International Journal on Advances in Internet Technology

2013 vol. 6 nr. 1&2

The *International Journal on Advances in Internet Technology* is published by IARIA.
ISSN: 1942-2652
journals site: http://www.iariajournals.org
contact: petre@iaria.org

Sponsored by IARIA
www.iaria.org

International Journal on Advances in Internet Technology
Volume 6, Number 1 & 2, 2013

Christophe Guéret, Vrije Universiteit Amsterdam, Nederlands
Frederic Guidec, IRISA-UBS, Université de Bretagne-Sud, France
Bin Guo, Northwestern Polytechnical University, China
Gerhard Hancke, Royal Holloway / University of London, UK
Arthur Herzog, Technische Universität Darmstadt, Germany
Rattikorn Hewett, Whitacre College of Engineering, Texas Tech University, USA
Nicolas Hidalgo, Yahoo! Research Latin America, France
Quang Hieu Vu, EBTIC, Khalifa University, Arab Emirates
Hiroaki Higaki, Tokyo Denki University, Japan
Eva Hladká, Masaryk University, Czech Republic
Dong Ho Cho, Korea Advanced Institute of Science and Technology (KAIST), Korea
Anna Hristoskova, Ghent University - IBBT, Belgium
Ching-Hsien (Robert) Hsu, Chung Hua University, Taiwan
Christian Hübsch, Institute of Telematics, Karlsruhe Institute of Technology (KIT), Germany
Chi Hung, Tsinghua University, China
Edward Hung, Hong Kong Polytechnic University, Hong Kong
Linda A. Jackson, Michigan State University, USA
Raj Jain, Washington University in St. Louis , USA
Edward Jaser, Princess Sumaya University for Technology - Amman, Jordan
Terje Jensen, Telenor Group Industrial Development / Norwegian University of Science and Technology, Norway
Yasushi Kambayashi, Nippon Institute of Technology, Japan
Georgios Kambourakis, University of the Aegean, Greece
Atsushi Kanai, Hosei University, Japan
Henrik Karstoft , Aarhus University, Denmark
Dimitrios Katsaros, University of Thessaly, Greece
Ayad ali Keshlaf, Newcastle University, UK
Reinhard Klemm, Avaya Labs Research, USA
Samad Kolahi, Unitec Institute Of Technology, New Zealand
Dmitry Korzun, Petrozavodsk State University, Russia / Aalto University, Finland
Evangelos Kranakis, Carleton University - Ottawa, Canada
Slawomir Kuklinski, Warsaw University of Technology, Poland
Andrew Kusiak, The University of Iowa, USA
Mikel Larrea, University of the Basque Country UPV/EHU, Spain
Frédéric Le Mouël, University of Lyon, INSA Lyon / INRIA, France
Nicolas Le Sommer, Université Européenne de Bretagne, France
Juong-Sik Lee, Nokia Research Center, USA
Wolfgang Leister, Norsk Regnesentral (Norwegian Computing Center), Norway
Clement Leung, Hong Kong Baptist University, Hong Kong
Man-Sze Li , IC Focus, UK
Longzhuang Li, Texas A&M University-Corpus Christi, USA
Yaohang Li, Old Dominion University, USA
Jong Chern Lim, University College Dublin, Ireland
Lu Liu, University of Derby, UK
Damon Shing-Min Liu, National Chung Cheng University, Taiwan
Michael D. Logothetis, University of Patras, Greece
Malamati Louta, University of Western Macedonia, Greece

Maode Ma, Nanyang Technological University, Singapore
Elsa María Macías López, University of Las Palmas de Gran Canaria, Spain
Olaf Maennel, Loughborough University, UK
Zoubir Mammeri, IRIT - Paul Sabatier University - Toulouse, France
Yong Man, KAIST (Korea advanced Institute of Science and Technology), South Korea
Sathiamoorthy Manoharan, University of Auckland, New Zealand
Chengying Mao, Jiangxi University of Finance and Economics, China
Brandeis H. Marshall, Purdue University, USA
Sergio Martín Gutiérrez, UNED-Spanish University for Distance Education, Spain
Constandinos Mavromoustakis, University of Nicosia, Cyprus
Hamid Mcheick, Université du Québec à Chicoutimi, Canada
Shawn McKee, University of Michigan, USA
Stephanie Meerkamm, Siemens AG in Erlangen, Germany
Kalogiannakis Michail, University of Crete, Greece
Peter Mikulecky, University of Hradec Kralove, Czech Republic
Moeiz Miraoui, Université du Québec/École de Technologie Supérieure - Montréal, Canada
Shahab Mokarizadeh, Royal Institute of Technology (KTH) - Stockholm, Sweden
Mario Montagud Climent, Polytechnic University of Valencia (UPV), Spain
Stefano Montanelli, Università degli Studi di Milano, Italy
Julius Müller, TU- Berlin, Germany
Juan Pedro Muñoz-Gea, Universidad Politécnica de Cartagena, Spain
Krishna Murthy, Global IT Solutions at Quintiles - Raleigh, USA
Alex Ng, University of Ballarat, Australia
Christopher Nguyen, Intel Corp, USA
Vlad Nicolicin Georgescu, SP2 Solutions, France
Petros Nicopolitidis, Aristotle University of Thessaloniki, Greece
Carlo Nocentini, Università degli Studi di Firenze, Italy
Federica Paganelli, CNIT - Unit of Research at the University of Florence, Italy
Carlos E. Palau, Universidad Politecnica de Valencia, Spain
Matteo Palmonari, University of Milan-Bicocca, Italy
Ignazio Passero, University of Salerno, Italy
Serena Pastore, INAF - Astronomical Observatory of Padova, Italy
Fredrik Paulsson, Umeå University, Sweden
Rubem Pereira, Liverpool John Moores University, UK
Mark Perry, University of Western Ontario/Faculty of Law/ Faculty of Science - London, Canada
Yulia Ponomarchuk, Far Eastern State Transport University, Russia
Jari Porras, Lappeenranta University of Technology, Finland
Neeli R. Prasad, Aalborg University, Denmark
Drogkaris Prokopios, University of the Aegean, Greece
Emanuel Puschita, Technical University of Cluj-Napoca, Romania
Lucia Rapanotti, The Open University, UK
Gianluca Reali, Università degli Studi di Perugia, Italy
Christoph Reinke, SICK AG, Germany
Jelena Revzina, Transport and Telecommunication Institute, Latvia
Karim Mohammed Rezaul, Glyndwr University, UK
Leon Reznik, Rochester Institute of Technology, USA

