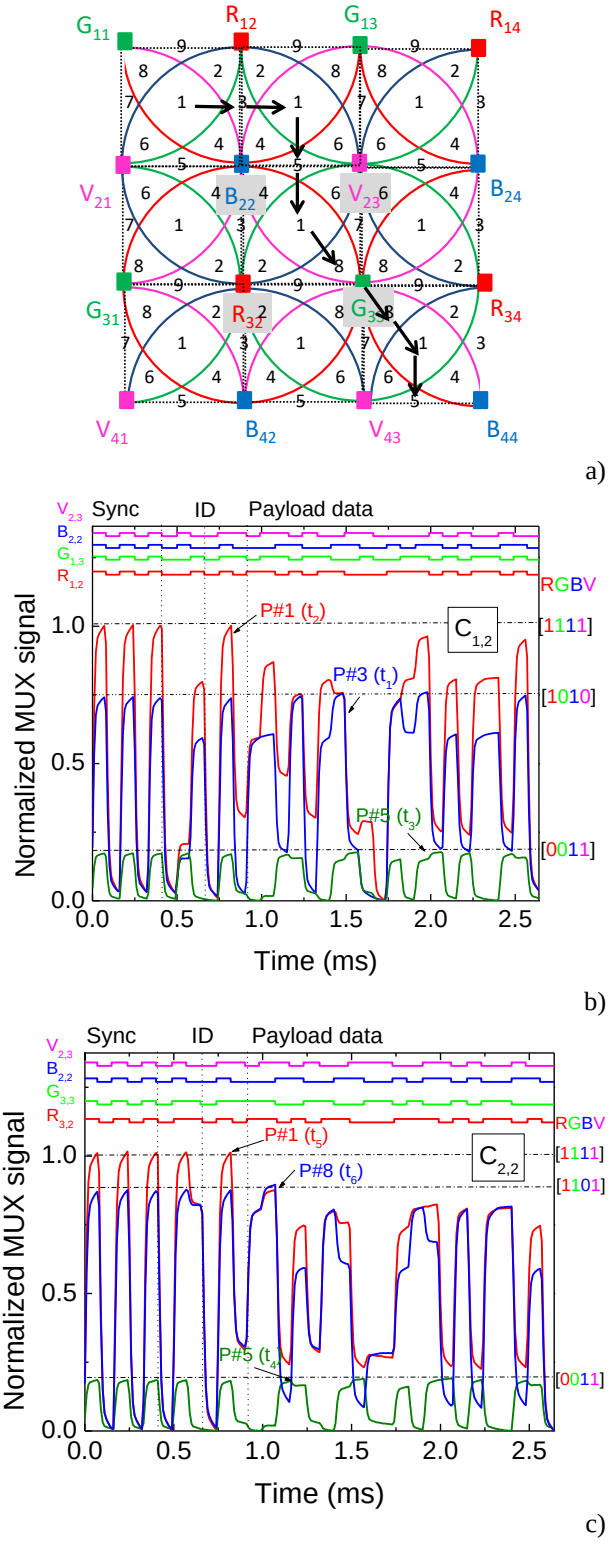


Figure 6 a) Fine-grained indoor localization and navigation in successive instants. Signal acquisition across cell $C_{2,2}$ at t_1, t_2 and t_3, On the top the transmitted channels packets are decoded [R, G, B, V].

Each decoded message carries the node address of the transmitter. The location in the network is decoded through the ID-BIT that carries the address of the node. Looking into the next block of six bits, their position in the network is assigned as: $R_{3,2}$ $G_{3,3}B_{2,2}V_{2,3}$ and linked with $C_{2,2}$ cell. Comparing both P#1 in Figure 6b and Figure 6c, we notice that although the position inside the navigation cell is the same (same level) the location inside the network is different. In Figure 6b the nodes $R_{1,2}$ [001 010], $G_{1,3}$ [001 011], $B_{2,2}$ [010 010] and $V_{2,3}$ [010 011] are recognized and linked respectively, to the cell $C_{1,2}$ while in Figure 6c the assigned nodes are $R_{3,2}$ $G_{3,3}B_{2,2}V_{2,3}$ belong to $C_{2,2}$. At t_4 and t_5 the same two channels are received and linked with P#5, in both figures, the assigned nodes are, $B_{2,2}$ $V_{2,3}$, same position but in different cells (same level in both figures).

Figures 7b and 7c display the MUX/DEMUX signals acquired, in a hexagonal environment (Figure 2b), at six successive instants. The receiver was moved along a known pattern path as described in Figure 7a, the arrows illustrate the simulated path in six successive instants (t_1 to t_6). Fine-grained indoor localization and LED based navigation is illustrated. At the right hand of both figures, the 4-bit binary codes are pointed out at the correspondent levels.

Taking into account Figure 1b and the frame structure (Figure 3b), results show that at t_1 the receiver was located at P#1 [1101]/ $R_{0,-1}$ $G_{-1,-1}$ $V_{0,0}$. At t_2, it arrives to P#7 [1111]/$R_{0,-1}$ $G_{-1,-1}$ $B_{1,0}V_{0,0}$, then, at t_3, moves towards P#2, [1011]/ $R_{0,-1}$ $B_{1,0}V_{0,0}$. At t_4, goes to P#7 [0111]/ $G_{1,1}$ $B_{1,0}V_{0,0}$ and at t_5, arrives to P#6 [1111]/$R_{2,1}$ $G_{1,1}$ $B_{1,0}V_{0,0}$. At t_6, the receiver enters in the second ring, where the 4-binary code [1110] locates the position inside the cell (P#A) and the ID-BIT of the received channels, $R_{2,1}$ $G_{1,1}$ $B_{1,0}$, its position inside the network. The main results show that, for both topologies, the location of a mobile receiver is achieved based on the LED-based navigation system. At the client's end, positioning bits are decided by the received MUX signal (wavelengths and ID address of the received channels). Fine grained localization was achieved by detecting the wavelengths of the received channels in each cell (Table I). Nine sub-regions fill each square cell while in the hexagonal, due to the existing XY symmetry, eighteen possible sub-regions can be designed.

The use of the square, hexagonal or both topologies depends on the layout of the environment. In Figure 8, and for a shopping mall, an example of the mix of both is illustrated. Here, the different areas transmit payload data according to the areas they are located. In concentric layouts, to fill all the space with hexagon, it presents advantages (cosmetics, machines and vegetables areas). Here, the client can move around and walk between the different rings toward the outside region. In an orthogonal layout (hall), the square topology is the best. It allows crossroads and the client can walk easily in the horizontal, vertical or both directions.

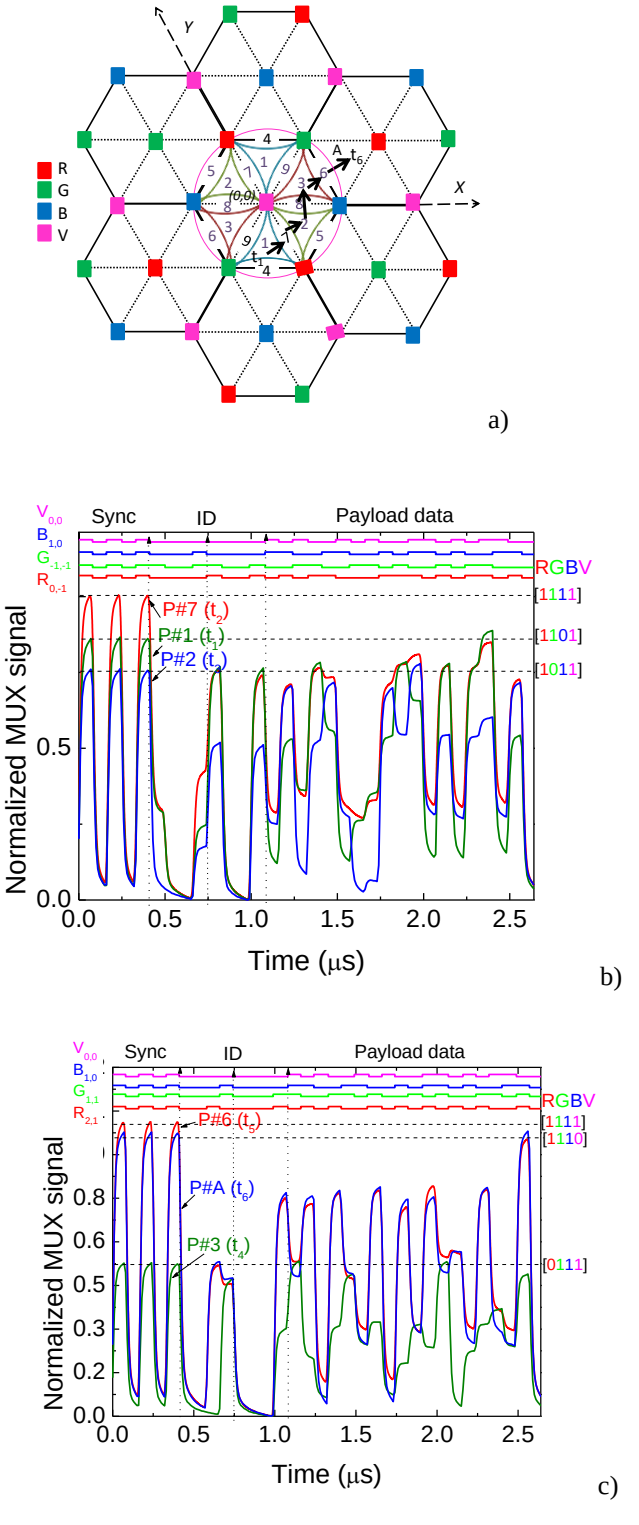


Figure 7 a) Fine-grained indoor localization and navigation, as illustrated by the arrows. b) Signal acquisition at t_1, t_2 and t_3. c) Signal acquisition at t_4, t_5 and t_6 On the top the transmitted channels packets [R, G, B, V] are decoded

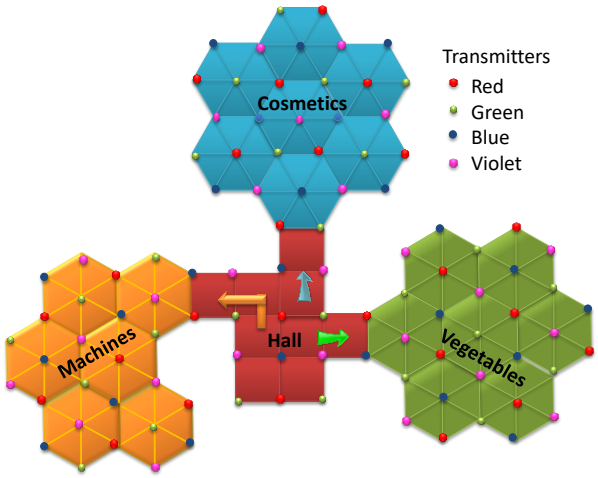


Figure 8 Fusion of hexagonal and square topologies. Example of a ceiling plan. in a shopping mall, with different layout areas. A four-code assignment for the RGBV LEDs is used as positioning technique.

C. *Bidirectional communication*

Bidirectional communication between infrastructure and the mobile receivers can be established. Downlink communication occurs from the ceiling lamps to the mobile receiver.

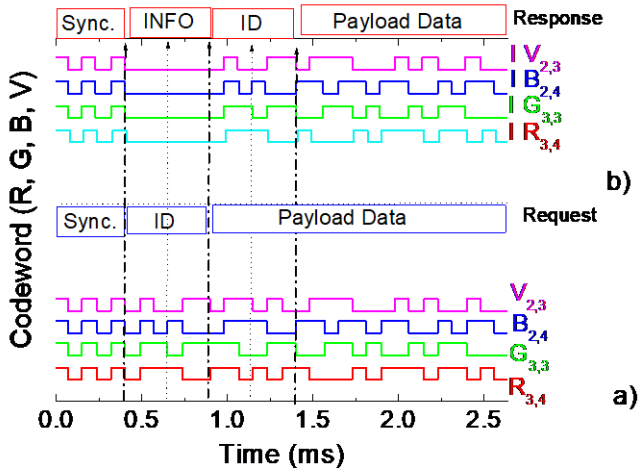


Figure 9 Frame structure representation. a) Codification used to in a request message in P#1. $R_{3,4}$, $G_{3,3}$, $B_{2,4}$ and $V_{2,3}$ are the transmitted node packet, in a time slot. b) Encoded message response of a local controller emitter to the traveler at P#1\ $R_{3,4}$, $G_{3,3}$, $B_{2,4}$ and $V_{2,3}$.

Uplink communication is defined from the receiver to a near indoor billboard linked to the control manager. Each emitter broadcasts a message with its ID and payload data which is received and processed by the receiver. Using a white polychromatic LED as transmitter, the receptor sends to the local controller a message “request” with its location (ID) and adds its queries as payload data (right track, shops,

products, etc,). For path coordination, the local controller emitter sends the "response" message with the required information.

In Figure 9a an example of the codification used in a "request" message is illustrated. Thus, $R_{3,4}$, $G_{3,3}$, $B_{2,4}$ and V_{23} are the transmitted node packets, in a time slot, inside the cell in position P#1. In Figure 9b, a "response" message from the controller emitter located at the billboard is displayed. The second block (INFO) in a pattern [000000] means that a response message is being sent by the controller manager. The third block (6 bits) identifies the receiver (ID) for which the message is intended. Here, the controller [000000] responds to a request of a traveller located in position P# 1 ($R_{3,4}$, $G_{3,3}$, $B_{2,4}$ and V_{23}) (see Figure 2a).

In Figure 10, the MUX signal assigned to a "request" and a "response" message are displayed. In the top the decoded information is presented. In the right side, the match between the MUX signal and the 4-binary code is pointed out.

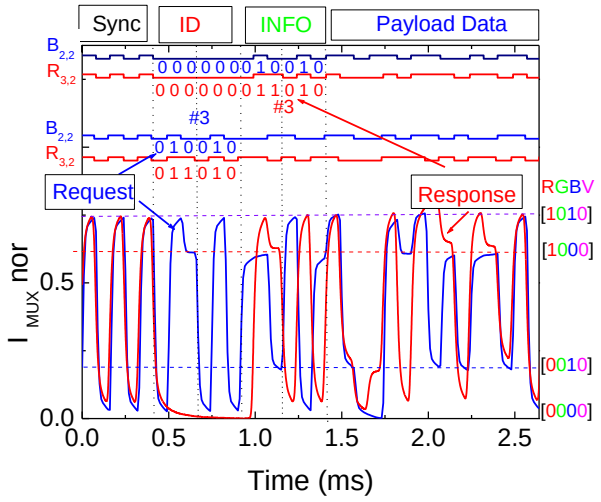


Figure 10. MUX/DEMUX signals assigned to a "request" and a "response" message. On the top the transmitted channels packets [$X_{i,j}$] are decoded.

Here, in a time slot, the traveler, in position #3 ($R_{3,2}$, $B_{2,2}$), sends to the central controller the message "request" in order to add the points of interest (shops or the right track). After that it is advised, through a "response" message, that the request was received, how to reach its destination and how to use location based advertising.

Taking into account the frame structure, results show that the codification of both signals is synchronized (Sync). The request message includes the complete address of the traveler (Sync+ID) and the help need (Payload Data). In the "response" message the block (ID), in a pattern [000000], means that a response message, from the local manager, is being sent. The next block (6 bits) identifies the address (INFO) for which the message is intended and finally in the last block appears the requested information (Payload Data). Here, the emitter controller [000000] responds to a request of a passenger located in position # 3 ($R_{3,2}$, $B_{2,2}$) and sends to him the requested information.

IV. Conclusions and Future Trends

We have proposed a VLC LED-assisted navigation system for large indoor environments. For illumination purposes, data transmission and positioning, white LEDs were used. An a-SiC:H/a-Si:H pin/pin SiC optical MUX/DEMUX mobile receiver decodes the data and based on the synchronism and ID of the joint transmitters it infers its path location. A four-code assignment for the LEDs was proposed. Two cellular networks were tested and compared: the square and the hexagonal. Results show that, in large indoor environments, the use of VLC technology allows different cellular topologies where locations together with data transmission are achieved. The choice of one or both topologies depends mainly on the layout of the environment. Bidirectional communication between the infrastructure and a mobile receiver was also tested.

Minding the benefits of VLC, it is expected that this type of communication will have an important role in positioning applications. Moving towards real implementation, the performances of such systems still need to improve. As a future goal, we plan to finalize the embedded application, for experimenting in several network layouts.

Acknowledgements

This work was sponsored by FCT – Fundação para a Ciência e a Tecnologia, within the Research Unit CTS – Center of Technology and systems, reference UID/EEA/00066/2013

The projects: IPL/2018/II&D_CTS/UNINOVA_ISEL and by: IPL/IDI&CA/2018/LAN4CC/ISEL, are also acknowledge.

Refeeences

[1] Vieira M., Vieira M. A., Louro P., Vieira P., Fantoni A., "Light-Fidelity (Li-Fi) Optical Sensing and Detection in Large Indoor Environments,"The Ninth International Conference on Sensor Device Technologies and Applications, SENSORDEVICES 2018.

[2] Yang, C. and Shao, H. R., "WiFi-based indoor positioning," IEEE Commun. Mag., vol. 53, no. 3, 150–157 (Mar. 2015).

[3] Lin, X. Y., Ho, T. W., Fang, C. C. , Yen, Z. S., Yang, B. J., and Lai, F., "A mobile indoor positioning system based on iBeacon technology," in Proc. Int. Conf. IEEE Eng. Med. Biol. Soc., 4970–4973 (2015).

[4] Huang, C. H., Lee, L. H., Ho, C. C., Wu, L. L., and Lai, Z. H., "Real-time rfid indoor positioning system based on Kalman filter drift removal and heron-bilateration

location estimation," IEEE Trans. Instrum. Meas., vol. 64, no. 3, 728–739, (Mar. 2015).

[5] Hassan, N. U., Naeem, A., Pasha, M. A., Jadoon, T., and Yuen, C., "Indoor positioning using visible led lights: A survey,"ACM Comput. Surv., vol. 48, 1–32 (2015).

[6] Harle, R., "A survey of indoor inertial positioning systems for pedestrians," Commun. Surv. IEEE Commun. Surveys Tuts., vol. 15, no. 3, 1281–1293, (2013).

[7] Sun, Y. X., Liu, M., and Meng. Q. H., "Wifi signal strength-based robot indoor localization," in IEEE International Conference on Information and Automation (2014).

[8] Bahl P. and Padmanabhan V. N., "Radar: an in-building rf-based user location and tracking system," in Proc. of IEEE INFOCOM, (2000).

[9] Liu M., and Siegwart, R., "Dp-fact: Towards topological mapping and scene recognition with color for omnidirectional camera," in Robotics and Automation (ICRA), IEEE International Conference, 3503 –3508 (2012).

[10] Liu, M., Qiu, K., Li, S., Che, F., Wu, L., and Yue, C. P., "Towards indoor localization using visible light communication for consumer electronic devices," in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Chicago, the USA (2014)

[11] Ozgur E., Dinc, E., Akan, O. B., "Communicate to illuminate: State-of-the-art and research challenges for visible light communications," Physical Communication 17 72–85, (2015).

[12] Armstrong, J., Sekercioglu, Y., Neild, A., "Visible light positioning: a roadmap for international standardization," Communications IEEE, vol. 51, no. 12, pp. 68-73, (2013).

[13] Panta K., Armstrong, J., "Indoor localisation using white LEDs,"Electron. Lett. 48(4), 228–230 (2012).

[14] Komiyama, T., Kobayashi, K., Watanabe, K., Ohkubo, T., and Kurihara, Y., "Study of visible light communication system using RGB LED lights," in Proceedings of SICE Annual Conference, IEEE, 2011, pp. 1926–1928.

[15] Wang, Y., Wang, Y., Chi, N., Yu, J., and Shang, H., "Demonstration of 575-Mb/s downlink and 225-Mb/s uplink bi-directional SCM-WDM visible light communication using RGB LED and phosphor-based LED," Opt. Express 21(1), 1203–1208 (2013).

[16] Tsonev, D., Chun, H., Rajbhandari, S., McKendry, J., Videv, S., Gu, E., Haji, M., Watson, S., Kelly, A., Faulkner, G., Dawson, M., Haas, H., and O'Brien, D., "A 3-Gb/s single-LED OFDM-based wireless VLC link using a Gallium Nitride μLED," IEEE Photon. Technol. Lett. 26(7), 637–640 (2014).

[17] O'Brien, D., Minh, H. L., Zeng, L., Faulkner, G., Lee, K., Jung, D., Oh, Y. and Won E. T., "Indoor visible light communications: challenges and prospects," Proc. SPIE 7091, 709106 (2008).

[18] Schmid, S., Corbellini, G., Mangold, S., and Gross, T. R., "An LED-to-LED Visible Light Communication system with software-based synchronization," in 2012 IEEE Globecom Workshops, 1264–1268 (2012).

[19] Jovicic, A., Li, J., and Richardson, T., "Visible light communication: opportunities, challenges and the path to market," Communications Magazine, IEEE, vol. 51, no. 12, pp. 26–32 (2013).

[20] S T. Komine and M. Nakagawa, "Fundamental analysis for visible-light communication system using led lights," Consumer Electronics, IEEE Transactions on, vol. 50, no. 1, pp. 100–107, (2004).

[21] Monteiro E., and Hranilovic, S., "Constellation design for color-shift keying using interior point methods," in Proc. IEEE Globecom Workshops, Dec., 1224–1228 (2012).

[22] Vieira, M., Louro, P., Fernandes, M., Vieira, M. A., Fantoni A., and Costa, J., "Three Transducers Embedded into One Single SiC Photodetector: LSP Direct Image Sensor, Optical Amplifier and Demux Device" Advances in Photodiodes InTech, Chap.19, 403-425 (2011).

[23] Vieira, M.A., Louro, P., Vieira, M., Fantoni, A., and Steiger-Garção, A., "Light-activated amplification in Si-C tandem devices: A capacitive active filter model" IEEE sensor journal, 12, NO. 6, 1755-1762 (2012).

[24] Vieira M. , Vieira, M. A., Vieira, P., Louro P., "Coupled data transmission and indoor positioning by using transmitting trichromatic white LEDs and a SiC optical MUX/DEMUX mobile receiver," Proc. SPIE. 10231, Optical Sensors 2017, 102310G. (May 16, 2017) doi: 10.1117/12.2265189 (2017).

[25] Vieira, M. A., Vieira, M., Silva, V., Louro, P., Costa, J., "Optical signal processing for data error detection and correction using a-SiCH technology" Phys. Status Solidi C 12, No. 12, 1393–1400 (2015).

[26] Vieira, M. A., Vieira, M., Silva, V., Louro, P. and Barata, M., "Optoelectronic logic functions using optical bias controlled SiC multilayer devices". MRS Proceedings, 1536, 91-96 (2013).

[27] Vieira, M. A., Vieira, M., Costa, J., Louro, P., Fernandes, M., Fantoni, A., "Double pin Photodiodes with two Optical Gate Connections for Light Triggering: A capacitive two-phototransistor model," in Sensors & Transducers Journal Vol. 10, Special Issue, February 2011, 96-120 (2011).

[28] Vieira, M. A., Vieira, M., Louro, P.; Silva, V., Vieira, P., "Optical signal processing for indoor positioning using a-SiCH technology," Opt. Eng. 55 (10), 107105 (2016), doi: 10.1117/1.OE.55.10.107105 (2016).

Similarity Measures and Requirements for Recommending User Stories in Large Enterprise Development Processes

Matthias Jurisch, Stephan Böhm, Maria Lusky

Faculty of Design – Computer Science – Media
RheinMain University of Applied Sciences
Wiesbaden, Germany
Email: {stephan.boehm, matthias.jurisch, maria.lusky}@hs-rm.de

Katharina Kahlcke

User Experience Consulting
DB Systel GmbH
Frankfurt, Germany
Email: katharina.kahlcke@deutschebahn.com

***Abstract*—In mobile application development projects, large enterprises have to face special challenges. To meet these challenges and to meet today's high expectations on user centered design, inter-project knowledge transfer of software artifacts becomes an important aspect for large software development companies. For supporting this kind of knowledge transfer, we propose two approaches based on textual similarity of user stories for a recommendation system: the first approach uses standard information retrieval techniques whereas the second approach uses a more recent approach from language modeling, namely word embeddings. We also present a three-step evaluation of these approaches, comprising of a data analysis, a survey and a user study. The results tend to support the information retrieval approach and not only show that user story similarity rated by users and rated by such an algorithm is connected, but also demonstrate a strong relation between user story similarity and their usefulness for inter-project knowledge transfer. Besides, our evaluation shows that using word embeddings showed worse results than the established information retrieval approach in the domain of large enterprise application development.**

Keywords–Mobile Enterprise Applications; User Stories; Recommendation Systems; User Centered Design.

I. Introduction

In recent years, the user centered design approach has become an integral part of software development, and also for mobile application (app) development. Often, at the beginning of an agile development process, requirements for an app are analyzed and are written down in the form of user stories. They are short requirement descriptions from the user's point of view. Based on these user stories, during the further development process other software artifacts, such as documentation, screen designs, or source code are created to support the development process. Especially in large enterprises, the reuse of these software artifacts can save time and resources, since large software development companies are facing several challenges: There are multiple development projects at the same time, resulting in a large number of software artifacts. Due to a lack of time, these artifacts are often not properly documented in order to support a reuse of these materials and if there is a documentation, the form and content are not standardized. Furthermore, team members often do not know if there is a project with similar requirements and which coworker they can contact about a reuse of software artifacts. In general, large software development companies deal with lack of transparency in development projects, contact persons, and software artifacts.

Saving time and resources through reuse is especially desirable for organizations in the Mobile Enterprise Application (MEA) market: due to digitalization trends, these enterprises are developing many apps for various customers at the same time. Nevertheless, quick time to market is important because of a rapidly changing mobile ecosystem. Additionally, enterprises can only access a limited number of specialists that should focus on demanding tasks and work on difficult problems that have not been solved in the company already. Given this background, in this paper we propose an approach that supports the reuse of software artifacts in mobile app development projects based on textual similarity of user stories. This paper is an extension of [1]. In a previous paper on this problem, we showed that similar user stories can be identified via classical methods of information retrieval [2]. In the following evaluation, we investigate how well these methods work in a real world dataset and compare it to a mor recent approach from the language modeling area, namely word embeddings. We also evaluate how useful our approach is for employees of a large software development company. Therefore, Section II provides an overview on related work. Section III introduces our approach. An evaluation is described in Section IV and its results are presented in Section V. These results are discussed afterwards in Section VI. Section VII concludes this paper and gives an outlook on further research.

II. Foundations and Related Work

Supporting reuse in the context of software development can be facilitated in many ways. One of these ways is best practice sharing, where good solutions for common problems are exchanged within a community of mobile app developers and designers. However, this requires a lot of work and time to find common problems and respective solutions. Especially in the context of MEA development, time is an important factor. Therefore, supporting this process with automated approaches seems to be promising. An automated approach in this area is the use of recommendation systems [4]. Recommendation systems recommend items to users based on item similarities or preferences of similar users. This idea can be applied to software engineering, where a system can recommend software engineering artifacts to developers [5]. The goal of these

recommendation systems is mainly to support the software development process, especially focused on the implementation phase and fixing bugs.

An important field in this area is issue triage. This field deals with supporting the management of bug reporting systems. This includes both recommending specific developers for a given bug report, as well as detecting duplicate bugs. Usually, standard information retrieval techniques or shallow machine learning approaches are used [6]: An approach by Runeson et al. [7] is based on information retrieval techniques and tries to detect duplicates. Other approaches build on including other information besides text, e.g., execution information [8]. In [9], a framework for building recommendation systems for issue triage is presented. This is done by linking both developers and bug reports to software components. Besides using classical information retrieval methods, more recent approaches use deep learning-techniques like word embeddings [10]. While this area also includes computing similarities between textual descriptions in the area of software development, there are some important differences: (1) bug reports are often written from a very developer-centric perspective. (2) They usually contain a lot of technical information like log output. (3) The main goal of issue triage is not to support reuse, but to support bug management tasks.

Another approach to improve the knowledge management in software development projects is to document and store project information in an accessible way, e.g., in architectural knowledge management tools [11]. These approaches have also been applied in industrial case studies and were deemed fit for usage in an industry context: [12] evaluated a semantic architectural knowledge management tool that is based on existing data on software design patterns and their application in software projects. However, if this kind of data is not already available the overhead for documenting usage of design patterns can be too high for an application in practice. This is especially an issue for the fast-paced market of mobile enterprise applications.

In the last decade, user stories as a user-centric representation of requirements were introduced [13]. A typical user story is at most two sentences long and consists of a reference to a certain type of user, a description of an activity this user wants to do with the software and a reason why this will help the user. As an attachment to the user story, acceptance criteria add more detailed information to the user story. Only few approaches exist to support software reuse in the context of user stories: [14] proposes a recommendation system based on user stories and evaluates this system on a project history. However, it is not clear how helpful these recommendations would be when actually working on a new project. In our previous work [2], we evaluated how well information-retrieval-based approaches can distinguish between two types of user stories and which aspects of the user story are important to it.

The established method for text representation in information retrieval is the vector space model which dates back to the 1960s: each document is represented by a vector, where each vector component represents how often a term occurs in the document. This is often accompanied by weighting the terms given their prevalence in the overall corpus. The similarity between documents is computed by the cosine of the angles of their vector representation. However, this approach has a significant drawback: the semantic of terms can not be taken into account: terms like ”pretty” and ”lovely” are treated as completely unrelated terms, the same way as terms like ”driving” and ”universe”. To also represent the meaning of words in search corpora, latent semantic indexing was introduced (LSI) [15]. LSI is based on a singular value decomposition of the term by document matrix, which is the matrix built from all document vectors. While LSI can deal with the synonymy problem in some cases, it still conceptualizes language at the abstraction level of documents and can not determine meaning on a lower level – terms used in similar documents will be regarded as semantically similar by LSI, regardless of their immediate context.

To overecome this issue, more recent approaches use word embeddings, where an embedding should represent a term's context on the level of sentences. One of the most popular embedding approaches, Word2Vec [3], is able to represent semantic information based on an unsupervised learning procedure. This learning procedure is based on two models, Skip-

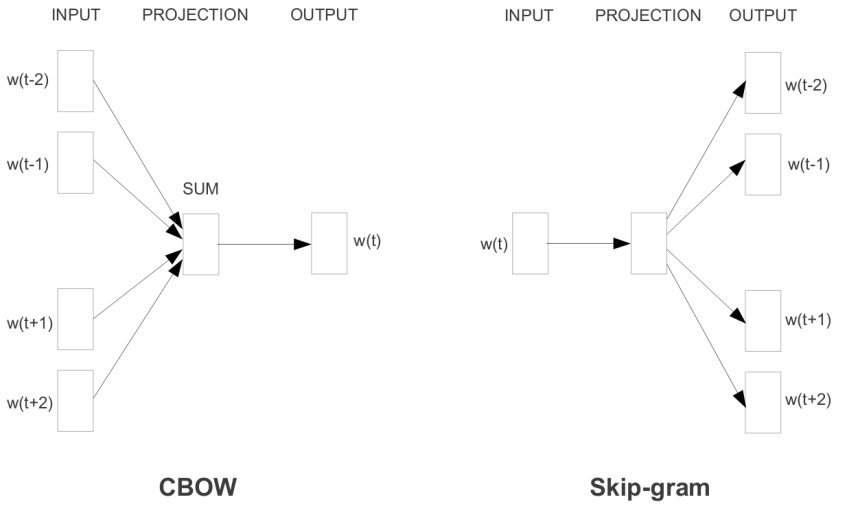


Figure 1. Continuous Bag of Words (CBOW) and Skip-Gram model [3]

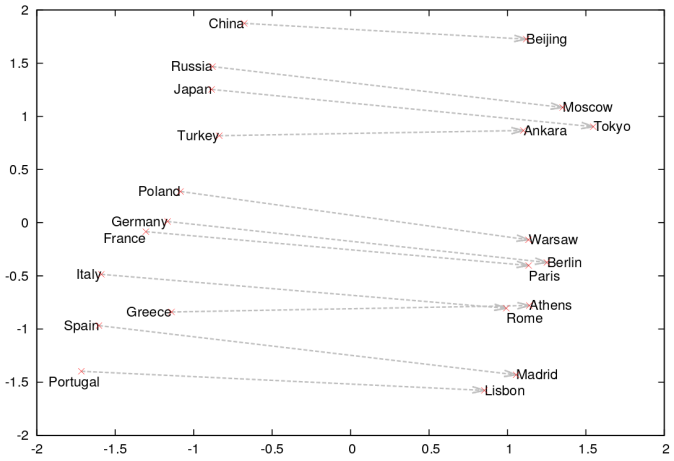


Figure 2. Principal component analysis of word embeddings for countries and capitals [16]

Gram and Continuous Bag of Words (CBOW). Both models are shallow neural network architectures depicted in Figure 1. Given a context of a few words (words $w(t-2) - w(t+2)$) CBOW predicts the word in the center ($w(t)$) using a weighted sum and a hidden embedding layer. E.g., given "The dog ran very fast" as context, CBOW would try to predict the term "ran". The Skip-Gram model is built on the reverse task: given a centroid word ("ran" in the previous example), this model tries to predict the context. As a byproduct of learning these models, entity-specific weight vectors are produced, which capture some semantic relations. An example for these semantic relations is shown in Figure 2. This figure shows selected vectors for countries and capital cities. The semantic relation "is capital of" can be observed through a shift operation in the vector space. This property holds for other relations, too. E.g., the vector from "king" to "queen" is very similar to the vector from "man" to "woman". To our knowledge, text similarity based on word embeddings has not been used for recommending similar user stories.

In this paper, we expand our previous work to an evaluation of how useful recommendations on a real-world software engineering dataset are and what information needs to be contained in these recommendations to make them actually helpful. We also evaluate, how established methods for information retrieval compare to modern approaches like word embeddings in relation to these aspects. This issue has not been addressed by the approaches we mentioned in this section and is required to make recommendations in the context of user stories usable in practice. The only way to evaluate the usefullness of recommendations is to conduct a survey with practitioners from the industry.

III. Recommendation Approaches

Our general approach to supporting reuse is to use similarity measures for documents to recommend textually similar user stories to the story a participant in an app development-project is currently working on. As similarity measures we use established techniques from information retrieval as well as a newer methods from the area of language modeling, namely word embeddings. The information that is attached to the recommended user stories (e.g., screen designs, textual documents or source code) can then be used to support current efforts. In this way, team members could reuse results from different projects without previously knowing about these projects.

A. Information Retrieval Methods

To find textually similar user stories based on established information retrieval methods, a search based on the well-known information retrieval approach Term Frequency-Inverse Document Frequency (TF-IDF) and stop word removal is used. Stop words are words that occur frequently in texts so that they do not contain useful information. Examples for stop words are "I", "the", "a", etc. These words are removed from the user stories before processing the user stories with TF-IDF, which represents texts as follows: Each document d (i.e., a user story) is represented by a vector $\boldsymbol{W}_d$, which contains an entry for each term used in the dataset. Each vector component $\boldsymbol{W}_{d,t}$ represents the importance of a term t for the document d. This representation is computed by the frequency of a given term in the document $tf_{d,t}$ multiplied by the inverse document frequency $\log \frac{N}{df_t}$, where N is the number of all documents and df_t is the number of occurrences for a given term in all documents. This yields the following formula for a document's vector representation:

$$\boldsymbol{W}_{d,t} = tf_{d,t} * \log \frac{N}{df_t}$$

To compute the similarity between documents, the cosine of the angle of two vectors is used. The naive approach for similarities would be to compute the euclidian distance between vectors, however, this would favour documents with similar lengths.

Cosine similarities are then used to order texts regarding their relative similarities. Thus, we do not use similarity scores as an absolute value, but only to distinguish between more and less similar documents. To find similar user stories to one given user story, the similarity is computed according to the described procedure. User stories are then ordered by their similarity and the user story with the highest score is considered the most similar.

B. Word embeddings

To find similar user stories based on word embeddings, first, a word embedding is needed. Word embeddings are usually trained on very large corpusses such as the Google News dataset, which contains around six Billion tokens [3]. Even in large companies, creating a text corpus of that size to learn embeddings is not realistic. Because this issue is not unique to the domain of large companies, pretrained sets of general purpose word embeddings are publically available on the web [17].

However, these word embeddings need to be transformed into a document embedding. This is usually done by averaging the word embeddings for a document:

$$\boldsymbol{e}_d = \frac{1}{|d|} \sum_{t \in d} \boldsymbol{e}_w$$

where $\boldsymbol{e}_d$ and $\boldsymbol{e}_w$ are embeddings of word w and document d and $|d|$ is the length of d. To compute the similarity

between documents, we use the same distance measure as for comparing TF-IDF documents, namely the cosine distance. As with TF-IDF, we can then order documents by their similarity and thus find the most similar user stories to each story.

For implementing this process, we use spaCy [18], a freely available language processing toolkit. The embeddings we use are pre-trained on two corpuses: the TIGER and WikiNER corpuses. The TIGER corpus [19] contains text from newspaper articles from the "Frankfurter Rundschau", a german newspaper. It consist of 50000 sentences containing 900000 tokens. WikiNER [20] is build upon Wikipedia articles from several languages, the german part used in the spacy model consists of 19.8 Million Sentences containing 208 Million Tokens.

IV. Evaluation

The aim of our evaluation is to assess the recommendation approaches described in Section III regarding their suitability in a real-world scenario. To do so, it is relevant to deepen the understanding of the reuse process in order to get a complete picture on the potential usefulness of a recommendation system in large enterprises. Therefore, our evaluation answers the following research questions:

1) Which kind of knowledge transfer is already being practiced?
2) Can an automated recommendation system be useful for supporting knowledge transfer?
3) Is there a relation between user story similarity and their usefulness?
4) How do information retrieval approaches compare to more recent language modeling approaches in this environment?

A. Methodology

To answer these research questions, we conducted an evaluation comprising three steps: In the first step, we analyzed a dataset of user stories from a large German software development company. In the second step, we distributed a questionnaire covering questions about practices in inter-project knowledge transfer in general. In the third step, we invited participants to single sessions where they were asked to solve tasks focusing on user stories in inter-project knowledge transfer.

The number of participants in this study was limited to a small number due to the testing requirements: all participants had to come from the same company with a specific expertise on the implementation of the user stories and as references for the similarity comparison. Thus, the results are far from representative and cannot be considered as an empirical validation. However, the results of this prestudy can five some critical and usable expert feedback on the potential usefulness and applicability of our approach.

B. Dataset

To evaluate the usefulness of recommendations based on user stories in the area of Mobile Enterprise Application Development, we used a dataset of real-world user stories out of nine app development projects provided by an industry partner. The dataset contains 591 user stories, of which 355 are long enough to contain meaningful information. User stories were considered long enough when they were at least 80 characters long, which is roughly two times the length of only the formal aspects of a user story description. This boundary was set by investigating example user stories. A histogram of story length (in characters) is shown in Figure 3. From the distribution of story length and the standard deviation, we can already conclude that the dataset is very heterogeneous, as could be expected in a real-world dataset. The data is not only heterogeneous regarding the textual length, but also regarding their specifity and their degree of abstraction. For example, some user stories describe fixing typos in data protection regulation informations, while others describe a high level view of a location-based service.

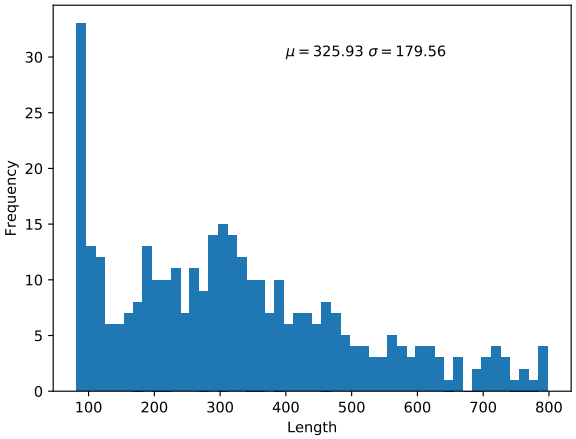


Figure 3. Distribution of User Story Length

For each user story in the dataset, we computed the top five most similar user stories according to TF-IDF, with cosine similarity both for stories from different projects as well as stories from the same project. Stories from the same project shold overall be more similar than stories from different projects, since user stories in one project can deal with overlapping topics. E.g., there can be one story describing a search function, one describing how the search can be accessed and another story describing how search results should be sorted and displayed. A histogram of cosine similarity values between all user stories is shown in Figure 4. Stories in the same project are given higher similarity values than stories from different projects, which indicates that it is possible to differentiate between projects using cosine similarities of TF-IDF vectors.

We followed the same procedure for vectors generated by the word embeddings procedure. Figure 5 shows a histogram of cosine similarities between stories in the same project as well as in different projects. When comparing Figure 4 and Figure 5, one can observe that while for TF-IDF, the average is close to zero, for embedding-based methods it is close to 1. This result can be expected from the nature of the vector spaces: Whereas a TF-IDF vector space has many dimensions and the vectors are relatively sparse, a word embedding space contains only a few hundred dimensions with dense vectors. Another notable difference is that embedding-based methods show a smaller overall variance. In general, the distribu-

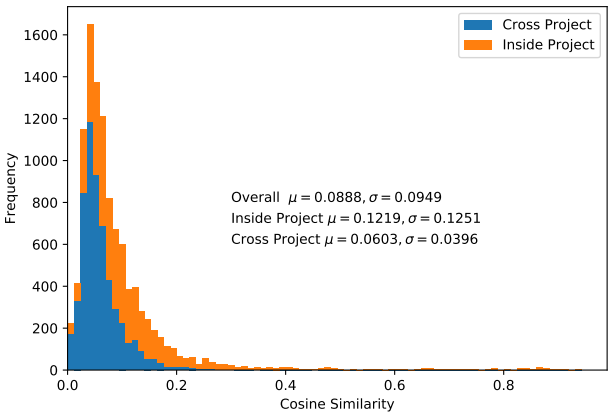


Figure 4. Distribution of TF-IDF User Story Similarities

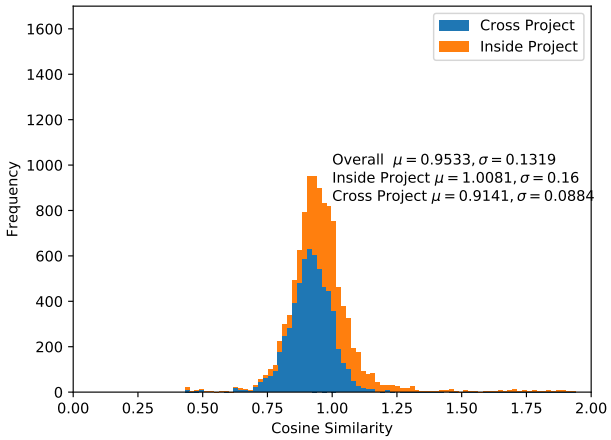


Figure 6. Distribution of Combined Similarities, Unscaled

tion of cross- and inside-project similarity values are more similar for embedding-based methods. This implies that the TF-IDF-based approach might be better suited to distinguish between projects. However, there is a difference between the distributions for cross- and inside project similarities for the embedding approach and the usefulness needs to be further evaluated.

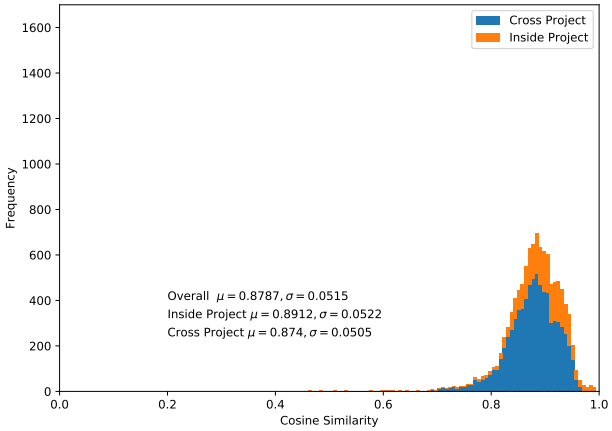


Figure 5. Distribution of Embedding-Based User Story Similarities

To test the effect of combined similarity measures, we repeated these experiments by simply adding the two similarity measures. Results of this approach are shown in Figure 6. Note that this combination of similarity measures has a domain of [0,2]. However, just adding the two similarity measures leads to more influence in the combined similarity measure for the similarity with a higher standard deviation and mean. Hence, with this unscaled version of the combined similarity measure, the similarity is influenced mainly by the embedding-based similarity.

To overcome this issue, we constructed a new similarity measure that scales both similarities to the same mean of 0 and to the same standard deviation. In this way, both similarity measures have an equal influence on the value of the combined similarity measure. The similarity distribution of this combined method is shown in Figure 7. This combined similarity measure is the variant with the least difference between means of Cross- and Inside-Project similarities. Hence, the combined method is the weakest method for distinguishing between projects. The highes differences can be found between TF-IDF scores.

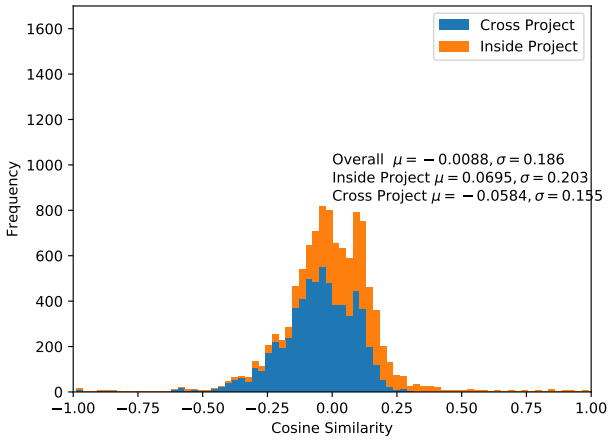


Figure 7. Distribution of Combined Similarities, Scaled

C. Survey

To get an overall overview of the requirements in user story-centered reuse, we designed a questionnaire that comprised ten questions about inter-project knowledge transfer. The questionnaire was online for 17 days and was distributed among the employees of a large German software development company. It was also used for recruitment of participants for the user study. First, the participants had to specify their field of activity. We then explained to them our approach for inter-project knowledge transfer that underlay the questions, which is the re-use of software artifacts and knowledge, such as user

stories, screen designs, documentation or source code, during development projects of mobile applications.

We asked the participants, how useful such a knowledge transfer is and in which way and how regularly it is already being practiced in their department. Further, they had to name obstacles that occur with inter-project knowledge transfer. Then, we asked them to rate the usefulness of particular software artifacts in this context and they had to assess the viability of such a knowledge transfer in their department. They were asked to rate the usefulness of a software that would support knowledge transfer. Lastly, the participants were asked to rate the importance of additional information to specific software artifacts on a five-point scale. Importance may differ from usefulness in certain situations, since it is used to prioritize between different potentially useful artifacts.

D. User Study

The user study was carried out in one-on-one sessions with employees of a large German software development company. Each session lasted 20 to 30 minutes. We selected three user stories for which related user stories were known to be in the dataset. We computed the most similar user stories with both similarity approaches for each of these *reference* user stories with varying levels of similarity: one user story that the algorithm listed most similar, one that it listed as medium similar, and one that it listed as less similar, which lead us to three user story groups, one for each reference story.

Based on the reference user stories, the participants were asked to solve three tasks. First, they had to rank the user stories obtained by the algorithm regarding their similarity to the reference user story from the most similar to the least similar one. Then, they should rate the usefulness of each of the similar user stories. To determine the usefulness of the user stories, participants were told to estimate how much artifacts (e.g., source code, design documents or documentation) produced during an implementation of a ranked user story could contribute to the implementation of the corresponding reference user story. Note that this is not included in similarity aspects: user stories can cover a roughly similar topic, however, different levels of abstraction, different user types or platforms or technical aspects could make it impossible to actually reuse the results of the implementation of a user story in a different context. Such user stories would be considered similar by users, but finding these stories would not actually support reuse of artifacts related to one user story to another. Concluding the session, they were asked to name additional information that should be provided by the recommendation system in order to support the implementation of the reference user story.

V. Results

The questionnaire was answered by nine employees of a large German Software development company. While nine participants are not enough to allow a detailed statistical analysis, this number is in general considered enough for usability testing [21]. Eight participants specified their field of expertise as conception, one as implementation.