Joel Rodrigues, Instituto de Telecomunicações / University of Beira Interior, Portugal
Simon Pietro Romano, University of Napoli Federico II, Italy
Michele Ruta, Politecnico di Bari, Italy
Jorge Sá Silva, University of Coimbra, Portugal
Farzad Salim, Queensland University of Technology, Australia
Sébastien Salva, University of Auvergne, France
Ahmad Tajuddin Samsudin, Telekom Malaysia Research & Development, Malaysia
Josemaria Malgosa Sanahuja, Polytechnic University of Cartagena, Spain
Luis Enrique Sánchez Crespo, Sicaman Nuevas Tecnologías / University of Castilla-La Mancha, Spain
Paul Sant, University of Bedfordshire, UK
Brahmananda Sapkota, University of Twente, The Netherlands
Alberto Schaeffer-Filho, Lancaster University, UK
Peter Schartner, Klagenfurt University, System Security Group, Austria
Rainer Schmidt, Aalen University, Germany
Thomas C. Schmidt, HAW Hamburg, Germany
Didier Sebastien, University of Reunion Island, France
Zary Segall, Chair Professor, Royal Institute of Technology, Sweden
Dimitrios Serpanos, University of Patras and ISI/RC ATHENA, Greece
Jawwad A. Shamsi, FAST-National University of Computer and Emerging Sciences, Karachi, Pakistan
Michael Sheng, The University of Adelaide, Australia
Kazuhiko Shibuya, The Institute of Statistical Mathematics, Japan
Roman Y. Shtykh, Rakuten, Inc., Japan
Patrick Siarry, Université Paris 12 (LiSSi), France
Jose-Luis Sierra-Rodriguez, Complutense University of Madrid, Spain
Simone Silvestri, Sapienza University of Rome, Italy
Åsa Smedberg, Stockholm University, Sweden
Vasco N. G. J. Soares, Instituto de Telecomunicações / University of Beira Interior / Polytechnic Institute of Castelo Branco, Portugal
Radosveta Sokullu, Ege University, Turkey
José Soler, Technical University of Denmark, Denmark
Boyeon Song, National Institute for Mathematical Sciences, Korea
Victor J. Sosa-Sosa, CINVESTAV-Tamaulipas, Mexico
Dora Souliou, National Technical University of Athens, Greece
João Paulo Sousa, Instituto Politécnico de Bragança, Portugal
Kostas Stamos, Computer Technology Institute & Press "Diophantus" / Technological Educational Institute of Patras, Greece
Vladimir Stantchev, SRH University Berlin, Germany
Tim Strayer, Raytheon BBN Technologies, USA
Masashi Sugano, School of Knowledge and Information Systems, Osaka Prefecture University, Japan
Tae-Eung Sung, Korea Institute of Science and Technology Information (KISTI), Korea
Sayed Gholam Hassan Tabatabaei, Isfahan University of Technology, Iran
Yutaka Takahashi, Kyoto University, Japan
Yoshiaki Taniguchi, Osaka University, Japan
Nazif Cihan Tas, Siemens Corporation, Corporate Research and Technology, USA
Alessandro Testa, University of Naples "Federico II" / Institute of High Performance Computing and Networking (ICAR) of National Research Council (CNR), Italy

Stephanie Teufel, University of Fribourg, Switzerland
Parimala Thulasiraman, University of Manitoba, Canada
Pierre Tiako, Langston University, USA
Ioan Toma, STI Innsbruck/University Innsbruck, Austria
Orazio Tomarchio, Universita' di Catania, Italy
Kurt Tutschku, University Blekinge Institute of Technology, Karlskrona, Sweden
Dominique Vaufreydaz, INRIA and Pierre Mendès-France University, France
Massimo Villari, University of Messina, Italy
Krzysztof Walkowiak, Wroclaw University of Technology, Poland
MingXue Wang, Ericsson Ireland Research Lab, Ireland
Wenjing Wang, Blue Coat Systems, Inc., USA
Zhi-Hui Wang, School of Softeware, Dalian University of Technology, China
Matthias Wieland, Universität Stuttgart, Institute of Architecture of Application Systems (IAAS),Germany
Bernd E. Wolfinger, University of Hamburg, Germany
Chai Kiat Yeo, Nanyang Technological University, Singapore
Mark Yampolskiy, Vanderbilt University, USA
Abdulrahman Yarali, Murray State University, USA
Mehmet Erkan Yüksel, Istanbul University, Turkey

International Journal on Advances in Internet Technology
Volume 6, Numbers 1 & 2, 2013

CONTENTS

Dynamic Nearest Neighbors and Online Error Estimation for SMARTPOS

Philipp Marcus, Moritz Kessel, and Martin Werner
Mobile and Distributed Systems Group
Ludwig-Maximilians-University Munich
Munich, Germany
{*philipp.marcus,moritz.kessel,martin.werner*}*@ifi.lmu.de*

***Abstract*—Location-based services are possibly the most popular services with respect to mobility, since they allow for the automated filtering of information relevant to the user. This paper presents a detailed evaluation of SMARTPOS, an indoor positioning system based on deterministic 802.11 fingerprinting and a digital compass. SMARTPOS is accurate enough to supply location estimates for indoor location-based services and can be deployed standalone on a mobile phone. Assuming that the mobile phone is held in front of the body, the system considers the user's orientation to avoid errors caused by the blocking effect of the human body. For location estimation it employes a kNN approach on that part of the fingerprint database that corresponds to the user's current orientation. As an extension to this approach, SMARTkNN is proposed which is based on dynamically selecting the number of nearest neighbors. This improved the mean position error to 1.10 meters and to a maximum position error of 2.65 meters in a 250 square meter environment in comparison to SMARTPOS which achieved a mean position error of 1.16 meters and a maximum position error of 2.74 meters. Furthermore, it is shown that the errors of SMARTPOS are normally distributed. Based on this fact, a novel online error estimator using bivariate Gaussians is proposed which gives the best approximation of the observed errors compared to existing methods. Additionally it was observed, that the density of the underlying radiomap strongly correlates to the maximum error and has a weaker impact on the observed mean error.**

Keywords-802.11 Fingerprinting, Orientation Filter, Mobile Phone Positioning, Location-Based Services.

I. INTRODUCTION

In recent years, a trend towards mobility can be recognized. Smartphones, small devices with comparatively high processing power and mobile Internet, make it possible to work while traveling, to stay connected to social networks, and to retrieve nearly any information anywhere at any time. One of the most popular mobile services are location-based services (LBS). These are value-added services, which utilize the location of the mobile to present the user with information about its surroundings. Navigation and information services, friend-finder, pet-tracker, and location-based games are only a small part of the number of services and applications filling the app-stores of the world.

The key enabler for LBS is the Global Positioning System (GPS). It enables accurate positioning in outdoor environments, the usage is free of charge, the system is globally available, and most of today's smartphones are equipped with a GPS-receiver. Unfortunately, GPS is not able to track people in indoor environments with acceptable accuracy. Signals might get lost due to attenuation effects of roofs and walls or lead to position fixes of very low accuracy due to multipath propagation.

Even worse, indoor location-based services require much higher precision guarantees than outdoor services. Errors should not exceed a few meters to allow for a differentiation between several floors or rooms. Otherwise, the service could provide information for places, which are quite far away from the actual position of the target. Despite these challenges, many users would appreciate indoor location-based services, especially in large and complex buildings such as museums, shopping malls, airports, hospitals, or university buildings.

Existing indoor positioning techniques can be grouped by their level of precision and the expenses for additional infrastructure. Dedicated indoor positioning systems such as ultra wide band or ultrasonic systems consist of several components with the sole purpose of determining the positions of possibly multiple targets in indoor environments. The precision is often high, but an expensive infrastructure is needed and hence the space where positioning is possible is usually limited to a small area, where higher accuracy compensates the high cost. Another class of systems is built on existing infrastructure such as WLAN, Bluetooth or inertial sensors for positioning. The precision of such systems is limited, but the system can be deployed with few additional expenses.

In this paper, we extend SMARTPOS [1], an indoor positioning system for smartphones based on deterministic WLAN fingerprinting and a digital compass. The system is self-positioning, meaning that the whole positioning process (including all measurements) is carried out on the phone. It achieves a high accuracy within few meters and therefore is able to provide interactive, non-background indoor location-based services with high quality location estimates at no additional expenses. SMARTPOS makes use of the smartphone's orientation (which should correspond to the user's orientation) to avoid errors caused by the blocking effect of the human body. Only those fingerprints are considered for location estimation that were measured while viewing in a similar direction like the user. As an extension to the

system, a detailed evaluation of the system's errors has been carried out. Based on the results of that evaluation a novel online error estimation scheme is proposed, which enables the system to provide an error estimator as a Gaussian probability density function for each position measurement. Furthermore, a method for dynamically choosing the number of nearest neighbors based on convergence criteria is proposed which obviates the need for empirically determining an optimal number for every environment.

The remainder of this paper is structured as follows: In the next section, an overview of existing indoor positioning systems with focus on WLAN fingerprinting is given. In Section III, the original SMARTPOS is presented and evaluated in detail, stressing the impact of several parameters and decisions on the design of the system. Whether weighted or non-weighted kNN (k-nearest neighbors) in signal space should be carried out, the influence of missing values on the algorithm and the performance gain of including the orientation on SMARTPOS and a Naive Bayesian Estimator are evaluated. In Section IV, the question of the reliability of the positioning method is researched and an online error estimation scheme introduced. Then, the influence of the density of fingerprints is analyzed in Section V and a novel algorithm for dynamically choosing the best number of neighbors for every position fix is presented in Section VI. Section VII concludes the paper and gives hints on future work.

II. Related Work

In the past 15 years, a variety of technologies for indoor positioning have been proposed. A good overview of existing indoor positioning systems using radio frequency (RF) technologies such as radio frequency identification (RFID), ultra wide band (UWB), ultra high frequency (UHF), WLAN and Bluetooth is given in [2]. However, the authors do not describe up-to-date systems, which have been developed since 2007. We therefore focus in this section on the recent development and work closely related to our research.