All of the participants rated the knowledge transfer described by us (that is, the re-use of software artifacts and knowledge, such as user stories, screen designs, documentation or source code, during development projects of mobile applications), using a five-point scale from 1 – not useful at all to 5 – very useful, as very useful or rather useful (median=5; maximum=5; minimum=4). The currently used types of knowledge transfer selected from a list of pre-made options are shown in Figure 8. All of the participants stated that they already practiced this kind of knowledge transfer, seven via one-on-one conversations or group conversations, four via e-mail, chats or by using knowledge bases, and three via phone calls. On average, each person practices three methods of knowledge transfer. Only one stated to practice it on a regular basis, and eight practice it as needed. Obstacles for knowledge transfer selected by participants from a list of possible obstacles are shown in Figure 9. The most often named obstacle was missing information about a contact person (eight), followed by missing documentation of content and materials and lack of time (six respondents each). The participants described further obstacles as being unaware of the existence of reusable materials, as well as not knowing where to look for information regarding reusable artifacts.

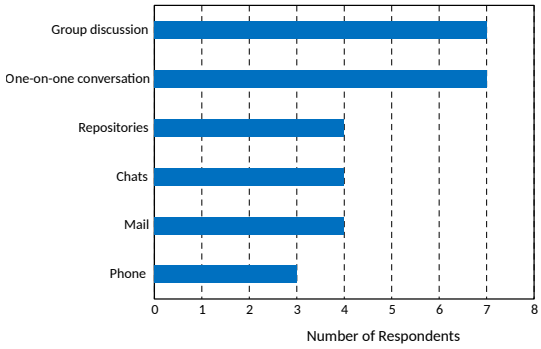


Figure 8. Currently Used Types of Knowledge Transfer

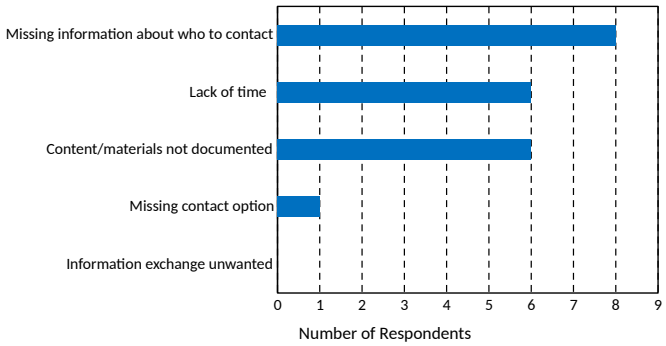


Figure 9. Existing Barriers for Knowledge Transfer

User ratings for usefulness of artifacts for Knowledge transfer on a five-point scale are shown in Figure 10. Screen designs were rated as most useful (median=5, maximum=5, minimum=3), followed by documentation of the software architecture (median=4, maximum=5, minimum=3). Ratings for potential usefulness, implementability and importance are shown in Figure 11. Regarding the viability of such a knowledge transfer in their department and in relation to specific software artifacts, the highest implementability was considered for screen designs, followed by documentation of the software

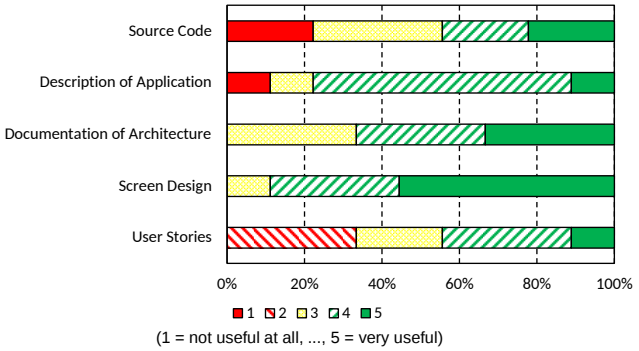


Figure 10. Perceived Usefulness of Artifacts

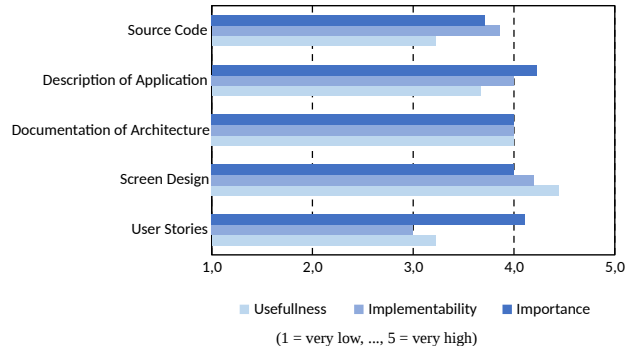


Figure 11. Responses on Potential Recommendation System Artifacts

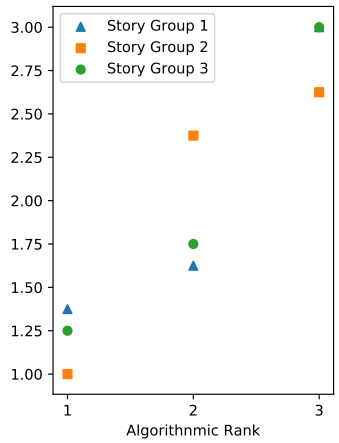


Figure 12. Ranking of user stories by TF-IDF and mean ranking by participants.

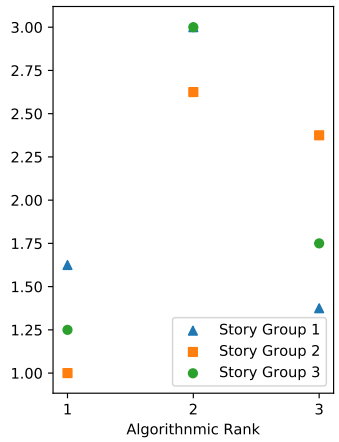


Figure 13. Ranking of user stories by Word-Embedding-Similarity and mean ranking by participants.

architecture and use case descriptions. Furthermore, the answers revealed that most knowledge transfer that is already practiced concerns screen designs (done by 4 participants), documentation of the software architecture (2 participants) and user stories (1 participant).

The participants rated a software solution for supporting knowledge transfer on a five-point scale as rather useful (median=4; maximum=5; minimum=1), with 6 participants considering it rather useful or useful. Regarding the importance of additional information, information on use case descriptions were rated as most useful (median=4; maximum=5; minimum=3). In general, any kind of additional information (e.g., source code, screen designs, architecture documentation) was rated as ”rather important” for all kinds of software artifacts.

Of the eight participants of the user study, all were working in conception respectively design. Results of the user study are shown in Figures 12-15. Results of the first task show that the participants ranked the user stories similar to the ranking of the algorithm. Figure 12 shows the ranking of story similarity to the reference story by the TF-IDF algorithm and the mean ranking by the participants. The data shows that user story similarity of the algorithm seems to resemble the perceived user story similarity by humans: The three user stories ranked as most similar by the algorithm also got the highest similarity rankings by the participants. The user stories ranked as second by the algorithm were partially ranked as more and partially as less similar, but in general reflect the algorithmic ranking. For stories that are not obviously the least or most similar, this result was to be expected. Accordingly, the three least similar user stories of the participants match those ranked by the algorithm.

The results of the same experiment with the embedding model are shown in Figure 13. While the stories rated as most similar by the algorithm were also perceived as rather similar by the users, some stories that where perceived as least similar by the algorithm where rated as very similar by the algorithm. This is especially the case in story group 1, where the story that was perceived as most similar by the users was ranked as least similar by the algorithm. The stories ranked as second most similar by the algorithm are ranked least similar by the users. Overall, there does not seem to be much agreement between user- and algorithmic ratings when using the embedding-based approach.

We also repeated this experiment with a combination of TF-IDF and Embeddings we already used in Section IV-B.

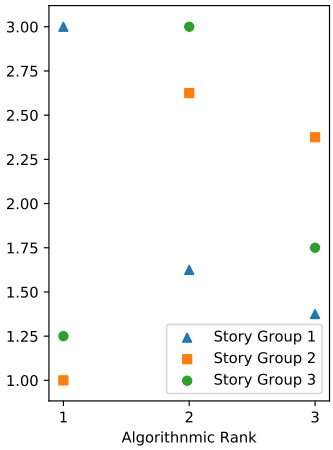


Figure 14. Ranking of user stories by Word-Embedding-Similarity combined with TF-IDF-Similarity and mean ranking by participants.

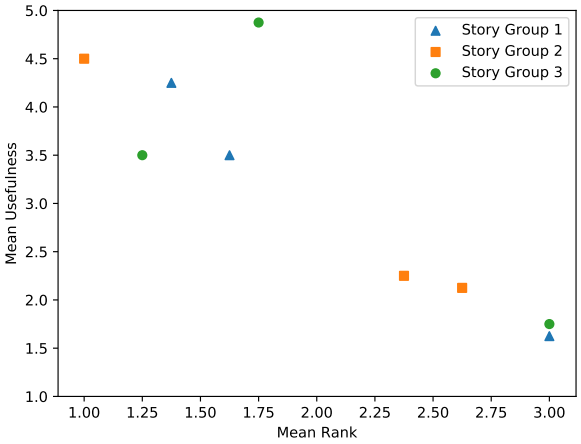


Figure 15. Mean estimated usefulness and mean ranking by participants per user story.

We only used the version with un-weighted addition of the combination, since this lead to slightly better results when distinguishing between projects. From the graph we can see no connection between algorithmic rank and average user scores. In contrast to the similarity measure using only word embedding similarities, where some connection between human and algorithmic rank was visible, no connection can be found.

Further on, for each user story in the three user story groups, we calculated the mean similarity ranking to the reference story given by the participants and the mean usefulness rating, according to the rated usefulness of a computed user story for the implementation of the reference story. As Figure 15 shows, there are two groups of user stories that are delineated from each other. The first group has higher usefulness values (3.50 to 4.88) and higher similarity rankings (1.00 to 1.75), while the second group has lower usefulness values (1.63 to 2.25) and lower similarity rankings (2.38 to 3.00). However, the user story with the highest usefulness (4,88) is ranked with medium similarity (1.75), while the two user stories with the lowest usefulness (1.63 and 1.75) are ranked as least similar.

VI. Discussion

The results of the questionnaire show that, in general, people appreciate knowledge transfer and that our approach for it meets the users' needs. This is also reflected in the fact that seven of our participants already practice this kind of knowledge transfer. Although, direct contact and personal conversations are the preferred ways of doing so. Electronic ways for contacting each other are rarely used. One reason for that might be that electronic methods, such as emails, chats or knowledge bases, do not meet the user needs for knowledge transfer. Nevertheless, for all these methods the user needs to know, which person can be contacted for further information – our approach takes this important feature into account, which was also the most often named obstacle for knowledge transfer. The second obstacle is insufficient documentation – this problem is also taken account of in our approach, since it simplifies documentation by searching existing software artifacts based on similarities. In general, the answers confirm that our approach addresses the right issues and helps to eliminate the obstacles for knowledge transfer that have been named by the participants. The software tool proposed by us was said to be most useful for screen design. This answer is not unusual, since almost all participants of our questionnaire work in design and conception. However, screen design is their main field of activity and thus, we consider it positive that our approach is rated as useful for this field. Further, the viability was rated highest for screen designs. These results give some evidence that our approach can create additional value in one of the most important fields for knowledge transfer. All in all, two thirds of the participants consider our approach useful or very useful.

The results of the user study show that the user story similarity of the TF-IDF algorithm seems to be connected to the perceived human user story similarity. For the embedding-based similarity algorithm this is not the case. The most likely cause of embeddings not working in this case is that our dataset contains a lot of words that are very specific to the domain of mobile application development. Therefore, they do not have a meaningful embedding or no embedding at all, since we used embeddings theat were pretrained on a general purpose dataset. However, these domain-specific words are often the words that carry a lot of meaning, since they express concepts that are highly relevant to a domain.

One mitigation for this issue would be retraining the word embeddings on a set of documents from the domain. However, given the small number of documents that use these words it is unclear if this can work. Even if documents of a relevant size were available, a significant amount of computing power would be required to train these embeddings.

Another mitigation would be a manual clustering of these domain specific important terms to capture their semantic relations. These clusters could then be used to improve search results for TF-IDF-based searches. However, finding these clusters and maintaining them requires a significant amount of manual work.

A combination of TF-IDF and Word-Embedding-based methods showed less connection between perceived similarity and algorithmic similarity. The most likely cause for this is that the disagreement between both similarity measures causes so much noise, so that the overall performance is impacted. This might be addressed by further tuning the wights of a combined approach, but given the poor performance of general-purpose embeddings for this use case, this does not seem like a very promising approach.

Furthermore, user story similarity mostly coincides with their usefulness. The data indicates that a low similarity implies a low usefulness. However, the most similar stories are not always the most useful ones, but to some extent a high similarity seems to be connected to higher usefulness. This confirms our assumption that similarity between user stories and usefulness for the implementation of another user story are not necessarily the same, since usefulness can be influenced by several dimensions of similarity: two user stories can share the same implementation technology, but not the same domain or vice versa. Two user stories can share the same domain and technology but one may use an outdated version of an API or adhere to a human interface guideline that has become obsolete. Hence, it is important to address these factors like this when building a recommendation system in the area of mobile enterprise application development.

VII. Conclusions and Further Research

In conclusion, the evaluation provided valuable findings, so that our research questions can be answered as follows:

1) Which kind of knowledge transfer is already being practiced?
   Mainly, knowledge transfer takes place in personal conversations between two people and groups. It is not carried out on a regular basis, but as required. Our results suggest that everyone does practice knowledge transfer in one way or the other.
2) Can an automated recommendation system be useful for supporting knowledge transfer?
   Our results show that an automated recommendation system is a useful tool for supporting knowledge transfer, especially for screen designs.
3) Is there a relation between user story similarity and their usefulness?
   The results of our user study indicate that similarity and usefulness are not necessarily the same, but there is a relation between user story similarity and their usefulness. Further, there also is a connection between user story similarity rated by an algorithm on the one hand and humans on the other hand.
4) How do information retrieval approaches compare to more recent language modeling approaches in this environment?
   On our dataset, established information retrieval approaches performed better than an embedding-based language modeling approach. The likely cause of this is the domain-specific vocabulary in this area.

As a next step, another iteration of the evaluation could be made in different companies, in order to receive results that are applicable in several contexts of work. Also, more approaches for computing similar user stories could be evaluated: A comparative study of textual similarity approaches such as word movers distance [22] or taking metadata into account, would provide valuable insights.

Acknowledgment

This work was funded by the German Federal Ministry of Education and Research, grant no. 03FH032PX5; the PROFRAME project at RheinMain University of Applied Sciences. All responsibility for the content of this paper lies with the authors.

References

[1] M. Lusky, M. Jurisch, S. Böhm, and K. Kahlcke, “Evaluating a User Story Based Recommendation System for Supporting Development Processes in Large Enterprises,” in CENTRIC 2018, The Eleventh International Conference on Advances in Human-oriented and Personalized Mechanisms, Technologies, and Services, 2018, pp. 14–18.

[2] M. Jurisch, M. Lusky, B. Igler, and S. Böhm, “Evaluating a recommendation system for user stories in mobile enterprise application development,” International Journal On Advances in Intelligent Systems, vol. 10, no. 1 and 2, 2017, pp. 40–47.

[3] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” CoRR, vol. abs/1301.3781, 2013. [Online]. Available: http://arxiv.org/abs/1301.3781

[4] M. Robillard, R. Walker, and T. Zimmermann, “Recommendation systems for software engineering,” IEEE Software, vol. 27, no. 4, 2010, pp. 80–86.

[5] D. Cubranic, G. C. Murphy, J. Singer, and K. S. Booth, “Hipikat: A project memory for software development,” IEEE Trans. Softw. Eng., vol. 31, no. 6, Jun. 2005, pp. 446–465. [Online]. Available: https://doi.org/10.1109/TSE.2005.71

[6] A. Goyal and N. Sardana, “Machine learning or information retrieval techniques for bug triaging: Which is better?” e-Informatica Software Engineering Journal, vol. 11, no. 1, 2017, pp. 117–141.

[7] P. Runeson, M. Alexandersson, and O. Nyholm, “Detection of duplicate defect reports using natural language processing,” Proceedings - International Conference on Software Engineering, 2007, pp. 499–508.

[8] X. Wang, L. Zhang, T. Xie, J. Anvik, and J. Sun, “An Approach to Detecting Duplicate Bug Reports using Natural Language and Execution Information,” Proceedings of the 30th international conference on Software engineering, 2008, pp. 461–470.

[9] J. Anvik and G. C. Murphy, “Reducing the Effort of Bug Report Triage: Recommenders for Development-Oriented Decisions,” ACM Transactions on Software Engineering and Methodology (TOSEM), vol. 20, no. 3, 2011, pp. 10:1–10:35.

[10] S. Mani, A. Sankaran, and R. Aralikatte, “Deeptriage: Exploring the effectiveness of deep learning for bug triaging,” arXiv preprint arXiv:1801.01275, 2018.

[11] R. Capilla, A. Jansen, A. Tang, P. Avgeriou, and M. A. Babar, “10 years of software architecture knowledge management: Practice and future,” Journal of Systems and Software, vol. 116, 2016, pp. 191–205.

[12] M. Sabou et al., “Exploring enterprise knowledge graphs: A use case in software engineering,” in European Semantic Web Conference. Springer, 2018, pp. 560–575.

[13] M. Cohn, User Stories Applied: For Agile Software Development. Redwood City, CA, USA: Addison Wesley Longman Publishing Co., Inc., 2004.

[14] H. Pirzadeh, A. D. S. Oliveira, and S. Shanian, “ReUse : A Recommendation System for Implementing User Stories,” in International Conference on Software Engineering Advances, 2016, pp. 149–153.

[15] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, “Indexing by latent semantic analysis,” Journal of the American society for information science, vol. 41, no. 6, 1990, pp. 391–407.

[16] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States., 2013, pp. 3111–3119. [Online]. Available: http://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality

[17] T. Mikolov, "Google code: Word2vec," https://code.google.com/archive/p/word2vec/, [retrieved: February 2019], 2013.

[18] Explosion AI, "spaCy – Industrial-Strength Natural Language Processing," spacy.io, [retrieved: February 2019], 2019.

[19] S. Brants, S. Dipper, P. Eisenberg, S. Hansen-Schirra, E. König, W. Lezius, C. Rohrer, G. Smith, and H. Uszkoreit, "Tiger: Linguistic interpretation of a german corpus," Research on language and computation, vol. 2, no. 4, 2004, pp. 597–620.

[20] J. Nothman, N. Ringland, W. Radford, T. Murphy, and J. R. Curran, "Learning multilingual named entity recognition from Wikipedia," Artificial Intelligence, vol. 194, 2012, pp. 151–175. [Online]. Available: http:/dx.doi.org10.1016/j.artint.2012.03.006

[21] J. Nielsen, "How many test users in a usability study?" https://www.nngroup.com/articles/how-many-test-users/, website. [retrieved: February, 2019], 2012.

[22] M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger, "From word embeddings to document distances," in International Conference on Machine Learning, 2015, pp. 957–966.

A Framework for Semantic Description and Interoperability across Cyber-Physical Systems

Amita Singh
Technical University Kaiserslautern
Email: amitas@kth.se

Fabian Quint
German Research Center for Artificial Intelligence (DFKI)
Email: mail@fabian-quint.de

Patrick Bertram
Technologie-Initiative SmartFactoryKL e.V.
Email: bertram@smartfactory.de

Martin Ruskowski
German Research Center for Artificial Intelligence (DFKI)
Email: martin.ruskowski@dfki.de

***Abstract*—With the advent of Industry 4.0 and human-in-the-loop paradigms, Cyber-Physical Systems (CPS) are becoming increasingly common in production facilities, and, consequently, there has been a surge of interest in the field. In production systems, CPS, which assist humans in completing tasks are called assistance systems. Most recent designs proposed for assistance systems in the production domain are monolithic and allow only limited modifications. In contrast, this work considers an assistance system to have a hybrid architecture consisting of a central entity containing the process description (or instructions) and one or more plug-and-play Cyber-Physical Systems to retrieve relevant information from the physical environment. Such a design allows the overall system capabilities to be adapted to the needs of workers and tasks. In this paper, a framework is presented for designing the CPS modules using Semantic Web technologies, which will allow (i) interpretation of all data, and (ii) interoperability among the modules, from the very outset. Furthermore, a knowledge description model and ontology development of a CPS module is described. Two different models of maintaining ontologies and the ecosystem are described along with their advantages and disadvantages. An approach is illustrated with the help of a use case for implementing the framework to design a module, data exchange among modules, and to build a sustainable ecosystem of ontologies, which enables rapid development of third-party CPS modules. An implementation is provided discussing hardware, software and communication design of such a module and future direction of research is discussed.**

***Keywords*—human-centered CPS; assistance systems; adaptive automation; ontology; interoperability.**

I. Introduction

An ever growing catalogue of products, [1] short product life-cycle, competitive product costs, and changing demographics have led to a demand of reactive and proactive production systems that can adapt to the changing needs [2]–[4]. According to the European Factories of the Future Research Association, human-centricity is a prerequisite for the production systems to be flexible and adapt to the changing demographics [5][6]. Thus, major efforts are being made to make adaptive human-centered CPS (H-CPS) where machines and automation adapt to the physical and cognitive needs of humans in a dynamic fashion [7][8].

In this paper, assistance systems are considered as H-CPS in production systems. Assistance systems assess the production process using sensors embedded in the environment and, based on the state of the process, provide instructions to workers through visualisation devices attached to them [9]. Although humans have unparalleled degree of flexibility, i.e., humans can adapt to varying production, major focus is being placed on increasing the flexibility of automation systems that help workers during processes. Emerging developments like modularity, Service-Oriented Architecture (SOA), interoperability by the virtue of common semantic description (e.g., administrative shell [10][11]), and edge-computing [12] are rarely applied to H-CPS.

In this paper, a CPS-based assistance system, which adapts to a worker's need by exploiting the benefits of such techniques is proposed. Such an assistance system has a central system and one or more CPS modules attached to it as shown in Figure 1. CPS modules feed information extracted from the environment to the central system. The central system, in turn, processes this information to assess the state of the process and the worker's needs.

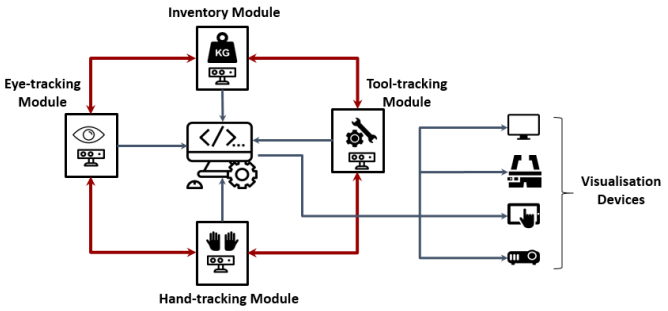


Figure 1. Schematic description of an assistance system.

To the best of the authors' knowledge, no design so far allows one module to access and use the data from other modules. In this paper, semantic design of modules and interoperability between different parts of an assistance system are discussed in detail, and consequently, a Semantic Description and Interoperability (SDI) framework is proposed.

In the remainder of the paper, first the state-of-the-art and related work is presented in Section II and then the concepts of modularity and interoperability are discussed in detail in Section III. Next, Section IV discusses the SDI framework for design of modules and Section V further explains ecosystem of ontologies. In Section VI, the development of such modules is discussed. Finally, the implementation of an assistance system is simulated using the proposed framework in Section VII, followed by the conclusion and potential future work.

II. Related Work

The vision of Computers in Manufacturing (CIM) of creating completely automated factories could not be realized due to the complexity involved in production processes [13]. The effort to implement CIM made it clear to engineers that completely automated factory is not a plausible solution as per the state-of-the-art. Humans are an indispensable part of production systems but automation at different stages of product is a necessity and a practical approach to the problems of increasing product variants, reducing product life-cycle and rising labour costs [5]. Thus, CIM established that it is important that the CPS systems developed should help humans instead of trying to replace them because with the current state of

the technology it is difficult to replicate human cognitive skills. This has lead to human-centric workplaces and the approach was coined human-in-the-loop.

Human-system interaction is an indispensable part of the production systems and acts as an enabler of the intelligent decision making process [14]. In complex production scenarios, the symbiotic man-machine systems are the optimal solution. This change in the nature of human-machine led to the paradigm shift from independently automated and manual processes towards a human-automation symbiosis called human cyber-physical systems. These systems are characterized by collaborative effort of workers and machines and aim at assisting workers being more efficient and effective [5]. These systems are based on a trusting and interaction-based relationship, which has human supervisory control and human situation awareness, leading to adaptive automation to improve knowledge of worker and help the process.

Production facilities are focusing on cyber-physical systems (CPSs) that can interact with human through many modalities. CPSs are a combination of interacting embedded computers and physical components. Both computation and physical processes work in parallel to bring about the desired output. Computers usually monitor the physical processes via sensors in real-time and provide feedback to actuators [15], [16]. A CPS consists of one or more micro-controllers to control sensors and actuators which are necessary to collect data from and interact with its environment. These systems also need communication interface to exchange data with other smart devices and a cloud. According to Jazdi [16], data exchange is the most important feature of cyber physical systems. CPSs connected over internet are also known as Internet-of-Things.

Its vision is to bring automation in the field of production and help combat the problems of increasing catalogue and labour costs. The information and communication technologies have trickled their way down to the production systems, paving the way for monolithic production systems to become modular and have decentralized control architectures. It is one of the most significant directions in computer science, information & communication technologies and manufacturing technologies. With the increasing sophistication of actuators and sensors available in the market, availability of data has increased many folds. The CPSs used to create flexible and re-configurable production systems called Cyber-Physical Production Systems (CPPSs). CPPSs are build on the principle of modularity and decentralized control. Thus, these modules are loosely coupled with each other.

Neither traditional nor fully automated systems can respond effectively and efficiently to dynamic changes in the system [17]. Hence, workers should be assisted, as needed, in their work, thus, including automation with human aptitude as a trouble shooter. Manual assembly stations with assistance systems are developed based on this concept. These stations are modular units, one in the chain of many automated/semi-automated stations. Products are assembled by workers at each station. Usually, production facilities are one piece flow. Depending upon the assembly plan of a product, one or more processes can be performed on a station.

This work brings together two different areas of research: development of CPS for production, as well as the semantic design of these systems. Related work in both areas are discussed separately and some aspects are discussed in detail.

A. Assistance Systems

Production facilities are focusing on CPS that can interact with human through many modalities. There is significant contemporary research interest in using sensor technology for developing context-aware CPS [9][18]–[20]. Nelles et al. have looked into assistance systems for planning and control in production environment [18]. Gorecky et al. have explored cognitive assistance and training systems for workers during production [19]. Pirvu et al. [21] talk about the engineering insights of human centered, yet highly automated, cyber-physical system while (i) keeping in mind adaptive control, (ii) cognitive assistance, and (iii) training in manual industrial assembly. The aim of such a system is to design a mobile, personal assembly work station which assists the worker in task solving in real time while understanding and inducing work-flows. Standardized abstractions and architectures help the engineers in the design phase by reducing the complexity involved in building such systems [22]. Zamfirescu et al. have also integrated virtual reality and a hand-tracking module to help workers during assembly processes [20]. Figure 1 shows a schematic description of such an assembly station. These stations are equipped with different visualization techniques and sensor technologies. Visualization techniques, like projectors and smart glasses as shown in Figure 1, help workers during assembly process by displaying instructions. Interactive screens can also be installed at assembly stations using which workers can interact with stationery computers when required. Assembly stations have areas dedicated for storing tools and parts used during assembly. Sensors can be employed to track usage of tools and control inventory of parts. RFID readers are installed on products and to know the current status of products in addition to the tools and parts as in the traditional workstations.

Assistance system derives its principles from the Operator 4.0 principle where workers are provided machines to aid their work. It helps workers by reading the product status available with products in machine readable format, collecting other information about the environment and helping the worker to decide the next step to be taken based on the information it receives. Hence, this system can be seen as a context aware human-centric cyber-physical system. As shown in Figure 1, this system consists of a central system and one or more CPS modules.

These CPS modules are built on the principle of plug-and-produce. The analogy is drawn from plug-and-play concept in computer science [23]. Plug-and-produce means a smart device can be easily added or removed, replaced without disrupting functioning of the system. The system should continue working while a CPS module is being added or removed. Additionally, the system should be able to recognize the newly added CPS. This process is different from the traditional processes in which systems need to be reprogrammed and machines are stopped for reconfiguration. Time taken in the complete process is counted as downtime. Similarly, in case of plug-and-produce systems maintenance can be done by removing only the required CPS module while the complete system continues working.

For this purpose, each CPS module should have its own environmental information and it should provide this information to the system to which it is being attached [23]. This gives central system the leeway to reconfigure and requires CPS modules to be smart and adaptive which demands CPS modules to have certain level of intelligence.

Very recently, Quint et al. have proposed a hybrid architecture of such a system, which is composed of a central system and modules which can handle heterogeneous data [9]. However, they do not explore standardizing the design of such modules. In this work, a framework for designing CPS modules and an ecosystem for ensuring interoperability across these modules is proposed.

B. Semantic Design

The Semantic Web is an extension of World Wide Web that promotes common data formats and exchange protocols on the Web through standards. Wahlster et al. [24][25] use Semantic Web technologies to represent and integrate industrial data in a generic way. Grangel et al. [11] discuss Semantic Web technologies in handling heterogeneous data from distributed sources using light-weight vocabulary. Semy et al. [26] describe these technologies as the key enabler for building pervasive context-aware system wherein independently developed devices and softwares can share contextual knowledge among themselves.

Recently, there have been some efforts towards discussing the need of bringing more semantics and data-driven approaches to Industry 4.0. Cheng et al. [27] identify varying degree of semantic approach and further provide guidelines to engineers to select appropriate semantic degree for different Industry 4.0 projects. Wahlster et al. [24] talk about the importance of semantic technologies in mass production of smart products, smart data and smart services. Semantic service matchmaking in cyber-physical production systems is presented as a key enabler of the disruptive change in the production logic for Industry 4.0. Obitko et al. [28] introduce the application of semantic web technologies in handling large volumes of heterogeneous data from distributed sources. Grangel et al. [11] describe an approach to semantically represent information about smart devices. The approach is based on structuring the information using an extensible and light-weight vocabulary aiming to capture all relevant information. Semantic Web technology formalisms, such as Resource Description Framework (RDF), RDF Schema and Web Ontology Language (OWL), help solve the major hurdle towards description and interoperability between CPS by annotating the entities of a system. Some of the major advantages of using RDF-based semantic knowledge representation are briefly discussed here:

Global unique identification. Semantic Web describes each entity within a CPS and its relations as a global unique identifier. According to the principles of Semantic Web, HTTP URIs/IRIs should be used as the global unique identifiers [29]. This ensures disambiguation, and retrieval, of entities in the complete system. As a consequence, a decentralised, holistic and global unique retrievable scheme of CPS can be established.

Interoperability. Interoperability is the ability to communicate and interconnect CPS from different vendors. It is vital in order to have cost effective rapid development. According to domain experts [11][25][30], RDF and Linked Data are proven Semantic Web technologies for integrating different types of data. Gezer et al. [31] mention that OWL-S ensures better interoperability by allowing services to exchange data and allowing devices to configure themselves.

Apart from the above mentioned advantages, by using RDF representation different data serialization formats, for example RDF/XML, RDF/OWL can be easily generated and transmitted over the network [11]. Further, data can be made available through a standard interface using SPARQL, a W3C recommendation for RDF query language [32].

Recently, Negri et al. [33] discussed requirements and languages of semantic representation of manufacturing systems and conclude that ontologies are the best way of such representations in the domain. The authors also highlighted importance of ontologies in providing system description in an intuitive and human-readable format, standardization not only in terms of definitions and axioms, but also standardizing Web-services and message-based communication. This not only makes engineering of the system streamlined but also facilitates interoperability between parts of the system. In his seminal work, Nocola Guarino formally defined ontologies both as a tool for knowledge representation and management, as well as a database for information extraction and retrieval [34]. In particular, he describes how ontologies can play a significant role during development, as well as run-time, for information systems.

Further, Niles et al. [35] highlighted the usefulness of upper ontologies in facilitating interoperability between domain-specific ontologies by the virtue of shared globally unique terms and definitions (HTTP URIs/IRIs) in a top-down approach of building a system. Semy et al. [26] also described mid-level ontologies as a bridge between upper ontologies and domain-specific ontologies, which encompass terms and definitions used across many domains but do not qualify as key concepts. Furthermore, Sowa et al. [36] discussed ontology integration and conflicts of data in the process. They conclude that ontology merge is the best way of ontology integration as it preserves complete ontologies while collecting data from different parts of the system into a coherent format. In the remainder of the paper, unless otherwise stated, the definition of ontologies and standards as given by W3C [32] are followed.

C. Ontologies

In computer science, ontologies were developed for the Semantic web. The aim of Semantic web is to help software agents interact and share information over the internet. This is done by encoding the data in a machine interpretable language using constraints defined in the domain ontology. This lets software agents locate resources to extract and use information on the web. This differentiates ontologies from other traditional languages, like UML and SysML, used to describe software structure.

Ontologies conceptualise a domain by capturing its structure. In this section, some features of ontologies, which are relevant for the proposed design are discussed. Ontologies are used to explicitly define entities and relations between entities. Figure 2 shows an example of a small ontology, an associated SPARQL query language, and query results obtained during a run-time. Ontologies provide unique global addresses to all entities and relations using HTTP URIs/IRIs. Hence, with the virtue of HTTP URIs/IRIs, entities and relations can be referred to easily from within and outside the system. Ontologies can also be *imported*, which is how definitions of entities and their relationships can be re-used during development time. This feature, as shown in the work later, is crucial in creating an ecosystem of ontologies. During run-time, *individuals* of the entities along with their relationships with each other are created.

In the remainder of the section, important features relevant for this work are discussed:

Upper ontologies are high-level, domain-independent ontologies, providing a framework by which disparate systems may utilize a

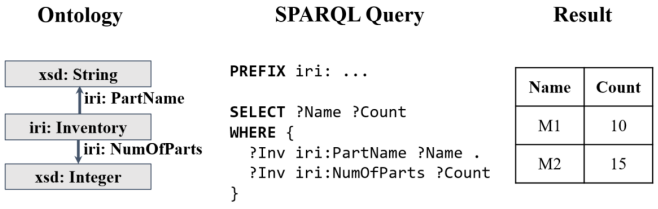


Figure 2. An example of ontology definitions and relations, SPARQL query and results.

common knowledge base and from which more domain-specific ontologies may be derived [26]. Thus, upper ontologies facilitate interoperability between domain-specific ontologies by the virtue of shared common terms and definitions [37]. They contain definitions and axioms for common terms that are applicable across multiple domains and provide principled forms of conceptual inter-linkage between data [38]. Thus, provide semantic integration of domain ontologies.

On the other hand, **domain ontologies** have specific concepts particular to a domain and represent these concepts and their relationships from a domain-perspective. Multiple domains can have the same concept but their representation may vary due to different domain contexts. Domain ontologies inherit the semantic richness and logic by importing upper ontologies.

Another important feature of upper ontology is the structure that they impose on the ensuing ontologies: they promote modularity, extensibility, and flexibility. According to Semy et al. [26], upper ontologies can be built using two approaches: top-down and bottom-up. They discuss benefits and limitations of both approaches. In a top-down approach domain ontology uses the upper ontology as the theoretical framework and the foundation for deriving concepts [26]. In a bottom-up approach, new or existing domain ontologies are mapped to an upper ontology. This approach also benefits from the semantic knowledge of upper ontology but the mapping can be more challenging as inconsistencies may exist between the two ontologies. For example, two teams may have different vocabulary for a similar semantic variable. In this case, mapping the two ontologies to an upper ontology would have inconsistencies. These inconsistencies are resolved as and when needed. However, usually a combination of both approaches is used to design upper ontologies.

The solution proposed to the problem of interoperability across modules relies heavily on the idea of upper ontologies. Upper ontology starts with defining a set of high level entities and then successively adding new content under these entities [35]. The solution incorporates both the top-down and bottom-up approaches. Depending on the need entities are added to the high level ontology.

Mid-level ontologies act as a bridge between basic vocabulary described in the upper ontology and domain-specific low-level ontology. This category of ontologies may also encompass terms and definitions used across many domains.

Ontology development can be seen as defining structure, constraints and data for other programs to use. Software agents and other problem solving methods can use these ontologies as ready-made data that can be fed to the program in order to understand the vocabulary and basic principles of the domain. The independently developed ontologies need to join to exchange data.

Ontology integration is the process of finding commonalities between two ontologies, for example Ontology A and ontology B, and a third ontology C is derived from it. This new ontology C facilitates interoperability between software agents based on ontologies A and B. The new ontology C may replace the old ontologies or may be used as only an intermediary between systems based on ontologies A and B are merged in a third ontology C. Ontologies can be integrated primarily in three ways depending on the amount to change required to derive the new ontology [39] [40]. In this work, we recommend ontology merge to integrate ontologies.

To know more about ontologies, the reader is encouraged to visit the W3C standards [32]. The described features are essential while designing and implementing the proposed SDI framework.

III. Modular Design and Interoperability

The assistance system should be designed to be adaptive and flexible, such that it should be possible to combine different CPS with very varying capabilities without requiring extensive configuration from the worker. This flexible design makes it possible to scale the intelligence of the overall system by adding/removing CPS. The paper assumes that the central system contains a process description model, which describes the instructions for a process. The model remains unchanged irrespective of addition or removal of CPS modules. Adding new CPS modules to the central system makes the complete assistance system more aware of its environment and consequently more intelligent.

An assistance system, considered in this work, has hybrid architecture which consists of CPS modules and a central system where each CPS module collects and preprocesses data and feeds information to a central decision-making entity as shown in Figure 1. The central system collects information from all the modules attached to it and decides the next step of the process depending upon the process description model. Next step in the process is conveyed to a worker with the help of visualisation devices as shown in Figure 1. In contrast to a completely centralised or decentralised architecture, in a hybrid architecture, the burden of *making* sense from the raw-data is divided between the CPS modules and the central system: the modules need to preprocess raw data and make minor decisions before reporting it to the central system. The preprocessing step may include operations like analog to digital conversion, computing a parameter which is a function of data from more than one sensor (e.g., numberofParts from totalWeight and weightPerPart), calculating a moving average of a sensor reading, etc. This avoids any computing overhead on both the central system and CPS modules, and consequently makes them more intelligent and context-aware. This division is discussed in detail in Section IV.

A modular design enforces separation of concerns: the central system will only rely on the information *provided* by the modules. As per the traditional modular design, the internal state of the modules, i.e., the implementation details, would ideally be made completely opaque and inaccessible to the central system and other modules. In contrast, in this work, a framework for designing the modules using ontologies is proposed, which will allow the modules to access and use information from each other.

There are several challenges which need to be addressed in order to allow for such interoperability. The paper shows how these can be overcome by semantically annotating the information in each module using ontologies. As discussed in the previous section, an outright advantage of using ontologies is that they can give a unique name, i.e., URIs/IRIs, to *each* piece of information in the complete system,

thus, making it immediately accessible using a simple declarative querying language (SPARQL) as shown in Figure 2 [41]. Moreover, other advantages come naturally with using ontologies, viz. self-documentation, automatic reasoning using description logic for free.

Using ontologies as the tool of choice, the following two questions are considered.

(i) **How to design and semantically annotate a CPS module?** This question is answered in Section IV.

(ii) **How to develop such modules using ontologies?** This issue is discussed in Section VI and in Section VII.

Remark. Note that the decision-making algorithm in the central system should be designed in such a way that it does not need to be adapted to accommodate the underlying frequently changing CPS modules, i.e., the assistance system should be able to function without all modules being attached to the system and the modules should be plug-and-play. However, the problem of designing the algorithm is out of the scope of this work.

IV. Framework for designing a CPS

In this section, a framework for designing a CPS module and its ontology is proposed as shown in Figure 7. It starts with *what* the module designer wants to achieve by adding a particular CPS to the system, and then determines its boundary, or scope, with respect to the central system. Next, decisions about the *intelligence* of the system are made which, in turn, influence the hardware choices for the module. Finally, a bottom up ontology of a CPS is created and its integration with the central system ontology is described. Examples inspired from Figure 1 are discussed throughout the paper. The framework, and its implementation, are explained with the help of a use case of an inventory module which is shown in Figure 4.

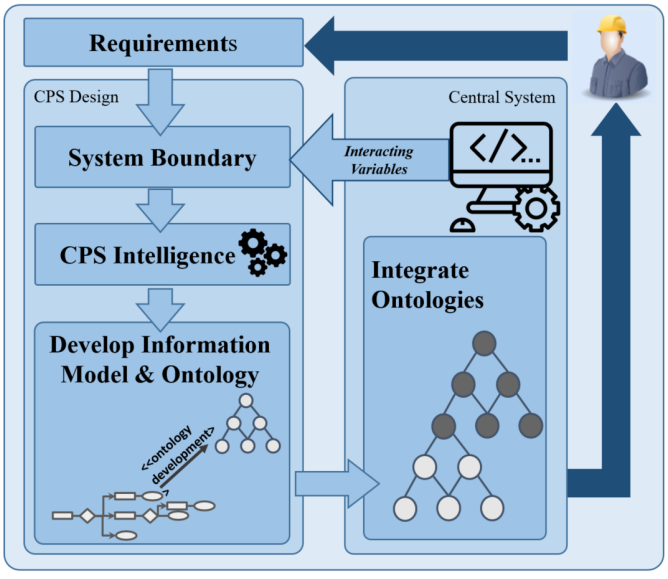


Figure 3. SDI framework for designing a CPS module.

A. Requirements

At the outset, it is important to understand *why* a CPS module is required. This decision determines the metric used for measuring the effectiveness of a module finally. This objective may range from general, e.g., "increasing the efficiency of a factory", to specific, e.g., "decreasing the number of errors for a particular assembly station".

For example, the requirement behind adding an inventory module can be to make the assistance system more aware of the environment in order to better understand the state of the process by the virtue of parts used in the process. This, in turn, improves the ability of an assistance system to help the worker. Keeping the requirements as specific as possible helps with the next step of the design.

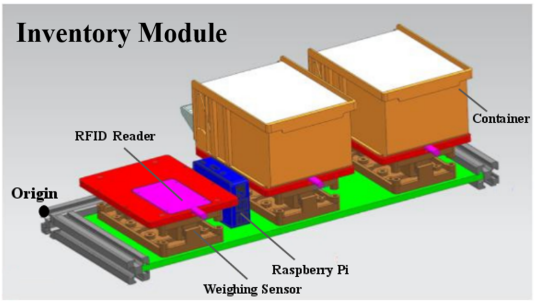


Figure 4. Schematic description of an inventory module

B. System Boundary

In the next step, the objective needs to be translated into a concrete piece of information that the central system needs from the CPS. An analogy can be drawn between the information which the central system needs and the idea of *minimal sufficient statistic*: the information should be *sufficient* for the central system to arrive at its objective. This information is the *interacting variable* between a CPS module and the central system.

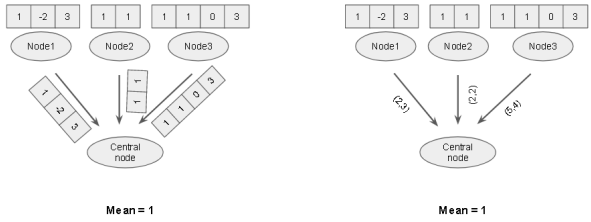


Figure 5. Two ways of calculating mean: in the first case complete raw data is provided to the central node to calculate mean whereas in the second case, only the sufficient statistic is provided.

In statistics, a statistic is sufficient with respect to a parameterized statistical model if no other statistic that can be derived from the same sample (e.g., raw sensor data) provides any additional information as to the value of the parameter. For example, consider the sufficient statistic to calculate mean of samples which are distributed across multiple nodes as shown in Figure 5. Each node only needs to report the sum of its samples and the number of samples to the central node doing the calculations. The central node then can calculate the total sum and the total number of samples and produce the mean without having the complete raw data (thereby saving computation and communication costs).

In terms of ontologies, the interacting variable needs to have the same URI/IRI in both the central system ontology as well as the ontology of the CPS module. The choice of sufficient static is driven by the data required by central system. In other words, the vocabulary, i.e., the terms defined by the central system, decide the

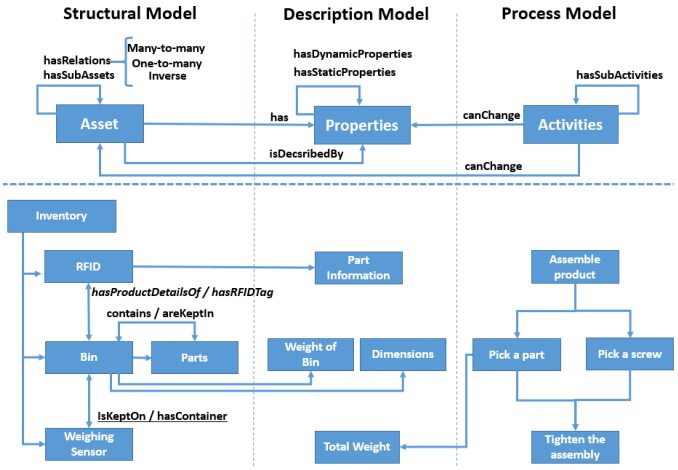


Figure 6. Information model contains structural, description and process models.

system boundary. This is ensured by defining the interacting variable in the upper ontology of an assistance system and the CPS module *importing* it.

For example, the central system may need the `total number of parts` for each part on the assembly station from an inventory module. This is the interacting variable for the CPS module.

C. CPS Intelligence

Once the system boundary is known, i.e., the interacting variable for a CPS module, it is necessary for the CPS to be *intelligent* enough to calculate this information from raw sensor readings. This *intelligence* is manifested in the accuracy/update frequency of sensors and the computational power afforded by the hardware (e.g., Raspberry Pi or Arduino) used to create the CPS module. Calculation of the value of the interacting variable effectively sets a lower bound on this system intelligence, i.e., a CPS should be able to process the data received through sensors to communicate the interacting variable whenever it is needed by the central system, e.g., calculating moving average of raw data every millisecond. The system intelligence can further be improved by using more sophisticated hardware and/or applying better algorithms while processing data, which improves the *quality* of the values calculated by the CPS module for the interacting variable.

Also, note that the CPS module should have the computational power to use ontologies during run-time. However, the restrictions placed by this requirement are mild because ontologies can be made light-weight during run-time [11].

D. Developing the Information Model & Ontology

After deciding on the hardware to use for a module, an *information model* which is an abstraction of the physical layer is created based on the structural and description models of the physical units present in a CPS module (as shown in Figure 6). The structural model defines physical assets present in a module: it lists all sensors, computational units, communication units and relations between them. The description model describes the properties of these assets. The process model is the process description that exists in the central system and is not changed on addition/removal of CPS modules. Structural and description models of the information model are used to explicitly define the hardware that was decided in the above steps. Figure 6 also shows the structural and description models of an inventory module and the process model contained by the central system.

The ontology of a CPS module is developed using the information model as a reference. In addition to the entities and relations defined in the information model, the ontology may also contain variables which are the result of *processing* the data gathered by sensors. Finally, the interacting variable(s), which were decided while determining the system boundary, are added to the ontology with appropriate relationships with other entities.

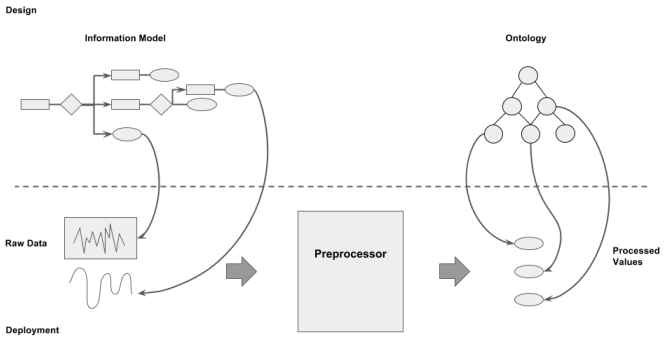


Figure 7. The information model (top-left), which is based on the physical setup of the system, is used to design the ontology (top-right) during the design phase. During implementation of the model, in the deployment phase, the sensors in the information model produce some raw data. This raw data is preprocessed by the CPS module (this is where system-intelligence comes into play) and is made ready for the ontology.

E. Ontology Integration

In the final step, ontology of the CPS module is merged with the central system ontology. The central system uses the interacting variable for making its own decisions, but also acts as a database for the complete assistance system during run-time. The modules, hence, can query the central system for not only the interacting variables of other modules, but also about the internal entities, which the central system does not explicitly use. The problem of how can the CPS modules be made aware of the various entities which can be accessed is addressed next.