Many pedestrian indoor positioning systems rely on WLAN fingerprinting algorithms [1], [3], [4], [5], which offer position estimates with sufficient accuracy (i.e., 1-3m) while utilizing the existing WLAN infrastructure and therefore avoiding high expenses. These algorithms belong to the area of pattern matching and work in two phases: The first phase is called the calibration phase, where a database is created by the collection of received signal strength indicator (RSSI) at certain reference positions from the surrounding access points (AP). The accumulated information of RSSI, AP and reference position at a specific time/interval is called a fingerprint. In the second phase, positioning is carried out by comparing current RSSI measurements with the previously stored values from the database. Different algorithms calculate the position as the reference position of the nearest fingerprint in signal space [3], the average of the k-nearest neighbors (kNN) with or without the distance in signal space as additional weight [1]. Some algorithms also utilize Bayesian methods [4], [5] based on probability distributions derived by multiple measurements over a length of time. While earlier systems utilize laptops for position determination, the recent trend goes towards smartphones. Martin et al. present one of the first WLAN positioning systems which integrates both offline and online phase on a mobile phone [6].

One of the first developed systems for WLAN fingerprinting, RADAR [3], includes already the impact of the user's orientation in the position calculation by obtaining empirical data for multiple orientations. Kaemarungsi et al. further analyze the effects of the user's presence and orientation on RSSI values in [7]. The results show that the attenuation effects of the human body can lower the RSSI by more than 9 dBm. COMPASS [5] is one of the first fingerprinting systems that addresses the problem of attenuation effects caused by the human body by adding a digital compass to the system. In the calibration phase, fingerprints for several selected orientations (typically each $45°$ or $90°$) are collected at reference positions. In the positioning phase, the user's orientation is measured by a digital compass and only those fingerprints with a similar orientation estimate are used for the positioning algorithm. COMPASS presents the most similar approach to the SMARTPOS System. However, we extend the work in serveral directions. By using kNN instead of a Bayesian estimator, the number of measurements carried out for fingerprint creation is massively reduced. While COMPASS reports 20 to 100 measurements for a single fingerprint to correctly estimate the Gaussians, we tested our system with 3-5 measurements. While the COMPASS approach might achieve an even higher accuracy due to the larger training dataset and the inclusion of the RSSI's second moment, it is not well suited for the self creation of databases by the user due to the high calibration effort. Chan et al. also present a system running on a mobile phone considering the orientation of the user in [8], but apply a technique called Newton Trust Region for further position refinement.

Most up-to-date systems combine WLAN fingerprinting with additional technologies such as inertial sensors to offer more accurate position estimates and continuous tracking functionality [9], [10]. In [9], the authors utilize a particle filter for fusing WLAN fingerprint location estimates with an accelerometer. For the utilization of the SMARTPOS system in Bayesian filtering techniques, a probability distribution needs to be given for each position calculation. Existing approaches [11], [12] often utilize grid based approaches, where the discrete probability distribution is directly obtained by the probability of all grid cells according to a Bayesian model. In [9], Evennou and Marx utilize a Gaussian distribution for particle weighting with the mean located on the WLAN position and a variance based on the deviation of the RSSI.

Lemelson et al. further investigate error estimation of WLAN fingerprinting based position determination in [13]. They propose different schemes which estimate the occuring error as a scalar that can be used to assess the trust of position estimates.

In this paper, we show that the position errors follow a Gaussian with high probability. Based on this result, an online error estimator is proposed that derives a Gaussian probability density function modelling an estimate for the ground truth position relative to the position fix. We then compare our own approach to slightly adapted proposals from Lemelson et al. [13]. In contrast to that paper, Beder et al. propose an offline error estimation method which allows for the calculation of the expected uncertainty of every possible position [14]. Similar to [9] a Gaussian distribution of RSSI values is assumed and the covariance matrix of the fingerprint is used to calculate the expected error.

In addition to the error estimation, this paper proposes a method for dynamically estimating the number of neighbors suitable for position estimation. Roshanaei and Maleki combine in [15] traditional RSSI-based fingerprinting with a method based on Angle of Arrival (AOA) to further reduce the set of the nearest neighbors to those which are located in a certain area. The area is determined by their AOA algorithm using an adaptive antenna array. Altintas and Serif enhance in [16] the neighbor selecton by k-means clustering. The candidate fingerprints are clustered according to their reference points and only the fingerprints of one cluster which has the smallest diameter are returned. Another approach is presented by Shin et al. in [17]. For their weighted nearest neighbor approach, the neighbors are picked from a set of all fingerprints with a distance in signal space below a certain threshold. Furthermore, the mean distance in this set is calculated and only those fingerprints are taken into consideration for position estimation whose distance is below this mean value. In contrast to related work, the method for dynamically choosing k presented in this paper is based on convergence criteria of the derived position estimates for different values of k. One advantage is, that it is completely independent from deriving thresholds in signal space.

III. SMARTPOS: A System for Self-Contained Mobile Positioning

In this section, we describe the original SMARTPOS system presented in [1], a system for accurate and self-contained indoor positioning based on deterministic 802.11 fingerprinting and a digital compass. The system runs standalone on a mobile phone and consists of a management module for the creation and maintenance of the fingerprint database and a module for location determination. The latter offers the possibility of modifying several parameters concerning the deterministic location estimation or allows a change of the positioning method to a room-based bayesian approach.