As discussed before, the interacting variables are described in an upper ontology and a mid-level ontology contains descriptions of the entities of *all* modules. To help the ecosystem develop, a committee which consists of all shareholders (central system designers, deployment point managers, module developers, etc.) which oversees the addition to new modules to the ontology would be needed. The upper ontology is kept minimal and is only extended with new interacting variables, i.e., when a new potential CPS module is identified which can aid the intelligence of the central system. The other entities which can be provided by the new module, but which are not needed by the central system, are described in the mid-level ontology. The mid-level ontology acts as a repository of all relevant entities described in all CPS modules. This simplifies the search by engineers for variables provided by other modules. CPS modules `<<import>>` the upper ontology to get the URIs/IRIs of interacting variables and mid-level ontologies to get the URIs/IRIs of the entities of *all* modules.

Instead of having a mid-level ontology, it is possible to have only an upper ontology and ontologies of CPS modules. In such a setting, if one module needs to query for the variables of other CPS module, it then `<<import>>s` the ontology of that particular

module. However, this scheme of ontology development may result in reinvention of entities. Thus, a centralised W3C committee like setup [32] which consists of all stakeholders is favoured.

V. Ecosystem of ontologies

This section describes two possible ways of creating and maintaining ontologies for a complete assistance system. These ecosystems can be classified mainly into the two following ways:

A. Decentralised scheme of ontologies

In this section, a decentralized organizational scheme for the ontologies is described. As shown in Figure 8, upper ontology of assistance system is designed. To recap, upper ontologies of modules are created from their information models. CPS module ontology is described using its information model. These upper ontologies consists of definitions of entities and relationships between them.

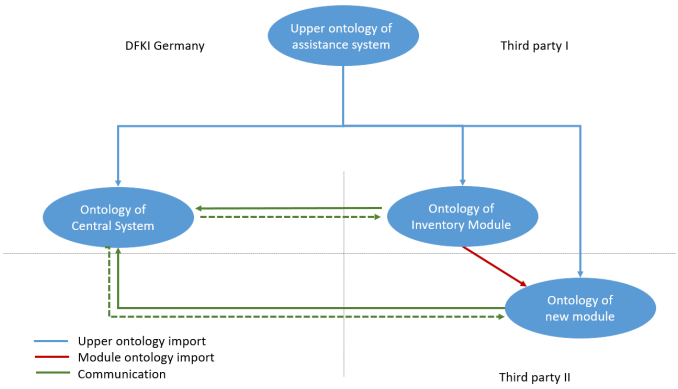


Figure 8. Schematic description of a decentralized ecosystem of ontologies.

So, the basic vocabulary described by upper ontology of assistance system is imported by the modules' ontologies. As information model maps all possible data, inventory module upper ontology contains definitions which are needed for the inventory module itself, but are not required by the central system ontology. An example of this can be position of container `(x, y, z)`. This allows for flexibility in implementation of inventory module. If another module requires data regarding container position, ontology of that particular module can import the inventory module ontology.

Pros & Cons. This design focuses on building a completely decentralized system. The central system's ontology only contain the minimal taxonomy of entities and properties which are necessary for the Central System to function, i.e., be able to use the information from the modules effectively. However, the individual modules are free to report any variable which they can measure and to report it to the central system. The central system will store that information even if it may not have explicit uses for the variables but can produce this knowledge if a different third party module requests for it through SPARQL queries.

However, such a setup has the disadvantage that independent teams may reinvent properties independently and since these properties will have unique IRI (e.g., `TeamA:hascoordinateX`, `TeamB:hasX` and `TeamC:hasPositionX`) but with the same semantic meaning. This would complicate interoperability across modules and for the same information from different inventory modules the a new module would have to query independently.

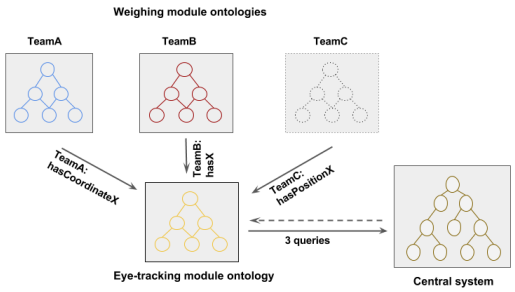


Figure 9. Example of possible reinvention of entities with same semantic meaning.

Figure 9 shows an example of such a situation where teams A, B and C independently define the variable for position of container as `hasCoordinateX`, `hasX` and `hasPositionX`. On the other hand, if the teams follow a particular nomenclature for defining variables would avoid reinventing similar variables which reduces the number of both imports and queries. Consolidation of the property names may also suffer due to the Not-Invented-Here syndrome [42].

A more subtle, and potentially more dangerous, side-effect of this design is compromised security of the data stored in the central system. In this design, the central system is completely unaware to the features which are being developed by independent modules. Hence, the central system needs to be excessively permissive when it comes to allowing arbitrary SPARQL queries by third party modules. A malicious module can very easily take advantage of vulnerability to obtain data on the central system.

B. Centralised scheme of ontologies

This section describes a centralized organizational scheme for the ontologies. As shown in Figure 10, upper ontology of assistance system is created which consists of the basic vocabulary for the complete system. Then a mid-level ontology is created. This mid-level ontology imports the upper ontology of assistance system. Further, the mid-level ontology describes the entities of all other modules. Depending on the engineers describing the mid-level ontology, all or some of the significant entities used by other modules are defined in the ontology.

The idea behind creating a mid-level is to create a repository of all relevant entities described in any CPS module. This simplifies the search by engineers for variables required by other modules. Mid-level ontology collects entities and their definitions described by upper ontologies of modules to facilitate exchange of data and this differentiates the approach from the previous approach. An assistance system upper ontology defines the minimal variables requires by modules to send data to the central system. This ontology is governed by the highest level committee and usually changes to it will be made when new modules are attached to the assistance system. On the other hand, modifications can be done easily in mid-level ontology which gives engineers the freedom to extend and access variables.

All modules' upper ontologies import the mid-level ontology. All modules need to import the mid-level ontology only once as it has all entities defined in the complete system.

During the design of the module, the interacting variable(s) were added in the upper ontology while the mid-level ontology was updated to include all entities which the module could provide, as agreed by all the stakeholders. For the purpose of exposition and to maintain complete generality, it is assumed in this section that the developer creating the module is a third party who intends to develop a newer version of the module from the specification.

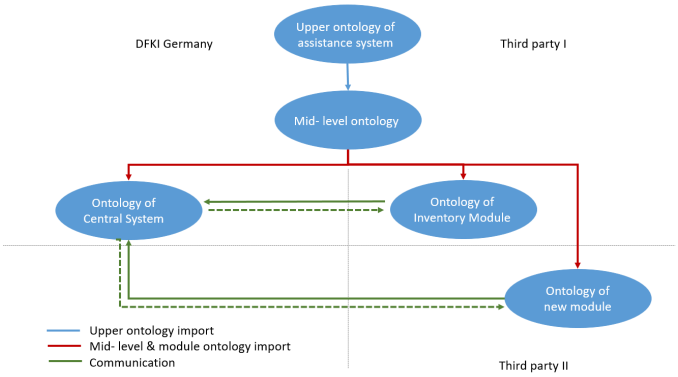


Figure 10. Schematic description of centralized ecosystem of ontologies .

Pros & Cons. In comparison with the more decentralized structure given in the previous section, the benefits and costs of this design are apparent. First of all, the mid-level ontology can be viewed as a *white-list* of entities and properties which can be read from the central system during execution through SPARQL queries. This prevents reinvention because a cursory check through the mid-level ontology will show that the properties already exist for the module being developed. Further, each module can individually extend the entities imported from mid-level ontology. These extensions are local and are not propagated to the mid-level ontology, thus, modules may again reinvent variables. These extensions can be made directly in the mid-level ontology to avoid reinvention on the next level. However, whether these extensions should be a part of the mid-level ontology is not discussed in detail in this work and can be seen as a future work.

Because of the white-list provided by the mid-level ontology, the central system can also put in place a system for authorizing certain (known) modules to have access to information which is not available to other modules. This allows security sensitive data to be *inaccessible* from potentially malicious or unknown modules. The exact authentication mechanism will depend on public key cryptography [43], which is out of scope of this work.

Pair-wise collaboration of teams is not encouraged in this setup. This can be a potential downside of the organizational scheme. This may increase the development time for a particular module if the properties it needs to import are not in the white-list already and the decision and procedure of whether to add these properties might take longer compared to the previous design scheme.

Based on the discussion is this section, centralized structure of ontolgies with mid-level onotlogies is selected for development of CPS module for the proof of the concept.

VI. Module Development

This section describes at a high level the development of a CPS module and the central system after the design for the module has been included into the upper and mid-level ontologies. During the design of the module, the interacting variable(s) were added in the upper ontology while the mid-level ontology was updated to include all entities which the module could provide, as agreed by all the stakeholders. For the purpose of exposition and to maintain complete generality, it is assumed in this section that the developer creating the module is a third party who intends to develop a newer version of the module from the specification.

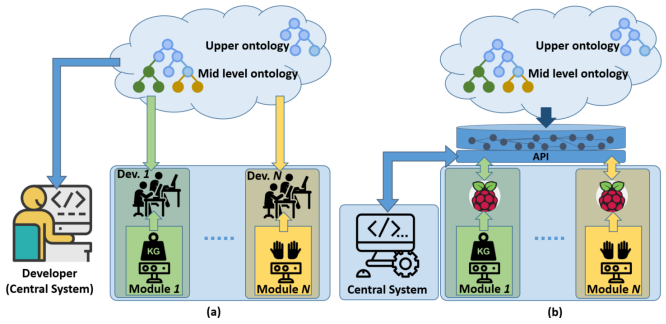


Figure 11. Ontology development of CPS modules.

In the next step towards development of the module, on the one hand, the developer (say, *Dev.* 1) studies the capabilities of the hardware available to her. Here, the developers can leverage the information model and ontology created during the design phase. On the other hand, the developer studies the upper (mid-level) ontology to determine what entities/values they should (could) provide to the central system. This part of the development process is illustrated in Figure 11(a). It should be noted that there is no need for communication or synchronisation between the developers of the different modules or between the developers and the central system developer. The developer `<<import>>s` the upper and mid-level ontologies and creates the module ontology with the remaining (local) ontological entities, and writes code which uses the central system's Application Programmable Interface (API) and SPARQL queries to update the central system database (as shown in Figure 11(b)).

Lastly, it is advised that Protégé should be used to create the module ontology as (i) it enhances interoperability by using OWL-S, and, (ii) it can automatically generate code using OWL API, which can ease the burden on the developer.

In this work, Protégé is used to create ontologies and the code generated is used to update the ontologies. Figure 12 shows an ontology created in Protégé and Figure 14 shows code generated by the API. In the next section, simulation of the central system and an inventory CPS module using Protégé is discussed.

VII. Implementation

This section describes implementation of a CPS module developed during this work. An inventory module as shown in Figure 1 is attached to the assistance system. The hardware implementation deals with reading data from sensors and RFID tags, processing the data and sending the required information to central system of assistance module. Sensor data and part information from RFID tags are used to provide information about the part in each container. In the presented scenario, Rapsberry Pi (RPi) attached to weighing module provides information about part name, total number of parts contained in each container and change in the number of parts for each container to a central system. RPi also sends a message to signal low inventory for parts if number of parts in a container drops below a threshold level.

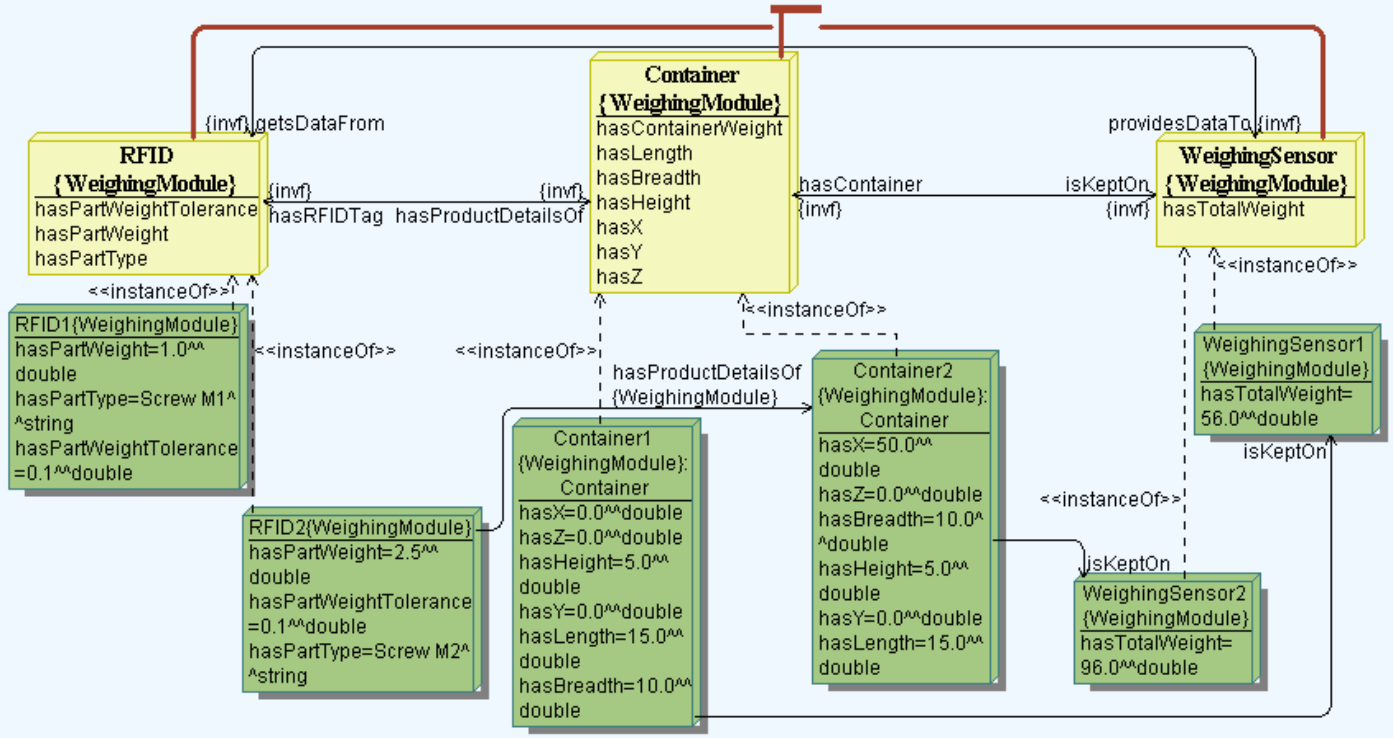


Figure 12. Example of classes and their relations as described in Protégé.

Figure 4 shows the setup of weighing module used in implementation. Weighing module has three weighing sensors with container kept on each sensor. Further, an RFID tag is attached to each bin. RFID tags contain data regarding parts, for example type of part, part name which can be read by RFID readers kept on weighing sensors as shown in the Figure 4. However, it is noteworthy that there is a scope of human error in this scenario as the part details are entered manually and while filling the container it should be ensured that container has the corresponding parts. This, indeed, can be one area of further improving the system and making it error proof.

This section discusses various hardware and communication choices available for implementation and makes recommendations based on the issues faced chronologically during implementation. First, different hardware options and procedures are discussed followed by communication between the central system and the inventory module and a few implementation recommendations are made.

A. Hardware Design

As discussed in Section I, an assistance system consists of a central system and CPS modules. In this implementation, weighing modules are the only kind of module attached to the central system.

Calibration of the sensors avoids any discrepancies in weights. Calibration follows a procedure wherein are given to calibrate the weighing sensor against dead-load. Since each sensor has a standard container and an RFID reader, the weights of these two entities are included in the dead-load of weighing sensor for the ease of calibration procedure. Including the weight of container and RFID reader in dead-load would lessen the complication in finding the number of parts as the weighing sensor will report only the weight of parts as opposed to the weight of the whole setup. However, weight of container is still a data node in our information model and ontologies in order to capture as much data as possible.

HEAD	0xF2
L	Length in bytes counting from the byte after the L-byte to the end including checksum byte and END
x	Command byte and literal values to the command
C	Checksum byte XOR function on all bytes preceding the checksum byte, not including HEAD byte.
END	0xF3

Figure 13. Example of definitions of command bytes.

This software implementation can be done for a micro-controllers, a Raspberry Pi (RPi) or a computer. RPi has more computational power than micro-controllers. It also has lower cost & lower power consumption than computers and is easy to use for programming. Hence, RPi was used for the implementation.

HEAD | L | x | C | END

Listing 1. An schematic layout of an encoded message to the weighing module. See Figure 13 for an explanation of each part of the command.

The inventory module uses communication protocol RS485 whereas the de-facto protocol for computer communications is `RS232`. There are multiple ways of converting `RS485` signals to `RS232`. An `RS485` shield, which sits on RPi, receives voltage corresponding to `RS485` and converts it RPi standard I/O voltages, is used in the implementation.The inventory module uses a LAN (*RJ45*) cable to power and to send/receive data to RPi.

A Python program is written to calibrate the module. Raw data from sensors are read in hexadecimal form and is converted to decimal for the ease of reading and understanding. The program sends byte-encoded messages to the sensor which responds by sending bytes back. The byte code message for different commands are provided

as the documentation for invnetory module. The documentation and code written for reading sensor data are provided on GitHub [44]. The module works on the principle that it gets a predefined encoded message from the user/RPi and depending on the value of message it returns the desired bytes. Listing 1 shows the general encoding of messages to/from sensors. Definitions regarding the command are provided thereafter in Figure 13.

Part information is read from RFID tags which are placed at the bottom of containers. RPi collects the data from sensors, part information from RFID tags and extract information regarding total number of parts and change in number of parts for each container. Python code also incorporates the detail of inventory threshold for each part. If inventory for a part goes below this threshold, a flag is raised to signal that refilling of parts is required.

B. Software Design

In the previous section, the development phase of ontologies was discussed. In this section, the simulation of an assistance system during run-time is discussed. The simulator in our implementation consists of two parts: a Python program for the central system and a Java program for an inventory module (see Figure 16) for the setup. The communication between the module and the central system uses ZeroMQ [45] and can be transparently done on a single machine or multiple machines. The details of the communication will be given in the next section.

Assistance system ontology is developed in Protégé, a free, open source ontology editor. The code generated via OWL API using Protégé (as shown in Figure 14) is used to simulate the behaviour of the CPS module. This implementation is written in Java. During execution, the ontology is *populated* by creating *individuals* locally on the module during execution. The central system may answer queries sent to it in SPARQL or may provide it using an alternate API. However, the use of unique URIs/IRIs to refer to entities in the ontologies is crucial to facilitate interoperability in all implementations.

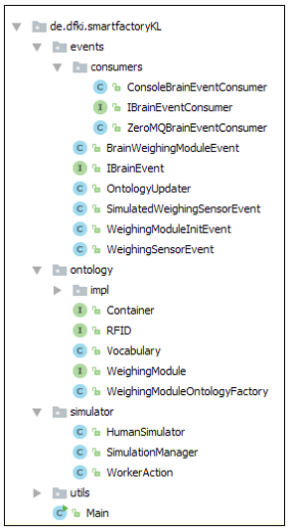


Figure 14. Classes generated by Protégé, based on OWL API. Central system discussed in the paper is referred to as `Brain`.

C. Communication Design

During execution, the system goes through three primary stages: (i) intialization, (ii) trigger, and (iii) update, which are shown in Figure 15, and are briefly discussed here:

- **Initialization.** When an assistance system is started, the central system sends an *init()* request to all CPS modules attached to it. This request contains the URI/IRI of the central system. This URI/IRI is address with which all modules identify the central system through the lifetime of the process. In case of hardware malfunction, system restart, or when a new module is attached to the system, the initialization step is executed again.

- **Trigger.** Triggers can be either timer-driven or event-based. Event-based triggers are reported by CPS modules to the central system whereas timer-driven triggers are generated by the central system. Event-based triggers can be events that change the present state of a system to another (valid) state of the system [9]. In case an event occurrence renders no valid state of the system, triggers are not generated. *Trigger()* request is either sent from modules to the central system, as shown in Figure 15, or may be generated internally by the central system clock.

- **Update.** Communication between the central system and CPS modules is pull-based. Upon a trigger, the central system sends a *getUpdate()* request to all modules. Modules send the complete, or a part of, ontologies with the new data values to the central system which, in turn, update its own ontology.

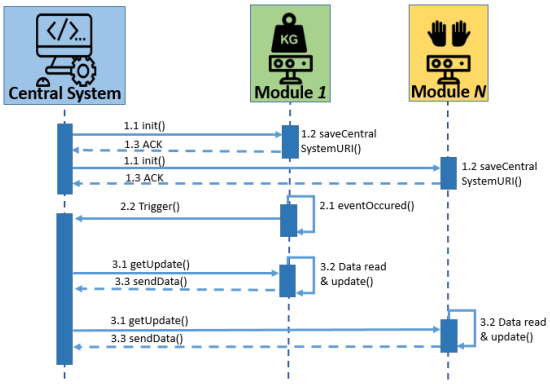


Figure 15. Communication between the central system and CPS modules.

An example implementation is available for download on GitHub [44]. The implementation therein simulates an inventory module (using code generated from Protégé), a central system, and then simulates human actions, updates the ontology on the inventory module using the OWL API, and shows the communication between the module and the central system.

D. Additional Recommendations

The inventory module must be developed keeping in mind the failures and errors that might happen while deploying the system. Thus, the system must have certain properties which make it error proof. Intertwined with the desirable properties of the system, some practical recommendations regarding implementation are also made in

this section. The inspiration for these recommendations comes from design of concurrent systems [46].

Safety property asserts that *nothing bad* happens. The foremost requirement to implement this property is the system should not be in an invalid state at any point in time. For CPS, it means that the module should never report values which are not computed from its sensor values. A discrepancy may result from the scenario that the module reports a value to the central system after it has probed one sensor but before probing another sensors, if the report contains values which were computed using both the sensor's readings (one of the sensors may have out-dated value). An invalid state can also be a deadlocked state where there are no outgoing transitions, such as an error state. Semaphores, mutex and locks should be judiciously used during development to avoid such scenarios. As a side-note, handling missing values (which is a subset of error states) gracefully is a potential future extension of the work.

Liveness property asserts that the system will perform its intended use *eventually*. In other words, liveness means that the system will continue to make progress. This implies ensuring that the semaphores and mutexes will be unblocked and the module will, eventually, send data to the central system. Though several race conditions can be avoided simply by using atomic operations exclusively, it is possible to end up in a live-lock. Say the module has a *hard* parameter which controls after how long a sensor's data is considered *stale*. Then say the module reads data from one sensor, and then while it is reading data from another sensor, the data from the first sensor becomes *stale*. So, the module will go back to re-reading data from the first sensor and in the meanwhile data from the second sensor becomes *stale*. This could bind the module into a sensor reading infinite loop. During implementation, such situations should be carefully thought about and the liveness of the module should be tested/verified under the most extreme of conditions.

Encapsulation is another way of making system more reliable. Encapsulation is restricting direct access of software components so that they cannot interfere with other subsystems or privileged-level software. It keeps faults from propagating which increases the reliability of the overall system.

Finally, in case everything fails, a **watchdog timer** (or a Heartbeat) can be used to detect the catastrophic failures and recover from it. The timer is regularly reset by computer during normal operation, but it timeouts if there is a hardware or software error. The timeout signal initiates a corrective measure by placing the system in a safe state or restoring normal operation. One of the ways this can be accomplished is by using a Hypervisor [47] which can simply restart the entire module in case the timer timeouts.

These are some necessary properties that the system must have, but not sufficient to ensure that it functions properly. In the end, the deployment and user feedback would be the final test of the module.

VIII. Conclusion

This work is focused on designing a human-centric assistance system used in production which can dynamically adapt to the needs of the workers and tasks using Semantic Web technologies. Assistance systems are considered as consisting of a central system and one or many CPS modules. An SDI framework is proposed to design CPS modules which makes the data of the complete system globally accessible by the virtue of HTTP URIs/IRIs. The SDI framework explained the steps used to decide the boundary between the central system and CPS modules, the performance requirements of hardware, describing modules with the help of information models and finally developing and merging ontologies. It also explains the ecosystem of ontologies consisting of upper, mid-level and module ontologies. Hardware and software implementation of an inventory module is completed. For the inventory module, the framework is implemented in Protégé using OWL-S. OWL API is used to simulate CPS behaviour and data exchange is demonstrated. Communication between a CPS module and the central system is also described. However, the proposed framework can be used to design CPS in general: the discussion in the paper was limited to designing a CPS for an assistance system for ease of both exposition and demonstration.

The work assumes that all vendors and third party development use SPARQL as the query language. Calbimonte et al. have discussed how such a problem of multi-vendor multi-querying language can be resolved [48]. It can be incorporated in the SDI framework to make it more robust. Knowledge mapped in ontologies may evolve over time due to modifications in conceptualisation and adaptation to incoming changes. Thus, in future, it is important to establish protocols for versioning of data on Semantic Web as well as understanding the missing data [49]. Another non-trivial task towards adoption of ontologies in real life is setting up committees which oversee the creation and maintenance of upper and mid-level ontologies [50].

The framework results in a repository of data from all modules of the system and interoperability between these modules, thus, laying the foundation of plug-and-play production systems. The next important step in the development of assistance systems is to develop a plug-and-play methodology for CPS modules, as alluded to in Section III.

Another important step to make the system deployable is to create global standards: either by defining design and communication standards specific to assistance systems, or by investigating the suitability of existing standards, e.g., RAMI 4.0 [51].

References

[1] A. Singh, F. Quint, P. Bertram, and M. Ruskowski, "Towards modular and adaptive assistance systems for manual assembly: A semantic description and interoperability framework," in *The Twelfth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2018)*, 2018.

[2] M. M. Tseng and S. J. Hu, "Mass customization," in *CIRP encyclopedia of production engineering*. Springer, 2014, pp. 836–843.

[3] F. Salvador and C. Forza, "Configuring products to address the customization-responsiveness squeeze: A survey of management issues and opportunities," *International journal of production economics*, vol. 91, no. 3, pp. 273–291, 2004.

[4] Y. Koren and M. Shpitalni, "Design of reconfigurable manufacturing systems," *Journal of manufacturing systems*, vol. 29, no. 4, pp. 130–141, 2010.

[5] D. Romero, O. Noran, J. Stahre, P. Bernus, and Å. Fast-Berglund, "Towards a human-centred reference architecture for next generation balanced automation systems: human-automation symbiosis," in *IFIP International Conference on Advances in Production Management Systems*. Springer, 2015, pp. 556–566.

[6] S. Tzafestas, "Concerning human-automation symbiosis in the society and the nature," *Intl. J. of Factory Automation, Robotics and Soft Computing*, vol. 1, no. 3, pp. 6–24, 2006.

[7] P. A. Hancock, R. J. Jagacinski, R. Parasuraman, C. D. Wickens, G. F. Wilson, and D. B. Kaber, "Human-automation interaction research: past, present, and future," *ergonomics in design*, vol. 21, no. 2, pp. 9–14, 2013.

[8] V. Villani, L. Sabattini, J. N. Czerniak, A. Mertens, B. Vogel-Heuser, and C. Fantuzzi, "Towards modern inclusive factories: A methodology for the development of smart adaptive human-machine interfaces," *22nd IEEE International Conference on Emerging Technologies and Factory Automation*, 2017.

[9] F. Quint, F. Loch, M. Orfgen, and D. Zuehlke, "A system architecture for assistance in manual tasks." in *Intelligent Environments (Workshops)*, 2016, pp. 43–52.

[10] E. Tantik and R. Anderl, "Integrated data model and structure for the asset administration shell in industrie 4.0," *Procedia CIRP*, vol. 60, pp. 86–91, 2017.

[11] I. Grangel-González, L. Halilaj, G. Coskun, S. Auer, D. Collarana, and M. Hoffmeister, "Towards a semantic administrative shell for industry 4.0 components," in *Semantic Computing (ICSC), 2016 IEEE Tenth International Conference on*. IEEE, 2016, pp. 230–237.

[12] J. Gezer, Volkan Um and M. Ruskowski, "An extensible edge computing architecture: Definition, requirements and enablers," in *The Eleventh International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2017)*, 2017.

[13] D. Zuehlke, "Smartfactory–from vision to reality in factory technologies," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 14 101–14 108, 2008.

[14] M. Gaham, B. Bouzouia, and N. Achour, "Human-in-the-loop cyber-physical production systems control (hilcp 2 sc): A multi-objective interactive framework proposal," in *Service orientation in holonic and multi-agent manufacturing*. Springer, 2015, pp. 315–325.

[15] E. A. Lee, "Cyber physical systems: Design challenges," in *11th IEEE Symposium on Object Oriented Real-Time Distributed Computing (ISORC)*. IEEE, 2008, pp. 363–369.

[16] N. Jazdi, "Cyber physical systems in the context of industry 4.0," in *Automation, Quality and Testing, Robotics, 2014 IEEE International Conference on*. IEEE, 2014, pp. 1–4.

[17] P. Leitão, "Agent-based distributed manufacturing control: A state-of-the-art survey," *Engineering Applications of Artificial Intelligence*, vol. 22, no. 7, pp. 979–991, 2009.

[18] J. Nelles, S. Kuz, A. Mertens, and C. M. Schlick, "Human-centered design of assistance systems for production planning and control: The role of the human in industry 4.0," in *Industrial Technology (ICIT), 2016 IEEE International Conference on*. IEEE, 2016, pp. 2099–2104.

[19] D. Gorecky, S. F. Worgan, and G. Meixner, "Cognito: a cognitive assistance and training system for manual tasks in industry." in *ECCE*, 2011, pp. 53–56.

[20] C.-B. Zamfirescu, B.-C. Pirvu, D. Gorecky, and H. Chakravarthy, "Human-centred assembly: a case study for an anthropocentric cyber-physical system," *Procedia Technology*, vol. 15, pp. 90–98, 2014.

[21] B.-C. Pirvu, C.-B. Zamfirescu, and D. Gorecky, "Engineering insights from an anthropocentric cyber-physical system: A case study for an assembly station," *Mechatronics*, vol. 34, pp. 147–159, 2016.

[22] D. Kolberg, C. Berger, B.-C. Pirvu, M. Franke, and J. Michniewicz, "Cyprof–insights from a framework for designing cyber-physical systems in production environments," *Procedia CIRP*, 2016.

[23] T. Arai, Y. Aiyama, M. Sugi, and J. Ota, "Holonic assembly system with plug and produce," *Computers in Industry*, vol. 46, no. 3, pp. 289–299, 2001.

[24] W. Wahlster, "Semantic technologies for mass customization," in *Towards the Internet of Services: The THESEUS Research Program*. Springer, 2014, pp. 3–13.

[25] M. Graube, J. Pfeffer, J. Ziegler, and L. Urbas, "Linked data as integrating technology for industrial data," *International Journal of Distributed Systems and Technologies (IJDST)*, vol. 3, no. 3, pp. 40–52, 2012.

[26] S. K. Semy, M. K. Pulvermacher, and L. J. Obrst. (2004) Toward the use of an upper ontology for us government and us military domains: An evaluation. Retrieved on 2018-09-20.

[27] C.-H. Cheng, T. Guelfirat, C. Messinger, J. O. Schmitt, M. Schnelte, and P. Weber, "Semantic degrees for industrie 4.0 engineering: Deciding on the degree of semantic formalization to select appropriate technologies," in *Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering*. ACM, 2015, pp. 1010–1013.

[28] M. Obitko and V. Jirkovskỳ, "Big data semantics in industry 4.0," in *International conference on industrial applications of holonic and multi-agent systems*. Springer, 2015, pp. 217–229.

[29] C. Bizer, T. Heath, and T. Berners-Lee, "Linked data: The story so far," in *Semantic services, interoperability and web applications: emerging concepts*. IGI Global, 2011, pp. 205–227.

[30] A. Schultz, A. Matteini, R. Isele, P. N. Mendes, C. Bizer, and C. Becker, "Ldif-a framework for large-scale linked data integration," in *21st International World Wide Web Conference (WWW 2012), Developers Track, Lyon, France*, 2012.

[31] V. Gezer and S. Bergweiler, "Cloud-based infrastructure for workflow and service engineering using semantic web technologies," *International Journal on Advances on Internet Technology*, pp. 36–45, 2017.

[32] S. Bechhofer, "Owl: Web ontology language," in *Encyclopedia of database systems*. Springer, 2009, pp. 2008–2009.

[33] E. Negri, L. Fumagalli, M. Garetti, and L. Tanca, "Requirements and languages for the semantic representation of manufacturing systems," *Computers in Industry*, vol. 81, pp. 55–66, 2016.

[34] N. Guarino, *Formal ontology in information systems: Proceedings of the first international conference (FOIS'98), June 6-8, Trento, Italy*. IOS press, 1998, vol. 46.

[35] I. Niles and A. Pease, "Origins of the ieee standard upper ontology," in *Working notes of the IJCAI-2001 workshop on the IEEE standard upper ontology*. Citeseer, 2001, pp. 37–42.

[36] J. F. Sowa *et al.* Building, sharing, and merging ontologies. Retrieved on 2018-09-20. [Online]. Available: http://www.jfsowa.com/ontology/ontoshar.htm

[37] R. Hoehndorf, "What is an upper level ontology?" *Ontogenesis*, 2010.

[38] E. Beisswanger, S. Schulz, H. Stenzhorn, and U. Hahn, "Biotop: An upper domain ontology for the life sciences," *Applied Ontology*, vol. 3, no. 4, pp. 205–212, 2008.

[39] J. F. Sowa *et al.*, *Knowledge representation: logical, philosophical, and computational foundations*. Brooks/Cole Pacific Grove, CA, 2000, vol. 13.

[40] H. S. Pinto, A. Gómez-Pérez, and J. P. Martins, "Some issues on ontology integration." IJCAI and the Scandinavian AI Societies. CEUR Workshop Proceedings, 1999.

[41] E. Prud *et al.* Sparql query language for rdf. Retrieved on 2018-09-20.

[42] R. Katz and T. J. Allen, "Investigating the not invented here (nih) syndrome: A look at the performance, tenure, and communication patterns of 50 r & d project groups," *R&d Management*, vol. 12, no. 1, pp. 7–20, 1982.

[43] R. E. Smith, *Authentication: from passwords to public keys*. Addison-Wesley Longman Publishing Co., Inc., 2001.

[44] A. Singh. Example implementation of the SDI framework. Retrieved on 2018-11-16. [Online]. Available: https://github.com/AmitaChauhan/SDI-Framework

[45] P. Hintjens, *ZeroMQ: messaging for many applications*. " O'Reilly Media, Inc.", 2013.

[46] G. R. Andrews, *Concurrent programming: principles and practice*. Benjamin/Cummings Publishing Company San Francisco, 1991.

[47] M. Masmano, I. Ripoll, A. Crespo, and J. Metge, "Xtratum: a hypervisor for safety critical embedded systems," in *11th Real-Time Linux Workshop*. Citeseer, 2009, pp. 263–272.

[48] J.-P. Calbimonte, H. Jeung, O. Corcho, and K. Aberer, "Enabling query technologies for the semantic sensor web," *International Journal On Semantic Web and Information Systems (IJSWIS)*, vol. 8, no. 1, pp. 43–63, 2012.

[49] M. C. Klein and D. Fensel, "Ontology versioning on the semantic web." in *SWWS*, 2001, pp. 75–91.

[50] I. Jacobs. World wide web consortium process document. Retrieved on 2018-09-20. [Online]. Available: https://www.w3.org/2018/Process-20180201/

[51] M. Weyrich and C. Ebert, "Reference architectures for the Internet of things," *IEEE Software*, vol. 33, no. 1, pp. 112–116, 2016.

Achieving Higher-level Support for Knowledge-intensive Processes in Various Domains by Applying Data Analytics

Gregor Grambow
Computer Science Dept.
Aalen University
Aalen, Germany
e-mail: gregor.grambow@hs-aalen.de

Abstract — **In many domains like new product development, scientific projects, or complex business cases, knowledge-intensive activities and processes have gained high importance. Such projects are often problematic and may suffer from various threats to successful and timely project completion. This is often caused by the involved knowledge-intensive processes because of their high dynamicity, complexity, and complex human involvement. In this paper, we describe an abstract framework capable of managing and supporting such projects holistically. This is achieved by applying various kinds of data analytics on the different data sets being part of the projects. Thus, processes can be implemented and supported technically utilizing the results and combinations of the data analytics. We furthermore illustrate the applicability of the abstract framework by describing two concrete implementations of this framework in two different domains.**

Keywords-data analytics; knowledge-intensive processes; process implementation; knowledge management

I. INTRODUCTION

This paper is an extension of the article "Utilizing Data Analytics to Support Process Implementation in Knowledge-intensive Domains" [1]. It adds a comprehensive evaluation of the approach describing two concrete technical applications of the envisioned framework in detail as well as an extended discussion of related work and extended scenarios, figures and explanations. In the last decades, the number and importance of knowledge-intensive activities has rapidly increased in projects in various domains [2][3]. Recent undertakings involving the inference of knowledge utilizing data science and machine learning approaches also require the involvement of humans interpreting and utilizing the data form such tools. Generally, knowledge-intensive activities imply a certain degree of uncertainty and complexity and rely on various sets of data, information, and knowledge. Furthermore, they mostly depend on tacit knowledge of the humans processing them. Hence, such activities constitute a huge challenge for projects in knowledge-intensive domains, as they are mostly difficult to plan, track and control. According to literature [4][5], knowledge-intensive processes are characterized as follows:

- They are a composition of prospective activities whose execution contributes to achieving a certain goal.
- They rely on knowledge workers performing interconnected knowledge-intensive activities.
- They are knowledge-, information-, and data-centric.
- They require substantial flexibility, at design- and run-time.

Typical examples for the applications of such activities and processes are business processes in large companies [2], scientific projects [6], and projects developing new products [7]. In each of these cases, responsibles struggle and often fail to implement repeatable processes to reach their specific goals.

In recent times, there has been much research on data storage and processing technologies, machine learning techniques and knowledge management. The latter of these has focused on supporting whole projects by storing and disseminating project knowledge. However, projects still lack a holistic view on their contained knowledge, information and data sets. There exist progressive approaches for storing data and drawing conclusions from it with statistical methods or neural networks. There also exist tools and methods for organizing the processes and activities of the projects. Nevertheless, in most cases, these approaches stay unconnected. Processes are planned, people execute complex tasks with various tools, and sometimes record their knowledge about procedures. However, the links between these building blocks stay obscured far too often.

In this paper, we propose a framework that builds upon existing technologies to execute data analyses and exploit the information from various data sets, tools, and activities of a project to bring different project areas closer together. Thus, the creation, implementation, and enactment of complex processes for projects in knowledge-intensive domains can be supported.

The remainder of this paper is organized as follows: Section II provides background information including an illustrating scenario. Section III distils this information into a concise problem statement. Section IV presents an abstract framework as solution while Section V provides concrete information on the modules of this framework. This is followed by an evaluation in Section VI, related work in Section VII, and the conclusion.

II. BACKGROUND

In the introduction, we use the three terms data, information and knowledge. All three play an important role in knowledge-intensive projects and have been the focus of research. Recent topics include research on knowledge management and current data science approaches. Utilizing

definitions from literature [8], we now delineate these terms in a simplified fashion:

- Data: Unrefined factual information.
- Information: Usable information created by organizing, processing, or analyzing data.
- Knowledge: Information of higher order derived by humans from information.

This taxonomy implies that information can be inferred from data manually or in a (semi-)automated fashion while knowledge can only be created by involving the human mind. Given this, knowledge management and data science are two fields that are complementary. Data science can create complex information out of raw data while knowledge management helps the humans to better organize and utilize the knowledge inferred from that information.

Processes in knowledge-intensive domains have special properties compared to others like simple production processes [9]. They are mostly complex, hard to automate, repeatable, can be more or less structured and predictable and require lots of creativity. As they are often repeatable, they can profit from process technology enabling automated and repeatable enactment [10].

In the introduction, we mentioned three examples for knowledge-intensive processes: scientific projects, business processes in large companies and new product development. We will now go into detail about the properties of these.

In scientific projects, researchers typically carry out experiments generating data from which they draw knowledge. The amount of processed data in such projects is rapidly growing. To aid these efforts, numerous technologies have been proposed, on the one hand for storage and distributed access to large data sets. On the other hand, many frameworks exist supporting the analysis of such data with approaches like statistical analyses or neuronal networks [11]. There also exist approaches for scientific workflows enabling the structuring of consecutive activities related to processing the data sets [12]. However, the focus of all these approaches is primarily the processing of the scientific data. A holistic view on the entire projects connecting these core activities with all other aspects of the projects is not prevalent. In addition, the direct connection from data science to knowledge management remains challenging.

Business processes in large companies are another example of knowledge-intensive processes. Such processes are often planned on an abstract level and the implementation on the operational level remains difficult due to numerous special properties of the context of the respective situations. Consider a scenario where companies work together in complex supply chains to co-create complex products like in the automotive industry. Such companies have to share different kinds of information. However, this process is rather complicated as the supply chains are often huge with hundreds of participants. A data request from the company at the end of the chain can result in thousands of recursive requests through the chain [13]. For each request, it must be separately determined, which are the right data sets that are needed and can be shared.

A third example are projects developing new products. As example, we focus on software projects because software projects are essentially knowledge-intensive projects [7]. For these, various tools exist from development environments to tools analyzing the state of the source code. In addition to this, usually a specified process is also in place. However, the operational execution relies heavily on individuals that have to analyze various reports and data sources manually to determine the correct course of action in order to create high quality software. This implies frequent process deviations or even the complete separation of the abstract planned process from its operational execution. Furthermore, due to the large amount of available data sets (e.g., specifications, bug reports, static analysis reports) things may be forgotten and incorrect decisions made.

We will now illustrate different problems occurring when trying to implement a software development process on the operational level. Therefore, we will utilize an agile software development process: the OpenUP. The process comprises the four phases Inception, Elaboration, Construction, and Transition as illustrated in Figure 1.

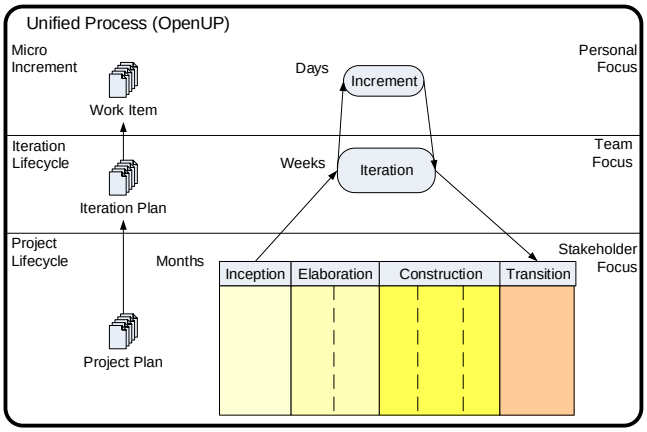


Figure 1. Software Development Process.

These phases cover the entire project lifecycle and are executed with a stakeholder focus. Each of the phases, in turn, may comprise an arbitrary number of iterations. In the latter, the focus lies on the team managing the scope of the iteration with an iteration plan. Each iteration contains different concrete workflows to support activities like requirements management or software development. Each participating person processes the concrete activities of these workflows working on one or more work items. The project lifecycle is managed in the granularity of months, the iterations are more fine grained. Finally, the processing of the work items is done on a daily basis. However, besides various concrete workflows and activities there are also various artifacts, tools, roles, and persons involved. We will now provide details on the OpenUP, its implementation, and issues regarding to it on the operational level as depicted in Figure 2. The figure shows the workflows of the iterations of the four phases. Each of the activities within these represents a sub-workflow containing more fine-grained activities. Every iteration has a sub-workflow for managing the iteration containing activities for planning, managing and assessing the iteration.

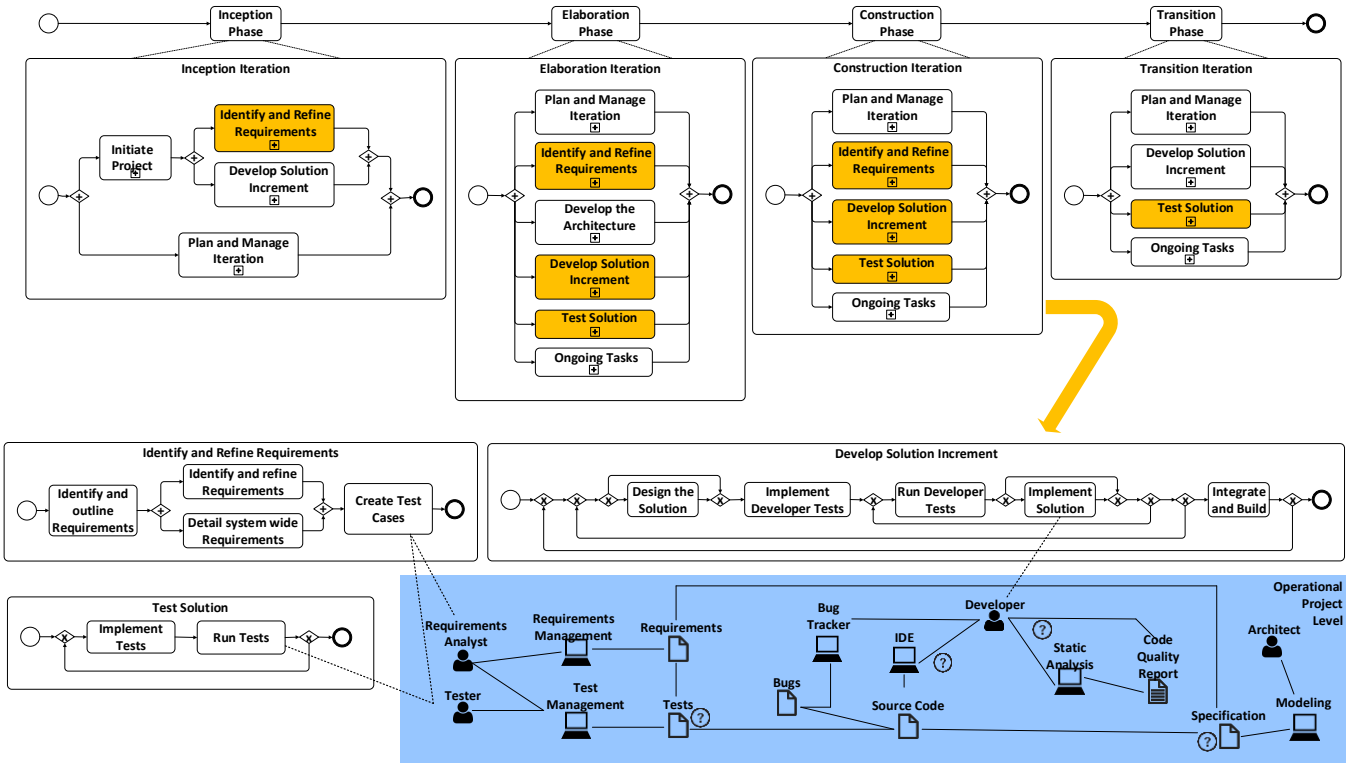


Figure 2. Scenario.

The iterations for all but the first phase also have a sub-workflow called ‘Ongoing Tasks’ for managing changes, e.g., in case the scope or the requirements change. The inception phase primarily deals with setting up the project and the requirements but also allows for creating the first increment of the envisioned solution. The elaboration phases’ iterations add activities for creating the architecture of the software and already the first testing while refining the requirements and continuing to create the solution. The construction phase, in turn, is the main development phase. In the transition phase, the development and testing is finalized to transfer the software to the client. In this phase no more requirement changes shall take place.

As examples, we also show three concrete workflows: ‘Identify and refine Requirements’ deals with the initial creation and refinement of the requirements. In addition, system wide technical requirements are detailed and the relating test cases must be created. ‘Develop Solution Increment’ covers operational software development. It contains concrete activities like ‘Implement Solution’ where the developer shall technically implement the solution (i.e., a specific feature of a software), which was designed before. ‘Test Solution’ contains a loop of creating and running tests for the created software. However, such activities are still rather abstract and have no connection to tasks the human performs to complete the activities. These tasks are performed with concrete tools, artifacts, and other humans depicted in the blue box of Figure 2. The figure indicates various issues: (1) Tasks performed with different tools like IDEs and static analysis tools are fine-grained and dynamic. Therefore, the workflow cannot prescribe the exact tasks to be performed [14]. Furthermore, the mapping of the numerous real world events to the workflow activities is challenging. (2) In various situations, the developer must derive decisions based on data contained in reports from different tools. One example are specific changes to improve the source code to be applied on account of static analysis reports. Goal conflicts (e.g., high performance vs. good maintainability) may arise resulting in wrong decisions. (3) In various cases, different artifacts (e.g., source code and technical specifications) that relate to each other may be processed simultaneously by different persons, which may result in inconsistencies [15]. (4) Unexpected situations may lead to exceptions and unanticipated process deviations. (5) The whole process relies on knowledge. Much of this knowledge is tacit and is not captured to be reused by other persons [16]. This often leads to problems.