A. Database Creation on a Mobile Phone

During the offline phase, active scans for WLAN signals from surrounding access points (APs) are executed with a mobile phone at several reference positions. The measured signal strength values are enhanced with the viewing direction and the pixel coordinates of the reference position on a bitmap of the floor. The viewing direction is obtained by the digital compass of the smartphone, the position is assigned by tapping on a zoomable and scrollable map displayed on the screen of the mobile. Finally, these values (in the following referred to as fingerprints) are stored in a database. At each reference position, four fingerprints are created, one in the direction of each axis of the specific building. The alignment along the axes of the building instead of the geographic directions is carried out to improve the accuracy of the application in tracking scenarios since most users move along the main axes of a building, e.g., when walking down a corridor. For each fingerprint, five scans are executed and the average of the received signal strengths is stored in the database to reduce the impact of short-time fluctuations. Furthermore, the orientation of the phone, which is derived from the mobile phone's compass, is averaged throughout the sampling time and also stored in the database. This is done to remedy the disturbances of the magnetic field inside of buildings, especially near electronic sources or large amounts of metal.

B. Deterministic Location Estimation

During the online phase, SMARTPOS utilizes a deterministic positioning algorithm based on weighted kNN to estimate the approximate position of the user. WLAN signal strength measurements are carried out in a continuous fashion and for each measurement m the current orientation o of the phone is measured by its digital compass.

The orientation is considered to represent the approximate viewing direction of the user and hence implicitly yields the information about the attenuation of his body. The online RSSI values should therefore not be compared to all fingerprints in the database due to possible influence of the human body, but only to those fingerprints that correspond to a similar viewing direction to o during the offline phase. Since the viewing direction is retrieved from the noisy readings of the compass, the orientation is averaged over the duration of each scan. This mechanism could also be replaced by advanced filtering algorithms to reduce the impact of outliers. SMARTPOS considers only a subset S of all fingerprints in the database containing those with a maximal deviation of 50° from o and is therefore able to reduce the number of fingerprints matched in the online phase to an extent of approximately 27.7% of the database size.

On the remaining subset S of filtered fingerprints, the nearest neighbours in signal space with respect to m are computed. SMARTPOS uses a sophisticated distance metric for the comparison of two RSSI measurements (i.e., the online measurement m and a fingerprint $f \in S$): Each measurement contains the information about all RSSI values with the MAC address of the AP, which sent the signal. Since at a given position only signals of a subset of all access points in the building can be received, the question arises how to treat missing signal strength information in one of two compared measurements. One possibility would be to assign a fixed value *MIN* to the RSSI of all access points missing in one measurement. This mechanism favors combinations of measurements, where signals by an AP are of very small strength in one measurement and missing in the other instead of combinations, where a high RSSI value in one measurement is missing a counterpiece in the other. The value of *MIN* should be below the minimal RSSI value measureable by the device. The other possibility is to ignore all signal strength information missing at least in one of the compared measurements. Based on the results of a detailed evaluation (see Section III-C) SMARTPOS utilizes the second approach, which is expected to be more robust in the case a new AP is turned on or an existing AP is turned off. Nevertheless, a minimum overlap of at least three APs is required to avoid choosing wrong neighbors due to propagation symmetries in larger environments.

Based on the Euclidean distance $d_i = \text{dist}(m, f_i)$ in signal space the subset $N \subset S$ of the k nearest neighbours is computed. In addition, SMARTPOS assigns a weight w_i to each fingerprint $f_i \in N, i \in \{1, \ldots, k\}$ which is indirectly proportional to the distance in signal space. It computes after the following formula:

$$w_i = \left(d_i \sum_{j=1}^{k} \frac{1}{d_j} \right)^{-1} \quad (1)$$

It is easy to see that the w_i are normalized since $\sum_{i=1}^{k} w_i = 1$. For the computation of the user's position l, SMARTPOS calculates the weighted average of $l_i, i \in \{1, \ldots, k\}$, l_i being the reference position of the fingerprint f_i:

$$l = \sum_{i=1}^{k} l_i w_i \quad (2)$$

C. SMARTPOS Evaluation

For the evaluation of the SMARTPOS system, two sets of fingerprints were manually collected under laboratory conditions, i.e., without anybody around, in a part of our university building. All RSSI information was gathered with a HTC Desire. The first set is arranged in an approximate grid of 79 reference positions with fingerprints measured in the direction of all four main axes of the building, which results in 316 fingerprints in total (the grey dots in Figure 1). The second set is a much smaller set of 64 fingerprints