III. PROBLEM STATEMENT

In Section II, we have defined different kinds of relevant information and shown examples from different domains in which a lacking combination of such information leads to problems with operational process implementation.

In scientific projects, data analysis tools aid humans in discovering information in data. However, the projects mostly neither have support for creating, retaining, and managing knowledge derived from that information, nor do they have process support beyond the data analysis tasks [16][17]. Complex business processes in large companies often suffer from lacking process support because of the high number of specific contextual properties of the respective situations. In new product development, problems often arise

due to the inability to establish and control a repeatable process on the operational level. This is caused by the high number of dynamic events, decisions, deviations, and goal conflicts occurring on the operational level.

In summary, it can be stated that process implementation in knowledge-intensive projects is problematic due to the high complexity of the activities and relating data. Processes can be abstractly specified but not exactly prescribed on the operational level. Thus, it remains difficult to track and control the course of such projects, which often leads to exceeded budgets and schedules and even failed projects.

In particular, the following points need to be addressed:

- Seamless integration of data analysis approaches into the projects. Data producers, data storage and data consumers should be integrated globally in projects.
- Integration of (semi-)automated data analytics with knowledge management.
- Integration of data analytics with automated process support to automatically adapt the process to changing situations.

IV. FRAMEWORK

In this paper, we tackle these challenges by proposing an approach uniting different kinds of data analytics and their connection to other project areas like knowledge management and process management. That way we achieve a higher degree of automation supporting humans in their knowledge-intensive tasks and facilities to achieve holistic and operational implementation of the projects process.

Because of the high number of different data sets and types and their impact on activities, we think it is not possible to specify a concrete framework suitable for all possible use cases of knowledge-intensive projects of various domains. We rather propose an extensible abstract framework and suggest different modules and their connections based on the different identified data and information types in such projects. The idea of this abstract framework builds on our previous research where we created and implemented concrete frameworks for specific use cases. Hence, we use our experience to extract general properties from these frameworks to achieve a broader applicability.

The basic idea of such a framework is a set of specific modules capable of analyzing different data sets and utilizing this for supporting knowledge-intensive projects in various ways. Each of these modules acts as a wrapper for a specific technology. The framework, in turn, provides the following basic features and infrastructure to foster the collaboration of the modules.

A simple communication mechanism. The framework infrastructure allows each module to communicate with the others to be able to receive their results and provide its results to the others.

Tailoring. The organization in independent modules facilitates the dynamic extension of the framework by adding or removing modules. That way the framework can be tailored to various use cases avoiding technical overhead.

Support for various human activities. The framework shall support humans with as much automation as possible. Activities that need no human intervention shall be executed in the background providing the results in an appropriate way to the humans. In contrast to this, activities that require human involvement shall be supported by the framework. All necessary information shall be presented to the humans helping them to not forget important details of their tasks.

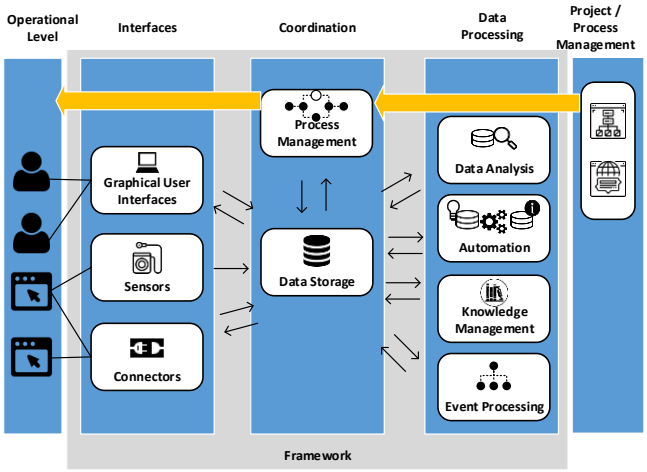


Figure 3. Abstract Framework.

Holistic view on the project. Various technologies for different areas of a project are seamlessly integrated. That way, these areas, like process management, data analysis, or knowledge management can profit from each other.

Process implementation. The framework shall be capable of implementing the process spanning from the abstract planning to the operational execution.

In the following, the structure of the framework and the interplay of its components are described. A more concrete description of each component follows in Section V. The framework is illustrated by Figure 3. We divide the latter into three categories of modules: Interfaces, Coordination, and Data Processing. The coordination category contains the modules responsible for the coordination of data and activities in the framework: The data storage module is the basis for the communication of the other modules by storing and distributing the messages between the other components. The process management module is in charge of implementing and enacting the process. Thus, it contains the technical representation of the processes specified at the project / process management level, which is outside the framework. Utilizing the other modules, these processes can be enacted directly on the operational level where concrete persons interact with concrete tools. This improves repeatability and traceability of the enacted process.

The interface category is comprised of three modules: Graphical user interfaces enable users to communicate with the framework directly, e.g., for controlling the process flow or storing and utilizing knowledge contained in the framework. The sensor module provides an infrastructure for receiving events from sensors that can be integrated into external software tools or from sensors from production machines. That way, the framework has access to real-time event data from its environment. The connector module

provides the technical interface to communicate with APIs of external tools to exchange data with the environment.

The data processing category provides modules relating to data processing and analytics, which enables the framework to automatically issue various actions and influence the process to fit to changing situations: The event processing module aggregates event information. This can be used, for example, for determining actions conducted in the real world. Therefore, sensor data from the sensor module can be utilized. By aggregating and combining atomic events, new events of higher semantic value can be generated. The data analysis module integrates facilities for statistical data analytics and machine learning. This can be utilized to infer information from raw data, e.g., coming from production machines or samples in scientific projects. The knowledge management component aids humans in managing knowledge derived from it. Both technologies can interact to support scientific workflows. E.g., incoming data can be analyzed and classified and the framework can propose an activity to a human for reviewing the data and record knowledge in a knowledge base.

Finally, the automation component enhances the automation capabilities of the framework. Therefore, various technologies are possible. As a starting point, we propose the following: rules engines for simple specification and execution of rules applying for the data or the project as a whole. One example use case is the automated processing of reports from external tools. Multiple reports can be processed creating a unified report by a rules-based transformation that, in turn, can be processed by other modules. A second important technology for automation are multi-agent systems. They enhance the framework by adding automated support for situations with goal conflicts. Consider situations where deviations from the plan occur and the framework shall determine countermeasures. Software refactoring is one possible use case: When the framework processes reports of static analysis tools indicating quality problems in the source code, software quality measures can help. However, mostly there are too many problems to tackle all and the most suitable must be selected. In such situations, agents perusing different quality goals like maintainability or reliability can autonomically decide on software quality measures that are afterwards integrated into the process in cooperation with the other modules [14].

V. MODULES

This section provides details on the different modules, their capabilities and the utilized technologies.

Data Storage. As depicted in Section IV, the first use case for this module is being the data store for the module communication. Messages are stored here and the modules can register for different topics and are automatically notified if new messages are available for the respective topic. This also provides the basis for the loose-coupling architecture. However, this module is not limited to one database technology but enables the integration of various technologies to fit different use cases. One is the creation of a project ontology using semantic web technology to store and process high-level project and domain knowledge that can be used to support the project actors.

Process Management. This module provides PAIS (Process-Aware Information System) functionality: Processes are not only modelled externally at the project management level as an idea of how the project shall be executed but can be technically implemented. Thus, the enactment of concrete process instances enables the correct sequencing of technical as well as human activities. Humans automatically receive activities at the right time and receive support in executing these. To enable the framework to react on dynamic changes we apply adaptive PAIS technology [18]. That way the framework can automatically adapt running process instances. Consider an example from software development projects: Software quality measures can be inserted into the process automatically when the framework detects problems in the source code by analyzing reports from static analysis tools [14]. This actively supports software developers in achieving better quality source code.

Sensors. This module comprises facilities for receiving events from the frameworks environment. These events can be provided by hardware sensors that are part of production machines. This can also be established on the software side by integrating sensors in the applications used by knowledge workers. That way, information regarding the processed artifacts can be gathered. Examples regarding our scenario from Section II include bug trackers and development tools so the framework has information about bugs in the software and the current tasks developers process.

Graphical User Interfaces. GUIs enable humans to interact with the framework directly. Firstly, this applies to the enactment of processes with the framework. The latter can provide activity information to humans guiding them through the process. In addition, humans can control the process via GUIs indicating activity completion and providing the framework with information on their concrete work. Another use case is storing knowledge in a knowledge store being part of the framework. To enable this, the GUI of a semantic wiki integrated into the framework as knowledge store can be exposed to let humans store the knowledge and annotate it with machine-readable semantics. That way, the framework can provide this knowledge to other humans in an automated fashion. However, GUIs are also used for configuring the framework to avoid hard-coding its behavior matching the respective use case. One example is a GUI letting humans configure the rules executed in the integrated rules engine. Thus, e.g., it can be configured, which parts of external reports shall be used for transformation to a unified report the framework will process.

Connectors. This module is applied to enable technical communication with external tools. Depending on the use case, interfaces can be implemented to call APIs of other tools or to be called by these. Consider an example relating to the projects' process: The process is externally modeled utilizing a process modeling tool. This process can be transformed (manually or automatically) to a specification our framework uses for process enactment. In the process enactment phase, the external tool can be automatically updated displaying the current state of execution.

Automation. For this module we proposed two technologies as a starting point: rules engines can be utilized for simple automation tasks. One use case is, as mentioned, automatic transformation of reports from multiple external tools into one unified report. Multi-agent systems are applicable in situations where goals conflicts apply. Consider the example regarding the quality of newly created software: In software projects, often multiple static analysis tools are executed providing metrics regarding the source code quality. Usually, there is not enough time to get rid of all issues discovered. It is often challenging for software engineers to determine the most important software quality measures to be applied. Such projects mostly have defined quality goals as maintainability or reliability of the source code. Quality goals can be conflicting as, e.g., performance and maintainability and different measures support different quality goals. For such situation, agents can be applied: Each goal gets assigned an agent with a different strategy and power. When a quality measure can be applied the agents utilize a competitive procedure for determining the most important quality measure to be applied.

Data Analysis. This module enables the integration of frameworks or libraries for semantic reasoning, statistical analysis, or machine learning frameworks like Scikit-learn [11]. The advantage of the integration in the framework infrastructure is option to execute such tools as part of a holistic process. Data that has been acquired by other modules can be processed and the results can also be stored in the frameworks data storage. Furthermore, other modules can be notified so humans can be involved. For example a process can be automatically executed where data is analyzed and the results are presented to humans that, in turn, can derive knowledge from them and directly manage this knowledge with the knowledge management component. That way, data analysis approaches can be seamlessly integrated at several points in the process to achieve a higher level of support and automation.

VI. EVALUATION

We now provide two concrete scenarios in which we have created and successfully applied concrete frameworks that implement our idea of this abstract framework. The first one comes from the software engineering domain. For this domain, we have implemented a comprehensive framework including all of the mentioned modules [14][15][17] as illustrated in Figure 4.

The framework is based on a loose-coupling architecture where different modules managed by an OSGI [19] infrastructure that communicate via events. Such events are stored and distributed by the data storage module. The latter is realized by a key-value store based on the XML database eXist [20]. The events are organized in several collections for which the modules can register to be automatically notified in case of new events regarding a certain topic. Communication with the frameworks environment is realized on the one hand by web-based GUIs and connectors to tools like bug trackers. On the other hand, the event extraction module applies the framework Hackystat [21] to be able to integrate sensors on various tools like IDEs or source control management tools. These sensors generate events in several situations like opening files or perspective switches in the IDEs. The events are sent to the data storage component, which, in turn, provides them to the event processing component. The latter applies complex event processing (CEP) utilizing the tool Esper [22]. That way, events with higher semantic value can be generated out of multiple low level events.

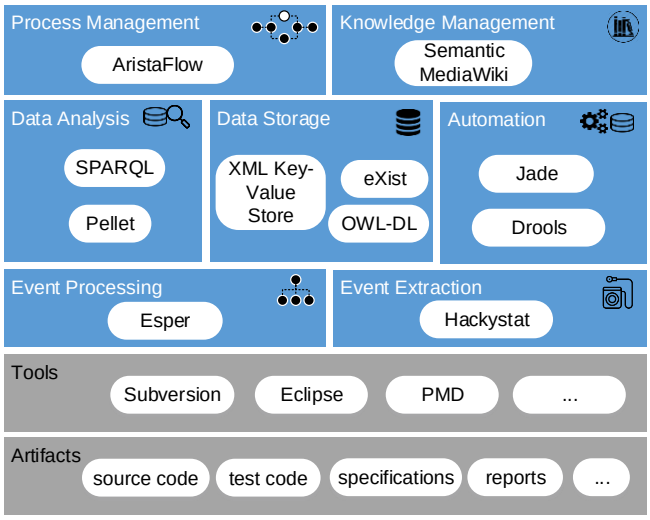


Figure 4. Framework Implementation for Software Engineering.

The framework also facilitates reasoning about higher level project information. Therefore, the framework integrates semantic web technology. This technology offers numerous advantages like advanced consistency checking or enhanced reuse possibilities among applications [23]. The data storage module therefore contains an OWL-DL (Web Ontology Language Description Logic) [24] ontology. That way, high-level project data can be stored in a standardized structured way. Moreover, ontologies provide capabilities for complex querying and the capability of reasoning about the contained data and inferring new facts. This is realized in the data analysis module with SPARQL [25] queries, SWRL [26] rule processing, and the reasoner Pellet [27]. This configuration fosters the integration of knowledge management with the high-level project information: The knowledge management component therefore integrates the Semantic MediaWiki [28]. Information entered in this wiki can be enhanced by machine-readable semantics enabling the framework to automatically access and distribute this information.

A framework aiming at holistic project support also needs components for automating as many tasks as possible. Therefor the automation module integrates the Jboss Drools [29] rules engine to execute simple automatisms, e.g., for converting reports. To support situations, in which goal conflicts arise, the framework also integrates the FIPA-compliant [30] multi-agent system (MAS) Jade [31]. Thus, it becomes possible to assign different goals to different autonomous agents that will pursue the respective goal.

To be able to support a software project holistically, its process and the various concrete workflows of the involved persons must also be managed in some way. To achieve this, the framework integrates the AristaFlow BPM suite [18][32] as process management module. That way, workflows can be composed out of existing services and human activities. The AristaFlow BPM suite guarantees correctness during modelling as well as enactment of the workflows. Furthermore, it has a feature crucial for process support in such a dynamic domain: Workflows can be adapted even when they are already running. Thus, their enactment is not tied to a pre-defined schema but can be tailored to the needs of the respective situation.

With this framework, we have implemented various use cases in order to achieve effective support of the involved persons. We will now provide details on one example of these being automatic provision of software quality measures. In that case, the framework automatically analyzed various reports regarding the quality of the source code and automatically selected matching software quality measures for existing problems. These measures were then automatically integrated into the developers' workflows in the best-matching situations. This was achieved by executing the following steps:

- Problem detection: To provide effective support, the awareness of existing problems is crucial. Therefore, the framework processes reports of external tools like PMD [33] for static source code analysis or Cobertura [34] for code coverage. Via rules processing the reports are transformed to a unified format and if defined thresholds for the contained metrics are exceeded, these are considered as problems and software quality measures are automatically assigned to them in the report. These are not the only problems that may exist but via the connections to various external tools using connectors and sensors the framework is aware of the execution of various tasks in the projects like testing or profiling and can detect their absence. The latter is also included in the problem report.
- Quality opportunity detection: To be able to distribute software quality measures automatically to concrete persons, the framework must be aware of their situation to not overburden them and provide quality measures matching their situation (e.g., the artifacts they are working on currently). This is enabled through the integration of the development process into the framework. Thus, the framework is aware of the planned activities and their timely planning as well as assignment to concrete persons. If a person finishes an activity early, the remaining time can be filled with a quality measure. As one cannot rely on people finishing their activities earlier as planned, there is also a quality overhead factor that allows for defining a certain percentage of the project to be reserved for quality activities.
- Measure tailoring: When the framework recognizes an opportunity for a quality activity, it triggers a measure proposal procedure. As a first step, the problems and assigned measures are strategically prioritized in line with the projects quality goals. This is achieved by an automated implementation of the Goal Question Metric (GQM) technique [35] realized with autonomous agents as illuistrated in Figure 5. Each agent pursues one quality goal like maintainability or reliability. Using the GQM structure, the agents can relate the metrics with violated thresholds to their respective goal. This is achieved by extending the standard GQM structure: Besides the goals, questions, and metrics the extended structure also incorporates measures and the agents. In addition, different levels of KPIs are integrated: The KPI aggregates the values of one or multiple metrics. The QKPI, which is assigned to a GQM question, aggregates the values of multiple KPIs. Finally, the GKPI that belongs to a certain goal aggregates multiple QKPIs. Utilizing these KPIs, each agent can calculate a concrete value representing the state of the goal it pursues. That way, the agents can autonomously prioritize concrete software quality measures. Each agent has a number of points he can distribute on the measures. For proactive measures, the agents use a competitive bidding process, in which each agent tries to bring measures relating to his goal to execution. For reactive measures, the agents utilize a cooperative voting process where the cumulated value of points spent by all agents on a measure is used for ranking the measures. After that, measures matching the respective person's situation must be selected to aid affective application of the measures. To find matching points in the various workflows a person processes, the latter are semantically annotated with extension points. These are points where a workflow can be extended by inserting new activities into it. Thus, specific properties of the persons, the measures, and the extension points can be matched to finally select the right measure for the right extension point of the right person.
- Measure application: At the end of the quality measure distribution the integration in the operational workflows of the respective persons must be done. This is achieved by the capabilities of the AristaFlow BPM suite: Workflow instances can be adapted during runtime even if their processing has already started. The process management module utilizes these capabilities to automatically and seamlessly integrate the measures into the selected workflows at the chosen extension points.
- Quality trend analysis: The final step of the procedure is the continuous analysis of the products' quality to assess the effectiveness of the applied quality measures. This is achieved by continuously analyzing reports from external tools. Thus, it can be determined if previously detected quality

problems disappear. Moreover, via the GQM structure the development of the quality goals can also be monitored.

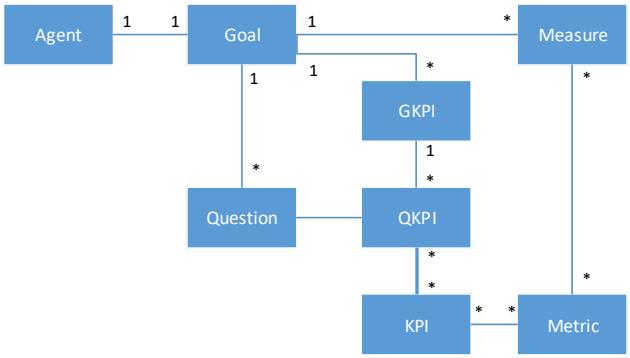


Figure 5. GQM Structure for Autonomous Agents.

Another use case was activity coordination: with the project ontology we determined relations of different artifacts and could automatically issue follow-up activities for example to adapt a software specification if the interface of a components' source code was changed and vice versa.

The integration of a semantic wiki enabled the following: Knowledge was recorded and annotated by humans and thus, the framework could automatically inject this knowledge into the process to support other humans in similar activities. In this project, we applied the framework in two SMEs and successfully evaluated its suitability. In fact, two teams used the framework in a certain project and reported on its usability. Thus, we gained insights in the advantages the framework could enable in real usage. The main advantage was the support of better software quality. With features like automated software quality measure distribution or activity coordination many aspects that would have been forgotten by humans could be automatically supported which can lead to better quality source code and less bugs.

The second scenario involves a business use case in which different companies in a supply chain have to exchange sustainability information regarding their production [13]. The producer of the end product has to comply with many laws and regulations and must collect information from the whole supply chain resulting in thousands of recursive requests. On the operational level, this process is very complex as it is difficult to determine, which information is important for sustainability, which one must be externally evaluated to comply, and which information should not be shared as it reveals internals about the production process. To implement such data exchange processes automatically, we applied a more tailored-down version of our framework [36] as illustrated in Figure 6.

In this case, the focus was different than the one described in the software engineering domain: The target was not holistic support of entire projects with various use cases but the support of one complex use case. The crucial components were the generic connection to various external tools in the supply chain to obtain the contextual properties influencing the data collection process and the automatic generation of the latter by analyzing the properties. Therefore, a more tailored-down implementation of our framework idea was suitable.

The connection to the frameworks environment was realized by a set of connectors and adapters. Thus, it was possible to gather information from various external tools and provide the frameworks functionality to others. The latter was enabled by exposing a Java Service Provider Interface (SPI). Data collection was realized by web service adapters to use data sources that provided web services. In addition to this, specific connectors were created for the most prevalent in-house solutions in this domain. With the different connectors the utilization of information from prevalent tools of this domain, like BOMcheck [37] or IMDS [38], was possible. Knowledge management was also realized in a less automated fashion for this use case with a wiki and a bulletin board to support users in storing and retrieving information regarding the sustainable supply chain communication.

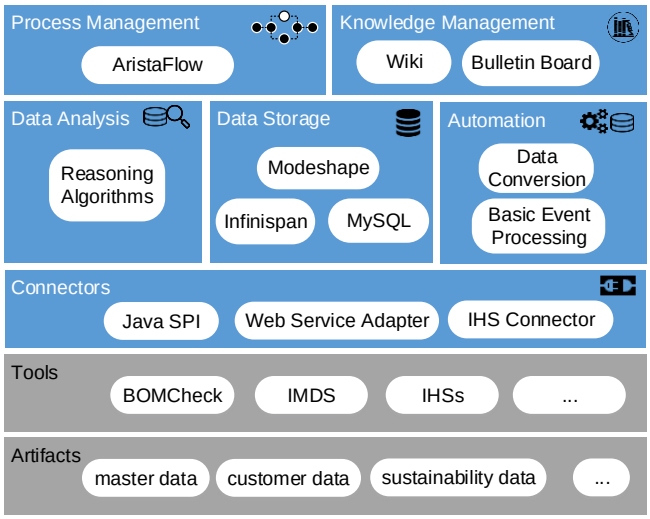


Figure 6. Framework Implementation for Supply Chain Data Collection.

The core of the framework was made up by the data storage, automation, and data analytics modules. Many different data sets had to be integrated as, e.g., master data, customer data, sustainability data and contents like reports. Furthermore, the ability to scale and provide high performance was crucial. Therefore, we opted for a combination of MySQL, Modeshape [39], and Infinispan [40]. The structured data with high focus on consistency was stored in MySQL while all relevant contents like documents and reports were stored in a content repository realized by Modeshape. For a performance increase we used the latter together with Infinispan, which acted as distributed cache. In the automation component, we implemented various data transformations to transfer the data obtained from external sources in a format our framework can process. Besides that, basic event processing was also integrated to automatically react to events in the changing environment. One example for this is the triggering of activities in case a certification of a customer was no longer valid, e.g., in case of changes to regulations.

For the data analysis module, we implemented reasoning algorithms that examine the obtained data from OEMs, regulations, the concrete requests, and suppliers. From this data, concrete context properties were extracted. By analyzing these properties and using the results to adapt processes, we were able to automatically create customized data exchange processes suiting different situations. For process management we opted for the AristaFlow BPM suite due to its capabilities for dynamic and correct process adaptation. In particular, we extended the process specifications with various properties to be matched with the context properties. To be able to build customized data exchange processes, we defined process fragments that were automatically composed to processes by our reasoning algorithms. Thus, it was not only possible to tailor these processes exactly to the respective situations but also to dynamically adapt long-running processes to changing situations. In this project, the framework was evaluated by a consortium of 15 companies and was later transferred to one of them to build a commercial tool from it.

These slightly different scenarios demonstrate the advantages of our approach: Its modules can be implemented matching the use case. The framework facilitates the communication between the modules and enables not only data analyses but also automated actions resulting from these supporting process and knowledge management.

VII. Related Work

To the best of our knowledge, there exists no directly comparable approach enabling holistic integration of various data analysis capabilities to support and operationally implement processes in knowledge-intensive domains. However, in different domains, there exist approaches to support projects and processes. One example are scientific workflow management systems [6][12]. Such systems support projects in the processing of large amounts of data. Their focus is the organization and parallelization of data-intensive tasks. Hence, they support the different steps taken to analyze data sets but are not able to support whole projects.

In the software engineering (SE) domain, there have also been numerous efforts to support projects and their processes. Early approaches include the Process-centered Software Engineering Environments (PCSEEs) [41][42]. These environments supported different SE activities and made process enactment possible. However, their handling was complex and configurability was cumbersome what made them obsolete. More recent approaches also exist but these frameworks focused on a specific areas of the projects. Examples are artifact-based support [43] and model-driven approaches [44]. Hence, these frameworks could not provide holistic support for entire projects.

Another area comparable to the current approach for supporting knowledge-intensive processes are tools and frameworks enabling the technical enactment of flexible processes like Provop [45], WASA2 [46], Worklets [47], DECLARE [48], Agentwork [49], Alaska [50], Pockets of Flexibility (PoF) [51], and ProCycle [52]. Provop provides an approach for modeling and configuration of process variants. WASA2 constitutes an example of adaptive process management systems. It enables dynamic process changes at the process type as well as the process instance level. Worklets feature the capability of binding sub-process fragments or services to activities at run-time, thus not enforcing concrete binding at design time. DECLARE, in turn, provides a constraint-based model that enables any sequencing of activities at run-time as long as no constraint is violated. Similarly, Alaska allows users to execute and complete declarative workflows. Pockets of Flexibility is a combination of predefined process models and constraint-based declarative modeling. Agentwork features automatic process adaptations utilizing predefined but flexible process models. Finally, ProCycle provides integrated and seamless process life cycle support enabling different kinds of flexibility support along the various lifecycle stages. As a matter of fact, all of these approaches enable the flexible technical enactment of processes which constitutes a crucial feature when trying to support knowledge-intensive processes. However, to achieve higher-level support, process changes must be applied automatically on account of current data. This requires components for automatic data analysis and automatic process changes which none of the mentioned approaches provide. Only Agentwork provides rudimentary capabilities for automation but lacks the general applicability of our approach.

The business domain also features complex knowledge-intensive processes. However, this domain is dominated by tools focusing on the processed data like ERP systems or specialized tools. One concrete example regarding the aforementioned sustainability data use case is BOMcheck [37], a tool that helps companies handling sustainability data. In particular, this tool contains current sustainability information on various materials but is not capable of supporting the process of data handling and exchange.

VIII. Conclusion and Future Work

In this paper, we presented a broadly applicable approach to support process implementation in knowledge-intensive domains. Based on our experience from prior research projects we suggested an extensible set of modules whose collaboration enables holistic support for projects. Furthermore, we proposed technologies, frameworks and paradigms to realize these modules with specific properties.

We have shown problems occurring in projects in different knowledge-intensive domains and provided an illustrative example from the software engineering domain. Such problems are mostly related to operational dynamics, complex data sets, and tacit knowledge. Our framework enables automatic processing of various data sets relating to the activities in such projects to not only support these activities but also their combination to a knowledge-intensive process. Thus, humans can be supported in transforming data to information and information to knowledge.

Finally, as evaluation, we have shown two concrete domains were we have successfully implemented concrete frameworks based on our idea of the abstract framework. In the software engineering domain we have shown how to

achieve holistic support and guidance for the involved persons encompassing entire projects. Therefore, we have implemented support for various complex use cases like automatic software quality management support, automated coordination, and knowledge management. The second scenario we presented relates to sustainable supply chain communication. We have shown how to implement a tailored-down version of the framework to support one complex use case spanning the whole supply chain: The recursive request of sustainability data from suppliers. To achieve this, we have analyzed various different data sets in order to customly and dynamically create data collection processes matching the properties of the respective situation.

As future work, we plan to extend the set of modules of our framework and to extend the technology options to realize these modules. We also want to specify concrete interfaces of the modules to enable standardized application and easy integration of new technologies. Finally, we plan to specify types of use cases and their mapping to concrete manifestations of our framework.

ACKNOWLEDGMENT

This work is based on prior projects at Aalen and Ulm universities in cooperation with Roy Oberhauser and Manfred Reichert.

REFERENCES

[1] G. Grambow, "Utilizing Data Analytics to Support Process Implementation in Knowledge-intensive Domains," Proc. 7th Int'l Conf. on Data Analytics (DATA ANAYLTICS 2018), pp. 1-6, 2018.

[2] M. P. Sallos, E. Yoruk, and A. García-Pérez, "A business process improvement framework for knowledge-intensive entrepreneurial ventures," The J. Technology Transfer 42(2), pp. 354–373, 2017.

[3] O. Marjanovic and R. Freeze, "Knowledge intensive business processes: theoretical foundations and research challenges," HICSS 2011, pp. 1-10, 2011.

[4] C. Di Ciccio, A. Marrella, and A. Russo, "Knowledge-Intensive Processes: Characteristics, Requirements and Analysis of Contemporary Approaches." J. Data Semantics 4(1), pp. 29-57, 2015

[5] R. Vaculin, R. Hull, T. Heath, C. Cochran, A. Nigam, and P. Sukaviriya, "Declarative business artifact centric modeling of decision and knowledge intensive business processes," 15th IEEE Int'l Conf. on Enterprise Distr. Object Computing (EDOC 2011), pp. 151-160, 2011

[6] J. Liu, E. Pacitti, P. Valduriez, and M. Mattoso, "A survey of data-intensive scientific workflow management," J. Grid Computing 13(4), pp. 457-493, 2015.

[7] P. Kess and H. Haapasalo, "Knowledge creation through a project review process in software production," Int'l J. Production Economics, 80(1), pp. 49-55, 2002.

[8] A. Liew, "Understanding data, information, knowledge and their inter-relationships," J. Knowl. Manag. Practice 8(2),pp. 1-16, 2007.

[9] O. Isik, W. Mertens, and L. Van den Bergh, "Practices of knowledge intensive process management: Quantitative insights," BPM Journal, 19(3), pp. 515-534, 2013.

[10] F. Leymann and D. Roller, Production workflow: concepts and techniques. Prentice Hall, 2000.

[11] G. Varoquaux et al.: Scikit-learn, "Machine learning without learning the machinery," GetMobile: Mobile Computing and Communications, 19(1), pp. 29-33, 2015.

[12] B. Ludäscher et al., "Scientific workflow management and the Kepler system," Concurrency and Computation: Practice and Experience, 18(10), pp. 1039-1065, 2006.

[13] G. Grambow, N. Mundbrod, J. Kolb, and M. Reichert, "Towards Collecting Sustainability Data in Supply Chains with Flexible Data Collection Processes," SIMPDA 2013, Revised Selected Papers, LNBIP 203, pp. 25-47, 2015.

[14] G. Grambow, R. Oberhauser, and M. Reichert, "Contextual injection of quality measures into software engineering processes," Int'l J. Advances in Software, 4(1 & 2), pp. 76-99, 2011.

[15] G. Grambow, R. Oberhauser, and M. Reichert, "Enabling automatic process-aware collaboration support in software engineering projects," Selected Papers of ICSOFT'11, CCIS 303, pp. 73-89, 2012.

[16] S. Schaffert, F. Bry, J. Baumeister, and M. Kiesel, "Semantic wikis," IEEE Software, 25(4), pp. 8-11, 2008.

[17] G. Grambow, R. Oberhauser, and M. Reichert, "Knowledge provisioning: a context-sensitive processoriented approach applied to software engineering environments," Proc. 7th Int'l Conf. on Software and Data Technologies, pp. 506-515, 2012.

[18] P. Dadam and M. Reichert: The ADEPT Project, "A Decade of Research and Development for Robust and Flexible Process Support - Challenges and Achievements," Computer Science - Research and Development, 23(2), pp. 81-97, 2009.

[19] OSGi Alliance: https://www.osgi.org/. [retrieved 02, 2019]

[20] W. Meier, "eXist: An Open Source Native XML Database," Web, Web-Services, and Database Systems, Springer, ,pp. 169-183, 2009.

[21] P. M. Johnson, "Requirement and Design Trade-offs in Hackystat: An In-Process Software Engineering Measurement and Analysis System," Proc. of the First International Symposium on Empirical Software Engineering and Measurement, IEEE Computer Society, 2007, pp. 81-90.

[22] Espertech: http://www.espertech.com/esper/. [retrieved 02, 2019]

[23] D. Gasevic, D. Djuric, and V. Devedzic, "Model driven architecture and ontology development." Berlin: Springer-Verlag, 2006.

[24] World Wide Web Consortium, "OWL Web Ontology Language Semantics and Abstract Syntax," (2004)

[25] E. Prud'hommeaux and A. Seaborne, "SPARQL Query Language for RDF," W3C WD 4 October 2006.

[26] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean,. "SWRL: A semantic web rule language combining OWL and RuleML," W3C Member Submission, 21, 79, 2004

[27] E. Sirin, B. Parsia, B. C. Grau, A. Kalyanpur, and Y. Katz, "Pellet: A practical OWL-DL Reasoner," J. Web Semantics, 2006.

[28] M. Krötzsch, D. Vrandecic, and M. Völkel: „Semantic mediawiki," Proc. Int'l Semantic Web Conference, pp. 935-942, 2006.

[29] P. Browne, "JBoss Drools Business Rules. Packt P. Browne. JBoss Drools Business Rules" Packt Publishing, 2009.

[30] P. D. O'Brien and R. C. Nicol, "FIPA — Towards a Standard for Software Agents", BT Technology Journal, 16(3), pp. 51-59, 1998

[31] F. Bellifemine, A. Poggi, and G. Rimassa, "JADE - A FIPA-compliant Agent Framework," Proc. 4th Intl. Conf. and Exhibition on The Practical Application of Intelligent Agents and Multi-Agents. London, 1999.

[32] M. Reichert et al., “Enabling Poka-Yoke Workflows with the AristaFlow BPM Suite,” Proc. BPM'09 Demonstration Track, 2009

[33] T. Copeland, “PMD Applied,” Centennial Books, ISBN 0-9762214-1-1, 2005

[34] Cobertura, http://cobertura.github.io/cobertura/. [retrieved 02, 2019]

[35] V. Basili, G. Caldiera, and H. D. Rombach, “Goal Question Metric Approach,” Encycl. of Software Engineering, John Wiley & Sons, Inc., pp. 528-532, 1994.

[36] N. Mundbrod, G. Grambow, J. Kolb, and M. Reichert, “Context-Aware Process Injection: Enhancing Process Flexibility by Late Extension of Process Instances,” Proc. CoopIS15, pp. 127-145, 2015.

[37] BOMcheck: https://www.bomcheck.net. [retrieved 02, 2019]

[38] International Material Data System: https://www.mdsystem.com/imdsnt/startpage/index.jsp. [retrieved 02, 2019]

[39] Modeshape: http://modeshape.jboss.org/. [retrieved 02, 2019]

[40] Infinispan: http://infinispan.org/. [retrieved 02, 2019]

[41] S. Bandinelli, A. Fuggetta, C. Ghezzi, and L. Lavazza, “SPADE: an environment for software process analysis, design, and enactment,” Software Process Modelling and Technology. Research Studies Press Ltd., pp. 223-247, 1994.

[42] R. Conradi, C. Liu, and M. Hagaseth, “Planning support for cooperating transactions in EPOS,” Information Systems, 20(4), pp. 317-336, 1995.

[43] A. de Lucia, F. Fasano, R. Oliveto, and G. Tortora, “Fine-grained management of software artefacts: the ADAMS system,” Software: Practice and Experience, 40(11), pp. 1007-1034, 2010.

[44] F. A. Aleixo, M. A. Freire, and W. C. dos Santos, U. Kulesza, “Automating the variability management, customization and deployment of software processes: A model-driven approach,” Enterprise Information Systems, pp. 372-387, 2011.

[45] A. Hallerbach, T. Bauer, and M. Reichert, “Capturing variability in business process models: the Provop approach,” J. Software Maintenance and Evolution: Research and Practice, 22(6 7), 2010, pp. 519-546

[46] M. Weske, “Flexible modeling and execution of workflow activities,” Proc. 31st Hawaii Int'l Conf. on System Sciences, 1998, pp. 713-722

[47] M. Adams, A.H.M. ter Hofstede, D. Edmond, and W.M.P. van der Aalst, “Worklets: A service-oriented implementation of dynamic flexibility in workflows,” On the Move to Meaningful Internet Systems 2006: CoopIS, DOA, GADA, and ODBASE, LNCS, 4275, 2006, pp. 291-308

[48] M. Pesic, H. Schonenberg, and W.M.P. van der Aalst, “Declare: Full support for loosely-structured processes,” Proc. 11th IEEE International Enterprise Distributed Object Computing Conference 2007, pp. 287-298

[49] R. Müller, U. Greiner, and E. Rahm, “AGENT WORK: a workflow system supporting rule-based workflow adaptation,” Data Knowlage Engineering, 51(2), 2004, pp. 223-256

[50] B. Weber, J. Pinggera, S. Zugal, and W. Wild, “Alaska Simulator Toolset for Conducting Controlled Experiments on Process Flexibility,” Proc. CAiSE'10 Forum, LNBIP, 72, 2011, pp. 205-221

[51] S. Sadiq, W. Sadiq, and M. Orlowska, “A framework for constraint specification and validation in flexible workflows,” Information Systems, 30(5), 2005, pp. 349-378

[52] B. Weber, M. Reichert, W. Wild, and S. Rinderle-Ma, “Providing integrated life cycle support in process-aware information systems,” Int'l J. Cooperative Information Systems (IJCIS), 18(1), 2009, pp. 115-165

Dynamic Knowledge Tracing Models for Large-Scale Adaptive Learning Environments

Androniki Sapountzi[1] Sandjai Bhulai[2] Ilja Cornelisz[1] Chris van Klaveren[1]
[1]Vrije Universiteit Amsterdam, Faculty of Behavioral and Movement Sciences, Amsterdam Center for Learning Analytics
[2]Vrije Universiteit Amsterdam, Faculty of Science, Department of Mathematics
Email addresses: a.sapountzi@vu.nl, s.bhulai@vu.nl, i.cornelisz@vu.nl, c.p.b.j.van.klaveren@vu.nl

Abstract— **Large-scale data about learners' behavior are being generated at high speed on various online learning platforms. Knowledge Tracing (KT) is a family of machine learning sequence models that use these data to identify the likelihood of future learning performance. KT models hold great potential for the online education industry by enabling the development of personalized adaptive learning systems. This study provides an overview of five KT models from both a technical and an educational point of view. Each model is chosen based on the inclusion of at least one adaptive learning property. These are the recency effects of engagement with the learning resources, dynamic sequences of learning resources, inclusion of students' differences, and learning resources dependencies. Furthermore, the study outlines for each model, the data representation, evaluation, and optimization component, together with their advantages and potential pitfalls. The aforementioned dimensions and the underlying model assumptions reveal potential strengths and weaknesses of each model with regard to a specific application. Based on the need for advanced analytical methods suited for large-scale data, we briefly review big data analytics along with KT learning algorithms' scalability. Challenges and future research directions regarding learners' performance prediction are outlined. The provided overview is intended to serve as a guide for researchers and system developers, linking the models to the learner's knowledge acquisition process modeled over time.**

Keywords- adaptive learning; big data applications; deep learning models; knowledge tracing; predictive analytics; sequential machine learning.

I. INTRODUCTION

Big Data Analytics (BDA) is becoming increasingly important in the field of online education. Massive Open Online Courses (e.g., Coursera), Learning Management Systems (e.g., Moodle), social networks (e.g., LinkedIn Learning), online personalized learning platforms (e.g., Knewton), skill-based training platforms (e.g., Pluralsight), educational games (e.g., Quizlet), and mobile apps (e.g., Duolingo) are generating various types of temporal, dynamic and large-scale data about learner's behaviors during their knowledge acquisition process of a skill over time [1]–[3]. To illustrate this with an example, the 290 courses offered by MIT and Harvard in the first four years of edX produced 2.3 billion logged events from 4.5 million learners. The emerging scientific fields of educational neuroscience [4] and smart-Education [5][6] can provide new insights about how people acquire skills using these new big data sources in education.

Artificial Intelligence (AI), Learning Analytics (LA), and Educational Data Mining (EDM) are three areas under development oriented towards the inclusion and exploration of big data analytics in education [2][7]–[9]. AI, LA, EDM, and big data technologies have been progressing rapidly, including developments towards the inclusion and exploration of BDA in education. Yet, specific advanced analytic methods suited for large, diverse, streaming, dynamic or temporal data are still being under development. EDM considers a wide variety of types of data, algorithms, and methods for modeling and analysis of student data, as categorized by [2][10][11]. A critical question in this area is whether complex learning algorithms or better data in terms of higher quality [12], well pre-processed, or bigger in size [8][13]–[15] is more important for achieving improved analysis results concerning either their predictive power or explainability.

For the aforementioned reasons, the implementation of BDA in education is considered to be both a major challenge and an opportunity in education [2][3][7]–[11][13][16][17]. Table I illustrates the models used in EDM. task of Knowledge Tracing has been modeled via Neural Networks and Probabilistic Graphical supervised learning models.

Knowledge Tracing (KT) is widely applied in adaptive learning systems, and to other modal sources of big data [11] such as online standardized tests, Massive Open Online Courses (MOOCs) data, and educational apps. KT is an EDM framework for modeling the acquisition of student knowledge over time, as the student is observed to interact with a series of learning activities. The objective of the model is to either infer the knowledge state, -which stands for the depth and robustness of the specific skill- or to predict the performance on all learning resources in the sequence that assess the skill acquisition process.

KT can thus be considered as a sequence machine learning model that estimates a hidden state (i.e., the probability that a certain concept of knowledge is acquired) based on a sequence of noisy observations (i.e., the interaction-performance pairs on different learning resources on consecutive trials). The estimated probability is then considered a proxy for knowledge mastery that is leveraged in recommendation engines to dynamically adapt the learning resources or feedback returned to the learner.

There are plenty of past similar review attempts regarding the modeling of knowledge acquisition:

i. algorithms [2][10][17] and empirical evidence-based [7] EDM,
ii. issues [8], applications [9], research methods [14], and learner models [11] concerning big data in knowledge acquisition process,
iii. design principles [3], AI algorithms [16], learner models [50][51][34] for online, adaptive, intelligent learning systems,
iv. evaluation metrics for measuring learner modeling quality [26][27],

This study provides an overview of currently existing representations of KT models focused on prediction of learner performance. The educational and technical angles, inspired by the above list are then used as guides to select and analyze the modeling quality of the learner model.

The literature distinguishes between the probabilistic and deep learning representations and both are widely used to represent complex, real-world phenomena in other domains apart from learning. The former is comprised by Hidden Markov Models and Dynamic Bayesian Networks, which model the knowledge of a learner as a local, binary, stochastic hidden state. The latter is constituted by deep Recurrent Neural Networks (RNN) models with Long Short-Term Memory (LSTM) and Neural Networks augmented with an external memory (MANN). In this representation, the learner's understanding of a skillset is treated as a distributed, continuous hidden state in the LSTM and as an external memory (value) matrix in the MANN; both are updated in a non-linear, deterministic manner. This study is not focused on specific variants of one model, rather it considers all these different models for the representation of the evolution of learner knowledge. Furthermore, it is important to note that, although the challenges and advantages are usually hold for general applications in other domains, in this review we refer only to the task of knowledge acquisition.

TABLE I. MODELS FOR EDUCATIONAL DATA MINING

Predictive Analytics Methodology		
Statistical & Machine Learning		Computational Intelligence (CI)
Machine Learning Models		
	Supervised	*Unsupervised*
Continuous Output	- Decision Trees -Regression Analysis	- K-Means - PCA - SVD
Categorical Output	- Decision Trees - Logistic Regression - Naïve Bayes	- Association Rule Mining
Neural Networks, Nearest Neighbors, SVM, Probabilistic Graphical Models, Anomaly Detection and Random Forest can be applied to both outputs and learning types.		

From an educational point of view, learner models should satisfy a set of properties [3][5][36][44][45] in order to work in an adaptive, intelligent, online learning system to improve knowledge acquisition. This study focuses on the following four:

i. recent engagement of a learner with the learning activities,
ii. dynamic sequences of learning activities,
iii. inclusion of student's differences, and
iv. inclusion of the dependencies among skills that are instructed via activities.

The first feature is highly predictive for modelling knowledge acquisition over time; the second is important for the decision-making part of the model prediction; the third is concerned with the application of personalized learning; and the forth is the content-related requirement for many hierarchical knowledge domains. Embarking from the baseline Bayesian model, and based on the desired principles listed above, we outline four of the model's most recent extensions. These include the individualization of learning pace among students, the incorporation of the prerequisite relationships among multiple skills, and the continuous representation of the knowledge state. The latter enables all of the four aforementioned features to be partly estimated. There are other challenges [3][36] that are not discussed throughout the paper such as the management of optimal instructional types of learning resources, up-to-date predictions, and the lack of control over both the user experience and offline learning behavior.

From a machine learning point of view [15][34], the assumptions in the different representations [19] and the potential pitfalls and advantages of optimization [28][29][30][31] and evaluation [26][27] methods are investigated. Each model faces different challenges regarding the estimation of parameters, overfitting issues, data sample efficiency, intensity of computational operations, and overall complexity. The general idea is that, by investigating these aspects, one can gain understanding why the considered models work the way they do, under which conditions one should be preferred against another, and in which cases can a model fail to accurately model knowledge acquisition. Furthermore, based on the fact that online education systems produce large-scale data, we also consider the scalability and computational speed of the algorithms implementation phase [5][8]. Specifically, we constrain this dimension via outlying the algorithms' requirements on the following aspects [46]:

i. size of training data set,
ii. number of model parameters, and
iii. level of domain-knowledge dependence.

The review intends to serve as a guide of dynamic KT models for researchers and system developers; whose goal is to predict future learner's performance based on historical achievement trajectories and develop adaptive learning experiences. It provides a structured comparison of different models and outlines their strengths and similarities concerning the KT task. The resulting fundament may serve as inspiration for the development of more sophisticated algorithms or ways of richer data collection. The

corresponding citations throughout the paper provide further guidance in implementing or extending a model for a specific data source or educational application. A shorter version of the review is available in [1].

This study proceeds as follows. Section II describes the representation component for the KT along with a brief introduction behind the probabilistic and deep learning sequential models. Section III introduces the baseline KT model and four extensions of it, after which the strengths, weaknesses, differences, and similarities are highlighted in Section IV. Section V discusses Item Response Theory (IRT), as it is the alternative family of models for predicting future performance. Section VI investigates the prospects and challenges for modeling knowledge acquisition over time, and Section VII provides with the conclusions.

II. DATA REPRESENTATION FOR KNOWLEDGE TRACING

Data representation refers to the choice of a mathematical structure that models the data, in this case, the hidden learner's knowledge state while interacting with learning resources. It embodies assumptions required for the generalization to new examples [19]. Identifying a sufficient dataset and representation for addressing the prediction problem is not a trivial task.

A good representation is one that can express the available kind of knowledge [15], meaning that a reasonably-sized learned representation can capture the structure of a huge number of possible input configurations. Other elements contributing to a good representation are outlined in [19]. A relevant point to note is that the choice of representation affects analytics that lie in distributed systems -a common case in BDA- in the sense of the data set decomposition into smaller components; so that analysis can be performed independently on each component.

A. Learning Task

Consider a learner interacting with an intelligent, adaptive learning platform with the purpose of acquiring a skill or a set of skills. In KT, two AI frameworks have been utilized to represent the available knowledge and disentangle the underlying explanatory factors of this process: the Bayesian (inspired by Bayesian probability theory and statistical inference) and the connectionist deep learning framework (inspired by neuroscience). Bayesian Knowledge Tracing (BKT) is the oldest -and still dominant- approach for modeling cognitive knowledge over time, while the deep learning approach to knowledge tracing, known as Deep Knowledge Tracing (DKT), is a more recently developed state-of-the-art model. Both AI approaches are used for modeling sequential data which implies that the data instances are not any more independent and hold a temporal pattern. The temporal pattern in KT is the dependency between learner's time engagement within and between learning resources.

Modeling knowledge acquisition with the objective to predict the next learner's performance, in its general form can be seen as a supervised time-series learning problem. Suppose a data set D consisting of ordered sequences of length T, to be trajectories of exercise-performance observation pairs $X = \{(x_{m,1}, y_{m,1}) \dots (x_{m,T}, y_{m,T})\}$ with $y_{m,t} \in \{0,1\}$ from the m^{th} student on trial $t \in \{1, .., T\}$, and with $x_{m,t}$ to be a label of a subskill that instruct one or more of N skills $S = \{S^1, S^2, \dots, S^N\}$. The objective in the Bayesian approach is to estimate the probability applying a skill S^1 correctly on an upcoming exercise. This is estimated based on the sequence of observed answers that tap S^1, as determined by the concept map. Similarly, the objective of the deep learning approach -and of the logistic models described in Section V- is to predict the probability that the student will give a correct response $y_{m,t+1} = 1$ on an upcoming exercise, which could belong in any subskill. Different from Bayesian methods, the exercises in deep learning models do not have the skill notation of each exercise, thereby S is a latent structure.

Such a difference in computation is located in the logic behind generative and discriminative approaches of algorithms [20]. The former models the joint probability of both unobserved (target) y and observed (input) x random variables: $P(x, y)$ developing thus a model of each y; while the latter estimates the probability distribution of unobserved variables y conditioned on observed variables: $P(y|x)$ developing thus a model of boundary between y; in case of deep learning models, this boundary is non-linear. Neural networks are discriminative models that map out a deterministic relation of y as a function of x. Fig. 3 depicts this mapping of x sequence vectors to y sequence vectors.

B. Domain-Knowledge Dependence

Domain knowledge dependence refers to the amount of human involvement necessary to tailor the algorithm to the learning task, i.e., specify the prior knowledge built into the model before training [46]. An important distinction between the probabilistic and deep learning KT approaches is located in the existence of the concept map and the notion of a skill.

The concept map or otherwise called expert model breaks down the subject matter to chunks of knowledge. It maps an exercise and/or exercises' step to the related skill. Each skill is divided into a hierarchy of relatively fine-grained subskills, also known as Knowledge Components (KC), which need to be acquired by a learner. Skills, also referred as concepts or competencies, are abstract but intuitive notions of ideas that the exercise instructs and assesses.

Each exercise may require one or more KCs so as to be solved; the latter case is known as multi-skill learning.

To illustrate the granularity level of a KC with an example, 'the location of Kyoto' is a fine-grained KC while 'the names and locations of countries' is a coarse-grained KC [34]. The granularity of KCs is a subject of experimental research. An example of a KC could be 'declare a variable in a function definition' which should be split into 'single-variable' and 'multivariable' KCs if the data indicates such a split is warranted.

In the probabilistic KT, the sequences of X are passed through the concept map that is assumed to be accurately labelled by experts. This ensures that students master prerequisite skills, before tackling higher level skills in the hierarchy [18]. It assumes that all the learning activities are of the same difficulty level, which implies that the observation of a student struggling on some resources is occurring because there are some subskill(s) that the student has yet to acquire. In the probabilistic KT with Hidden Markov Models, a different model is initiated for each new skill.

Rather than constructing a separate model for each skill, the deep learning approach, as well as Dynamic Bayesian Networks (DBN), model all skills jointly. In contrast to deep learning models, a DBN demands a detailed concept map with the conditional relationships representing prerequisites among the KCs. The deep learning models just require exercise tags that denote the related KCs while the underlying skill and the related KCs belonging to the same skill are unknown. Examples of tags include the 'Pythagorean theorem I', 'mode', 'mean', and 'slope', which can be considered as coarse-grained KCs.

The network of DKT is presented with the whole trial sequence of exercises for all the skills practiced. The sequences are passed through featurization; that is the distributed hidden units in the hidden layers that relate the input to the output sequences. This distributed featurization is the core of the generalizing principle and is used to induce features and hence discover the similarity and prerequisite relationships among exercises.

The MANN model can also automatically discover the correlations among the KCs and cluster them based on the skill they instruct. It uses the inner product of the exercise tag with the embedding matrix that contains all KCs and passes it through the SoftMax activation function as described later in eq. (5α). Compared to DKT, which requires both a threshold to cluster the similar KC's and the network to be represented with the whole trial sequence at once, this model directly assigns exercises to concepts.

C. Probabilistic Sequence Models: statistical learning machines

The problem of KT was firstly posed as a 1st order, 2-state, 2-observation Hidden Markov Model (HMM) with a straightforward application of Bayesian inference. A DBN, also referred as Two-Timeslice Bayesian Network (BN), is employed to solve for a multi-skill learning task.

HMMs and DBNs are generative models called Probabilistic Graphical Models (PGM). These are statistical learning models that embody assumptions about the data generation process by modeling conditional dependencies (i.e., interactions) between random variables. Formally, a PGM is a graph $G = (X, E)$, where:

i. the random variables X are represented as nodes in a graph, and

ii. the conditional dependencies between X are described by the edges E (i.e., graph topology).

A graph is a powerful representation that can model a variety of data types by simply changing the definitions of nodes and edges. However, inference and learning over graphs is considered a difficult task. DBN is a Directed Acyclic Graph (DAG); directed graphs are useful for expressing causal relationships between random variables [20].

HMM is used to model sequences of possible latent random variables X that form a Markov process, in which the Markov property holds; that is, 'the past is independent of the future given the present' $X_{t+1} \perp (X_{t-1}, X_{t-2}, \dots, X_1)|X_t$. X have arrows pointing to the observed variables Y which are conditionally independent to each other, given the input variables X.

The interesting part of HMM is that X have unobserved states h, also referred to as *hidden* or *latent* states, which can store information for longer times in the sequence. The states h have their own internal dynamics, described by transitions between h, which are stochastic and controlled by a matrix A. At each timestep, these can take only one of N possible values. The outputs produced by an h are stochastic and hidden, in the sense that there is no direct observation about which state produced an output, much like a student's mental process. However, h produce as observables the emission probabilities Φ that govern the probability distribution of h.

The HMM model is characterized by its transition probability A, emission probability Φ and prior distribution Π. The parameters that need to be evaluated and learned are $\lambda = \{\Pi, A, \Phi\}$, where Π is the initial latent variable x_1, which is the only variable that does not depend on some other variable. Firstly, the evaluation problem is solved $P(Y|\lambda)$: the probability that the observations are generated by the parameters λ of the model, where Y is a sequence of learning activities attempts Y = $\{Y_t\}$, $t \in \{1, \dots, T\}$; and secondly the learning problem is being solved: how should λ be adjusted so as to maximize the $P(Y|\lambda)$. In the probabilistic setting of KT, X represents *the entire single skill,* h represent the knowledge states, which are 2 standing for a *mastered* and for a *not yet mastered*, as shown in Fig. 1. A indicates the learning or forgetting rate, i.e., the evolution of student's knowledge state and Φ includes the probability for a guessed or slipped answer. A detailed explanation of the HMM is provided by [23], [24].

DBNs generalize the HMM models by including a collection I of interacting input variables X linked by directional edges. The internal structure among X and E, called as graph topology, is repeated in the exact same way at each time step. The parent node set of X in G, denoted by $pa(X)$ is the set of all nodes from which an edge points to node X in G. These models hold the directed Markov property, which is $X_I \perp (non-descendants\ (X_I))|pa(X_I)$. In KT, $pa(X)$ carry the meaning of a prerequisite relationship, the nodes X indicate different KCs or skills and their realization x_i indicate the

knowledge state of a student for skill i at a specific t. In KT, an observed variable is linked always to just one latent variable.

The question now becomes how the evolution of a knowledge state is modelled in a DBN. The value of a x_i at time slice t is calculated from not only the graph topology at t, described above, but also from the previous value of x_i at time $t-1$. Thereby, a DBN links knowledge state variables to each other over adjacent time steps to indicate learning or forgetting rate, as shown in Fig. 2.

In order to infer the probability distribution across the total number of hidden states h, there is a marginalization of the latent variables x over h. This is equivalent to $P(y, h|\ \theta) = \prod p(X|pa(X))$, denoting that the joint distribution of the observed Y and unobserved H variables is given by the product over all of the variables, i.e., nodes X of the graph conditioned on the $pa(X)$, where θ includes λ parameters together with the conditional edges of the graph topology. A detailed explanation of the computations in the DBN is provided by [20], [21].

D. Neural Network Sequence Models: computational intelligence learning machines

Deep RNN with LSTM internal memory units and Memory-Augmented Neural Networks (MANNs) have only recently been employed to the KT task to solve for the binary (i.e., h can take only one value), highly structured (assumptions about data generation) and memoryless (i.e., *Markov processes*) representation of the hidden knowledge state.

RNN and MANN are a family of Artificial Neural Networks (ANN) that can take variable length of inputs and the hidden state acts as a memory able to capture the temporal structure among sequences. MANN uses an external memory matrix to encode the temporal information, while LSTM uses an internal hidden state vector.

Deep learning, as it is primarily used, is a computational intelligence technique for classifying patterns (e.g. similarities found in data instances) to different targets y, based on large training data, using ANN with multiple layers [48].

ANN is a discriminative model that relates the input units x, which are amplified with weights w, to the output units y through a series of hidden layers: $y = f_1(\vec{w}_1, f_2(\vec{w}_2, \dots, f_n(\vec{w}_n, \vec{x})))$. Each hidden layer is comprised by hidden units, which are triggered to obtain a specific value by events found in x and -in case of RNN- also patterns that are found in previously hidden states. This process of triggering is implemented by a non-linear activation function f in the hidden layer.

RNNs are layered ANNs that share the same parameters w, through the activation function f. This property is illustrated in Fig. 3, with the formation of directed edges between hidden units. RNNs are powerful, as they combine the two following properties, not found in PGM's:

i. The distributed hidden state allows them to forget and store a lot of information about historical trajectories, such that they can predict efficiently.
ii. The non-linear activation functions allow them to update the hidden state in complicated ways, which can yield high-level structures found in the data (if available).

Instead of having a single hidden neural network layer (e.g., hyperbolic tangent) repeating at each step, LSTM is a type of hidden units that additionally includes *Forget, Input, and Output gates* repeating at each step. The interaction of the gates with each other is used to adjust the flow of information over time. The hidden state acts as a memory able to hold bits of information for longer periods of time and hence capable of learning complex functions from '*remembering*' even longer sequences of data (i.e., long-term dependencies).

MANN refers to the class of external-memory equipped networks rather than the inherent memory-based architectures, such as LSTM. It is a special kind of RNN and it is advantageous for

i. rapid learning from sparse data,
ii. its computational efficient storage capacity, and
iii. meta-learning tasks (i.e., it does not only learn how to solve a specific task but it also captures information on the way the task itself is structured).

Instead of a distributed hidden vector, MANNs have an external memory to model the hidden state. The external memory contains two parts, a memory matrix that stores the information and a controller that communicates with the environment and reads or writes to the memory allowing it to forget information. These operations also make use of non-linear activation functions.

In Fig. 1, Fig. 2, Fig. 3, the blue circular nodes capture the hidden students' knowledge state per skill, while the orange rectangles denote the exercise-performance observations associated with each skill. The nodes in the probabilistic models denote stochastic computations, whereas in the RNN indicate deterministic ones.

III. Dynamic Models Applied In Knowledge Tracing

The dynamic models applied in KT are described below.

A. Standard Bayesian KT: skill-specific discrete states

The BKT model [18] includes four binary parameters that are defined in a skill-specific way. The model emits two performance-related parameters:

i. S-slip, the probability that a student will make an error when the skill has been acquired, and

ii. G-guess, the probability that a student will guess correctly if the skill is not acquired;

The model additionally distinguishes between two learning-related transition parameters:

i. $P(\theta_{t-1}) = P(\theta_0)$, the probability of knowing the skill a priori, and

ii. $P(T)$ represents the transition probability of learning after practicing a specific skill on learning activities. The knowledge acquired is estimated using equations (1a), (1b), (1c) and (1d) as illustrated below. The acquired knowledge $P(\theta_t)$ on trial t is updated according to (1c) with $P(T) = P(\theta_{t+1} = 1 | \theta_t = 0)$. The probability of a correct or incorrect attempt is computed using equation (1a) and (1b), respectively. Equation (1d) computes the probability of a student applying the skill correctly on an upcoming practicing activity. The equations are as follows:

$$P(\theta_{t+1}|y_t = 1) = \frac{P(\theta_t) \cdot (1 - P(S))}{P(\theta_t) * (1 - P(S)) + (1 - P(\theta_t)) \cdot (P(G))} \quad (1a)$$

$$P(\theta_{t+1}|y_t = 0) = \frac{P(\theta_t) \cdot P(S)}{P(\theta_t) \cdot P(S) + (1 - P(\theta_t)) \cdot (1 - P(G))} \quad (1b)$$

$$P(\theta_{t+1}) = P(\theta_{t+1}|y_t) + \left(1 - P(\theta_{t+1}|y_t)\right) \cdot P(T) \quad (1c)$$

$$P(KC_{t+1}) = P(\theta_t) \cdot \left(1 - P(S)\right) + \left(1 - P(\theta_t)\right) \cdot P(G) \quad (1d)$$

At each t, a student m is practicing a step of a learning activity that taps a single skill S. The process of a student trying to acquire knowledge about S^1 is illustrated in Fig. 1 over one-time step. The learner state can be in one of the two states and can emit one observable. Given a series of y_t, and t for the student m and skill S^1, the learning task is the likelihood maximization of the given data $P(y|\lambda)$, where $\lambda = \{P(S), P(G), P(T), P(\theta_t)\}$. This is done through Curve Fitting or Expectation Maximization and evaluated via Mean Absolute Error.

The key idea of BKT is that it considers guessing and slipping in a probabilistic manner to infer the current state during the practicing process. Even though BKT updates the parameter estimates based on dynamic student responses, it assumes that all of the four parameters are the same for each student. It follows that, the data of all students practicing a specific skill are used to fit the BKT parameters for that skill, without conditioning on certain student's characteristics.

B. Individualized BKT: student-specific states on learning rates

Individualizing towards the learning rates $P(T)$ provides higher model accuracy and better parameters interpretability [23]. The Individualized BKT (IBKT) model [23] is developed by splitting the BKT parameters into two components *(i)* λ^k-the skill-specific, and *(ii)* λ^u-the student-specific; and combining them by summing their logit function $l(p) = log\left(\frac{p}{1-p}\right)$, and using the sigmoid function $\sigma(x) = \frac{1}{(1+e^{-x})}$ to transform the values again to a probabilistic range. These two procedures are illustrated in (2*a*):

$$\lambda = \sigma\left(l\left(\lambda^k\right) + l\left(\lambda^u\right)\right) \quad (2a)$$

Finding the gradients of the parameters λ is done via forward and backward variables. Updating the gradients is possible using the chain rule, as illustrated in (2b) for the student-specific component of the parameter

$$\frac{\partial L}{\partial \lambda^u} = \frac{\partial L}{\partial \lambda} \frac{\partial \lambda}{\partial \lambda^u} \quad (2b)$$

where L simply indicates the loss function.

Fig. 1 depicts the structure for the HMM model of both BKT and IBKT. Although the underlying HMM model -and hence the process of a student practicing exercises- remains the same, the fitting process is different, i.e., λ^u is learned for each student separately.

Both the BKT and IBKT assume independent skills because thus they cannot deal with hierarchical structures. This assumption is restrictive, because it imposes that different skills cannot be related and, as a result, observing an outcome for one skill is not informative for the knowledge level of another skill. However, the expert model in educational domains is frequently hierarchical and should allow for multi-skill learning. DAG is the optimal data representation for describing the expert model in adaptive learning systems that incorporate parallel scalable architectures and BDA [3].

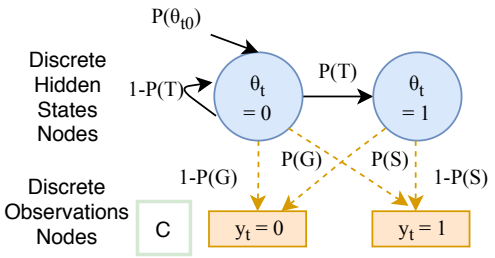


Figure 1. *Baseline and Individualized Bayesian Knowledge Tracing represented as a Hidden Markov Model over one time step*. In IBKT, the parameter {T} is learned seperately for each student.

C. Dynamic Bayesian Network: hierarchical, skill-specific, discrete states

DBN is a DAG implemented for the KT task [21] to allow for the joint representation of dependencies among skills.

On contrast to the previous models, at each timestep t, a student m receives a quiz-like assessment that contains problem steps or exercises that belong to different skills. The structure of the Bayesian network is repeating itself at each time step t with additional edges connecting the knowledge state on a skill at t to $t + 1$. This is the learning or forgetting rate, previously denoted as $A = \{T\}$. Same as in BKT and IBKT, once a certain threshold for a skill mastery is reached, the user can start practicing the less mastered skills.

The enhancement of the model is based on the fact that it is possible to infer the knowledge state for a skill, say S^3,

even without having observed certain outcomes for that skill y^3. To illustrate that, consider the example model depicted in Fig. 2. It depicts that, the probability of skill S^3 being mastered at t_2 depends not only on the state of S^3 at the previous time-step t_1, but also on the states of S^1 and S^2 at t_2.

The set of variables X contains all skill nodes S as well as all observation nodes Y of the model, while H denotes the domain of the unobserved variables, *i.e.*, exercises that have not yet been attempted by students and hence their corresponding binary skill variables S are latent. Suppose that a student solves a learning activity associated with S^2 at step t_2; then the hidden variables at t_2 will be $h_m = \{S^1, S^2, S^3, y^3, y^1\}$ while the observed variables will be y^2. The objective is to estimate the parameters λ that maximize the likelihood of the joint probability $p(y_m, h_m|\lambda)$. The likelihood loss is reformulated using a log-linear model to obtain a linear combination of a lower dimensional representation of features F, as shown in (3):

$$L(w) = \sum_m \ln\left(\sum_{h_m} exp(w^{\mathrm{T}}\varphi(y_m, h_m) - \ln(Z))\right) \qquad (3)$$

where φ: $\mathcal{Y}$ x $\mathcal{H} \rightarrow R^F$ denotes a mapping from the latent space $\mathcal{H}$ and the observed space $\mathcal{Y}$ to an F-dimensional feature vector. Z is a normalizing constant and w denote the weights that can be directly linked to the parameters λ.

DBNs rely on an accurate graph topology and can handle only simple topologies. Additionally, this model grapples with the limitations of the binary representation of student understanding, the lack of student differences, and the requirement for an even more detailed concept labeling and parameter constrain sets. RNNs have only recently tried to model student understanding in order to lessen or break the aforementioned assumptions.

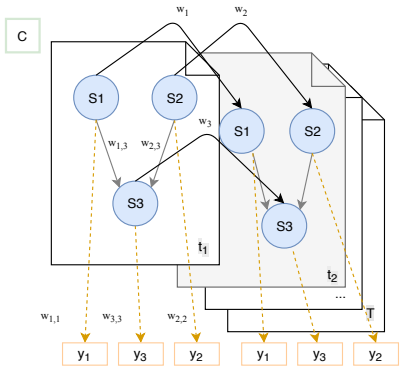


Figure 2. Bayesian Knowledge Tracing represented as a Dynamic Bayesian Network unrolled over T time steps. The hierarchical relationships between the skills (grey lines) are incorporated to the estimation of the learning rate (arrow lines) between adjacent time steps.

D. Deep Recurrent Neural Networks: continuous exercise-specific states and discovery of exercise dependencies

The complex representation in DKT is chosen based on the grounds that learning is a complex process [25] that should not rely only on simple parametric models as these models cannot capture enough of the complexity of interest, unless provided with the appropriate feature space [22].

The continuous and high dimensional representation of the latent knowledge state h_t in the hidden layer, learns the properties of sequences of the observed student interactions $x_t = \{(a_{m,0}, q_{m,0}) \ldots (a_{m,T}, q_{m,T})\}$, where a_t denotes the correctness of the response on a learning activity, which is denoted as q_t. In the deep learning context q_t denotes the corresponding activity tag, which can be roughly considered as a KC label.

DKT can discover exercise dependencies, i.e., prerequisites. Given that the knowledge state for a KC is represented as a hidden unit, the hidden-to-hidden connections encode the degree of overlapping between exercises. The researchers assign an influence metric J_{ij} on each directed pair of exercises i,j based on the correctness of the previous exercise i in the pair. They computed the correctness conditional dependencies between exercises $y(i)$, as shown in (4a):

$$J_{ij} = \frac{y(j|i)}{\sum_k y(j|k)} \qquad (4a)$$

where k is a predetermined threshold used to cluster the exercises that instruct the same skill. The possible skill labels for the clusters are manually provided.

DKT [25] exploits the utility of vanilla RNNLSTM whose fully and recurrent connections allow them to retain information of x_t for many time steps. The below equations describe the simple vanilla RNN and not the LSTM gates. Equation (4b) states that each hidden unit is activated via the hyperbolic tangent, which employs information on both the input x_t and on the previous activation h_{t-1} ,

$$h_t = tanh(W_{hx}x_t + W_{hh}h_{t-1} + b_h) \qquad (4b)$$

where b_h is the bias term and W_{hx}, W_{hh} are the weights of units corresponding to the input and hidden layers. The non-linear and deterministic output h_t will be passed to the sigmoid function σ to give the probability of getting each of the T learning activities correct $\hat{y}_t = (y_0, .., y_T)$ in the students' next interaction $t + 1$, as shown in (4c):

$$\hat{y}_t = \sigma(W_{yh}h_t + b_y) \qquad (4c)$$

Finally, the loss for a single student will be the negative log-likelihood, as shown in (4d):

$$L = \sum_{t} l(\hat{y}^T \delta(q_{t+1}), a_{t+1}) \quad (4d)$$

where l is the binary cross entropy $-(ylog(\hat{y})) + (1 - y)(1 - log\,(\hat{y}))$, δ denotes the one hot encoding transformation of the input $h_t = \{a_t, q_t\}$, which represents the categorical variables as binary vectors, and x_t is assigned to h_t. Compressing sensing is a suitable preprocessing step for larger data sets.

Fig. 3 depicts an example architecture of RNN, where X represents the entire sequence of exercises, which belong to multiple KCs, in the order the student receives them. After feeding X to the network, each time the student answers an exercise, the model predicts what KCs they are able to solve on their next interaction.

DKT requires large amounts of training data and it is prone to overfitting. Furthermore, it summarizes a student's knowledge state of all KCs in one hidden state vector, which makes it difficult to trace how much a student has mastered a certain skill over time. Zhang et al. (2017) [47] proposed a parameterization of MANN to address these two issues.

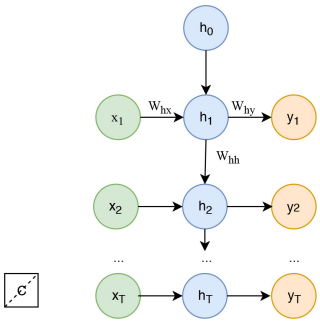


Figure 3. Deep Knowledge Tracing represented as a Recurrent Neural Network over 2 trials represented by the input $x_t = (a, q)$ and output y_t that denotes the probability of getting each of the q correctly. The lines in the hidden layer represent the learning rate between adjacent hidden knowledge states (blue nodes).

E. Memory Augmented Neural Networks: skill-specific states & discovery of exercise clusters

In the KT learning task, at each timestamp a MANN model takes a discrete exercise tag q_t, outputs the probability of response p($r_t|q_t$), and then updates the memory with the tuple (q_t, r_t). The MANN is extended to utilize a key-value, rather than a single memory matrix [47] because the exercise tags and the responses have different data types. The key component M^k, which is a static matrix, attends the latent skills underlying the exercises. The value matrix M_t^v stores, forgets, and updates the student's understanding of each skill h (skill state) via the read and write operations; and it changes over time.

Hence, the so-called Dynamic Key Value Memory Network (DKVMN) traces the knowledge of a student by reading and writing to the value matrix using the correlation weight w_t, which is commonly annotated by experts in the probabilistic KT framework. This is computed by taking the SoftMax activation of the inner product of the input exercise k_t^T and the key matrix M^k, as shown in (5α):

$$w_t(i) = softmax\,(k_t^T M^k(i)) \quad (5\alpha)$$

where i indicates the memory slot and k_t^T, arises after the multiplication of q_t with an embedding matrix so as to obtain a continuous embedding vector of the appropriate dimensionality size.

At each timestamp t, the learner solves an exercise tagged with q_t, the model finds that q_t requires the application of let's say skill $S1$ and reads the corresponding skill state h_{t-1}^{S1} from the read content r_t. This acts as a summary of the mastery level of the student for this exercise, as shown in (5b):

$$r_t = \sum_{i=1}^{N} w_t(i) M_t^v\,(i) \quad (5b)$$

Then it predicts p_t, which is the probability that the student will answer q_t correctly, as shown in (5c):

$$p_t = sigmoid(W_2^T f_t + b_2) \quad (5c)$$

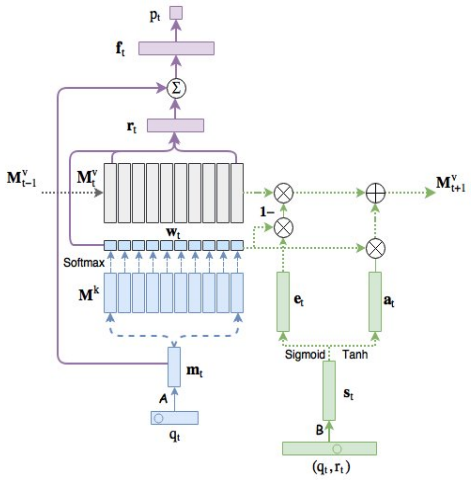


Figure 4. Deep Learning for Knowledge Tracing represented as a Dynamic Key Value Memory Network for one time step. The blue components denote the process of the correlation computation between the exercise and the underlying latent concepts, the purple components indicate the prediction process, and the green describe the update process that takes place after the students' interaction.

where f_t is a vector that contains both the student's mastery level and the exercises prior difficulty. It is calculated by a fully connected network as shown in (5d):

$$f_t = tanh(W_1^T[r_t, k_t] + b_1) \quad (5d)$$

where T and b indicate the transpose operation and the biases vectors respectively. After the student response is given, the model updates the values of h_{t-1}^{S1}. In this way, each time the student answers an exercise, the model not only predicts what exercises they are able to solve on their next interaction but also maintains a student's mastery level of each skill. The DKVMN makes use of the one-hot encoding preprocessing step and the binary cross entropy loss function. In Fig. 4 [47], the read and write processes of the model are described as purple and green components, respectively. It is implied that, the inaccurate estimation of $w_t(i)$ can lead to inaccurate predictions and updates.

IV. Comparison & Summarization of the Models

Tables III, IV, V outline important aspects of the models described above. Three dimensions are used as a guide for summarizing the models. The first is the machine learning algorithms' components [15], depicted in Table III, the second is the algorithmic scalability [46], illustrated in Table IV, and the third includes human learning related challenges faced by adaptive learning systems [36], described in Table V.

A. Machine learning components

The criterion of choosing the right algorithm is a combination of the efficiency of the available data along with the learning components of the algorithm; these are the representation, evaluation, and optimization [15]. In the below paragraphs, we briefly describe each of these components considering the KT task. The representation component has been already introduced in Section II.

1) Evaluation of the predictions

Model evaluation metrics analyze the performance of the model via the computation of training and the out-of-sample error; and are widely discussed in the context of machine learning applications including educational ones [26]–[28]. Even though, no experimental data are presented throughout the review, choosing for a metric is an open question in EDM including KT [27] for the assessment of the quality of the learner model. It depends highly on the intended use of the model and on whether absolute or relative predictions are important for this use. The metrics and their intended use are summarized in Table II based on findings from previous research [26]. This table can be used as a guide for assessing the quality of KT modeling concerning the evaluation metrics linked to it.

In KT, the metrics used for probabilistic understanding of errors include the Mean Absolute Errors (MAE) and Root Mean Square Error (RMSE). The former is considered an insufficient metric because it is biased towards the majority of classes whereas the latter is a proper score [26]. From the perspective of model comparison, the important part is only the sum of squared errors and not the square root. Note that RMSE has demonstrated a high correlation to the log-likelihood function and the *'moment of knowledge acquisition'* [26], which is highly important for mastery learning applications.

The RMSE without the squared error is sometimes referred to as the Brier Score and can give further insight to model behavior via decomposing it into three additive components. These are the following:

i. reliability, which measures the difference between predicted and observed probabilities,

ii. resolution, which captures the difference of the predictions from the base rate (proportion of positive classes), and

iii. uncertainty, which quantifies the inherent uncertainty of events.

An ideal model would, therefore, minimize the reliability term, while maximizing the resolution term. Assume that q_k are the model's predictions that they can take a set of different values c or values from c classes, n_k is the number of predictions that belong to the same category, and $f_k = \sum_{i,p_i = q_k} {}^{o_i}/_{n_k}$ is the frequency of observations. The Brier score is used by DBN whose formula is depicted in Table II.

As opposed to the probabilistic understanding of errors, values of qualitative metrics, i.e., either the prediction is correct or incorrect (0-1 loss), depend on the choice of the classification threshold. In the reviewed models, only classification accuracy was employed to evaluate the number of correctly predicted successes and failures on exercises. This measure reflects the proportion of true positives (TP) and true negatives (TN) as proportion of the total number of predictions (N). However, accuracy is not a reliable metric when the targeted classes are imbalanced. Recall is then a better metric to use, as it reflects the proportion of relevant incidents predicted correctly by the algorithm over the number of total relevant incidents. Commonly, this measure is used together with precision in F1 score, which is a more reliable metric than accuracy.

The Receiver Operating Characteristic (ROC) curve summarizes the qualitative error of the prediction model over all possible thresholds, so it summarizes performance even over those thresholds for which the algorithm would never be practically used. The predictions are considered relative to each other, and therefore the area under the ROC curve, called AUC, is better to be used as an additional metric for the evaluation of an algorithm's ability to distinguish correct from incorrect performances on exercises. It is interesting to note that, when the overall AUC is computed by averaging the per-skill AUC, namely weighing all skills equally, then its value is going to be smaller than by weighing all trials equally. This effect roots in two situations: i) the model performs poorly on a skill with only a few observations, and ii) it predicts the relative accuracy of different skills [22].

DKT and DKVMN employ the AUC on a per-trial basis instead of per-skill. Different from Bayesian methods, the deep learning models do not have the skill notation of each question and thus they cannot evaluate the results per skill.

Overfitting is a common source of error in machine learning models. That is, the model memorizes the training data and cannot generalize to out-of-sample data. The more increasing the number of model parameters, the more the danger of overfitting, since there will be less data for each subset of parameters that have to be estimated. All the aforementioned models apart from the standard BKT, compute the errors on Cross Validation (CV) so as to lessen the issue of overfitting. The folds are selected such that the mean performance of students is approximately equal to all folds, a technique referred as student-stratification.

TABLE II. EVALUATION METRICS AND APPROPRIATE USES FOR KNOWLEDGE TRACING

Metric	EDM Uses
Probabilistic	***Parameter Fitting & Model Comparison***
MAE	$1/n \sum_{i=1}^{n} \lvert y_i - \hat{y}_i \rvert$
RMSE	$\sqrt{1/n \sum_{i=1}^{n} (y_i - \hat{y}_i)^2}$
Several Numbers	***Model Comparison & Behavior***
Brier Score	$\frac{1}{N} \sum_{k} n_k \left[(q_k - f_k{}^2)(f_k - f^2) \right] + f(1-f)$
Qualitative	***Evaluation of Classification Tasks***
Accuracy	(TP+TN)/N
Recall	TP/(TP+FN)
Ranking of examples	***Interpretability of Results***
AUC	x-axis: (FP+TN), y-axis: Recall

The predictive ability of learner models is mainly a mean for improving the behavior of educational systems and for getting insight into the learning process. The automated evaluation metrics do not correlate with learning outcomes, namely, they cannot consider the fact that an adaptive learning system may not improve actual learning (e.g., decreasing learning curves). Therefore, frameworks oriented to adaptive learning systems should arise [27].

2) Optimization, Identifiability and Degeneracy

The optimization function derives the optimal values for the parameters of the objective function, which in KT is the log-likelihood function. Unlike in most other optimization problems, the function that generates the data and should be optimized is unknown and hence training error surrogates for the out-of-sample error [15]. The optimization of the log-likelihood function is performed using Curve Fitting (CF), Expectation Maximization (EM), Constrained optimization, and Gradient Descent (GD) methods.

Incremental optimization algorithms are suitable for large-scale data. GD on mini-batches is an incremental algorithm, which updates the weights using batches of data, and thus can avoid shallow local maxima. For instance, the IBKT model is built in an incremental manner by adding λ^u in batches and evaluating these additions on CV performance. It is also possible to improve the overall accuracy by incrementally updating the λ^k once a new group of students finishes a course or a course unit.

In addition to the big data handling, GD allowed IBKT to introduce student-specific parameters to BKT, without expanding the structure of the underlying HMM model and thus without increasing the computational cost of fitting. Researchers computed the gradients of the log-likelihood function given individual student and skill data samples with respect to every parameter. On every odd run, gradients are aggregated across skills to update skill component of the parameters; whereas in every even run, the gradients are aggregated across students to update respective student components. This block-coordinate descent is performed until all parameter values stabilize up to a pre-defined tolerance criterion.

In the procedure of training deep learning models, gradients tend to be unstable in the earlier layers as they either explode or vanish. Certain activations functions can cause this behavior. The exploding gradient problem refers to the large increase in the norm of the gradient, and hence it takes much time to converge to the parameters. To deal with the exploding problem, the DKVMN uses 'norm clipping' which is a function thresholding the values of the gradients before performing a gradient descent step; while DKT manually put thresholds to the values of the back-propagated gradients. The vanishing gradient, refers to the opposite behavior, when there is a small increase in the norm of the gradient, making it impossible for the model to learn from training data, i.e., find the correlation between temporally distant events. An interesting fact is that the LSTM model implemented by researchers in the DKVMN achieved better AUC than in the original paper of DKT. This could be probably because the DKVMN used 'norm clipping' and 'early stopping' instead of 'dropout' and manual thresholds to gradients.

In contrast to GD, EM takes longer to converge because even if it needs fewer evaluation steps, the computations are more difficult. In addition, due to the expectation step it does not directly maximize the likelihood of the learner's observations [23].

Constrained optimization is used in DBN [21] to ensure the interpretability of the constrained parameters and avoid the intractability issue, which can be caused by approximating the objective function. It obtains the joint

distribution as a product of the exponential terms, which translates to a weighted linear combination of feature vector entries in the exponent. This implies that the search of model parameters is done to a specific direction, in order to find the feature that is responsible for a prediction outcome.

The probabilistic KT models are susceptible to the identifiability and model degeneracy issues [20][28]. An identifiable model is considered the one that converges to the true values of the parameters, given an *infinite* number of observations. Model degeneracy occurs when the same combination of model parameters fits the data equally well. Hence, the resulted model parameters can lead to paradoxical behavior [28][30]. An example of a paradoxical behavior is the probability that the student acquired the instructed knowledge after three correct answers in a row [29]. Appropriate initialization conditions of the models' parameters and constrain values of the emission parameters, i.e., guess and slip, are used as techniques to resolve these two issues [18][28]–[31]. The identifiability and degeneracy are not relevant to deep learning frameworks, since they use approximation function and cannot pinpoint the input signal that lead to specific values for the model parameters.

TABLE III. COMPARISON OF KT MODELS: SUPERVISED MACHINE LEARNING COMPONENTS

Model	**Extension**	**Representation**	**Optimization**	**Evaluation**
BKT	Baseline-Binary skill Knowledge State	HMM	Curve Fitting or Expectation Maximization	MAE
IBKT	Learning Rate Personalization	HMM	Stochastic Gradient Descent on Minibatches	RMSE
DBN	Multi-Skill, binary Knowledge State	DBN	Constrained Latent Structure	RMSE, AUC, Brier Score
DKT	Continuous Knowledge State & Discovery of concept map	RNN-LSTM	Stochastic Gradient Descent on Minibatches	AUC, Accuracy
DKVMN	Skill Knowledge State & Discovery of concept map	Memory Augmented Neural Networks with Key-Value Matrices	Stochastic Gradient Descent on Minibatches	AUC, Accuracy

DKT, IBKT and DKVMN are prone to overfitting and need large-scale training data to optimize the objective function. Compared to the DKT, DKVMN does not require such large-scale data for training and is less susceptible to overfitting. Stopping the training before the weights have converged (i.e., 'early stopping'), and dropping out units (i.e., 'dropout') are methods that address the overfitting problem. Another issue present in DKT, is the alternation between mastered and not-yet-mastered state instead of the state transiting gradually over time [31], known as the waviness of the objective function. In addition to that, the model sometimes fails to reconstruct the input, which implies that even when a student performs well on a KC, the prediction of that KC's mastery level decreases instead, and vice versa [31]. According to the current literature, potential paradoxical behaviors while fitting the DKVMN are not yet investigated.

B. Scalability of the learning algorithms

Online education platforms create large-scale and diverse learning behavior data. An important aspect of BDA is the algorithm's ability to scale as new data or new features come into the model. In contrast to scaling towards the number of features, scaling towards the number of learners is an easier task that can be solved via using parallel infrastructures. In this study, we just scratch the surface of algorithmic scalability. Questions of large-scale data representation typically have much more complicated extensions in algorithmic, statistical, and implementation or systems aspects that are intertwined and need to be considered jointly.

1) Computational & Statistical Efficiency

Computational efficiency refers to the number of computations during training [46]. These include the numbers of the following:

i. iterations of the optimization algorithm,
ii. model parameters, and
iii. resources (i.e., the number of hidden units).

In Table IV, we note only the number of model parameters; this is not necessarily the most appropriate measure of model complexity. Nonlinear functions and large datasets increase the model complexity while offering flexibility in data fitting [20].

Implementing the DKVMN and especially the DKT models demand high numbers of computational resources. Nowadays, there are many parallel and distributed computing infrastructures and there is active research on parallel algorithms that can be used to boost the efficiency of data-intensive tasks. Parallel and scalable algorithms are often utilized for BDA. DKT, DKVMN, and IBKT models took the advantage of parallel computing infrastructures.

Compared to HMM model, DBN is computationally more expensive due to its complex loopy hierarchical structure [21]. Comparing the two deep learning models, LSTM is computationally more expensive than MANN, based on the ability of the latter to not increase the number of parameters when there is increasing number of memory slots. In general, the probabilistic models are computationally more efficient than the deep learning models even in other domains apart from online skill acquisition.

Statistical efficiency refers to the number of training examples required for good generalization performance. The volume of training data examples required to establish convergence, is depicted in Table IV as learnability requirements. Learnability, which is part of statistical efficiency, appears to present a daunting challenge for deep learning. Regarding the probabilistic KT models, the inclusion of large-scale training data can prevent identifiability problems. It is also important to note that, the parameter estimates and the behavior of the probabilistic KT models should be researched in different prior cases $P(\theta_0)$

and in scalability cases in either the number of students or the increased number of interaction examples per student [30].

2) Domain-Knowledge Dependence

In the educational domain, domain-knowledge or otherwise called human involvement is not only expensive to obtain, but there are also many differences in content representations beliefs among experts. The model is more flexible if it is less domain-dependent, implying that its performance is less prone to additional error. In contrast to the probabilistic models, the deep learning models are highly flexible. Beside deep learning models, DAG models can also learn the structure of the concept map as a network. However, the relationships between content are difficult to establish and to represent in the model, even given expert labels [36].

Adding or deleting pieces of modules or content in expert models is also an important scalability dimension related to the content representation. Rule-based algorithms fail to scale while flexible graph representations are much easier to scale. DAG models are considered the optimal representation of the relationships between concepts/skills (abstract but intuitive notions of ideas that the content teaches and assesses) and KCs/content units (pieces of content). A similar scalable graph ontology based on concepts and content units is used in Knewton, a well-known, online, adaptive learning tool.

C. Adaptive Learning Properties

There are many adaptive learning properties [36] and in this study the focus is placed on individualized learning rates [23], multi-skill learning [21], recency engagement, and skill discovery [25] [47].

1) Student Differences: Prior Knowledge, Learning Rate & Recent Engagement

Modeling parameters on an individual level results in significant changes regarding instructional or mastery decisions [45]. Researchers found that the inclusion of student-specific parameters has a significant positive effect on prediction accuracy and interpretability [23][24], as well as in dealing with overfitting [23]. Researchers [24] added Dirichlet priors for the initial mastery θ_{t-1}, while IBKT [23] extended their work and found that adding variables of learning rates $P(T)$ for individual learners, provides higher model accuracy. DKT and DKVMN allow for differences in learning ability of the student by conditioning on the average accuracy of recent learner's performance across trials and skills.

Learners' data include temporal dependencies, namely, there is a correlation in time engagement within or across learning resources and student's performance on activities, as described in Section II. Furthermore, recent performance is more predictive than past performance. DKT and DKVNM inherently are more sensitive to recent trials, allow for long-term learning and can capture temporal dependencies.

The probabilistic KT tends to predict practice performance over brief intervals where forgetting the acquired knowledge is almost irrelevant; though extensions of BKT towards this direction have been proposed. These include forgetting from one day to the next and not on a much shorter time scale [22]. DBN includes the forgetting property, but it would be more insightful if the model was compared with the equivalent BKT extension that includes forgetting.

Beyond the student knowledge that is reviewed throughout the paper, there are some single-purpose KT models augmented with non-performance data such as meta-cognitive [42], affect [43], and other student differences [44][45] apart from the learning rates that we reviewed.

2) Skill Dependencies & Adaptive Instructional Policies

The effect of sequencing learning resources is an important adaptive property of learning systems. Each skill has some degree of influence on the learning of other skills, especially in hierarchical domains of knowledge, such as algebra and physics. The paths through learning resources such as exercises, KCs, or abstract skill concepts taken by learners can influence their knowledge acquisition process.

TABLE IV. COMPARISON OF KT MODELS: SCALABILITY

Model	**Learnability requirements**[i]	**Efficiency**[ii]	**Domain Knowledge Dependence**[iii]	**Limitations**
BKT	↓↓↓	4 /skill, ↑↑↑	↑	Prone to bias, independent skills assumption, local transitions between states
IBKT	↑↑	4 /skill + 1 /student *(a)* ↑↑	↑	Independent skills assumption, local transitions between states
DBN	↓	4 /skill + 2^{n-1} for n skills ↑↑	↑↑	Hard-coded skill dependencies, complex, tractable only for simple models, local transitions between states
DKT	↑↑	250K with 200 hidden units & 50 skills *(b)* ↓↓	↓	Highly complex, prone to overfitting, not interpretable
DKVMN	↑	130K with 200 states & 50 skills ↓	↓	Highly complex, not interpretable, local transitions between states

i. the higher, the more complex the model,
ii. the higher, the less complex the model,
iii. the higher, the less flexible the model,
a. only the learning rate is individualized,
b. 4(input size+1) * output size + output size^2.

DKT and DKVMN can discover the inter-skill similarities and exercise prerequisites without requiring any domain knowledge apart from the exercise tags. A set of skill

labels are manually provided but the annotation part is done by the model. They can return the sequence of learning resources to a student that maximizes the expected knowledge state of that student. An interesting question is whether a sequence should contain exercises that belong to different skills or refer to one skill only. A trained DKT can be used in a Markov Decision Process for testing this scenario given a further time horizon (i.e., long-term learning). They found that presenting the exercises in an interleaved order of skills yields higher predicted knowledge after solving fewer problems, relative to presenting the exercises in a blocked order of the same skill.

Currently, the deep learning models can capture the relationships among exercises within a skill. It is worth mentioning that although deep learning models suffer from the lack of hierarchical input data structures [48], recently there is a lot of research in the direction of graph based deep learning models able to address hierarchical structures [52].

BKT and IBKT assume that each skill is independent and thus cannot be directly used to infer adaptive instructional policies; since they cannot keep the absolute sequence of exercises, given that they are not implemented in a mastery-learning way. A student's raw trial sequence is parsed into skill-specific subsequences that preserve the relative ordering of exercises within a skill, but discard the ordering relationship of exercises across skills.

TABLE V. COMPARISON OF KT MODELS: ADAPTIVE HUMAN LEARNING COMPONENTS

Model	Inclusion of forgetting rate	Inter-Skill Similarity & Instructional Policies	Learner individual differences	Multi-skill learning
BKT	✗	✗	✗	✗
IBKT	✗	✗	✓	✗
DBN	✓	✓	✗	✓
DKT	✓	✓	✓	✗
DKVMN	✓	✓	✓	✗

On the other hand, DBN allows for modeling hierarchical skill-dependencies, given a detailed expert model and can yield meaningful instructional policies. It can be used to offer an adaptive number of exercises that need to be solved for skill mastery. This is a mastery learning setting where the effort (number of practice opportunities needed to pass a skill) and score (percentage of correct observations after having the skill passed) of a learner are optimized [27][45]. Though, DBN is computationally tractable only for the simplest topologies among skills since exact inference is exponential in the number of parents a node has. Given a large-scale dataset, approximate inference can be used to exchange accuracy with computational time.

D. Comparison of KT models' applications & performance

All the models track at each time step the evolution of student knowledge state in real time for each skill separately [18][23], for a set of skills [21][47], or for a set of exercises [25][47]. This implies that, after a students' interaction with an exercise, the models update the knowledge state of each student in a skill [18][23] or skillset [21][25][47] way. To illustrate this with an example, let's assume there are fifty exercises where "bivariate data frequencies", "linear models of bivariate data", "plotting the line of best fit", "interpreting scatter plots", and "scatter plot construction" correspond to distinct labels of five exercises. DKT and DKVMN take the past performance of students on the sequence of 50 exercises and the current performance of a student on an exercise, and it will predict the probability of getting each of the exercise correct in their next interaction. Both models will cluster these exercises in one cluster named 'Scatter Plots' which can be roughly considered as a single skill. Different from DKT, the advantage of DKVMN is that it is more powerful in storing past performance of a learner since it can maintain the knowledge state per skill instead of only per distinct exercise label.

The three probabilistic models [18][21][23] cannot capture the relationships between the exercises. BKT and IBKT assume that, the labels provided in the example above correspond to different fine-grained skills, each one separately modelled. Each time an answer is provided for an exercise, the model will give the probability of the level of skill acquisition at time t given the probability of the level of skill acquisition on an exercise on the previous time step. Commonly, once a certain mastery level is reached, the learner can move to the next skill. Different from that, DBN will output a collection of probability distributions specifying the knowledge level for each skill or KC label. IBKT is the only model that individualizes toward learning rates of students.

KT models are commonly applied in curriculum sequencing and mastery learning frameworks, both axes are useful for smart and adaptive learning environments [5][7][10]. The former is used to either return or recommend to a learner a dynamic, optimal sequence of learning resources, whereas the latter is used to estimate the point of time that a certain skill is acquired [18] and from that point, learners are considered able to handle more advanced concepts. The DKT and DKVMN are used for curriculum sequencing while the HMMs and DBN for mastery learning. However, this study [49] suggests that KT models are better suited for discovery of concept and exercises relations, already included in DKT and DKVMN, rather than mastery learning applications. Mastery learning applications are threshold-dependent since it is difficult to automatically define an optimal threshold value.

DKT and DKVMN are complex, flexible and non-transparent models. DKT led to 25% increase in AUC when compared to the relatively simple BKT [25]. However, its success is attributed to its flexibility in capturing statistical regularities directly present in the inputs and outputs, instead of representation learning [22] which is the fundament advantage of deep learning models. When the performance of the DKT model and variations of BKT is compared [22], it is found that both models perform almost equally well. These variations allow for more flexibility in modeling statistical regularities that DKT has already the ability to explore because of the LSTM structure. DKT is presented with the whole trial sequence and thus it can discover aspects within interactions; whereas probabilistic models are given one student interaction at each time step. DKVMN performed better than the MANN baseline, DKT, BKT, and some BKT variations, as authors reported in [47].

Deep learning models can be superior towards the probabilistic ones, only if they are fed with more complex input data instead of exercise-performance interactions. Thus, they can take the full advantage of featurization and learn the representation of knowledge acquisition. These models can work only in platforms with a relatively large number of students and interactions, while not requiring significant domain expertise. All models can perform better when a bigger number of students and interactions is available to train the algorithm.