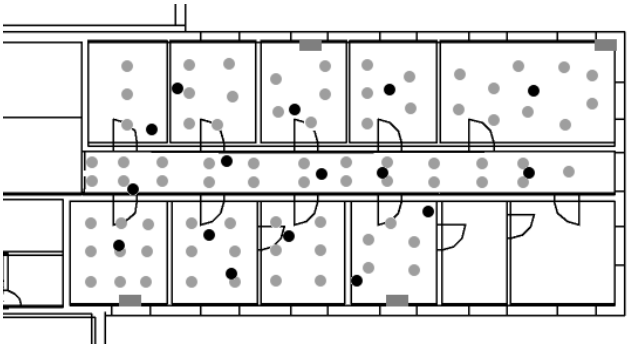
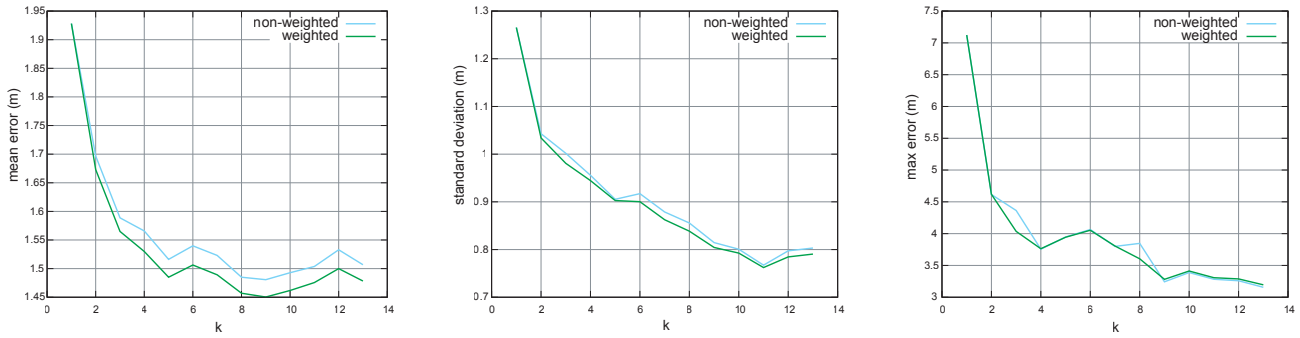


Figure 1: Reference database (gray dots) and online testset (black dots). APs are displayed as grey rectangles.

at 16 pseudo-randomly distributed reference positions (again measured in the direction of all four axes) within the coverage of the database and is used as substitution for online measurements (the black dots in Figure 1). This ensures that our results originate from an identical setting for all the different location estimators. The estimators are evaluated in respect to four criteria according to [2]: the accuracy as the mean position error, the precision as the maximal error and the standard deviation, and the complexity as the number of compared fingerprints. The question of scalability, cost and robustness is not considered, since the scalability and the cost are the same in all systems and the robustness is hard to measure. In the following, the results from a detailed evaluation of SMARTPOS in the described setting are presented and discussed. SMARTPOS is evaluated as follows: First, the deterministic kNN approach is analyzed and the settings of several parameters compared to each other. The questions of assigning a weight to the nearest neighbors and whether missing signal strength information should be considered or ignored are discussed and the impact of the user's orientation on accuracy and precision presented. In a consecutive step, an optimal value for k is determined for SMARTPOS. Finally, the usage of orientation information in a Naive Bayesian Estimator is analyzed.

1) Weighted or Non-Weighted kNN: When using a kNN approach together with WLAN fingerprinting one has to decide whether just to compute the mean of the nearest neighbors or to add a weight to each of the k-nearest neighbors according to the distance in signal space and then calculate the center of mass. With SMARTPOS, we evaluated both approaches for variable k. Figure 2 shows the results. The weighted approach behaves similarly, but performs better for each $k > 1$. The same applies for the deviation while the maximum error shows no significant difference except for two outliers ($k = 3$ and $k = 8$), for which the weighted approach also performs better. SMARTPOS therefore utilizes a weighted kNN as described in Section



(a) The mean error in meter (b) The standard deviation in meter (c) The maximum error in meter

Figure 2: Comparison of weighted and non-weighted kNN.

III-B.

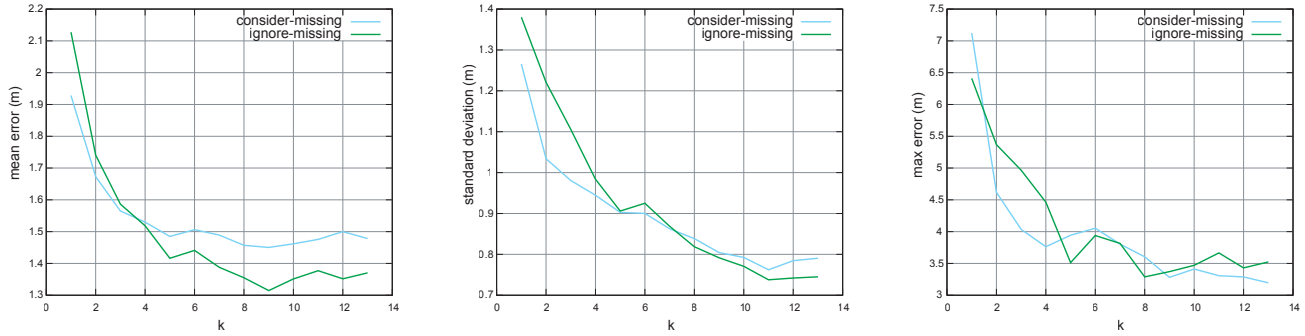
2) Treatment of Missing RSSI: In Section III-B, two approaches for the treatment of missing signal strength information when comparing two RSSI measurements are described. One considers the information by assigning a minimal value of -100 dBm for the missing RSSI information, the other ignores all RSSI values from APs measured only in one of the two compared measurements. Both approaches were tested for a variable k and the results are presented in Figure 3. The accuracy of a system ignoring missing values is higher than the accuracy of a system considering the information for each $k > 3$ and also offers a minimum mean error for $k = 9$. The deviation only becomes smaller for each $k > 7$ with the minimum for $k = 11$, while the maximum error oscillates and therefore adds little information. Hence, SMARTPOS ignores missing RSSI values as long as signals of at least three common APs have been measured in fingerprint and measurement.