DBN led to significant improvements in prediction accuracy compared to BKT, and the logistic models of Additive Factor Model, and Performance Factor Analysis [21]. Researchers suggest that the performance differences between DBN and BKT, need to be investigated further. DBN is a highly structured and hierarchical model that can work well in hierarchical domains of the instructed concepts; and given the availability of accurate domain expertise for the detailed development of skills topology and complex constraint sets. It can infer a student's mastery on skills even if there are not any observed interactions linked to these skills, given there are interactions on other related skills. They can perform well in mastery learning applications.

IBKT also performed better compared to BKT and BKT variation of prior knowledge individualization [24]. IBKT is the only model that allows for wide variations among students.

IBKT as well as DBN are single-purpose models [32]. Combining the benefits of skill hierarchies and accounting for student differences could introduce a more holistic model. However, probabilistic models use conditional probability tables to make inferences of a learners' state, whose time and space complexity grows exponentially with the number of states and features. IBKT used logit functions to lessen this issue and incorporate user-specific features. An efficient framework allowing the integration of general features into KT via logistic models is introduced in [32].

Table VI outlines potential applications of each model in an adaptive learning platform and the effects that each model can infer.

V. ITEM RESPONSE THEORY FOR PREDICTING FUTURE PERFORMANCE

This review focuses on KT, thereby ignoring the only available alternative, which is Item Response Theory (IRT) [34][35]-[37]. Theoretically, IRT models differ from KT in that it focuses on summative tests in which no learning occurs, or on modeling very coarse-grained skills where the overall learning is slow [33]. This implies that IRT is a static model where student's knowledge does not change over time. Technically, IRT uses logistic models, i.e., discriminative algorithms discussed in Section II and cross-sectional data where learners' interactions directly estimate the ability parameter. An important advantage of logistic models that follows up is their ability to keep a linear algorithmic complexity while integrating a variety of features into the model; but this comes with the expense of the large-scale training data requirement. Hence, especially the more sophisticated IRT variants can be directly used for multi-skill learning and to account for variability in student a-priori abilities or guesses.

TABLE VI. APPLICATIONS OF KT MODELS

Model	IBKT	DBN	DKT	DKVMN
Appli-cation	Individualized learning pace, Personalized feedback on progress	Adaptive number of learning resources, Feedback on progress & effort minimization	Adaptive order of educational activities, Feedback on exercises progress	Adaptive order of educational activities, Feedback on concept & exercises progress
Effects	Student differences on learning rates	Multi-skill learning	Student's differences on performance Discovery of exercise relationships	Student's differences on performance Discovery of exercise relationships

The baseline IRT Rasch model, known as the One Parameter (1PL) IRT, assumes that the probability of a correct response is mathematical function of the difference between student knowledge on skill θ and an item difficulty β, as depicted in (6). The responses to items are independent and occur at constant average rate. The items are considered conditionally independent to each other and β is better estimated when there is a large amount of data to calibrate them.

$$p_i(y = 1|\theta) = \left(1 + exp\left(-(\theta - \beta_i)\right)\right)^{-1} \quad (6)$$

The most sophisticated of 1PL IRT descendants include the Additive Factor Model (AFM), which incorporates features of learning rates and skills, and its extension the

Performance Factor Analysis (PFA). The literature has already compared the models of PFA and BKT, both in theoretical [34] and in technical [35] terms *(i.e., predictive accuracy and parameter plausibility*).

AFM is depicted in (7), where $q_{ki} = 1$ if item i uses skill k, and 0 otherwise, and γ_k and T_k denote the learning rate and the number of exercises the student has solved for skill k, respectively. This model is better estimated when there is a large number of learning responses available for calibration.

$$p_i(\theta) = (1 + \exp(-(\theta + \sum q_{ki}(\beta_k + \gamma_k \cdot T_k))))^{-1} \quad (7)$$

The PFA model developed to differentiate correct from incorrect responses. It is highly predictive but not useful for adaptive environments in the sense that it cannot optimize the subset of items presented to students according to their historical performance [27]. The PFA is depicted in (8),

$$p_i(\theta) = (1 + \exp(-(\theta + \sum q_{ki}(\beta_k + \gamma_k \cdot S_k + \rho_k \cdot F_k))))^{-1} \quad (8)$$

where S_k and F_k denote the number of correctly and incorrectly solved items for a student at skill k, respectively. The fixed effects γ_k and ρ_k, therefore, denote the learning rates associated with correct and incorrect responses, respectively.

IRT models, which need to estimate simultaneously the entire interaction trajectory for each student with item parameters [37], or require large samples for calibration [33], are considered difficult to implement in an online environment; and together with KT are rarely evaluated with respect to real-time prediction performance [36].

It is interesting that the equation (2a) of IBKT incorporates the intuition of IRT, when summing the logistic functions to incorporate skill and student-specific parameters. AFM and PFA models are found [21] to achieve high resolution in Brier Score when compared to DBN because they are directly fitting a curve over time while the AFM achieve bad reliability, most probably because it does not differentiate correct from incorrect answers.

VI. Prospects And Challenges

The quality of a KT model is measured by its ability to predict learner performance. However, its key use is to recommend dynamic instructional policies like deciding sequences of learning resources, so as to guide learners towards achieving optimal learning outcomes in an efficient way. This raises five challenges and future directions.

Firstly, in case of instruction recommendations, deep learning KT models should be transparent and inform the learner about the underlying intuition of the recommended decisions. This requirement is also present as a "right to explanation" in the General Data Protection Regularization law in European countries. There is research oriented to explainable AI, such as the usage of a knowledge graph as reasoning evidence for the predictions of deep learning models [52] or the development of frameworks for interpretable machine learning models [53]. Both are still in early stages and commonly oriented to other domains than that of education.

Because of the importance of generalizing to new examples, which depends both on the right representation model and the sufficiency of data, it is useful to briefly approach KT from a data-centric side. In general, modeling knowledge acquisition is a complex task as human learning is grounded in the complexity of both the human brain and knowledge organization. From a social science perspective, learning is influenced by complex interactions, including affect [38], motivation [39][40], and even social identity [41]. Though, the data used as input for the described KT models are not complex; since predicting student knowledge with the mere observation of correct versus incorrect responses to learning activities provides weak evidence. As educational apps and smart learning environments increase in popularity, it may be possible to collect valuable, diverse and vast amounts of student learning data, able to capture the reality of learning; and hence create opportunities, as well as new challenges, in the utilization of the deeper insights of each learner's knowledge acquisition trajectory.

Therefore, the second future direction concerns the inclusion of data beyond student performances. These could be conventional patterns like hint usage, exercise skipping [54], exercise difficulty perception [55], response times, involvement in discussion forums, leverage of personalized or social comparison feedback [56], or sensory patterns of facial expression, body temperature, eye movement, and body language. A shift to deep learning modeling will offer superior results only given data more behavioral and complex. It is worthy also to note that, until now, most of KT models model hint usage and exercise skipping as an incorrect answer, which is mathematically convenient but loses information about learners' behavior.

Another example of rich data could be the inclusion of learners' input such as their diagnosis of their prior knowledge about the instructed topic or their learning intention for topic acquisition. Such features can be included as a prior to the Bayesian or as an additional input to the network. This is important for adaptive learning systems, which face difficulties on making accurate inferences and sensible recommendations, when little data about the student is available (i.e., cold-start problem in recommendation systems) or for learners who were previously inactive for a long time [36]. Obviously, such a task is not trivial and raises other questions such as what kind of questions should be asked so as to both not overwhelm the learner and at the same time gather the maximum amount of information regarding their level of knowledge.

Both probabilistic and deep learning models are not easily scalable to include richer student and skill-specific features due to 'the curse of dimensionality'. Large-scale datasets are necessary for alleviating this issue. Scalable frameworks and

incremental, parallel algorithms are open research topics in the field of AI.

Thirdly, open issues remain the automatic setting of adaptive thresholds in mastery learning and the definition of optimality in dynamic learning paths which are conditioned on continuous learning behaviors.

The fourth future direction is related to the evaluation metrics of learner models, which should be more directed at measuring performance with respect to the learning outcomes. Frameworks and metrics specifically oriented to adaptive learning purposes should thus arise. Another related challenge is that none of these models have been evaluated on online recommendation tasks.

The fifth future direction is concerned with the expert model that describes the content relationships. The Bayesian models depend on accurate domain knowledge but defining content relationship and designing exercises is not only hard to accomplish, due to their high hierarchy and diversity, but also subject to human opinions across the globe. Deep learning models do not need domain experts but this may also be considered as an extreme in the educational domain. Hence, the utilization of knowledge graphs, crowdsourcing, or semi-supervised techniques that learn the topology of the content could possibly be considered as safer paths than the either highly structured or abstract representation of skills.

VII. CONCLUSIONS

Modeling learner's skill acquisition and predicting future performance is an integral part of online adaptive learning systems that drive personalized instruction. Knowledge Tracing has the capability to infer a student's dynamic knowledge state as the learner interacts with a sequence of learning activities. In this review, we described the probabilistic and deep learning AI approaches that are used to model the evolution of knowledge acquisition. We outline their technical and educational requirements, advantages, and limitations with respect to adaptive human learning, supervised sequential machine learning, and algorithmic scalability.

The deep learning approach models a continuous learner's state for multiple skills and can explicitly induce temporal aspects related to adaptive learning without being knowledge-domain dependent. Predictions and learning recommendations can be enhanced by including more complex data. The usage of frameworks towards AI explainability is also a beneficial step. The incorporation of regularization techniques can help in overfitting issues and inconsistent predictions.

Equivalent features in the probabilistic models are the incorporation of more flexible content representations and justified assumptions about the knowledge state dynamics. A Bayesian approach models a binary state either for one or multiple skills and is highly domain-knowledge dependent, especially in the latter case. Optimization algorithms in the Bayesian models are susceptible to local optima and multiple global optima, where proper parameter initialization and constraints have shown to alleviate these issues. Importantly, the performance of probabilistic models depends highly on the setting of a good prior probability.

Specifically, the IBKT and DBN are single-purpose models; The IBKT can be used to infer individualized learning paces while the DBN can be used for multi skill learning. The latter is tractable only for simple skill topologies. Approximation inference can improve the running time of the algorithm where justified constraint sets should be defined to ensure the interpretability and accurate estimates of parameters.

The DBN together with the DKT and the DKVMN can be used for adaptive instructional policies. The DKT is the most complex and together with the DKVMN are the only models that are non-transparent and that can discover inter-skill similarities. The IBKT, DKT, and DKVMN models require a relatively larger amount of training data than the BKT and DBN models.

An open issue regarding all models is the leverage of rich, general features and their corresponding algorithmic scalability. Furthermore, the choice of evaluation metric should be chosen based on the intended use of the model that adheres to the ultimate purpose of improving learning experiences. Lastly, to the best of our knowledge, there is a research gap on whether KT models are better for the offline discovery of exercises and skills relationships rather than the online decision-making part of mastery learning or instructional policies.

REFERENCES

[1] A. Sapountzi, S. Bhulai, I. Cornelisz, and C. van Klaveren, "Dynamic Models for Knowledge Tracing and Prediction of Future Performance," IARIA, ThinkMind, In Proceedings of the 7th International Conference on Data Analytics, pp. 121-129, Nov. 2018.

[2] C. Romero and S. Ventura, "Educational Data Mining: A Review of the State of the Art," IEEE Transactions on Systems, Man, and Cybernetics Part C Applications Rev., vol. 40, no. 6, pp. 601–618, Nov. 2010.

[3] A. Essa, "A possible future for next generation adaptive learning systems," Smart Learning Environments, vol. 3, no. 1, p. 16, Dec. 2016.

[4] K. W. Fischer, U. Goswami, and J. Geake, "The Future of Educational Neuroscience," Mind, Brain, Education, vol. 4, no. 2, pp. 68–80, Jun. 2010.

[5] Z.-T. Zhu, M.-H. Yu, and P. Riezebos, "A research framework of smart education," Smart Learning Environments, vol. 3, no. 1, p. 4, Dec. 2016.

[6] S. Kontogiannis et al., "Services and high level architecture of a smart interconnected classroom," in IEEE SEEDA-CECNSM, Sep. 2018, unpublished.

[7] Z. Papamitsiou and A. Economides, "Learning Analytics and Educational Data Mining in Practice: A Systematic Literature Review of Empirical Evidence," Journal of Educational Technology & Society, International Forum of Educational Technology & Society, vol. 17, pp. 49–64, 2014.

[8] B. K. Daniel, "Big Data and data science: A critical review of issues for educational research," Review, Wiley, Br. J. Educ. Technol., Nov. 2017.

[9] K. Nadu and L. Muthu, "Application of Big Data in Education Data Mining and Learning Analytics -A Literature Review ICTACT J. Soft Computing, vol. 5, no. 4, pp. 1035–1049, Jul. 2015.

[10] C. Romero and S. Ventura, "Educational data science in massive open online courses," Wiley Interdisciplinary Review Data Mining Knowledge Discovery, vol. 7, no. 1, pp. 1-12, Jan. 2017.

[11] Z. A. Pardos, "Big data in education and the models that love them," Current Opinion in Behavioral Science, vol. 18, pp. 107–113, Dec. 2017.

[12] I. Taleb, R. Dssouli, and M. A. Serhani, "Big Data Pre-processing: A Quality Framework," in IEEE International Congress on Big Data, pp. 191–198, 2015.

[13] P. Prinsloo, E. Archer, G. Barnes, Y. Chetty, and D. Van Zyl, "Bigger data as better data in open distance learning" Review, Int. Rev. Res. Open Distributed Learning, vol. 16, no. 1, pp. 284-306, Feb. 2015.

[14] D. Gibson, "Big Data in Higher Education: Research Methods and Analytics Supporting the Learning Journey," Technology Knowledge Learning, vol. 22, no. 3, pp. 237–241, Oct. 2017.

[15] P. Domingos, "A few useful things to know about machine learning," Communication. ACM, vol. 55, no. 10, p. 78, Oct. 2012.

[16] K. Colchester, H. Hagras, D. Alghazzawi, and G. Aldabbagh, "A Survey of Artificial Intelligence Techniques Employed for Adaptive Educational Systems within E-Learning Platforms," J. Artificial Intelligence, Soft Computing, Res., vol. 7, no. 1, pp. 47–64, Jan. 2017.

[17] A. Peña-Ayala, "Educational data mining: A survey and a data mining-based analysis of recent works," Expert Systems Applications, vol. 41, no. 4, pp. 1432–1462, Mar. 2014.

[18] A. T. Corbett and J. R. Anderson, "Knowledge tracing: Modeling the acquisition of procedural knowledge," User Model User-Adapted Interactions, vol. 4, no. 4, pp. 253–278, 1995.

[19] Y. Bengio, A. Courville, and P. Vincent, "Representation Learning: A Review and New Perspectives," IEEE Transactions Pattern Analytics Machine Intelligence, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.

[20] C. M. Bishop, Pattern recognition and machine learning, Editors: M. Jordan J. Kleinberg B. Scholkopf, Springer, 2006.

[21] T. Kaser, S. Klingler, A. G. Schwing, and M. Gross, "Dynamic Bayesian Networks for Student Modeling," IEEE Transactions Learning Technologies, vol. 10, no. 4, pp. 450–462, Oct. 2017.

[22] M. Khajah, R. V. Lindsey, and M. C. Mozer, "How deep is knowledge tracing?," arXiv preprint arXiv:1604.02416, Mar. 2016.

[23] M. V. Yudelson, K. R. Koedinger, and G. J. Gordon, "Individualized Bayesian Knowledge Tracing Models," in International Conference on Artificial Intelligence in Education, pp. 171–180, 2013.

[24] Z. A. Pardos and N. T. Heffernan, "Modeling Individualization in a Bayesian Networks Implementation of Knowledge Tracing," Springer, Berlin, Heidelberg, pp. 255–266, 2010

[25] C. Piech et al., "Deep Knowledge Tracing," in Advances in Neural Information Processing Systems, NIPS, pp. 505–513, 2015.

[26] R. Pelanek, "Metrics for Evaluation of Student Models.," J. Educational Data Mining, vol. 7, no. 2, pp. 1–19, 2015.

[27] J. P. González-Brenes and Y. Huang, "Your model is predictive-but is it useful? Theoretical and Empirical Considerations of a New Paradigm for Adaptive Tutoring Evaluation," in Proceedings of the 8th International Conference on Educational Data Mining, pp. 187-194, 2015.

[28] J. E. Beck and K. Chang, "Identifiability: A Fundamental Problem of Student Modeling," In Proceedings of the 11th International Conference on User Modeling pp. 137–146, 2007.

[29] R. S. J. d. Baker, A. T. Corbett, and V. Aleven, "More Accurate Student Modeling through Contextual Estimation of Slip and Guess Probabilities in Bayesian Knowledge Tracing," in International Conference on Intelligent Tutoring Systems, pp. 406–415, 2008.

[30] Z. A. Pardos, Z. A. Pardos, and N. T. Heffernan, "Navigating the parameter space of Bayesian Knowledge Tracing models: Visualizations of the convergence of the Expectation Maximization algorithm.", In Proceedings of the 3rd International Conference on Educational Data Mining, pp. 161–170, 2010

[31] C.-K. Yeung and D.-Y. Yeung, "Addressing Two Problems in Deep Knowledge Tracing via Prediction-Consistent Regularization," In Proceedings of the 5th ACM Conference on Learning @ Scale, vol. 5, pp. 1-10, Jun. 2018.

[32] J. Gonzalez-Brenes, Y. Huang, and P. Brusilovsky, "General Features in Knowledge Tracing to Model Multiple Subskills, Temporal Item Response Theory, and Expert Knowledge," in Proceedings of the 9th International Conference on Educational Data Mining, pp. 84–91, 2014.

[33] R. Pelánek, "Applications of the Elo rating system in adaptive educational systems," Computers & Education, vol. 98, pp. 169–179, Jul. 2016.

[34] R. Pelánek, "Bayesian knowledge tracing, logistic models, and beyond: an overview of learner modeling techniques," User Model. User-adapt. Interact., vol. 27, no. 3–5, pp. 313–350, Dec. 2017.

[35] Y. Gong, J. E. Beck, and N. T. Heffernan, "Comparing Knowledge Tracing and Performance Factor Analysis by Using Multiple Model Fitting Procedures," Springer, Berlin, Heidelberg, pp. 35–44, 2010.

[36] C. Ekanadham and Y. Karklin, "T-SKIRT: Online Estimation of Student Proficiency in an Adaptive Learning System," arXiv preprint arXiv:1702.04282, 2017

[37] K. H. Wilson, Y. Karklin, B. Han, and C. Ekanadham, "Back to the basics: Bayesian extensions of IRT outperform neural networks for proficiency estimation Acknowledgements," arXiv preprint arXiv:1604.02336, 2016.

[38] E. A. Linnenbrink, P. R. Pintrich, and P. R. Pintrich, "Role of Affect in Cognitive Processing in Academic Contexts," pp. 71–102, Jul. 2004.

[39] A. J. Elliot and C. S. Dweck, Handbook of competence and motivation. Guilford Press, 2007.

[40] B. Fogg and BJ, "A behavior model for persuasive design," in Proceedings of the 4th International Conference on Persuasive Technology - Persuasive', p. 1., Sep. 2009

[41] G. L. Cohen and J. Garcia, "Identity, Belonging, and Achievement: A Model, Interventions, Implications," Current Directions in Psychological Science, Sage Publications, Inc. Association for Psychological Science, vol. 17, pp. 365–369, 2008.

[42] I. Roll, R. S. Baker, V. Aleven, B. M. Mclaren, and K. R. Koedinger, "Modeling Students' Metacognitive Errors in Two Intelligent Tutoring Systems 1 Metacognition in Intelligent Tutoring Systems." In: Proceedings of User Modeling, pp.

379–388, 2005

[43] S. Spaulding and C. Breazeal, "Affect and Inference in Bayesian Knowledge Tracing with a Robot Tutor." Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction, pp. 219-220, USA 2015

[44] M. Khajah, R. M. Wing, R. V Lindsey, and M. C. Mozer, "Incorporating Latent Factors into Knowledge Tracing to Predict Individual Differences in Learning," Proceedings of the 7th International Conference on Educational Data Mining, Educational Data Mining Society Press, pp. 99–106, 2014.

[45] J. I. E. Lee, "The Impact on Individualizing Student Models on Necessary Practice Opportunities.," Int. Educ. Data Min. Soc., In Proceedings of the 5th International Conference on Educational Data Mining, pp. 118–125, Jun. 2012

[46] Y. Bengio and Y. LeCun, "Scaling learning algorithms towards AI", Large-Scale Kernel Machines, MIT Press, 2007

[47] J. Zhang, X. Shi, I. King, and D.-Y. Yeung, "Dynamic Key Value Memory Networks for Knowledge Tracing," In WWW, vol. 2, pp. 765–774, 2017.

[48] Marcus, G. Deep learning: A critical appraisal. arXiv preprint arXiv:1801.00631, 2018.

[49] Pelánek, R., & Řihák, J.: Experimental analysis of mastery learning criteria. In: UMAP, ACM, pp. 156-163, 2017.

[50] Brusilovsky, P., & Millán, E.: User models for adaptive hypermedia and adaptive educational systems. In: The Adaptive Web: Methods and Strategies of Web Personalization, , Springer-Verlag, Vol. 4321, pp. 3-53, 2007.

[51] Desmarais, M., & Baker, R. S., "A review of recent advances in learner and skill modeling in intelligent learning environments", User Modeling and User-Adapted Interaction, vol. 22, no. 1, pp. 9-38, Apr. 2012

[52] Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard Zemel, "Gated graph sequence neural networks," In International Conference on Learning Representations (ICLR), vol. 1, pp. 1-20, 2016.

[53] Lundberg, S. M. and Lee, S.-I. A unified approach to interpreting model predictions. In Conference Neural Information Processing Systems (NIPS), pp. 4768–4777, 2017.

[54] Savi, A. O., Ruijs, N. M., Maris, G. K. J., and van der Maas, H. L. J. Delaying access to a problem-skipping option increases effortful practice: Application of an a/b test in large-scale online learning. Computers & Education, vol. 119, pp. 84– 94, 2018.

[55] Cornelisz, I. and Klaveren, C.: Student engagement with computerized practicing: Ability, task value, and difficulty perceptions. Journal of Computer Assisted Learning, vol. 34, pp. 828-842, 2018.

[56] Davis, D., Jivet, I., Kizilcec, R.F., Chen, G., Hauff, and C., Houben, G.J., "Follow the successful crowd: raising MOOC completion rates through social comparison at scale", ACM, In: Proc. of LAK'17, pp. 454–463, 2017.

Modeling, Verification and Code Generation for FPGA with Adaptive Petri Nets

Carl Mai, René Schöne, Johannes Mey, Michael Jakob, Thomas Kühn and Uwe Aßmann

Technische Universität Dresden
Dresden, Germany
Email: {carl.mai, rene.schoene, johannes.mey, michael.jakob, thomas.kuehn3, uwe.assmann} @tu-dresden.de

***Abstract*—Petri nets are a formalism used to model the behavior of systems. Modeling systems with context dependent behavior is more complex and no suitable model exists, which can be used for formal verification, graphical modeling and program synthesis. With our extension, "Adaptive Petri nets", it is possible to directly model adaptive systems while still being able to utilize their expressiveness and existing model checking tools. In this work, the utilization of Adaptive Petri nets in the context of controller synthesis for Field Programmable Gate Arrays (FPGA) is demonstrated. A full workflow from an Adaptive Petri net Model to an FPGA will evaluate the system in its usability over the three components modeling, verification and code generation.**

Keywords–Petri nets; Reconfigurable Petri nets; Inhibitor Arcs; Analysis, Exceptions, FPGA, VHDL, Code Generation

I. Introduction

With Adaptive Petri Nets (APN) a framework was developed, which allows to change the behavior of a Petri net at runtime. Parts of the net can be enabled or disabled based on the number of tokens in designated places. To integrate well in the existing tool landscape of Petri nets, APN are built in a way that they can be flattened in normal Petri nets. It was proven in our work submitted for ADAPTIVE 2018 that each APN can be flattened to a Petri net with inhibitor arcs [1].

In this work, we will show multiple ways to use APN to model and synthesize a digital controller. By this, it is evaluated how APN are utilized and integrated with existing tools.

The developed APN was designed with following goals in mind:

1) **Usability:** the APN syntax should be easy to use and should require a minimal learning effort.
2) **Flattening:** an APN should be flattened into an equivalent Petri net with inhibitor arcs.
3) **Small Overhead:** flattening should not significantly increase the net in size.

With *usability* as our first goal, we hope to avoid that APN remain only a theoretical concept without practical use. For this, we developed multiple representation methods to define an APN (graphical, mathematical, composition based). Our second goal, *flattening*, is supposed to allow the reuse of existing Petri net tools. Flattening also improves the usability since an existing Petri net based project can use APN just on top without further modifications to their infrastructure. Having *small overhead* as our goal, influences decisions of the defined semantics, so a flattened APN does not explode in size and is still usable. However, there is a trade-off between usability and small overhead.

Developing a controller on an FPGA provides various challenges for a programmer. Due to its parallel and asynchronous nature, most logical controllers require some synchronization points. These synchronization points are used, e.g., to execute routines consecutively [2] or wait for sensors and actuators [3]. While the most commonly used models for this are state machines, they can handle only one state at a time. Therefore, state machines cannot be used well in systems where the state is dependent on multiple contexts [4], [5].

When modeling a system with reconfigurable behavior, e.g., reconfigurable manufacturing systems, the system not only has to handle multiple contexts, but has to adapt its behavior to contexts [6]. State machines and Petri nets fall short in this kind of scenario, since modeling of context dependent behavior cannot be directly expressed [7]. With Adaptive Petri nets (APN) [1], we proposed a Petri net extension, which adds a syntactic, semantic, and graphical extension to Petri nets to support modeling self-adaptiveness.

The remainder of the paper is structured as follows. In Section II, the related work is reviewed. It has three focus points, the extension of Petri nets to support adaptivity and the use of Petri nets for circuit synthesis as well as the intersection of both. Section III is a background chapter and will contain the formal background of Petri nets and introduces the concept of APN. Next, in Section IV, our proposed workflow from APN to circuit is described. In the end, in Section V, we will give an example of a circuit controller, which is modeled, verified, and then synthesized into a circuit with VHDL (Very High Speed Integrated Circuit Hardware Description Language) according to our workflow. Finally, an outlook and conclusion is given.

II. Related Work

In this section, we will survey three types of related work. The first type of related work, which we present here, covers Petri nets with adaptive behavior changes. The other type of related work covers the synthesis of Petri nets to circuits or HDLs (Hardware Description Languages). And finally, we survey the related work, which is a combination of the prior types, i.e., Adaptive Petri nets synthesized to circuits or HDLs.

A. Petri nets with changing runtime behavior

While Petri nets themselves already express runtime behavior, there is no construct to express changes in runtime behavior.

It is possible to express changing runtime behavior directly within Petri nets. However, this will model the adaptivity on the same layer as the business logic, and complicates the final designs because of an intermingling of concerns.

In the following, we review existing work concerning Petri nets, which can change their behavior at runtime.

Object Petri nets [8] are Petri nets with special tokens. A token can be a Petri net itself and therefore, nets can be moved inside a main net. This type of net can be used for modeling multiple agents, which move through a net representing locations. The agents change their internal state and have different interactions based on the location inside the net. This approach extends the graphical notation of Petri nets. Analysis of object Petri nets is possible with the model checker Maude [9] and by conversion to Prolog. It was not shown that object Petri nets can be flattened to standard Petri nets, though.

Reconfiguration with graph-based approaches is a topic of Padberg's group. They developed the tool **ReConNet** [10], [11] to model and simulate reconfigurable Petri nets. A reconfiguration is described as pattern matching and replacement that are evaluated at runtime. This notation is generic and powerful, but cannot be represented in the standard notation of Petri nets. It was also not a goal to flatten them into standard Petri nets. Verification is possible with Maude.

Another graph-based reconfiguration mechanism is **net rewriting systems** (NRS) [12]. The reconfiguration happens in terms of pattern matching and replacements with dynamic composition. The expressive power was shown to be Turing equivalent by implementation of a Turing machine. Additionally, an algorithm for flattening to standard Petri nets was provided for a subset of net rewriting systems called reconfigurable nets. This subset constrains NRS to only those transformations, which leave the number of places and transitions unchanged, i.e., only the flow relation can be changed. Flattening increases the size of transitions significantly, i.e., by the number of transitions multiplied by the number of reconfigurations. With **improved net rewriting systems** [13], the NRS were applied in logic controllers. The improved version of NRS constrains the rewrite rules to not invalidate important structural properties, such as liveness, reversibility, and boundedness.

Self-modifying nets [14] were already introduced in 1978 to permit reconfiguration at runtime. Arcs between places and transitions are annotated with a weight specifying the number of tokens required inside the place until the transition becomes enabled. To achieve reconfiguration, these weights are made dynamic by linking them to a place. The number of the weight is then determined by the number of tokens inside this referenced place. This mechanism allows the enabling and disabling of arcs and therefore, can change the control flow at runtime. However, the authors state that reachability is not decidable [14].

Guan et al. [15] proposed a dynamic Petri net, which creates new structures when firing transitions. To achieve this, the net is divided in a control and a presentation net. In the control net annotations on its nodes instruct the presentation net for structural modifications. Verification and reducibility were explicitly excluded by the authors.

A practical example was shown in **Bukowiec et al.** [16], who modeled a dynamic Petri net, which could exchange parts of the net are based on configuration signals. Defining reconfigurable parts was done with a formalism of hierarchical Petri nets. The dynamic parts of the nets were modeled with subnets to generate code for a partially reconfigurable Field Programmable Gate Array (FPGA). Since this work was of more practical nature, the reconfiguration and transformation were not formalized. However, it was shown by Padberg et al. [10] that this kind of net can be transformed into a representation, which can be verified using Maude.

Dynamic Feature Petri nets (DFPN) [17] support runtime reconfiguration by annotating the Petri net elements with propositional formulas. These elements are then enabled or disabled based on the evaluation of these formulas at runtime. The formulas contain boolean variables, which can be set dynamically from transitions of the net or statically during initialization. Their model extends the graphical notation with textual annotations. It was shown that they can be flattened to standard Petri nets [18]. Compared to Adaptive Petri nets, this type of net is problem specific and has the limitation of indirection by boolean formulas. A boolean formula cannot express numbers easily, only by encoding them in multiple boolean variables. In DFPN the net is modified by firing transitions, while in Adaptive Petri nets the net is modified by the number of tokens inside a place.

With **Context-adaptive Petri nets** [19], ontologies were combined with Petri nets to model context dependent behavior in Petri nets. These nets are included in an existing Petri net editor. By this, context-adaptive Petri nets support modeling, simulation and analysis. It is unclear whether this approach would also work on larger nets, since it was not detailed how the analysis is implemented. Additionally, the flattening of these nets is not supported.

Hybrid Adaptive Petri nets [20] are a Petri net extension coming from the field of biology. These nets extend non-standard Petri nets with a special firing semantic. A transition can fire discrete, which will consume and produce a single token and then wait a specified delay for the next firing. In continuous mode a transition will not have a delay. This Petri net is adaptive by switching between those two modes. Compared to our work this is out of scope since non-standard Petri nets are used and adaptivity is restricted to transitions only.

There exist two surveys, which also summarized the related work on this topic. In the work of Gomes et al. [21], the change of behavior at runtime is classified as dynamic composition. It is characterized as "*rare*", arguing that it "*radically changes the Petri net semantics* and *complicates the available analysis techniques*". A more thorough overview of the related work can be found in Padberg et al. [7].

B. Circuit synthesis from Petri nets

Transforming Petri nets into circuits was already done in the 1970s [22] only a few years after the concept of Petri nets was published by Carl Adam Petri [23]. This already highlights the strong relationship between Petri nets and circuit design. The complete history, for Petri net synthesis into circuits, can be read in [5].

A noticeable trend since the 1970s until today, is the more abstract view on hardware. Especially with the introduction of HDLs like VHSIC (Very High Speed Integrated Circuit) Hardware Description Language (VHDL) and Verilog, but also with the wide availability of FPGA, the gap between theoretical designs and practical implementations diminished. Furthermore,

the supported net classes increased over time. While in the beginning only basic net classes were supported, nowadays exist synthesis algorithms for high level nets, too. It can be observed that the interest in Petri nets as a design aid for digital systems has increased [5]. This is attributed to two reasons. Petri nets naturally capture the relations, concurrency and conflicts of digital systems. Additionally, Petri nets are very simple but expressive and formally founded [5, p.4]. The development of FPGA can be seen as the main contributor in this field. On one hand, it can be used to rapidly prototype and test algorithms and technologies, on the other hand, it is an easier to reach target for synthesized circuits. With this, the technological stack for implementing new tools around Petri net matured in the last decade much faster than before.

Several surveys were done in this field. A very early overview was given by Agerwala in 1979, where it was mentioned as part of a survey for practical Petri net applications [24]. The survey focused on the synthesis algorithms by Dennis' group in the 70s [22]. For Finite State Machine (FSM) an in-depth survey was done by Moore and Gupta [25]. Not only use-cases and approaches were surveyed, also Petri net types and analysis methods. A more practical article on FSM implementation in Verilog was written by [26]. In 1998, two more surveys were published, the survey from Yakovlev and Koelmans [4] and from Marranghello [5] concerning asynchronous and synchronous synthesis of embedded controller, respectively. After that only very small surveying was done, as part of related work in [27], [28], [29], [30].

Petri net synthesis can be classified into three general classes: type of implementation, type of encoding and type of Petri net.

The separation between **synchronous and asynchronous implementation** is the already the focus of two surveys from 1998 [5], [4]. The decision is largely dependent on the use-case. The **type of encoding** is well elaborated in [5] and similarly in [4]. There exist three different types. *Direct encoding* also called *one-hot encoding* or *isomorphic places encoding* [31], is the 1:1 mapping of Petri net places into a circuit element (e.g., a flip-flop). This encoding guarantees a circuit and has the shortest synthesis time, as there are no complex calculations involved. A disadvantage is the higher number of flip-flops required. To tackle the problem, *logical encoding* gives each state or transition in a (sequential) Petri net a code and represent this state in the circuit by complex logic. Depending on the encoding, this is either named *place-based* or *transition-based* encoding. The state-space explosion problem [32] might result in a failed synthesis. To mitigate this problem, the net can be partitioned in multiple subnets with macro nets [33] or by using Binary Decision Diagrams (BDD) for a more efficient representation [34]. The last encoding method is by building a specialized hardware, which takes a Petri net and computes the firing. This solution is the most space efficient for large nets and allows the highest grade of reconfiguration. Disadvantages are their higher initial effort and slower execution speed [27], [35].

Overlooked in the previous taxonomies, the **Petri net type** is also a distinctive characterization. Most implementation work on safe Petri nets, some on k-bounded and some with colored tokens. Additionally, often the non-deterministic feature of Petri nets is not supported when synthesizing to circuits or needs special care [36].

C. Circuit synthesis from Petri nets with changing runtime behavior

While the synthesis of Petri nets into circuits is researched for a long time, only in recent years the research focuses also on Petri nets with changing runtime behavior. The first work, looking at this topic is [16]. Here, a non-formal Petri net model, which allows reconfiguration at runtime, is synthesized into VHDL for a partial reconfigurable FPGA. Relatively similar is [37], in which a state machine is synthesized for a partial reconfigurable FPGA. Both approaches switch the runtime behavior based on the context. The use-cases are based on an industrial and smart home scenario, respectively.

Similar research is also performed outside of FPGA synthesis. In [38] the ReConNet of Padberg et al. [10] is utilized to synthesize a reconfigurable manufacturing system (RMS). Here, a formal approach was used to model the system, verify the Petri net properties and then synthesize the RMS.

III. Preliminaries

This section defines the preliminaries, used in this work. The mathematical notation of Petri nets is explained in this section together with some properties, which can be verified with model checking. The concept of Adaptive Petri nets is explained together with an algorithm to flatten APN to Petri nets with inhibitor arcs.

A. Petri net definitions

Definition 1: A **Petri net** [32] is a directed, bipartite graph and can be defined as a tuple $\Sigma = (P, T, F, W, M_0)$. The two sets of nodes are P for places and T for transitions, where $P \cap T = \emptyset$ and $P \cup T \neq \emptyset$. F is a set of arcs, describing the flow relation with $F \subseteq (P \times T) \cup (T \times P)$. $W : F \to \mathbb{N}$ is a weight function. $M_0 : P \to \mathbb{N}$ is the start marking.

Referencing an element of the tuple is done in dot notation: for a Petri net Σ, we reference the places P by $\Sigma.P$.

Definition 2: For an element $x \in P \cup T$, $\bullet x = \{y | (y, x) \in F\}$ and $x\bullet = \{y | (x, y) \in F\}$.

E.g., $t\bullet$ with $t \in T$ refers to the set of places, which are connected with an arc originating from t. We call those preset and postset, respectively.

Definition 3: A **marking** is defined as a function $M : P \to \mathbb{N}$.

A Petri net is a static model, in which only the marking changes. M_0 is the start marking. After firing a transition, the marking changes.

Definition 4: A transition $t \in T$ is **enabled** if all places $p \in \bullet t$ have a marking of at least $W(p, t)$ tokens, where $W(p, t)$ is the weight for the arc between p and t.

Definition 5: Iff a transition t is enabled, it can **fire** and the marking of each $p \in t\bullet$ is incremented by $W(t, p)$ and the marking of each $p \in \bullet t$ is decremented by $W(p, t)$.

Definition 6: If there exists a $k \in \mathbb{N}$ for a $p \in P$ such that, starting from M_0, every reachable marking $M(p) \leq k$, we speak of p as **k-bounded**.

A bounded place never contains more than k tokens. If k equals 1, this place is called **safe**.

B. Inhibitor arcs

To model the negation inside Petri nets, e.g., "fire this transition only when less than x tokens are inside this place", inhibitor arcs can be used. With inhibitor arcs, the flow relation of Petri nets is extended with an arc, which disables a transition when the connected place has more than a specified number of tokens in it. A Petri net with inhibitor arcs can implement a Turing machine [39], while this is not possible with standard Petri nets. Because of the change of expressiveness, the available tools for model checking are reduced, for example, the halting problem cannot be solved in general for Turing complete languages.

Definition 7: An **Inhibitor Petri net** is a tuple $\Sigma = (P, T, F, I, W_I, W, M_0)$. With the same defintions as previously mentioned. Additionally this Petri net contains the set of **inhibitor arcs** $I : (P \times T)$ and a weight $W_I : I \to \mathbb{N}$

To simplify notation, we define the inhibiting set of a transition t as $\circ t = \{(p, t) \in I\}$.

Definition 8: A transition t is **enabled**$_i$, iff all places connected by an inhibitor arc are below the weight $M(p) < W_I(p, t)$ for all $p \in \circ t$ and the transition is enabled as defined in Def. 4.

1) Flattening to a Petri net without inhibitor arcs: In general, a Petri net with inhibitor arcs is Turing complete. When a place with an inhibitor arc is bounded, the inhibitor arc can be replaced with a semantic preserving structure without an inhibitor arc [40].

C. Graphical notation

Places are drawn as circles: ◯, their marking is drawn as black dots ⊙. Transitions are drawn as black rectangles (horizontal or vertical) ▮. The flow relation is drawn with directed arcs between places and transitions ⟶. Inhibitor arcs are only drawn from places to transitions and get a circle head: —○.

D. Properties and analysis of Petri nets

Petri nets support various ways to verify its properties. The most commonly used analysis techniques check for reachability, boundedness, deadlocks and liveness [32]. With these properties, it is possible to verify the correctness of the model according to its specifications. In this section we will first describe these properties and then two tools used for analysis.

The basis for most model checking techniques in Petri nets is **reachability**. This technique answers the question, whether there exists a firing sequence to get from a marking M_1 to M_2. There exists also the sub-marking reachability, which ignores the marking of some places [32]. Besides for Petri nets with inhibitor arcs, the reachability is decidable but requires exponential space and time [41].

Boundedness is used to determine, whether the marking of a particular place is such that the number of tokens is always lower than a k with $k \in \mathbb{N}$ (see Def. 6). Boundedness is very important for synthesis of Petri nets to guarantee that no buffer will overflow.

The property **liveness** refers to a Petri net, in which, starting with any marking M_0, there exists a firing sequence such that all transitions can be fired. This is a very strong property, which can be checked on five different levels (L0-L4), where each level adds some relaxation [32].

A **deadlock** in the context of Petri net refers to a marking, in which no transition can fire [32].

For analyzing Petri nets, several tools exist. Here, shortly two tools are explained. For low level analysis and especially for checking reachability, we chose LoLA (Low Level Analyzer) [42], [43] as it is multiple times the winner in the Petri net model checking contest in the category of reachability [44]. To check for reachability, LoLA accepts formulas in either temporal logic (either linear temporal logic (LTL) or computation tree logic* (CTL*)). LoLA automatically uses the fastest temporal logic for a given query, therefore, we will draw no distinction here. LTL and CTL* both build on top of the propositional calculus and extends it with temporal quantors. Such quantors are X for next state, G for global state, F for finally (e.g., a state will be reached in a finite number of steps). One of the design goals of LoLA is to keep the architecture relatively clean. That is why, only the basic type of Petri nets with no inhibitor arcs or colored tokens is supported.

The second tool, regarded in this work is Tina [45]. This tool is much more high level than LoLA. It comes bundled with a graphical editor and simulator, it can convert many different Petri net formats and has an interpreter, which can check for most basic properties like liveness, deadlocks and boundedness. However, the interpreter is much slower than LoLA.

E. Adaptive Petri nets

Adaptive Petri nets (APN) extend Petri nets with a concept to change the behavior of the net at runtime. This is done by defining one or more *configuration points*, which in turn consist of a set of nodes, which are configured and a place (*configuration place*) together with a marking, which enables or disables the set of nodes. When the set of nodes is enabled, the behavior of the Petri net is not changed. When the set of nodes is disabled, no new tokens can be emitted from outside into the set of nodes.

Following this informal description, the definition and semantics is given here.

Definition 9: An APN is a tuple $\Sigma = (P, T, F, W, M_0, C)$, based on Petri nets of Def. 1, with $C = \{c_1, c_2, \ldots\}$ as the set of configuration points.

Definition 10: A **configuration point** is a tuple $c = (p, w, N, E)$ referencing the nodes of a containing Petri net Σ.

- $p \in \Sigma.P$, a place that we will call *configuration place*.
- $w : \mathbb{Z} \setminus \{0\}$, a weight
- $N \subseteq (\Sigma.P \cup \Sigma.T)$, the nodes that are configured
- $E \subseteq N$ the external nodes of the configured net, which are reachable, even if the configured net is disabled

Definition 11: The set of **external nodes** ($E \subseteq N$) are nodes of N which are connected to nodes outside of N. Usually defined like this - but a custom definition is possible, too: $E = N \cap \{x | (x \in \bullet n \cup x \in n\bullet) \forall n \in ((P \cup T) \setminus N)\}$

Definition 12: The set of **internal nodes** for a configuration point is calculated by $G = N \setminus E$.

With these definitions, the structural part of APN is described. — In the next definitions, the runtime semantics of APN are described.

Definition 13: A configuration point $c \in C$ is **enabled**, iff $(c.w > 0 \wedge M(c.p) \geq c.w) \vee (c.w < 0 \wedge M(c.p) < |c.w|)$.

Algorithm 1 Flattening of an Adaptive Petri net

```
1: procedure FLATTEN((P, T, F, W, M, C, I))
2:     for ∀c ∈ C do
3:         for ∀p ∈ c.E ∩ P do
4:             for ∀t ∈ p• ∩ c.G do
5:                 ConnectByArc((⊤, c, t, F, I, W))
6:             end for
7:         end for
8:         for ∀t ∈ c.E ∩ T do
9:             if (t• ∩ c.N ≠ ∅) ∨ (•t ∩ c.E ≠ ∅) then
10:                t_2 ← Duplicate(t, P, T, F, W, C, I, W_I)
11:                F ← F \ ((t_2 × c.N) ∪ (c.E × t_2))
12:                ConnectByArc((⊤, c, t, F, W, I, W_I))
13:                ConnectByArc((⊥, c, t_2, F, W, I, W_I))
14:            end if
15:        end for
16:        C ← C \ {c}
17:    end for
18: end procedure
```

With *M* being the marking function of Def. 3. As a shorthand, the set of enabled configuration points is defined as $C_e \subseteq C$.

An enabled APN is not changing the behavior of the Petri net. A disabled APN stops the flow of tokens from E to N. By this, the definition of fire Def. 5 must be modified as well as the definition of enabling Def. 4. These modifications are defined in Defs. 16 and 17, respectively.

Definition 14:
- The set of configuration points a node belongs to is defined by the function $B^N : (P \cup T) \rightarrow \mathbb{P}(C)$ with $B^N(n) = \{c | c \in C \wedge n \in c.N\}$.
- The set of configuration points, in which a node is external, is defined by the function: $B^E : (P \cup T) \rightarrow \mathbb{P}(C)$ with $B^E(n) = \{c | c \in C \wedge n \in c.E\}$.
- The set of configuration points, in which a node is internal, is defined by the function: $B^G : (P \cup T) \rightarrow \mathbb{P}(C)$ with $B^G(n) = \{c | c \in C \wedge n \in c.G\}$.

Definition 15: The *configured postset* and *configured preset* of a transition t is defined as $t\bullet_c = t\bullet \setminus \{p | c \in (B^E(t) \setminus C_e) \wedge p \in c.N\}$ and $\bullet_c t = \bullet t \setminus \{p | c \in (B^E(t) \setminus C_e) \wedge p \in c.E\}$, respectively.

Definition 16: Iff a transition t with $B^E(t) \neq \emptyset$ is enabled, it can **fire**$_a$ and the marking of each $p \in t\bullet_c$ is incremented by $W(t,p)$ and the marking of each $p \in \bullet_c t$ is decremented by $W(p,t)$. The fire semantics of all other transitions are following Def. 5.

Definition 17: A transition $t \in T$ is **enabled**$_a$, iff it is enabled according to Def. 4 and the following condition holds true $\{p | p \in \bullet t \wedge p \in c.E;\ \forall c \in (B^G(t) \setminus C_e)\} = \emptyset$.

For a disabled configuration point, the movement of tokens from E to N is prohibited in Def. 17 for transitions in N. The movement of tokens to places in N is prohibited with Def. 16.

An APN can be flattened to a Petri net with inhibitor arcs [1]. Furthermore, it was shown that in some cases no inhibitor arcs are created and that, with the algorithm of [40], an inhibitor arc from a k-bounded places can be flattened to a Petri net without inhibitor arcs. When the place is 1-bounded, the overhead is minimal with just one additional place.

Algorithm 2 Helper method to enable or disable a transition by a configuration place

```
1: procedure CONNECTBYARC((e, c, t, F, I, W, W_I))
2:     if ((c.w > 0) ∧ (e = ⊤)) ∨ ((c.w < 0) ∧ (e = ⊥)) then
3:         if (c.p, t) ∈ F ∨ (t, c.p) ∈ F then
4:             if (t, c.p) ∈ F then
5:                 F ← F ∪ {(c.p, t)}
6:                 W(c.p, t) ← |c.w|
7:                 W(t, c.p) ← |c.w| + W(c.p, t)
8:             end if
9:         else
10:            F ← F ∪ {(c.p, t), (t, c.p)}
11:            W(c.p, t) ← |c.w|
12:            W(t, c.p) ← |c.w|
13:        end if
14:    else
15:        if (c.p, t) ∈ I then
16:            if W_I(c.p, t) > |c.w| then
17:                W_I(c.p, t) ← |c.w|
18:            end if
19:        else
20:            I ← I ∪ {(c.p, t)}
21:            W_I(c.p, t) ← |c.w|
22:        end if
23:    end if
24: end procedure
```

Algorithm 3 Helper method to duplicate a transition

```
1: procedure DUPLICATE((t, P, T, F, W, C, I, W_I))
2:     T ← T ∪ {t_2} with t_2 ∉ (P ∪ T)
3:     F ← F ∪ {(t_2, p) | p ∈ P ∧ (t, p) ∈ F}
4:     F ← F ∪ {(p, t_2) | p ∈ P ∧ (p, t) ∈ F}
5:     I ← I ∪ {(p, t_2) | p ∈ P ∧ (p, t) ∈ I}
6:     W ← W ∪ {(t_2, p) | p ∈ P ∧ (t, p) ∈ W}
7:     W ← W ∪ {(p, t_2) | p ∈ P ∧ (p, t) ∈ W}
8:     W_I ← W_I ∪ {(p, t_2) | p ∈ P ∧ (p, t) ∈ W_I}
9:     for ∀c ∈ C do
10:        if t ∈ c.N then
11:            c.N ← c.N ∪ {t_2}
12:        end if
13:        if t ∈ c.E then
14:            c.E ← c.E ∪ {t_2}
15:        end if
16:    end for
17: end procedure
```

1) Multiple configuration points: When multiple configuration points are configuring a set of nodes, the intersection of internal nodes of these configuration points are only enabled, when all configuration points are enabled. Therefore, the logical operator *and* is represented with the combination of multiple configuration points.

F. Scalability of the flattening approach

We argue that one of its strengths of Adaptive Petri nets is the ability to flatten it and then utilize existing model checking tools. This will only be possible, when the flattening itself will not increase the state-space of the resulting net exponentially, such that the model checking can not work in reasonable time

for larger nets.

To clarify the three stages of flattening we perform, the type of Petri net is marked in the sub-script of the set. I.e, places of an APN are denoted as P_{APN}, places of a Petri net with inhibitor arcs are denoted as P_{inh}, and places of a Petri net without inhibitor arcs are denoted as $P_{p/t-net}$. The same syntax is also used for transitions.

The worst-case increase of places, when flattening from an APN to a Petri net with inhibitor arcs is: $|P_{APN}| \in o(|P_{inh}|)$, the number of places does not increase when flattening an APN. When flattening a safe (1-bounded) Petri net with inhibitor arcs, it is $2 \cdot |P_{inh}| \in o(|P_{p/t-net}|)$. Calculating the worst-case increase for the number of transitions might be misleading, as it hardly reflects reality since it will assume many overlapping subnets: $2^{|C|} \cdot |T_{APN}| \in o(|T_{inh}|)$. With each configuration point the number of transitions can double. When flattening a safe Petri net with inhibitor arcs, the amount of transitions does not increase.

For model checking tools, the most critical criteria to solve a net, is the size of the state-space. The state-space in turn is heavily influenced by the number of places a net contains. For a safe Petri net, the state space is in the worst case $o(2^{|P|})$.

With Adaptive Petri nets, the size of places does not increase when flattening to a Petri net with inhibitor arcs. Although, the size of transitions can increase exponentially to the number of configuration points. For scalability, the most limiting factor is the flattening of inhibitor arcs, which can result in an exponential amount of additional places and transitions. Since the semantics of the APN is just boolean (enabled or disabled), users should be able to model the net in a way, that all configuration places are 1-bounded. This will only add one additional place per configuration place. From practical experience, we never found the model checking as our limiting factor.

1) Improvements to previously published work: After the publication in [1], some improvements were found. To better compare these works, we will list the changes here.

Internal nodes of C were defined as I, which was ambiguous to the set of inhibitor arcs. Now the symbol G is used.

The set of external nodes E was previously set implicitly with a formula. Now it is part of the definition of an APN. This must be done to support commutativity in evaluation and flattening, when combining multiple configuration points over an intersecting subnet. This change can be especially noticed in Algorithm 3 and Algorithm 2.

IV. Workflow from Adaptive Petri nets to FPGA

Modeling a circuit with an FSM or Petri net has many advantages, already described in Section II. We propose an architecture, which generates valid VHDL code from an APN. The whole workflow is depicted in Figure 1, described later in this chapter, and finally evaluated with a practical example of a coffee machine in Section V.

In Figure 1, the transformation chain is depicted. It can be read from the left (input) to the right (output). Circles and ovals depict artifacts, e.g., files, while arcs and rectangles are transformations and computations.

A. Input: Petri net

The transformation chain is started with various inputs. The only mandatory input is a Petri net, named *Base PN*. This Petri net can already be an APN or contain inhibitor arcs. To help with separation of concerns, a *composition* system can be used to separate the configured nodes from the base net (we employ name-based composition and net addition rules [46], [47]. For this, a *Composition Specification* may be required to describe how the multiple parts are combined. For the compositional approach, the base net contains the core functionality, which is enhanced by several features, named *PN Feature*. This concept is similar to feature-oriented programming [48].

B. Input: Context net

The input *context net* is specified by the developer and used to separate concerns. While it is not strictly necessary for APN, we found that a separate handling of the configuration points helps when designing the nets. On one hand, it is used for separating the context information from the base net. On the other hand, it can be constructed in a way to guarantee that all places of this net are 1-bounded, simplifying the flattening of inhibitor arcs.

For the context net, three options were investigated. First a simple Petri net, which does not provide a lot of abstractions. The second investigated model is the context Petri net [49]. Context Petri nets are first described by a domain specific language (DSL), which is setting multiple contexts in relationship to each other. A relationship can be exclusion, inclusion, implication, etc. This DSL is then transformed in a Petri net, which handles the activation and deactivation based on the relationship. E.g., when a context is activated, which is in an exclusion relationship with another context, the other context is then deactivated. The third option is a modified state machine. A state machine is defined as a 4-tuple $STM = (Q, s, \Sigma, f)$. With Q as a finite set of states, s as the starting state, Σ a finite input alphabet and $f : S \times \Sigma \rightarrow S$ the state transition function. For our use-case, we also add the set of contexts C to the state machine. Each state can have a subset of C assigned to it. An example can be seen in Figure 3. Such a state machine has the advantage that it is very concise but still can be transformed to a Petri net with little overhead by transforming all states Q and events Σ into places, all state transition functions f into transitions connecting the input and output state-places correspondingly and also adding the event-place as input.

C. Composition

Utilizing a composition system gives two advantages. It enables us to use a context net in the first place. Furthermore, it can be used to simplify the definition of configuration points, when each composed net is interpreted as the set N, the *configured nodes*, while the composed nodes are the *external nodes*.

For composition, two systems were used. A rather pragmatic approach, based on node fusion [50] via name-unification. All nets that have to be composed are put into one large net. Then all places with the same name are fused together, performing an addition of their tokens and merging the input and output arcs. Similarly, this is done for transitions. As a wildcard, the "*"-character can be used at the beginning or end of a string, which then merges with all prefixes and suffixes of that string depending on the position of this character.

The other utilized system is based on net additions [46]. Net additions consist of a DSL, in which the names of the composed Petri nets are first listed, followed by a list of node fusion sets. A node fusion set is either a set of places or a set of transitions from any composed Petri net, referenced by their name and with a new name. For example, the node fusion set $(a/b/c \rightarrow d)$ is merging the nodes a, b and c to a node named d. We extended net additions, to specify a configuration place together with the marking next to the name of a Petri net, such that the Petri net becomes the set N of the configuration point [47].

D. Adaptive Petri net and flattening

After the composition step, an APN is the result. Either because the *base net* was already an APN, or because the composition added configuration points to the net. The APN is then *flattened* with the algorithm of [1] to a *Petri net with inhibitor arcs*.

E. Model checking

To utilize existing model checking tools, the inhibitor arcs can be *flattened* when the source node of this arc is bounded [51]. Model checking can be performed on user-provided rules (*Model Check: custom* with *LTL/CTL* Formulas*) and with generic rules, like deadlock detection, unreachability, unboundedness and invariants (*Model Check: generic*). These generic checks can also be used, to *Optimize* the circuit model. For example, to eliminate dead code or remove redundant places from the invariant analysis. All model checking results can be inspected by the developer to fix bugs and inconsistencies in the modeled Petri nets (*Check Results*).

F. Circuit model

The flattened APN, a Petri net with inhibitor arcs, is transformed in a *Circuit Model*. Our circuit model consists of: connections, basic gates like AND and OR, as well as a counter. Currently not implemented is the step from Adaptive Petri nets to the circuit model. It is planned to use the dynamic reconfiguration capabilities of FPGAs for this [16], [52].

```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3
4  entity place is
5      generic (max: integer := 1; def: integer := 0);
6      -- I=Increment, D=Decrement, O=Out
7      port (I: in std_logic; D: in std_logic; O: out std_logic
           ; clk: in std_logic);
8  end place;
9  architecture dataflow of place is
10     signal memory: integer range 0 to max := def;
11 begin
12     process(clk)
13     begin
14         if rising_edge(clk) then
15             if D = '1' then memory <= memory - 1;
16             elsif I = '1' then memory <= memory + 1;
17             end if;
18         end if;
19     end process;
20     O <= '1' when memory > 0 else '0';
21 end dataflow;
```

Listing 1. VHDL of a place

Here, we use the following Petri net synthesis class: (see Section II): *synchronous*, *one-hot* encoded, with *k-bounded* places, *1-bounded* arcs, *inhibitor* arcs and without indeterministic constructs. Most of these restrictions are purely for pragmatic reasons, to keep the transformation to the circuit model simple. In the future, it is planned to extend the synthesis to asynchronous circuits and to support k-bounded arcs. The most important transformations can be seen in Figure 2. The transformation is modeled closely to [53]. Each place and each transition gets a one-to-one mapping in the circuit. Inhibitor arcs are represented with the logical *not*.

G. HDL-code generation

After the circuit model was optimized, an *HDL (Hardware Description Language) Model* is generated. This model is an abstract representation of the textual VHDL code. The VHDL implementation of a place can be seen in Listing 1. From this, two HDL files are generated. One implementation file, which contains all the logic to run the Petri net and an *HDL Skeleton* is generated, which the developer can use to

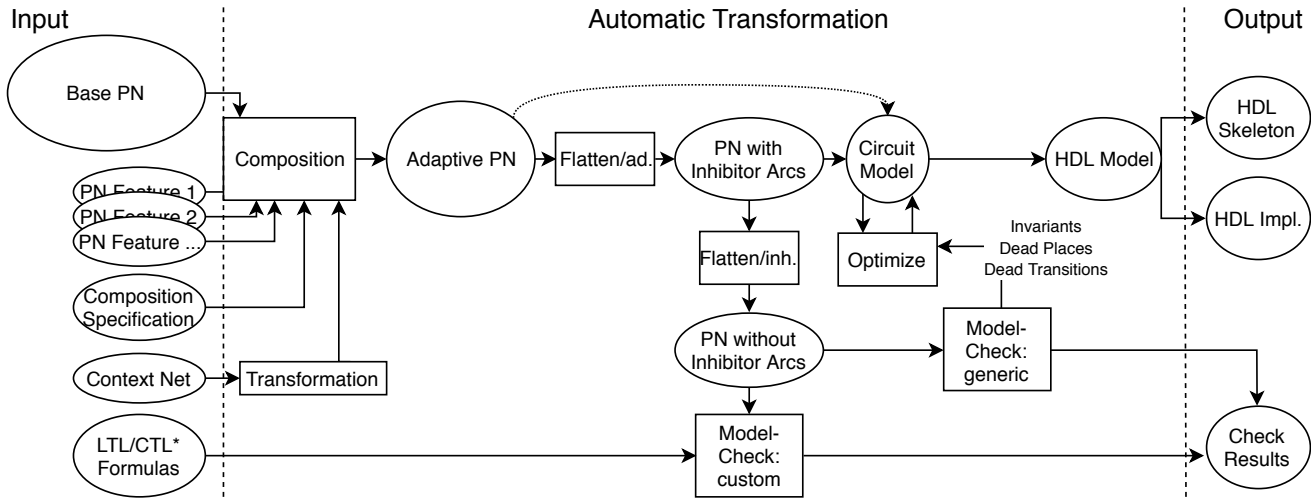


Figure 1. Transformation workflow. PN = Petri net. Ovals are artifacts, rectangles are processes.

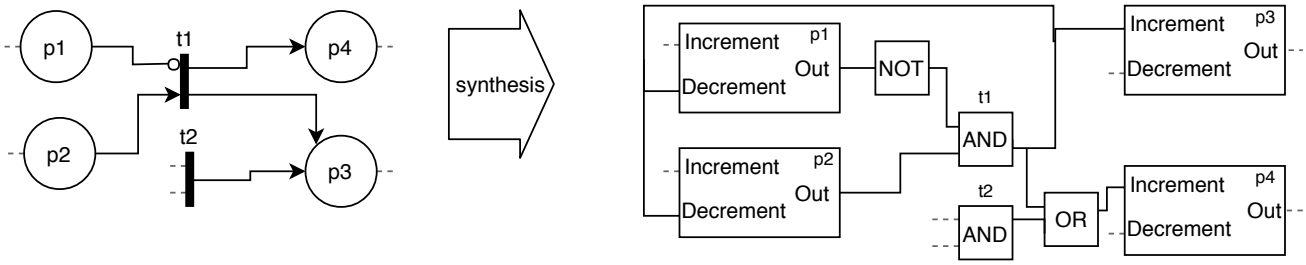


Figure 2. Petri net circuit synthesis with *one-hot encoding*

implement their functionality. The skeleton consists of the Petri net implementation and an API, exposing all important places and transitions. Unimportant nodes are those, which were created automatically. Internally this is done by prefixing the nodes with a special keyword. For implementation, transitions can be used for influencing the Petri net execution by either blocking or continuing the net-flow. Places can be used as impulses for the VHDL program to trigger the execution or directly power an actuator of the circuit, e.g., a place will start a small engine or letting an LED light blink.

V. Synthesizing an Adaptive Petri net to an FPGA

The general workflow, described in the previous section, will be demonstrated with a realistic use-case of a coffee machine. While the example is simple enough to understand, it also demonstrates most aspects of the workflow to show how the synthesis can be extended for more complex designs.

A. Use-case description: Coffee machine

The behavior of the coffee machine can be described in two phases: a *configuration phase*, which awaits user-input for the type of coffee they want and a *running phase*, where the machine will prepare and dispense the coffee. During the configuration phase, the configuration places are set. When the configuration phase is finished by pressing the start button, the runtime phase starts and executes the coffee machine adapted to the configuration.

Regarding the workflow of Figure 1, the input consists of a context net, a base Petri net and LTL/CTL* formulas. The composition specification is done implicitly, by composing the nodes with a unification of the names (nodes with the same name are merged, while a * will match anything).

The coffee machine consists of 3 models: the Petri net model, the context model, and the APN model. All three models are created separately. The APN model and Petri net model are only separated because of the current technical limitation that APN cannot be represented within PNML. The separation of the context model is not required but gives a nicer overall architecture as described also in Section IV-B.

The coffee machine itself operates in two phases: (I) beverage selection; (II) beverage dispensing. In Phase I, the customer can select from 5 buttons: Coffee, Cappuccino, Milk, Espresso, Start. Except for the start button, each selection will fill the place with the same name as the button with one token. This place is then used as the context configuration. Phase II can be reached, when the customer presses the start button. In this phase, the buttons are disabled and the machine starts dispensing according to the previous selection. The internal processes of the machine are controlled by the Petri net. Utilizing Adaptive Petri nets, only those parts are activated which are defined by the contexts. The Petri net can be seen in Figure 4.

B. Modeling

The beverage selection is done with a state machine, as it can be used to represent the selection logic in a simplified and extendable way. This state machine is modeled in our STN (state transition net) notation. It consists of states (circles), events (arcs) contexts (rectangles) and a start state. In Figure 3, the state machine can be seen.

The state machine starts in the *None* state and will move to the next state when the coffee or espresso event is triggered. It will then move to the *Coffee* or *Espresso* state, respectively. The state machine is converted into a Petri net with a simple conversion algorithm, which converts STN states to Petri net places prefixed with *state_*, events to places with the *event_* prefix and STN contexts to Petri net places without a prefix. Finally, all arcs between states are converted to a transition with the previous state and event as input and the next state as output. When transforming the example of Figure 3 into a Petri net, it results in 20 transitions and 12 places.

The Petri net model can be read from different formats. Notably PNML (Petri net Markup Language), which is a standard most Petri net tools support.

The coffee machine is modeled with the Petri net of Figure 4. This net already integrates the state machine from Section IV-B with the places *event_**, *state_None*, *Coffee*, *Espresso* and *Milk*. The net starts with a token inside place *stopped*, which allows to trigger the transitions starting with *req.*. The *-sign matches *req.Coffee*, *req.Espresso* and *req.Milk*. The net continues with the transition *ingredients*, if no token is inside any *event_** place and no token in *state_None*. This is required, so that our state machine is not in an intermediate state with unprocessed

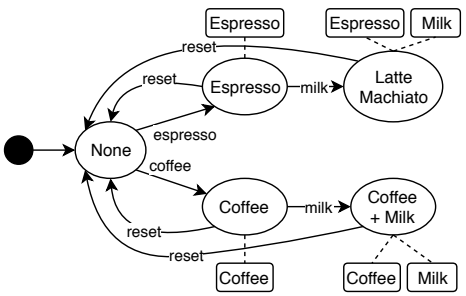


Figure 3. State machine for the selection inside the coffee machine

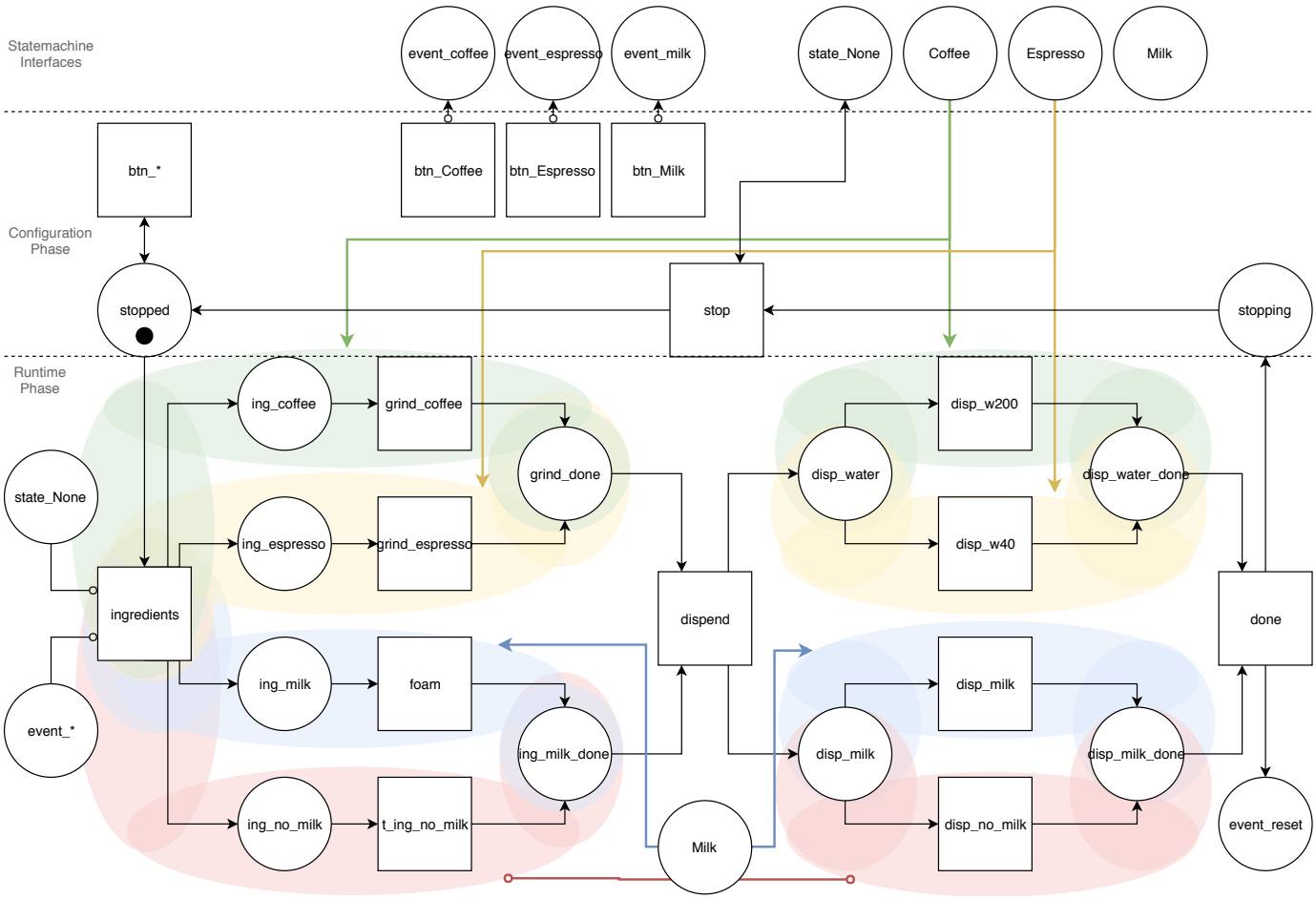


Figure 4. Petri net for the coffee machine with 4 configuration points. C_1={Coffee, 1, {ing_coffee, grind_coffee, disp_w200},{ingredients, grind_done, disp_water, disp_water_done}C_2={Espresso, 1, {ing_espresso, grind_espresso, disp_w40},{ingredients, grind_done, disp_water, disp_water_done}C_3={Milk, 1, {ing_milk, foam, disp_milk},{ingredients, ing_milk_done, disp_milk, disp_milk_done}C_4={Milk, -1, {ing_no_milk, t_ing_no_milk, disp_no_milk},{ingredients, ing_milk_done, disp_milk, disp_milk_done}

events and is also not in the *None*-state. After that, a token is put into all following places, representing ingredients for coffee, espresso and milk. The places and transitions will converge into the *dispend* transition in the middle of the figure. The ingredients will be later annotated with APN-structures. Similarly the subnet between *dispend* and *done* dispenses water and milk according to the specification and configured by the APN-structures. The coffee making process is finishing with the *done*-transition, which creates a token in *event_reset* to reset the state machine on the *None*-state and a token inside *Stopping*. The initial *stopped* state is reached, when the *stop*-transition fires, which only happens when a token is inside *state_None*.

C. Flattening

The resulting composed Adaptive Petri net consists of 35 transitions and 24 places with 4 configuration points. When this net is then flattened to a Petri net with inhibitor arcs, the size increases to 50 transitions and 24 places. The number of transitions increases a lot, because the transition *ingredients* is an incoming external node in four configuration points. This requires to duplicate this transition 2^4 times. However, the flattening algorithm is not yet optimizing the duplication. It will not prune illegal configurations (e.g., espresso and coffee can never be selected simultaneously).

When size is critical, the designer should watch out that the incoming external nodes are not transitions like it is done with the configuration points at the *disp* step (i.e., disp_water and disp_milk). Here, no new transitions or places are added to the net.

After the APN is flattened, it is a Petri net with inhibitor arcs. This can be further flattened to remove all inhibitor arcs with the algorithm of [40]. As prerequisite for flattening inhibitor arcs, the place connected to the inhibitor arc must have a known, finite bound. We know from all places that they are 1-bounded because they are the result of our state machine. After flattening, the net contains 58 transitions and 32 places, an increase of 8 transitions and 8 places.

D. Model checking

A flattened Petri net can be model checked. We utilize Tina, which we chose because it has good support for PNML, can convert it to other formats, has a graphical editor, and checks the net for basic properties in a well readable format. Additionally, we utilize LoLA (Low Level Analyzer), which is winner in several model checking competitions and allows to build complex LTL/CTL* formulas [43]. There are two kinds

of checks performed: automatically generated and manual tests. We will not describe all tests, but instead give two examples of each category. For automatically generated tests, we classify theses tests into those for user feedback and those for net optimization. Checks for User feedback is testing the net for reversibility, boundedness, and deadlock freeness. Those are all checked by default in Tina. In LoLA, the deadlocks are checked by *EF DEADLOCK*. Checks for optimizations are searching for invariants. In LoLA such a check would look like this: *AG((A = 1 AND (B = 1)) OR (NOT(A = 1) AND NOT(B = 1)))* with *A* and *B* being places. The quantifier *AG* modifies the temporal predicate that this formula is only true, if all states within the state-graph of the net conform to this rule.

The coffee machine net has 4 invariants, which are all inside the state machine, e.g., Coffee = State_Coffee OR State_Coffee_Milk. With an invariant, not every place needs to be represented with a memory, but can be represented with a logical expression instead.

Besides the automatic formulas, the user can also specify manually what is of interest to him, which requires domain knowledge. In the following, two manually specified rules:

- LoLA rule: *AGEF(Coffee = 1 AND Running = 1 AND AF(Grind_Coffee = 1))* — when Coffee is selected, the grind_coffee place is always selected afterwards.
- LoLA rule: *AGEF(Coffee = 1 AND Milk = 0 AND Running = 1 AND NOT AF(Milk_Heating = 1))* — when Coffee is selected, Milk is always deselected, place Running contains a token, and we will not reach the place Milk_Heating.

E. Generation of VHDL code

Based on the flattened Adaptive Petri net, the workflow will also create a coarse circuit model. This circuit model is generated with the transformation described in Figure 2. Here, each place is transformed into a counter and each transition in a logical AND. Currently, only synchronous circuits are generated, but there exist implementations, which do not need a clock and therefore, work asynchroneously. This coarse circuit model is then optimized to minimize the number of connections and gates. Furthermore, the optimization step receives input from the model checker, to remove invariants and dead nodes.

From the coarse circuit model, an abstract representation of the VHDL code is generated, which is transformed to actual source code in a last step. The source code is divided in two parts: an implementation part, which contains all the logic to run the Petri net and a skeleton, which contains the interface places and transitions as signal declarations. The skeleton can be later utilized by the programmer to implement additional logic. The resulting skeleton is 150 lines (3 lines for each place and 2 lines for each transition). The resulting Petri net implementation code has a length of 280 lines. Within the skeleton, the engineer has write access to the setter of all places, read access to the boolean output of all places, and write access to all transitions, where a low-signal can stop the transition from firing.

A small example of both files is given in Listing 2 and Listing 3, which consists of a single place connected to a transition. While the implementation itself must not be understood by the developer, it is still printed in a readable format. The skeleton must only be changed beginning on Line 20 for runtime behavior.

```
1  library IEEE;
2  use std.textio.all;
3  use IEEE.STD_LOGIC_1164.ALL;
4  entity main is
5      PORT(
6          -- custom ports go here (i.e. I/O)
7          btnL : in std_logic; -- button left
8          btnU : in std_logic; -- button up
9          led : out std_logic_vector(0 to 15); -- 16 leds
10         sw : in std_logic_vector(0 to 15); -- 16 switches
11         clk : in std_logic;
12     );
13 end main;
14 architecture behavior of main is
15     signal clk :std_logic := '0'; -- in
16     signal ps_ing_coffee :std_logic := '0'; -- in
17     signal t_grind_coffee :std_logic := '1'; -- in
18     signal p_ing_coffee :std_logic; -- out
19
20 begin
21     -- instantiation of entities
22     testbench : entity work.testbench port map(clk => clk
23         , ps_ing_coffee => ps_ing_coffee
24         , t_grind_coffee => t_grind_coffee
25         , p_ing_coffee => p_ing_coffee);
26     -- connection of entities by their ports
27     -- custom code here
28     t_start_transition <= '1' when (btnC = '1' and sw(14) =
           '0') else '0';
29     t_reqd_Milk <= '1' when (btnL = '1' and sw(14) = '0')
           else '0';
30     -- 3 further transitions are bound to a button
31
32     led(1)  <= (sw(1) and p_Milk_Heating) or (sw(0) and
           p_Stopped);
33     led(0)  <= (sw(1) and p_Preparing_milk_heating_out) or (
           sw(0) and p_Stopping);
34     -- 13 further places are bound to an LED + Switch
35 end;
```

Listing 2. VHDL skeleton code for place ing_coffee connected to transition grind_coffee

```
1  library IEEE;
2  use std.textio.all;
3  use IEEE.STD_LOGIC_1164.ALL;
4  entity testbench is
5      PORT(
6          clk : in std_logic := '0';
7          p_ing_coffee : out std_logic := '0';
8          ps_ing_coffee : in std_logic := '0';
9          t_grind_coffee : in std_logic := '1'     );
10 end testbench;
11 architecture behavior of testbench is
12     signal ing_coffeeir :std_logic;
13     signal ing_coffeeo2 :std_logic;
14 begin
15     ing_coffee : entity work.place generic map(1, 0) port
           map(ps_ing_coffee, ing_coffeeir, ing_coffeeo2, clk)
           ;
16     ing_coffeeir <= (ing_coffeeo2 and t_grind_coffee); --
           grind_coffee
17     p_ing_coffee <= ing_coffeeo2;
18 end;
```

Listing 3. VHDL (internal) implementation code for place ing_coffee connected to transition grind_coffee

Finally, in our test-setup we utilized Vivado-SDK-2016.2 to synthesize the bitstream for the Basys3 Artix-7 FPGA, which contains 33,280 logic cells in 5200 slices, with each slice containing four 6-input LUTs and 8 flip-flops. With this setup, the size-impact of the Petri net can be described as marginal, as can be seen in Figure 5.

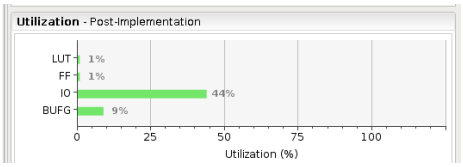


Figure 5. Resource utilization of the Petri net on a Basys 3 Artix-7 FPGA. The high I/O usage is due to our test-setup. The only required I/O is a clock.

VI. Conclusion and Future Work

In this article, we showed how Adaptive Petri nets can be embedded in a workflow to synthesize Petri nets for context adaptive circuits. Adaptive Petri nets support a Petri net developer with a new tool, which helps to express intentions more directly and make context-awarenes a higher level language construct of Petri nets. We claim that directly expressing the adaptivity behavior of the net, allows developers to better collaborate and communicate with each other. By maintaining the ability to flatten these nets into standard Petri nets with inhibitor arcs, existing tools and model checking solutions can still be applied on this new class of nets. Because of the specific structure of APN, inhibitor arcs can be removed in most cases to extend the suitable tools and model checking capabilities even further.

Compared to the initial paper on APN, the concept was slightly improved to support commutative flattening of multiple APN configurations. Further, this article proposed a methodology of development for context adaptive FPGA-based applications. The algorithm to flatten Adaptive Petri nets to FPGA is extending the existing work of code generation from Petri nets for FPGA, not only by supporting a new class of nets, but also by supporting new kinds of composition operations and supporting the usage of statemachines as input.

The workflow, for synthesizing Petri nets to FPGA, is generic and allows an instantiation with several tools and techniques. We showed how a coffee machine model can be transformed. The coffee machine is context dependent on the user input and changes its behavior based on the selection. The transformation workflow utilizes model checking to verify the correctness through automated checks, manual checks, and to optimize the resulting circuit by eliminating dead places and transitions as well as invariants. It was shown that the resulting circuit is relatively small compared to the size of modern FPGA.

While the coffee machine was utilized here as an illustrative example, we already experimented with utilizing Adaptive Petri nets for human-aware robotic control [54], [55] by implementing the Haddadin automaton [56] as the controlling net for an Adaptive Petri net. In the future, Adaptive Petri nets should be directly synthesized on the FPGA, with partial dynamic reconfiguration, which most modern FPGA support. We are implementing further semantics for exception handling [54], which allows to set and reset the tokens inside a configuration point. Utilizing the similar runtime semantics of Adaptive Petri nets and role oriented programming languages to model and verify these languages [57].

Acknowledgment

We gratefully acknowledge support from the German Excellence Initiative via the Cluster of Excellence "Center for advancing Electronics Dresden" (cfAED).

This project has received funding from the Electronic Component Systems for European Leadership Joint Undertaking under grant agreement No 692480. This Joint Undertaking receives support from the European Union's Horizon 2020 research and innovation programme and Germany, Netherlands, Spain, Austria, Belgium, Slovakia."

References

[1] C. Mai, R. Schöne, J. Mey, T. Kühn, and U. Aßmann, "Adaptive Petri nets – a Petri net extension for reconfigurable structures," in The Tenth International Conference on Adaptive and Self-Adaptive Systems and Applications (ADAPTIVE 2018). IARIA XPS Press, 2018, pp. 15–23.

[2] J. Deepakumara, H. M. Heys, and R. Venkatesan, "FPGA implementation of MD5 hash algorithm," in Canadian Conference on Electrical and Computer Engineering 2001. Conference Proceedings (Cat. No. 01TH8555), vol. 2. IEEE, 2001, pp. 919–924.

[3] F. B. Schneider, "Implementing fault-tolerant services using the state machine approach: A tutorial," ACM Computing Surveys (CSUR), vol. 22, no. 4, 1990, pp. 299–319.

[4] A. V. Yakovlev and A. M. Koelmans, "Petri nets and digital hardware design," in Lectures on Petri Nets II: Applications. Springer, 1998, pp. 154–236.

[5] N. Marranghello, "Digital systems synthesis from Petri net descriptions," DAIMI Report Series, vol. 27, no. 530, 1998.

[6] M. Salehie and L. Tahvildari, "Self-adaptive software: Landscape and research challenges," ACM Transactions on Autonomous and Adaptive Systems (TAAS), vol. 4, no. 2, 2009, p. 14.

[7] J. Padberg and L. Kahloul, "Overview of reconfigurable Petri nets," in Graph Transformation, Specifications, and Nets. Springer, 2018, pp. 201–222.

[8] R. Valk, "Object Petri nets," in Lectures on Concurrency and Petri Nets, ser. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, 2003, pp. 819–848.

[9] S. Eker, J. Meseguer, and A. Sridharanarayanan, "The Maude LTL model checker," Electronic Notes in Theoretical Computer Science, vol. 71, 2004, pp. 162–187.

[10] J. Padberg and A. Schulz, "Model checking reconfigurable Petri nets with Maude," in Graph Transformation, ser. Lecture Notes in Computer Science. Springer, 2016, pp. 54–70.

[11] J. Padberg, "Reconfigurable Petri nets with transition priorities and inhibitor arcs," in Graph Transformation. Springer, 2015, pp. 104–120.

[12] M. Llorens and J. Oliver, "Structural and dynamic changes in concurrent systems: Reconfigurable Petri nets," IEEE Transactions on Computers, vol. 53, no. 9, 2004, pp. 1147–1158.

[13] J. Li, X. Dai, and Z. Meng, "Improved net rewriting systems-based rapid reconfiguration of Petri net logic controllers," in 31st Annual Conference of IEEE Industrial Electronics Society IECON., 2005, pp. 2284–2289.

[14] R. Valk, "Self-modifying nets, a natural extension of Petri nets," in Automata, Languages and Programming. Springer, 1978, pp. 464–476.

[15] S.-U. Guan and S.-S. Lim, "Modeling adaptable multimedia and self-modifying protocol execution," Future Generation Computer Systems, vol. 20, no. 1, 2004, pp. 123–143.

[16] A. Bukowiec and M. Doligalski, "Petri net dynamic partial reconfiguration in FPGA," in Computer Aided Systems Theory - EUROCAST, ser. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, 2013, pp. 436–443.

[17] R. Muschevici, D. Clarke, and J. Proenca, "Feature Petri nets," in Proceedings 1st International Workshop on Formal Methods in Software Product Line Engineering (FMSPLE 2010), 2010.

[18] R. Muschevici, J. Proença, and D. Clarke, "Feature nets: Behavioural modelling of software product lines," Software & Systems Modeling, vol. 15, no. 4, 2016, pp. 1181–1206.

[19] E. Serral, J. De Smedt, M. Snoeck, and J. Vanthienen, "Context-adaptive Petri nets: Supporting adaptation for the execution context," Expert Systems with Applications, vol. 42, no. 23, 2015, pp. 9307 – 9317.

[20] H. Yang, C. Lin, and Q. Li, "Hybrid simulation of biochemical systems using hybrid adaptive Petri nets," in Proceedings of the Fourth International ICST Conference on Performance Evaluation Methodologies and Tools. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2009, pp. 42:1–42:10.

[21] L. Gomes and J. P. Barros, "Structuring and composability issues in Petri nets modeling," IEEE Transactions on Industrial Informatics, vol. 1, no. 2, 2005, pp. 112–123.

[22] S. S. Patil, "Coordination of asynchronous events," Ph.D. dissertation, Massachusetts Institute of Technology, 1970.

[23] C. A. Petri, "Kommunikation mit Automaten," Ph.D. dissertation, Universität Hamburg, 1962.

[24] T. Agerwala, "Special feature: Putting Petri nets to work," Computer, vol. 12, no. 12, 1979, pp. 85–94.

[25] K. Moore and S. Gupta, "Petri net models of flexible and automated manufacturing systems: a survey," International Journal of Production Research, vol. 34, no. 11, 1996, pp. 3001–3035.

[26] C. E. Cummings, "The fundamentals of efficient synthesizable finite state machine design using nc-verilog and buildgates," in Proceedings of International Cadence Usergroup Conference, 2002, pp. 1–27.

[27] S. Chevobbe, R. David, F. Blanc, T. Collette, and O. Sentieys, "Control unit for parallel embedded system." in ReCoSoC, 2006, pp. 168–176.

[28] N. Marranghello, "A dedicated reconfigurable architecture for implementing Petri nets," in M. Adamski (Ed.) Proceedings of the 2nd IFAC International Workshop on Discrete Event Systems Design, 2004, pp. 189–193.

[29] M. Adamski and M. Wegrzyn, "Petri nets mapping into reconfigurable logic controllers," Electronics and Telecommunications Quarterly, vol. 55, 2009, pp. 157–182.

[30] J. Carmona, J. Cortadella, V. Khomenko, and A. Yakovlev, "Synthesis of asynchronous hardware from Petri nets," in Lectures on Concurrency and Petri Nets. Springer, 2004, pp. 345–401.

[31] I. Grobelna, "Control interpreted Petri nets-model checking and synthesis," in Petri Nets - Manufacturing and Computer Science, P. Pawlewski, Ed. INTECH Open Access Publisher, 2012.

[32] T. Murata, "Petri nets: Properties, analysis and applications," Proceedings of the IEEE, vol. 77, no. 4, 1989, pp. 541–580.

[33] T. Kozlowski, E. Dagless, J. Saul, M. Adamski, and J. Szajna, "Parallel controller synthesis using Petri nets," in Computers and Digital Techniques, IEE Proceedings-, vol. 142. IET, 1995, pp. 263–271.

[34] E. Pastor and J. Cortadella, "Efficient encoding schemes for symbolic analysis of Petri nets," in Proceedings of the Conference on Design, Automation and Test in Europe, ser. DATE '98. IEEE Computer Society, 1998, pp. 790–795.

[35] S. Bulach, The design and realization of a custom Petri net based programmable discrete event controller. Aachen : Shaker, 2002.

[36] L. Gomes, "On conflict resolution in Petri nets models through model structuring and composition," in INDIN'05. 2005 3rd IEEE International Conference on Industrial Informatics, 2005. IEEE, 2005, pp. 489–494.

[37] R. Wiśniewski, G. Bazydło, L. Gomes, and A. Costa, "Dynamic partial reconfiguration of concurrent control systems implemented in FPGA devices," IEEE Transactions on Industrial Informatics, vol. 13, no. 4, 2017, pp. 1734–1741.

[38] L. Kahloul, S. Bourekkache, and K. Djouani, "Designing reconfigurable manufacturing systems using reconfigurable object Petri nets," International Journal of Computer Integrated Manufacturing, vol. 29, no. 8, 2016, pp. 889–906.

[39] D. Zaitsev and Z. Li, "On simulating turing machines with inhibitor Petri nets," IEEJ Transactions on Electrical and Electronic Engineering, 2017, pp. 147–156.

[40] N. Busi, "Analysis issues in Petri nets with inhibitor arcs," Theoretical Computer Science, vol. 275, no. 1, 2002-03-28, pp. 127–177.

[41] R. Lipton, "The reachability problem requires exponential space. department of computer science," Research Report 62, Yale University, Tech. Rep., 1976.

[42] K. Schmidt, "LoLA a low level analyser," in Application and Theory of Petri Nets, ser. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, 2000, pp. 465–474.

[43] K. Wolf, "Petri net model checking with LoLA 2," in International Conference on Applications and Theory of Petri Nets and Concurrency. Springer, 2018, pp. 351–362.

[44] F. Kordon, H. Garavel, L. Hillah, E. Paviot-Adet, L. Jezequel, F. Hulin-Hubard, E. G. Amparore, M. Beccuti, B. Berthomieu, H. Evrard, P. G. Jensen, D. L. Botlan, T. Liebke, J. Meijer, J. Srba, Y. Thierry-Mieg, J. van de Pol, and K. Wolf, "MCC'2017 - the seventh model checking contest," Transactions on Petri Nets and Other Models of Concurrency (ToPNoC), vol. XIII, 2018, pp. 181–209.

[45] B. Berthomieu, P.-O. Ribet, and F. Vernadat, "The tool TINA – construction of abstract state spaces for Petri nets and time Petri nets," International Journal of Production Research, vol. 42, no. 14, 2004, pp. 2741–2756.

[46] J. P. Barros and L. Gomes, "Net model composition and modification by net operations: A pragmatic approach," in 2nd IEEE International Conference on Industrial Informatics, INDIN, 2004, pp. 309–314.

[47] M. Volkmann, "Integration von adaptiven Petrinetzen in ein Petrinetz Kompositions-system," Bachelor's thesis, Technische Universität Dresden, 2018.

[48] C. Prehofer, "Feature-oriented programming: A fresh look at objects," in European Conference on Object-Oriented Programming. Springer, 1997, pp. 419–443.

[49] N. Cardozo, J. Vallejos, S. González, K. Mens, and T. D'Hondt, "Context Petri nets: Enabling consistent composition of context-dependent behavior." PNSE, vol. 12, 2012, pp. 156–170.

[50] L. Gomes and J. P. Barros, "Structuring and composability issues in Petri nets modeling," IEEE Transactions on Industrial Informatics, vol. 1, no. 2, 2005, pp. 112–123.

[51] N. Busi and G. M. Pinna, "Synthesis of nets with inhibitor arcs," in CONCUR'97: Concurrency Theory. Springer, 1997, pp. 151–165.

[52] M. Liu, W. Kuehn, Z. Lu, and A. Jantsch, "Run-time partial reconfiguration speed investigation and architectural design space exploration," in 2009 International Conference on Field Programmable Logic and Applications. IEEE, 2009, pp. 498–502.

[53] E. Soto and M. Pereira, "Implementing a Petri net specification in a FPGA using VHDL," in Design of embedded control systems. Springer, 2005, pp. 167–174.

[54] M. Jakob, "Extending adaptive Petri nets with a concept for exception handling," Master thesis, Technische Universität Dresden, 2019.

[55] H. Schole, "Modellierung von sensitivem roboterverhalten in szenarien der mensch-roboter-interaktion auf basis von kollaborationszonen," Master thesis, Technische Universität Dresden, 2019.

[56] S. Haddadin, M. Suppa, S. Fuchs, T. Bodenmüller, A. Albu-Schäffer, and G. Hirzinger, "Towards the robotic co-worker," in Robotics Research. Springer, 2011, pp. 261–282.

[57] T. Kühn, M. Leuthäuser, S. Götz, C. Seidl, and U. Aßmann, "A metamodel family for role-based modeling and programming languages," in International Conference on Software Language Engineering. Springer, 2014, pp. 141–160.

Governing Roles and Responsibilities in a Human-Machine Decision-Making Context: A Governance Framework

Koen Smit
Digital Smart Services
HU University of Applied Sciences Utrecht
Utrecht, the Netherlands
koen.smit@hu.nl

Martijn Zoet
Optimizing Knowledge-Intensive Business Processes
Zuyd University of Applied Sciences
Sittard, the Netherlands
martijn.zoet@zuyd.nl

***Abstract*—Proper decision-making is one of the most important capabilities of an organization. Therefore, it is important to have a clear understanding and overview of the decisions an organization makes. A means to understanding and modeling decisions is the Decision Model and Notation (DMN) standard published by the Object Management Group in 2015. In this standard, it is possible to design and specify how a decision should be taken. However, DMN lacks elements to specify the actors that fulfil different roles in the decision-making process as well as not taking into account the autonomy of machines. In this paper, we re-address and - present our earlier work [1] that focuses on the construction of a framework that takes into account different roles in the decision-making process, and also includes the extent of the autonomy when machines are involved in the decision-making processes. Yet, we extended our previous research with more detailed discussion of the related literature, running cases, and results, which provides a grounded basis from which further research on the governance of (semi) automated decision-making can be conducted. The contributions of this paper are twofold; 1) a framework that combines both autonomy and separation of concerns aspects for decision-making in practice while 2) the proposed theory forms a grounded argument to enrich the current DMN standard.**

Keywords-Decision-Making; DMN; RAPID; Autonomy.

I. INTRODUCTION

In September 2015, the Object Management Group (OMG) released a new standard for modelling decisions and underlying business logic, DMN [2]. In line with the DMN standard, a decision is defined as: "*A conclusion that a business arrives at through business logic and which the business is interested in managing.*" [3]. Furthermore, business logic is defined as: "*a collection of business rules, business decision tables, or executable analytic models to make individual decisions."* [2].

Proper decision-making is one of the most important capabilities of an organization [4]. In the previous decades, decision making was a capability only executed by human actors. However, given the technical developments in computer hard- and software, the possibilities to automate decision-making increases. Examples of techniques applied during automated decision making are: business rules systems, expert systems, and neural networks [5]. To achieve proper decision-making, organizations must design and specify their decisions and decision-making processes. One aspect that influences the specification of the decision and the decision-making process is the level of automated decision-making. Machines can execute decisions only when the decision and the underlying business logic is specified formally [6]. Furthermore, when organizations choose to specify their decisions and decision-making processes, the level of detail is of importance. This is based, amongst others, on the type of decision and the actor that executes the decision. For example, a strategic decision needs to be specified on a different level of detail compared to an operational decision and therefore needs a different type of specification and a different decision-making process.

While DMN is mainly applied to express operational decisions that will be automated, it can also be used for manual (strategic) decision-making. In this paper, the focus is on operational decision-making. Yet, the current DMN standard lacks a formal concept to specify a governance structure for each decision. In this context, a governance structure is defined to express the roles and responsibilities relevant to a decision and the underlying decision-making process. This becomes important when a decision is executed by instantiating a decision-making process that features both human and machine actors. Research on specifying a proper governance structure for decision-making already concluded that assigning clear roles and responsibilities are the most important steps in the design and specification of decisions and result in better coordination and quicker response times [3][6].

Another aspect of designing and specifying decisions and decision-making is the use of machine actors instead of human actors. Assigning machine actors to parts of the decision-making process requires organizations to evaluate the autonomy of the machine. Machine autonomy refers to the system's capability to carry out its own tasks and making decisions [8]. As Parasuraman, Sheridan and Wickens [9] stated in their work, the question now is: "*which system functions should be automated, and to what extent?*" For example, when possible, do we want to let a machine decide whether a person should or should not be admitted to enter a given country, based on the premise that the machine is more

accurate compared to a human actor in determining the eligibility of a person.

One reason why it is essential to include proper governance structure when designing and specifying decisions and decision-making processes are the increasingly stricter laws and regulations on digital privacy and data regulation, i.e., the Health Insurance Portability and Accountability Act (title II) and the General Data Protection Regulation [10]. Such laws and regulations can prohibit the use of machine actors in decision-making, and when it allows organizations to include them, poses exactly what is allowed and what is not allowed. For example, how exactly personal data is processed, and which roles have access to it. Thus, to design compliant decisions and decision-making, an organization must be able to define exactly what actors are responsible for what, and when a machine is made responsible, how autonomous it will operate.

In literature, studies are conducted that resulted in a model to define, for example, the autonomy of a machine in decision-making [8][10][11]. Moreover, studies are conducted that specify the roles that are used to design decision-making processes between stakeholders [3][12]. However, to the knowledge of the authors, no studies exist that combine both. One notable industry in which roles and, to some extent, responsibilities are explored and made explicit is the medical domain. Examples are [14][15][16], and [17]. However, these are, to the knowledge of the authors, not explored in a human-machine context.

Therefore, in this paper, a model is proposed that includes the roles and responsibilities aspect, taking into account human-machine interaction, while also including the autonomy level of a machine as part of the human-machine interaction in decision-making. To be able to do so, the following research question is addressed: "*How can a governance structure of the decision making process be made explicit?"*

The remainder of this paper is organized as follows. First, a literature overview is presented in section two in which the existing models that define the possible interaction between a human and a machine are explored and compared. This is followed by the construction of the model in section three. Next, in section four, the case to demonstrate and validate the model is described, which is followed by the actual demonstration of the model. Lastly, the conclusions are drawn and we propose directions for future research in section five.

II. Background and Related Work

The DMN standard consists of two levels; the Decision Requirements Level (DRD) and the Decision Logic Level (DLL). The DRD level consists of four concepts that are used to capture essential information with regards to decisions; 1) the decision, 2) business knowledge, which represents the collection of business logic required to execute the decision, 3) input data, and 4) a knowledge source, which enforces how the decision should be taken by influencing the underlying business logic, see Figure 1. The contents of the DLL level are represented by the business knowledge container in the DRD level. In the current version of DMN, two standard languages are suggested for expressing business logic, FEEL and SFEEL. However, it also allows the use of other, more adopted languages like JavaScript, Groovy, and Python. Still, the language selected to represent the decision logic does not influence the decision requirements level. Analysis of the DMN standard reveals that no formal elements exist to specify roles in the decision-making process. To add to the DMN standard, roles and responsibilities should be taken into account.