3) Impact of Orientation Information: The most profound innovation of SMARTPOS is the usage of orientation information in a deterministic location estimation system on a smartphone. With the filtering of the fingerprints in the offline database with respect to the orientation information of the user, the complexity of the online matching can be quartered (when using the state of the art four directions for each reference position) and the accuracy and precision increased by a considerable amount. Figure 4 shows the results of the tests. The mean error is much smaller when using the orientation information and also reaches its minimum of 1.16 m for $k = 4$, while the approach without orientation information reaches its minimum of 1.31 m for $k = 9$. The minimal deviation of 0.57 m for $k = 6$ is also much smaller than the minimal deviation of 0.74 m for $k = 11$ without considering the orientation. The same is true for the maximum error, which is minimal for $k = 5$ with a value of 2.65 m when considering the user's orientation, whereas without the orientation information the minimum is 3.29 m for $k = 8$. The much smaller number of k when using the orientation approach can be explained by the fact that the number of fingerprints for comparision is quartered and each online measurement has at most 4 neighbors in the grid, while without the filtering of the user's orientation the number of neighbors can increase to a total of 16 neighbors, because 4 fingerprints are stored for each reference position. In conclusion, SMARTPOS utilizes the orientation information of the user to improve accuracy and precision of the location determination, while reducing the complexity at the same time.

4) Determination of k: Based on our experiments with SMARTPOS, we recommend utilizing an orientation-based weighted kNN approach with k, the number of neighbors, set to 4. For the comparison of measurements one should ignore all signal strength information of each AP which is missing in at least one of the measurements when at least three APs are in common. With these parameters, the system offers the lowest mean error of 1.16 m of all possible fixed assignements for k with an acceptable deviation of 0.66 m and a small maximum error of 2.74 m. However, it is shown in Section VI that by dynamically choosing k for each measurement separately the error can be further reduced.

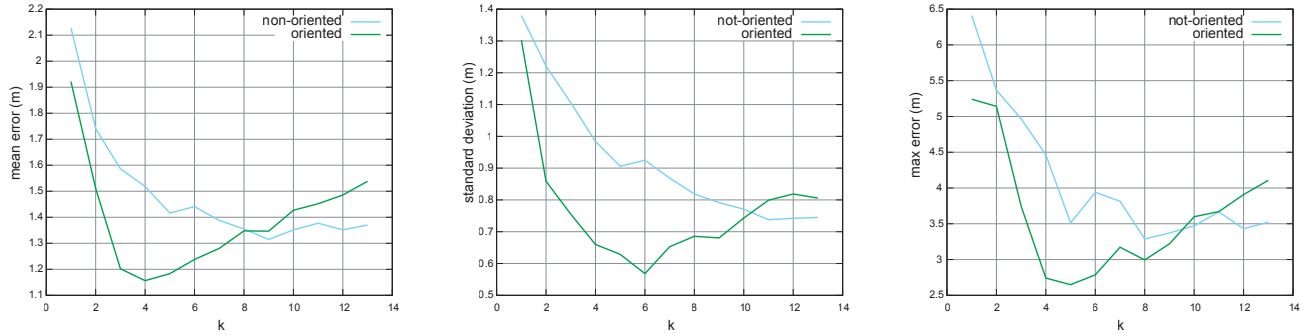
5) Orientation and the Naive Bayesian Estimator: The influence of filtering fingerprints according to their orientation on deterministic kNN positioning has been described. To get a deeper understanding of what influence the reduction of the search space according to the viewing direction has on indoor positioning, we chose to evaluate on the most simple (and often most effective) way of inducing a position from given measurements: Assuming that measurements are normally distributed, we estimate the mean and variance of a set of measurements taken in the same room and reuse this information for identification.

In order to do so, we assigned a label with each fingerprint specifying the room that it lies in. The long corridor has been cut into three rooms to reduce the variance of measurements in this long area as depicted in Figure 5. Using this labeled data we constructed a Bayesian Estimator, which calculates for each pair of AP and room label the mean RSSI, its standard deviation, weighted sum and precision and reuses them



(a) The mean error in meter (b) The standard deviation in meter (c) The maximum error in meter

Figure 3: Comparison of considering and ignoring missing RSSI values.



(a) The mean error in meter (b) The standard deviation in meter (c) The maximum error in meter

Figure 4: Comparison of considering and ignoring the user's orientation.

for classification. We tested the classification performance with 10-fold stratified cross-validation training on 90% and evaluating on the remaining 10% of the data.

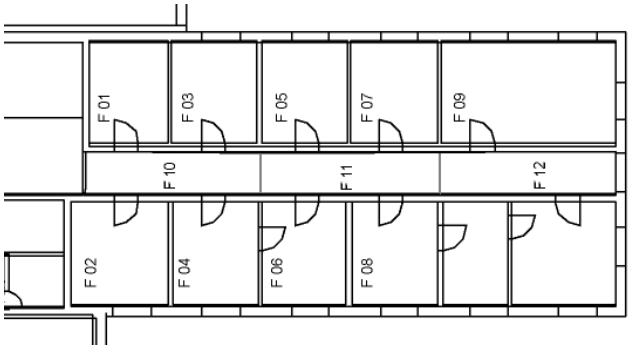


Figure 5: Labeled rooms for the Naive Bayesian Estimatior.

We used this technique on five different datasets: A dataset for each quadrant and a dataset where a random subset of 25% of all measurements in all directions were taken. In this way we achieve comparable training set sizes.

The results from this experiment are negative: A Bayesian classification of room-labels performs better on the total set of measurements than on the direction-dependent subsets. The results are given in Table I. Hence, for a system based on Bayesian estimation theory, we propose not to use the direction as a filter.