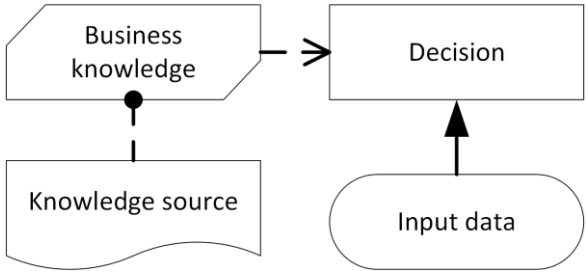


Figure 1. DRD-level elements

A. Roles and responsibilities in decision-making

In the current body of knowledge, frameworks that define roles and responsibilities in decision-making processes exist. These studies focus on different perspectives in the decision-making process. For example, there are studies that focus on the influences of decision-making roles, i.e., family/collegial pressure and gender or cultural preferences [13][14]. In addition, there are also studies that focus on specific application areas for decision-making, i.e., transportation, medical, financial and governance [15][16]. For example, in a patient-doctor context where a treatment has to be decided, multiple roles are relevant, i.e., the patient, different medical specialists, the doctor, a nurse, and in some cases family members of the patient [21].

Another research stream in decision-making comprises group-based decision-making. Group-based decision-making is explored because the context comprises multiple stakeholders that should be taken into account during decision-making, thus fulfilling roles and having responsibilities during the decision-making process. To the knowledge of the authors, a lot of contributions have been published on group-based decision-making processes, e.g., on group-based decision-making in the utility industry to determine wind farm site locations [22], trip planning as part of the transport industry [23], the allocation of primary health care services [24], group-based R&D project selection [25], and the performance of group-based decision making [26].

However, as the scope of this paper lies on the creation of a framework which can be applied to define the governance structure of any decision, a more generic set of roles and responsibilities is required.

The work of Rogers and Blenko [4] features a generic model titled RAPID, which presents five different roles that are applied during the decision-making process. However, one limitation of the original study is the focus on decisions

that are only executed by human actors. To ground our framework construction, a detailed description of the RAPID framework is provided here.

RAPID focuses on assigning a set of specific roles with regards to a decision. This framework is characterized by a simple, yet grounded in practice approach and consists of five different roles and underlying responsibilities that are related to a decision. The first role is ***Recommend***, which is responsible for making a proposal and gathering input for decision-making. This role communicates with the input role to ensure their viewpoints are embedded in the recommendation. The second role is ***Agree***, which is responsible for evaluating a proposal provided by the recommender. This role has veto power over the recommendation. When this role declines a recommendation, a modified proposal has to be made. The third role is ***Input***, which is responsible for providing input (data) to make the decision and are typically consulted on the decision. The opinion of this role is non-binding, but should be taken into account to ensure the decision does not falter during its execution. The fourth role is **Decide**, which is responsible as the formal decision maker and is accountable for the decision and its results. This role has the most authority compared to the other roles as it is able to resolve the decision-making between the previous roles by making the actual decision. By doing so, this role has the power to commit an organization to action based on decision-making. Lastly, the fifth role stands for **Perform**, which is responsible for executing the actual decision of the organization after it is decided by the previous role.

Based on RAPID, Taylor [13], in a professional article, adapted the RAPID model but made a distinction between a human and a machine for decision-making processes in which he stresses that the action component can be different between these two. For example, when a decision must be executed in an organization, human actors perform the actual decision and also handle possible exceptions. When a machine executes decisions, exceptions are filtered out and send to human actors for further examination. Another significant difference between a human and a machine actor is the explicitness of business rules that a machine must be able to execute, and therefore must be maintained adequately versus the implicit knowledge for the decision-making utilized by human actors in the actual decision-making process.

A non-generic framework which originates from the military domain is the Observe, Orient, Decide and Act (OODA) loop [27]. The OODA loop is arguably the basis for decision-making for many succeeding decision-frameworks in the military domain and also shown to influence decision-making processes and frameworks outside of the military domain as well [28][29]. OODA features four activities that represent the roles and responsibilities that should be adhered to in order to make grounded decisions in military situations. The comparison shows that RAPID and OODA show overlap in roles, e.g., decide (identical in both frameworks) and Act (OODA) versus Perform (RAPID). Multiple extensions have been proposed, based on the original OODA framework [30]. These extensions are proposed due to the fact that OODA is considered 1) a very high-level representation with abstract concepts that do not provide the kind of details needed for the OODA loop to be used as an analytical tool for improving decision-making, and 2) It has no representation of the feedback or feed-forward loops needed to effectively model dynamic decision-making [31]. The latter, however, is included in the RAPID framework.

B. Autonomy level of stakeholders in human-machine interaction

Machine autonomy broadly refers to a machine's capability to carry out its own processes and tasks, along with the decision-making needed to do so [8].

With regards to machine autonomy, also referred to as robot autonomy or computer autonomy, many authors added a framework to the body of knowledge that defines autonomy levels. Both general and context-specific frameworks for levels of autonomy (LOA) exist, while some define very detailed levels of autonomy, others utilize autonomy as a concept without exactly defining the spectrum of autonomy [32]. In this paper, the focus is on generic LOA frameworks. Regarding generic LOA frameworks, the work of Sheridan and Verplanck [33] and later Parasuraman, Sheridan and Wickers [9] defined ten levels of autonomy for decision-making with automation (i.e., machines/computers), also abbreviated to LOADAS. Their classification ranks from full human decisions and actions (level 1) until full autonomy without interaction with humans (level 10) and takes into account several variants with alternatives. For example, veto voting by human actors and the level of interaction between a machine and human actor. This LOA framework is, to the knowledge of the authors, the most popular work as it is cited numerous times and used in the construction of many other theoretical and practical constructs. However, the ten LOA levels described in the work of Parasuraman, Sheridan and Wickers [9] are too much prone to interpretation, which can be concluded by how the different authors of subsequent LOA frameworks and related work described this framework. For example, the work of Endsley and Kaber [34] describes that the first of the ten levels is not fully manual as it is handed over to the machine to execute it. This is in contrast with the interpretation and description by Miller and Parasuraman [35], which describes that a human actor is responsible for everything in the decision-making process, including the execution of the decision. A second example of an interpretation that is not specific enough with regards to this framework is the notion of levels one and two in the work of Beer, Fisk and Rogers [8], which states that these two levels are exactly the same. This would mean that the model contains a redundant level.

Endsley and Kaber [11] defined in their work ten categories of the level of automation along with definitions for the level of autonomy for each category, based on earlier work by Endsley [34]. However, the ten levels, which are all activity focused, are grounded by five levels of autonomy defined by Endsley [34], which are: 1) manual support, 2) decision support, 3) consensual AI, 4) monitored AI, and 5) full automation. This framework's strength is its simplistic

approach to autonomy, which is also its drawback. Compared to the framework of Parasuraman, Sheridan and Wickers [9], this framework lacks proper detail with regards to the possibilities a machine nowadays has. For example, based on the five levels of autonomy it is based on, it is unclear how recommendations are provided and how the human actor is informed about executing the actual decision or the result of the decision after execution by a machine.

A third generic framework is the Autonomy Levels For Unmanned Systems (ALFUS) [12]. This framework includes increasingly complex environments in which a machine makes decisions and executes actions. The LOA levels included in ALFUS, range from zero (remote control) to ten (full intelligent autonomy). At the lowest LOA, there is 100% interaction between a human and machine actor, while at the 10th LOA, almost no interaction between a human and machine actor is present. While ALFUS describes in more detail the amount of interaction between human and machine actors, the composition of this interaction is left implicit as it requires the ALFUS generic framework to be instantiated into program specific ALFUS frameworks [12].

The currently available frameworks very accurately describe what levels of autonomy could be taken into account and how the interaction is possible between human and machine actors. However, as pointed out earlier, the existing frameworks lack the exact separation of tasks and responsibilities in complex human-machine interaction environments. Therefore, in the next section, a framework is proposed that combines both the roles relevant for decision making with the different levels of autonomy possible for machines in human-machine interaction to overcome this gap.

III. GOVERNANCE FRAMEWORK CONSTRUCTION

For the construction of our framework that fills the gaps identified in the previous section, two perspectives have to be merged: detailed decision-making roles and detailed LOA's. Regarding the decision-making roles, the RAPID framework [4] is adopted due to its generic nature, thus is applicable in all contexts. Then, with regards to autonomy, the LOADAS framework [8] has been adopted due to the fact that it is utilized by many newer autonomy frameworks. However, the low level of detail and different interpretations of this framework and those that preceded LOADAS were already considered a drawback for the design and specification of decisions and decision-making as discussed in the previous section. Therefore, these theories have been analyzed to identify Situational Factors (SFs) that need to be taken into account for the construction of the governance framework. By doing so, the governance framework adopts all essential constructs from related work on the subject of autonomy. Analysis of the models resulted in five SF's. The five SFs identified from the literature are: 1) type of actor, 2) alternatives, 3) veto, 4) inform, and 5) deadline.

The first SF is the type of actor, see for example "*The **computer** informs the human only if asked*" [9]. Simply stated, when decision-making is defined, a choice has to be made whether this should be performed by a human actor only (variant one), a combination of a human and a machine actor (variant two) or solely by a machine (variant three). The second SF concerns the alternatives and the number of alternatives that are provided by a machine actor to the human actor, see for example "*The computer narrows the selection down to a few **alternatives***" [9]. This SF comprises three possible variants. The machine actor could provide a full list of possible alternatives to the human actor, offering no filtering or selection at all (variant one). In the second variant, the machine actor could provide a selected set of alternatives for evaluation by a human actor. This means that the machine actor already filtered out one or more alternatives. The amount of alternatives in this variant depends on the context of the decision-making, and therefore is not fixed compared to the first and third variant. Lastly, the machine actor could provide one alternative to the human actor, which means that the machine actor performs the complete selection for the human actor, which only has to decide whether to execute the provided alternative or not (variant three). The third SF is veto, which encompasses the time a human actor is provided by the machine actor to activate a veto over the decision-making by the machine actor, see for example "*Allows the human a restricted time to **veto**…*" [9]. The amount of time provided by the machine actor to veto depends on the context of the decision-making, which results in two possible variants, decision-making including a veto possibility regardless of the time specified to do so (variant one) or decision-making without the possibility to veto (variant two). The fourth SF comprises the interaction between the human and machine actor regarding the output of the decision-making, see for example "***Informs** the human only if the computer decides to*" [9]. This interaction could entail four possible variants. The first variant requires the machine actor to always inform the human actor with the result of the decision-making by the machine actor. The second variant requires the human actor to file a request for information about the decision-making by the machine actor. The third variant leaves the responsibility to inform the human actor about the decision-making in the hands of the machine actor, which has to decide whether it is necessary. For example, this could be determined by the machine actor based on pre-programmed or self-learned exceptions. The fourth variant is a fully autonomous state regarding decision-making by the machine actor, ignoring the human actor. The fifth SF comprises the maximum amount of time (predetermined or calculated) a role has to execute a certain activity, e.g., the input role gathering and sensing decision-making essential data, which must be completed within a timeframe of 24 hours. This SF must be considered for each step in the decision-making process as a decision can be time-critical for an organization, ranging from a product or service that need to be delivered in a normal timeframe to military [36] or High Frequency Trading (HFT) [37] contexts in which decisions need to be executed within a minute or even a second [36].

Combining the RAPID roles and the five identified SFs a framework is created that supports the detailed design for a governance structure, see Figure 3. In the governance framework, each role involved (five in total) is characterized

by five SFs in the decision-making process and should be specified accordingly.

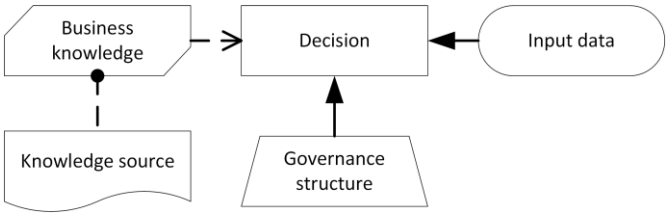


Figure 2. Governance structure to complement DMN 1.1

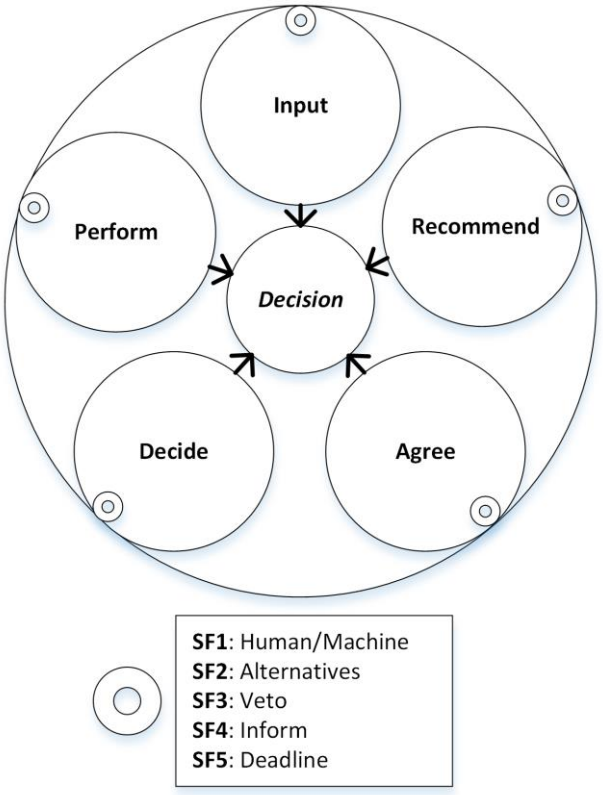


Figure 3. Governance Framework for Decision-making

Based on Figure 3, a governance structure for each decision can be taken into account. Therefore, an additional element to enrich the current DMN standard is proposed, see Figure 2.

IV. CASE DESCRIPTION & APPLICATION

The hypothesized application of the framework is demonstrated using three scenarios with three variants each, the first two variants are based on case study data, while the third variant is based upon a real-world situation, but is not an exact real-life organizational interpretation of it (simulation). In the next section, a demonstration on case study data is applied. This allows us to use data from an actual case while fully controlling the execution of the framework and input variables. The selection of the scenario's was based on three criteria: 1) the scenario must be a decision on the operational level, 2) the scenario's must significantly differ from each other in terms of industry/application, and 3) the data must be accessible for the research team.

A. Description of scenario

The first scenario used to demonstrate the framework embodies a governmental institution that is responsible for providing digital services to apply for child benefits, see Figure 4. In this scenario, civilians need to provide information for the governmental institution to be assessed whether the household is eligible to receive child benefits, and when this is the case, the amount of the child benefits and for what period the child benefits can be received. In this scenario, a citizen applies for child benefits, see for example [38].

The second scenario used to demonstrate the framework embodies trading and High-Frequency Trading (HFT). In this context, the focus lies on the decision to buy or sell stocks. To be able to do so, certain criteria need to be taken into account. When humans are involved, HFT is not possible, however, in this scenario, we cover the differences between human and machine decision making roles in the determination to buy/sell stocks, see for example [39].

The third scenario used to demonstrate the framework embodies the usage of drones by a military institution. In this scenario, a drone is utilized to determine whether a target should be terminated or not. This scenario progresses from the full control by human actors towards fully autonomous control by the drone itself, see for example [40].

B. Application of the model

The application of the framework is demonstrated by using three variants per scenario. Each of the variants is characterized by a different composition of roles and corresponding SFs. In the context of this demonstration, three steps are required before the framework can be demonstrated; 1) the decision has to be modelled in DMN. In this context, this means that the DRD for this particular decision has to be established (the decision, its input data, its ruleset and relevant sources), see Figure 1. 2) The governance structure element has to be added to the DRD, connected to the appropriate decision, see Figure 2. Lastly, 3) The roles and SFs need to be specified. An example template to do so is presented in Table I.

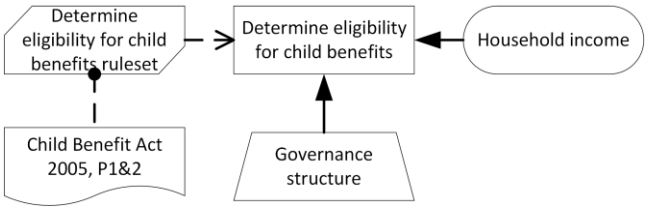


Figure 4. DRD for the decision: determine eligibility for child benefits

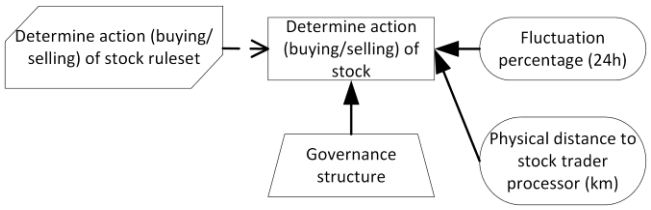


Figure 5. DRD for the decision: Determine action (buying/selling) of stock

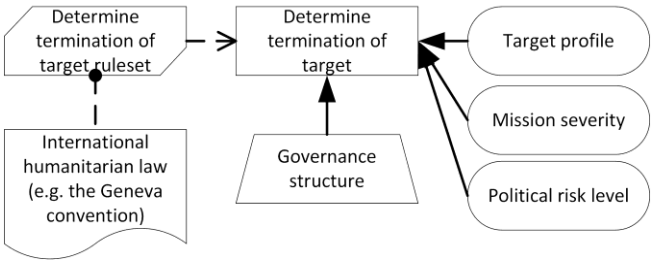


Figure 6. DRD for the decision: Determine termination of target

To demonstrate the usefulness of this template, the governance structure for the scenario in this demonstration is also specified in Table I. For each variant, the design is changed and depicted in a new table.

Scenario 1: Governmental service context

Variant 1: Manual human decision-making

TABLE I. GOVERNANCE STRUCTURE FOR VARIANT ONE

	SF1:	SF2:	SF3:	SF4:	SF5:
I	Human (applicant)	N.A.	N.A.	Always	N.A.
R	Human (template)	N.A.	N.A.	Never	N.A.
A	Human (manager)	N.A.	N.A.	Never	7 days
D	Human (employee)	N.A.	N.A.	Always	7 days
P	Human (employee)	N.A.	N.A.	Always	14 days

In the first variant, the applicant fills in a paper template and delivers it to the governmental counter (**Input**). Then, the governmental employee assesses the situation by analyzing the information in the template (**Recommend**) and decides for which benefits the household is eligible (**Decide**) based on a discussion about the case with the manager (**Agree**). In practice, it can be the case that one actor fulfils multiple decision-making roles. When the decision is made, the governmental employee enters the outcome into the governmental system (**Perform**). This allows the applicant to, on a monthly basis, pick up the appointed benefits at the governmental counter. Lastly, the applicant is informed by letter regarding the outcome of the decision and is able to make an appeal within two weeks.

The template used contains information about the different benefits available and thus guides the decision-making for both the input and decide roles.

Variant 2: Machine-supported decision-making

In this variant, see Table II, the applicant fills in an application template and uploads it to the online governmental portal (**Input**). Then, the governmental employee receives a notification of the system, which also provides a suggestion (**Recommend**) with regards to the eligibility of the application. The governmental employee decides (**Decide)** based on a discussion about the case with the manager **(Agree**), taking into account the suggestion of the system. Next, the system notifies the applicant and transfers the benefits automatically once a month (**Perform**).

In this variant, the machine generates a suggestion and is provided with the result of the decision as it needs to apply machine-learning to increase and maintain the accuracy of suggestions.

TABLE II. GOVERNANCE STRUCTURE FOR VARIANT TWO

	SF1:	SF2:	SF3:	SF4:	SF5:
I	Human (applicant)	N.A.	None	Always	N.A.
R	Machine (system)	One	None	Always	10 seconds
A	Human (manager)	N.A.	N.A.	Always	7 days
D	Human (employee)	N.A.	N.A.	Never	7 days
P	Machine (system)	N.A.	None	On request	1 day

Variant 3: Autonomous decision-making

TABLE III. GOVERNANCE STRUCTURE FOR VARIANT THREE

	SF1:	SF2:	SF3:	SF4:	SF5:
I	Machine (system)	None	None	Always	364 days
R	Machine (system)	None	None	Never	1 day
A	Human (citizen)	None	30 days	Always	30 days
D	Machine (system)	None	None	On request	1 day
P	Machine (system)	None	None	Always	1 day

In this variant, see Table III, the citizen's data (all digitally available) is evaluated on a yearly basis by a machine to determine the eligibility for benefits (**Input**). Based on this, the citizen is informed about the pre-filled applications and is able to veto the data in the pre-filled applications or veto the eligibility in general. For this

example, the time to veto is one month (**Agree**). When no veto is cast by the citizen, the system decides to process the relevant benefits (**Recommend & Decide**) and the benefits are automatically transferred once a month (**Perform**).

In the last variant, the citizen is informed about his/her pre-filled and analyzed data on top of the actual confirmation after the benefits are approved after no veto has been cast by the citizen.

Scenario 2: (High-Frequency-)Trading context

Variant 1: Manual human decision-making

In the first variant, see Table IV, the stock trader collects information by conducting a technical and financial analysis of corporate performance, which is used in determining whether to buy or sell stocks per given portfolio (**Input**). Then, based on a holistic overview of information, experience, and gut feeling, the stock trader aims to buy a large amount of stock of an organization, for which the stock trader contacts the financial office he or she works for to provide a recommendation (**Recommend**). This is required because the financial organizations' policy states that very large buy orders should be verified by a second opinion (human) before being processed (**Agree**). Based on the collected input of the stock trader and the received agree on the decision, the stock trader processes the buy order, but with a reduced order amount of 25% (**Decide**). This is followed by a verification of the stock exchange against certain financial rules of conduct. When the result of this verification process is positive, the stock trader processed the buy order into the system of the financial organization he or she works for (**Perform**).

TABLE IV. GOVERNANCE STRUCTURE FOR VARIANT ONE

	SF1:	**SF2:**	**SF3:**	**SF4:**	**SF5:**
I	Human (Trader)	N.A.	N.A.	Always	2 minutes
R	Human (Trader)	N.A.	N.A.	Always	2 minutes
A	Human (Risk Officer)	Three	N.A.	Always	5 minutes
D	Human (Trader)	N.A.	N.A.	Always	5 minutes
P	Human (Trader)	N.A.	N.A.	Always	5 minutes

Variant 2: Machine-supported decision-making

In this variant, see Table V, the stock trader is provided technical and financial information from a machine that collects data from multiple in-company and online sources (**Input**). Based on the data collected and provided to the stock trader, the machine also provides a best next action (**Recommend**). By default, this action is processed by the machine after 60 minutes (**Decide**), however, only when no veto is cast by the stock trader (**Agree**). When no veto is cast, the machine processes the buy order into the system of the financial organization (**Perform**).

In this variant, the machine generates a suggestion and is provided with the result of the decision as it needs to apply machine-learning to increase and maintain the accuracy of recommendation.

TABLE V. GOVERNANCE STRUCTURE FOR VARIANT TWO

	SF1:	**SF2:**	**SF3:**	**SF4:**	**SF5:**
I	Machine (system)	N.A.	None	Always	1 minutes
R	Machine (system)	One	None	Always	5 micro seconds
A	Human (Trader)	None	60 minutes	On request	5 minutes
D	Machine (system)	None	None	Always	1 minute
P	Machine (system)	None	N.A.	On request	1 micro second

Variant 3: Autonomous decision-making

In this variant, see Table VI, one trading machine collects data (e.g., financial, performance, sentiment) 24/7 (**Input**). Based on the data, the machine considers buy/sell orders per stock in the portfolio. In this variant, another machine, solely focused on calculation tasks, provides predictions that represent recommendations for the algorithm to take into account (**Recommend**). Furthermore, as is usual with HFT, performance is critical for the profit margin, so redundant activities should be prevented as much as possible. However, another machine of the stock trader simultaneously needs to, independently, come to the same conclusion regarding the considered stock in order to execute a buy or sell order (**Agree**). When both machines agree, the order is sent (**Decide**). Because the machine (and its underlying algorithms) is validated for its compliance, the stock exchange does not have to verify the transaction and can instantly process the change of ownership of the given stocks. Then, the machine processes the buy order into the system of the financial organization (**Perform**).

TABLE VI. GOVERNANCE STRUCTURE FOR VARIANT THREE

	SF1:	**SF2:**	**SF3:**	**SF4:**	**SF5:**
I	Machine 1 (system)	None	None	Never	1 minute
R	Machine 2 (system)	One	None	On request	5 micro seconds
A	Machine 3 (system)	None	None	Never	5 micro seconds
D	Machine 4 (system)	None	None	On request	1 micro second
P	Machine (system)	None	None	On request	1 micro second

Scenario 3: Military use of drones context

Variant 1: Manual human decision-making

In the first variant, see Table VII, a human operator fully controls the military drone that is on patrol in a conflict territory, defending certain strategic assets. Human mission specialists provide data that the drone and its human operator requires to operate in the mission area (**Input**). When on patrol, the drone's infrared sensor detects two heat signatures and alerts the human operator, providing two possible scenarios that could be relevant in the given context. Based on the data and sensor readings, the drone provides two recommendations with probability percentages (**Recommend**). The human operator considers the recommendations, assesses the situation via the drone's sensors, and considers to execute a given action (**Agree**). Depending on the situation at hand, the human operator controlling the drone could veto the agree role, e.g., when the context drastically changes in a very short amount of time in combination with human assets that could be at risk. Based on all data relevant to making the decision, the human operator, which is always the highest ranking employee present, decides upon the best next action to proceed (**Decide**), and orders the drone to eliminate the targets (**Perform**).

TABLE VII. GOVERNANCE STRUCTURE FOR VARIANT ONE

	SF1:	**SF2:**	**SF3:**	**SF4:**	**SF5:**
I	Human (specialist)	N.A.	None	N.A.	12 hours
R	Machine (drone)	Two	None	Always	5 minutes
A	Human (operator)	N.A.	None	N.A.	5 minutes
D	Human (highest rank)	N.A.	None	N.A.	10 minutes
P	Machine (drone)	None	None	Always	1 minute

Variant 2: Machine-supported decision-making

In this variant, see Table VIII, the drone receives input data from mission specialists beforehand, using machine parameters so that the machine can operate autonomously (**Input**). In this variant, the human operator does not control the drone constantly as described in the previous variant. However, the human operator controls the drone only when an alert is generated by the drone indicating a situation that needs human attention. Before the alert is generated, the drone autonomously calculates, based on mission and sensor data, one next best action with a probability percentage (**Recommend**). Then, the human operator consults the highest ranked employee present to ask permission to execute a given action (**Agree**). Based on the previous interaction the human operator approves or rejects the recommended next best action proposed by the drone (**Decide**). The outcome of the decision is executed by the drone, in this case resulting in either returning to patrol pattern, keep monitoring the situation or eliminating the target (**Perform**).

TABLE VIII. GOVERNANCE STRUCTURE FOR VARIANT TWO

	SF1:	**SF2:**	**SF3:**	**SF4:**	**SF5:**
I	Human (specialist)	N.A.	None	N.A.	1 hour
R	Machine (drone)	None	None	Always	2 minutes
A	Human (highest rank)	N.A.	None	N.A.	5 minutes
D	Human (operator)	N.A	None	N.A.	1 minute
P	Machine (drone)	None	None	Always	1 minute

Variant 3: Autonomous decision-making

In this variant, see Table IX, the drone collects mission parameters and data from different military sources autonomously to assess the mission context (**Input**). The drone's sensors detect suspicious behavior and generates likely scenarios and corresponding recommendations in terms of actions (**Recommend**). Based on these scenarios, several additional data sources are evaluated and a next best action is calculated by the drone. The drone communicates to mission command that it detected suspicious behavior in the mission area and reports upon the derivation towards the next best action and the corresponding actions the drone is going to execute (**Agree**). Then, mission command has three minutes to evaluate the situation and the drone's decision and veto the decision if required (**Decide**). When no veto is cast by the human operators' part of mission control, the drone executes the next best action, returning to the original patrol protocol (**Perform**).

TABLE IX. GOVERNANCE STRUCTURE FOR VARIANT THREE

	SF1:	**SF2:**	**SF3:**	**SF4:**	**SF5:**
I	Machine (drone)	N.A.	None	On request	1 minute
R	Machine (drone)	None	None	Always	1 minute
A	Human (operator)	N.A.	3 minutes	N.A.	3 minutes
D	Machine (drone)	N.A.	None	Always	5 seconds
P	Machine (drone)	None	None	Always	30 seconds

The three scenario's each accompanied by three variants provide an overview of a decision-making process, the role

distribution between humans and machines, the autonomy of the machine, and SF's that have to be taken into account. The framework can also be applied to guide the creation of a roadmap, as it shows how decision-making processes can be further automated and plan accordingly.

V. FRAMEWORK VALIDATION

To validate the framework, a qualitative research approach is selected given the first cycle of validation required [41]. Qualitative research aims to capture phenomena and its relationships using rich data sources. Data sources are always real-world context-based, and therefore support the exploration of a phenomenon in its natural context [42]. One widely-accepted qualitative research technique is a focus group.

A focus group is a qualitative face-to-face data collection technique that allows for broad interactions on a topic [43]. It is a more efficient method of data collection than qualitative interviews because, physically, more participants can be involved at a given point in time. Furthermore, utilizing focus groups also allows for cross-participant discussion about a subject to achieve a greater sense of detail about that subject as well as shared decision-making, i.e., validating artifacts [43]. Before a focus group can be executed, a number of factors need to be considered; 1) the goal of the focus group, 2) the selection of participants, 3) the number of participants, 4) the selection of the facilitator, 5) the information recording facilities, and 6) the protocol of the focus group [43], [44].

(1) For the research team, the goal of the focus group was to validate the framework.

(2) The selection of participants should be based on the group of individuals, organizations, information technology, or community that best represents the phenomenon studied [42]. In this study, organizations and individuals that deal with (semi)automated decision making processes at a large scale form the phenomenon studied; examples are financial and governmental institutions. To find relevant experts on this topic, the research team requested that the framework could be discussed during the monthly meeting of the Business Rules Management (BRM) expertise forum. This group consists of experts working for different Dutch governmental institutions, namely the Dutch Tax and Customs Administration, Dutch Immigration and Naturalization Service, Netherlands Enterprise Agency, Dutch Employee Insurance Agency, Dutch Education Executive Agency, Ministry of Education, Culture and Science, the Department of Waterways and Public Works, and Dutch Social Security Office. All of such governmental institutions are responsible for executing law and regulations.

(3) In total, six experts were present during the meeting that agreed to participate. Each participant represented one Dutch governmental institution and are all involved in designing semi(automated) decision making. The respondents had following roles: one enterprise architect, two business rules analysts, business rules architect, one business analyst, and one BRM project manager. Each of the participants had at least five years of experience within the domain of decision-making using BRM.

(4) Delbecq and van de Ven [44] and Glaser [45] state that the facilitator should be an expert on the topic and familiar with group meeting processes. The selected facilitator has a Ph.D. in BRM, has conducted seven years of research on the topic, and has facilitated many (similar) focus group meetings before.

(5) The focus group could not be recorded due to confidentiality of the decision-making cases discussed alongside the framework. However, the facilitation made notes regarding a prepared set of questions per participant. The duration of the focus group was approximately one hour.

(6) The focus group had a protocol that consisted of three phases. The first phase comprised the preparation of the participants where they were invited to already study the framework, its concepts and their definitions. The framework's documentation was sent three days in advance to the participants. The second phase comprised the actual focus group in which the following questions where addressed: 1) "*Do you believe that the framework adds value for the governance of decision management?"* 2) "*Are the roles described recognizable?*", 3) "*Are additional roles needed, and why?*", 4) "*Are all SF's recognizable?*", 5) "*Are there SF's that are missing?*", and 6) "*Do you believe that DMN will be enriched using the proposed element?*"

The facilitator started with a short presentation about the framework and its components (i.e., the roles, their responsibilities, and the SF's that need to be taken into account per role). Regarding question one, the participants agreed with each other that the information in the framework needs to be captured, thus are recognizing the need for such an addition for DMN. Note that DMN is becoming an accepted standard, especially in the Dutch governmental. An example mentioned that also shows the need to structure and capability to share decision-making data is the new General Data Protection Regulation [10] which states that automated decisions must be explainable to both regulators, but more importantly to, European civilians. Furthermore, from a theoretical point of view, the participants agreed that a lot of research is conducted and published regarding the design and production of decisions and underlying rules, but lacking contributions regarding the governance of decisions. Then, with regards to question two and three, the participants stated that the roles were recognizable and that none are missing. This was mainly because the participants were aware of a close variant of the RAPID model, the RACI model. However, there was some discussion about the absence of a dedicated role for informing relevant stakeholders, when necessary. When the facilitator explained that the ability to inform is actually a separate SF designed to be taken into account for each role

the respondents agreed that it is not an actual role but indeed a situational factor. Furthermore, there was some discussion regarding the labelling of the roles. The main discussion was about the fact that specific roles, e.g., recommend and perform, are formulated as activity names and not real role names. Although three participants identified this as a problem, the other three did not agree and thought the role labels were clear. As we adopted these best practice labels from existing literature, the research team chose to not change the labels as is. Lastly, the participants argued that, depending on the input of a given decision, the stakeholders can differ in practice. The participants discussed the possibility to define multiple governance structures based on the input for the same decision, however, this would lead to (too) much extra administration, i.e., when more than two or three variants need to be defined. This is followed by question four and five. Discussion regarding both questions mainly was about the SF inform. This is due to the fact that informing stakeholders can be done on different levels. The framework does not take this into account. An example is the difference between informing a stakeholder about the outcome of the decision made versus informing about the outcome of the decision made in addition with extra information, for example, information on how the decision is executed, how the decision has been made as well as which data is used in the decision-making process. The participants added that this difference significantly affects how the decision-making process is facilitated by both tooling as well as the stakeholders involved, and should be taken into account as part of the inform SF. Furthermore, regarding the inform role, when multiple stakeholders from different organizations are involved in decision-making, the framework should take into account possible conflicts of interest and provide the possibility to specify how stakeholders are involved. As our current definition of the SF inform does not dictate who to inform and how the actual role/person should be informed. Organizations are free to apply additional localized business rules on the framework, thereby managing conflict of interest. For example, one organization can define inform to only inform customers about the outcome of the decision, while other organizations want to inform their customers on a different level, by communicating the outcome of the decision-making as well as the data and rules utilized. Lastly, one of the participants argued that the deadline SF is not always relevant and should be interchangeable with other SF's. While the other participants disagreed, on this topic the framework allows to change SF's (the example of budget was mentioned by the participant as a replacement for deadline). With regards to question six, the participants agreed that DMN could benefit from the element proposed to support the registration of important governance information about decisions modelled.

One general remark was about the presentation of the governance framework and its contents. Although not in scope of this study, the participants added that the presentation is important for acceptance, as the contents are usually read and utilized by people instead of machines. They argued that the current proposed element for DMN presented in Figure 2 seems simple yet very appropriate.

VI. Discussion and Conclusion

Since the DMN standard is getting more commonly utilized in practice, more decisions are being modelled explicitly for documentation or automation. However, the current DMN standard does not take into account roles and autonomy regarding decisions and the underlying decision-making process. In this paper, a governance structure framework is being proposed to complement the design and specification of decisions in the DMN standard. To do so, the theoretical constructs of decision-making roles (RAPID) and autonomy levels together with five SFs (LOADAS) are combined to answer the following research question: '*How can a governance structure of the decision making process be made explicit?*'. One could solely consider the currently available models and frameworks (i.e., RAPID and OODA) to answer this question. However, this results to an incomplete assessment of the situation. To illustrate this finding we this base our example on the drone usage by military institutions. When an analysis of this situation is made based on the RAPID model, an overview of the different stakeholders is provided in the decision to assess the use of lethal force. The autonomy of each role is not described. In a normal military operation this is tackled by the normal hierarchy of command. However, machines (killer drones) are increasingly being utilized and their decision power progressively becomes larger. As such drones are designed to analyze and act themselves, without human intervention. So, in the context of the military usage of drones, it is unclear what the drone can decide on its own and whether it should or should not inform human operators, since only the roles and their activity is clear.

The other way around, when solely considering autonomy levels for machines in decision-making (i.e., LOADAS and ALFUS), it is explicit how machines operate in a decision-making process. For example, what responsibilities the drone has with regards to informing human operators after executing lethal force to eliminate targets or the whether a human operator has the possibility to override a decision made by the drone. However, in such a situation it is unclear what roles and responsibilities are involved in the decision-making and how they work together to achieve a certain added value. Thus, for this example, the drone does not know which role is able to veto the decision and therefore the combination adds value

The proposed governance structure framework has been presented using three scenario's each based on three variants. For each variant, the roles, responsibilities and SF's (human-machine, alternatives, veto and inform) are different. These variants demonstrate that various choices in decision-making processes lead to design considerations that should be taken into account. For example, when machines autonomously decide on which benefits are relevant, what is the best method of informing humans in a specific context, or the

appropriate timeframe applicable to veto a decision by a human, in a specific context.

The suggested framework has its limitations. The framework is a suggested solution derived from the existing knowledge base in the area of decision management, decision-making and machine autonomy, and thereby the result of a 'generate design alternative' phase [46]. However, we believe that the proposed framework reached a level of maturity such that it can enter a detailed validation phase. In a planned study, a collection of cases will be used to further validate the framework and to further demonstrate its practical usefulness. We note that the framework is widely applicable if every decision-making context can be modelled so that all stakeholders are aware of their roles and responsibilities in a given decision-making context.

Lastly, several future research directions are described, which are based on the theoretical findings as well as the focus group conducted.

The first direction comprises the need for a practical approach when a decision has multiple, i.e., more than three, variants of which the governance structure must be made explicit with the framework. For example, the decision-making to grant a work visa for a county could be very diverse based on the data inserted by the applicant. When an applicant enters that a work visa has been revoked earlier, additional criteria, actors and decision-making factors (such as deadlines or the possibility to veto) are relevant, yet for the same decision. Future research should therefore focus on the incorporation (and how that could be achieved) of multiple layers for the same decision, as the Subject Matter Experts (SME's) suggested that the framework could become difficult to use in practice otherwise.

The second research direction comprises the presentation of the element in DMN (DRD level) as well as the presentation of the governance information in the matrices e.g., in tables IV-VI. Although not in the scope of this research study, the SME's stated that this is an important factor to take into account. This partly overlaps with the previous research direction as the presentation of multiple possible variants of the same decision needs to be presented effectively, according to the SME's. It is therefore likely that the current proposed matrix changes to accommodate effective information transferal.

As this study proposes an addition to enrich the DMN standard, future steps should focus on approaching the OMG to discuss incorporation of governance structures in the next version of the DMN standard. However, before such steps are taken, it is imperative that the framework undergoes more validation rounds to ensure more SME's and even whole organizations endorse the framework. Future research would therefore mean that more SME's are included as well as from industries other than the governmental setting, which was the demarcation of the SME selection for the focus group in this study. Involving different industries for the validation of the framework would probably yield other interesting improvements as well as future research directions.

REFERENCES

[1] K. Smit and M. Zoet, "A Governance Framework for (semi) Automated Decision-making," in *Proceedings of the Tenth International Conference on Information, Process, and Knowledge Management (eKNOW)*, 2018, pp. 83–88.

[2] Object Management Group, "Decision Model And Notation (DMN), Version 1.1," 2016.

[3] Object Management Group, "ArchiMate® 3.0 Specification," 2016.

[4] P. Rogers and M. Blenko, "Who has the D?," *Harv. Bus. Rev.*, vol. 84, no. 1, pp. 52–61, 2006.

[5] M. Zoet, *Methods and Concepts for Business Rules Management*, 1st ed. Utrecht: Hogeschool Utrecht, 2014.

[6] B. Hnatkowska and J. M. Alvarez-Rodriguez, "Business Rule Patterns Catalog for Structural Business Rules," in *Software Engineering: Challenges and Solutions*, 1st ed., Springer International Publishing, 2017, pp. 3–16.

[7] M. W. Blenko, M. C. Mankins, and P. Rogers, "The Decision-Driven Organization," *Harv. Bus. Rev.*, vol. 88, no. 6, pp. 54–62, Jun. 2010.

[8] J. Beer, A. D. Fisk, and W. A. Rogers, "Toward a framework for levels of robot autonomy in human-robot interaction," *J. Human-Robot Interact.*, vol. 3, no. 2, p. 74, 2014.

[9] R. Parasuraman, T. B. Sheridan, and C. D. Wickens, "A model for types and levels of human interaction with automation," *IEEE Trans. Syst. man, Cybern. A Syst. Humans*, vol. 30, no. 3, pp. 286–297, 2000.

[10] European Commission, "Protection of personal data - GDPR," 2017. [Online]. Available: http://ec.europa.eu/justice/data-protection/. [Accessed: 14-Aug-2017].

[11] M. R. Endsley and D. B. Kaber, "Level of automation effects on performance, situation awareness and workload in a dynamic control task," *Ergonomics*, vol. 42, no. 3, pp. 462–492, 1999.

[12] H. M. Huang, K. Pavek, B. Novak, J. Albus, and E. Messin, "A framework for autonomy levels for unmanned systems (ALFUS)," in *Proceedings of the AUVSI's Unmanned Systems North America*, 2005, pp. 849–863.

[13] J. Taylor, "Who has the 'D' when the 'D' is automated?," 2007. [Online]. Available: http://www.beyeblogs.com/edmblog/archive/2007/02/who_has_the_d_w_2.php. [Accessed: 01-Dec-2017].

[14] A. Edwards and G. Elwyn, *Shared decision-making in health care: Achieving evidence-based patient choice*. Oxford University Press, 2009.

[15] H. M. Davey *et al.*, "Medical tests: women's reported and preferred decision-making roles and preferences for information on benefits, side-effects and false results," *Heal. Expect.*, vol. 5, no. 4, pp. 330–340, 2002.

[16] J. R. Adams, R. E. Drake, and G. L. Wolford, "Shared decision-making preferences of people with severe mental illness," *Psychiatr. Serv.*, vol. 58, no. 9, pp. 1219–1221, 2007.

[17] N. K. Arora and C. A. McHorney, "Patient preferences for medical decision making: who really wants to participate?," *Med. Care*, vol. 38, no. 3, pp. 335–341, 2000.

[18] B. W. Husted and D. B. Allen, "Toward a model of cross-cultural business ethics: The impact of individualism and

collectivism on the ethical decision-making process," *J. Bus. Ethics*, vol. 82, no. 2, pp. 293–305, 2008.

[19] A. Ho, "Relational autonomy or undue pressure? Family's role in medical decision-making," *Scand. J. Caring Sci.*, vol. 22, no. 1, pp. 128–135, 2008.

[20] P. S. Scherrer, "Directors' responsibilities and participation in the strategic decision making process," *Corp. Gov. Int. J. Bus. Soc.*, vol. 3, no. 1, pp. 86–90, 2003.

[21] C. Charles, A. Gafni, and T. Whelan, "Decision-making in the physician–patient encounter: revisiting the shared treatment decision-making model," *Soc. Sci. Med.*, vol. 49, no. 5, pp. 651–661, 1999.

[22] P. V. Gorsevski, S. C. Cathcart, G. Mirzaei, M. M. Jamali, X. Ye, and E. Gomezdelcampo, "A group-based spatial decision support system for wind farm site selection in Northwest Ohio," *Energy Policy*, vol. 55, pp. 374–385, 2013.

[23] M. Sigala, "The impact of geocollaborative portals on group decision making for trip planning," *Eur. J. Inf. Syst.*, vol. 21, no. 4, pp. 404–426, 2012.

[24] P. Jankowski, N. Andrienko, and G. Andrienko, "Map-centred exploratory approach to multiple criteria spatial decision making," *Int. J. Geogr. Inf. Sci.*, vol. 15, no. 2, pp. 101–127, 2001.

[25] Q. Tian, J. Ma, C. J. Liang, R. C. W. Kwok, O. Liu, and Q. Zhang, "An organizational decision support approach to R and D project selection," in *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*, 2002, pp. 3418–3427.

[26] N. L. Kerr and R. S. Tindale, "Group performance and decision making," *Annu. Rev. Psychol*, vol. 55, pp. 623–655, 2004.

[27] J. Boyd, "A discourse on winning and losing," 1987.

[28] D. G. Ullman, "OO-OO-OO!" the sound of a broken OODA loop," *CrossTalk-The J. Def. Softw. Eng.*, pp. 22–25, 2007.

[29] E. Shahbazian, D. E. Blodgett, and P. Labbé, "The extended OODA model for data fusion systems," in *Proceedings of 4th International Conference on Information Fusion*, 2001.

[30] R. Breton and R. Rousseau, "The C-OODA: A cognitive version of the OODA loop to represent C2 activities," in *Proceedings of the 10th International Command and Control Research Technology Symposium*, 2005.

[31] R. Rousseau and R. Breton, "The M-OODA: A model incorporating control functions and teamwork in the OODA loop," in *Proceedings of the 2004 Command and Control Research Technology Symposium*, 2004, pp. 14–16.

[32] C. Bartneck and J. Forlizzi, "A design-centred framework for social human-robot interaction," in *Proceedings of the 13th IEEE International Workshop on Robot and Human Interactive Communication*, 2004, pp. 591–594.

[33] T. B. Sheridan and W. Verplank, "Human and Computer Control of Undersea Teleoperators," Cambridge, MA, 1978.

[34] M. R. Endsley, "The application of human factors to the development of expert systems for advanced cockpits.," in *Proceedings of the Human Factors Society Annual Meeting*, 1987, pp. 1388–1392.

[35] C. A. Miller and R. Parasuraman, "Designing for flexible interaction between humans and automation: Delegation interfaces for supervisory control," *Hum. Factors*, vol. 49, no. 1, pp. 57–75, 2007.

[36] R. Azuma, M. Daily, and C. Furmanski, "A review of time critical decision making models and human cognitive processes," in *Aerospace Conference*, 2006, pp. 1–9.

[37] E. Budish, P. Cramton, and J. Shim, "The high-frequency trading arms race: Frequent batch auctions as a market design response," *Q. J. Econ.*, vol. 130, no. 4, pp. 1547–1621, 2015.

[38] Canadian Government, "Apply for Child Benefits," 2018. [Online]. Available: https://www.canada.ca/en/revenue-agency/services/child-family-benefits/canada-child-benefit-overview/canada-child-benefit-apply.html. [Accessed: 14-Aug-2018].

[39] R. J. Kuo, C. H. Chen, and Y. C. Hwang, "An intelligent stock trading decision support system through integration of genetic algorithm based fuzzy neural network and artificial neural network," *Fuzzy sets Syst.*, vol. 118, no. 11, pp. 21–45, 2001.

[40] N. Sharkey, "Saying 'no!'to lethal autonomous targeting," *J. Mil. Ethics*, vol. 9, no. 4, pp. 369–383, 2010.

[41] A. Hevner and S. Chatterjee, *Design research in information systems: theory and practice*, 22nd ed. Springer Science & Business Media, 2010.

[42] A. Strauss and J. Corbin, *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*, 3rd ed., vol. 3. Thousand Oaks, CA: SAGE Publications Ltd., 2015.

[43] D. L. Morgan, *Focus groups as qualitative research*, 16th ed. Sage publications, 1996.

[44] A. L. Delbecq and A. H. Van de Ven, "A group process model for problem identification and program planning," *J. Appl. Behav. Sci.*, vol. 7, no. 4, pp. 466–492, 1971.

[45] B. G. Glaser, *Theoretical sensitivity: Advances in the methodology of grounded theory*. Sociology Press, 1978.

[46] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design Science in Information Systems Research," *MIS Q.*, vol. 28, no. 1, pp. 75–105, 2004.

www.iariajournals.org

International Journal On Advances in Intelligent Systems
issn: 1942-2679

International Journal On Advances in Internet Technology
issn: 1942-2652

International Journal On Advances in Life Sciences
issn: 1942-2660

International Journal On Advances in Networks and Services
issn: 1942-2644

International Journal On Advances in Security
issn: 1942-2636

International Journal On Advances in Software
issn: 1942-2628

International Journal On Advances in Systems and Measurements
issn: 1942-261x

International Journal On Advances in Telecommunications
issn: 1942-2601