Table I: Evaluation results

Dataset	Number of Fingerprints	Success Rate
All directions	78	79%
North	72	62.5%
West	77	70.13%
East	82	65.85%
South	82	71.95%

IV. Online Error Estimation

Several factors influence the occuring positioning errors of WLAN fingerprinting systems and thus also affect the SMARTPOS system. For many application scenarios, these errors make an online error estimator necessary such as the application of SMARTPOS in Bayesian filters or scenarios of location-based access control [18]. In order to propose a good estimator for SMARTPOS, we first examine the properties of occuring errors in the first part of this section. In the second part, the online error estimator, which has been developed for the SMARTPOS system, is presented and evaluated.

A. The error distribution of SMARTPOS

In order to determine the real occuring errors, a cross-validation on the recorded reference positions has been performed. In this experiment, each of the 316 fingerprints of the reference database has been used as the current measurement m once and was blacklisted in the process of nearest neighbor selection. This tends to cause slightly increased errors, as the reference positions are reduced by m, but the larger size of samples allows to derive a stronger assumption of the error distribution. The position estimation was based on weighted kNN with $k = 4$, considered the orientation and did not punish missing RSSI, i.e., the optimal SMARTPOS setting. The results are shown in Figure 6 as a 2-dimensional histogram, representing the observed error vectors of position estimates relative to their ground truth position. The

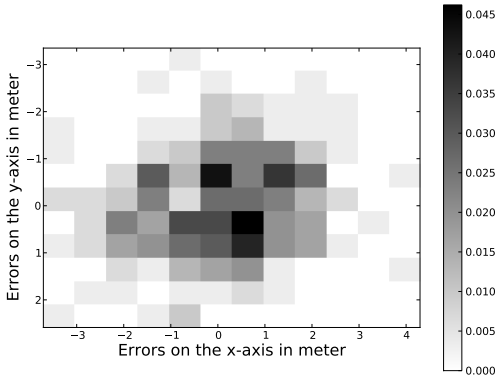


Figure 6: Histogram of the observed error distributions on the x- and y-axis. On each axis, 12 bins have been created for the 316 results.

results give good incidence to assume that the errors on each axis follow approximately a Gaussian distribution. We ignore any correlations on the axes and even the (obviously existing) differences concerning the deviation, since most services working with inaccurate position information expect a circle as error estimator. Thus we model the occuring errors as twodimensional univariate Gaussians, which is also an important constraint for the proposed estimator as shown later in this section. The individual distribution for each axis is shown in Figure 7. Obviously, the errors on the x-axis tend to have a higher standard deviation which is caused by the fact that the recorded reference positions have a larger extent on the x-axis, as depicted in Figure 1. In order to fortify the assumption of normally distributed errors a Wilk-Shapiro test has been performed for each axis for 50 randomly selected samples of the measured errors. The results showed a test statistic of $W_x = 0.960$ for the x-axis and $W_y = 0.955$ for the y-axis. Given a level of significance of $\alpha = 0.05$, the critical value of W is 0.947 for $n = 50$ which is lower than W_x and W_y. Thus, the assumption of normal distribution can not be rejected for the given level of

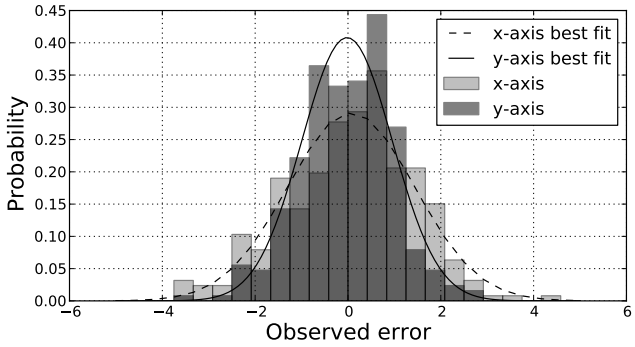


Figure 7: Histogram of the observed error distributions on the x- and y-axis.

significane, which allows to define an online error estimator for SMARTPOS based on Gaussians.

B. Estimating positioning errors

As the previous section indicated a normal distribution of errors, this section aims at giving an estimation of errors with Gaussians. The mean values correspond to the specific position fix and for each fix, an empirically estimated standard deviation is derived. For $k > 2$, three different error estimation schemes have been defined by Lemelson et. al [13] based on the coordinates $(l_1, l_2, \ldots, l_k)$ in $\mathbb{R}^2$ of k nearest neighbors in signal space. These methods estimate the error, i.e., the geographic distance of the position fix to the ground truth position. For this task, the first method σ_{m1} computes the average geographic distance of the second up to the k-th fingerprint to the nearest neighbor:

$$\sigma_{\mathrm{m1}}(l_1, \ldots, l_k) = \frac{1}{k-1} \sum_{i=2}^{k} \|l_1 - l_i\|_2 \qquad (3)$$

Another modification computes the error estimate as the maximum geographic distance of any selected neighbor to the nearest neighbor:

$$\sigma_{\mathrm{m2}}(l_1, \ldots, l_k) = \max \left(\bigcup_{i \in \{2,\ldots,k\}} \{\|l_1 - l_i\|_2\} \right) \qquad (4)$$

Finally, a third version estimates the error as the maximum geographic distance of any two fingerprints in the sequence of selected nearest neighbors:

$$\sigma_{\mathrm{m3}}(l_1, \ldots, l_k) = \max \left(\bigcup_{(i,j) \in \{2,\ldots,k\}^2} \{\|l_i - l_j\|_2\} \right) \qquad (5)$$

As proposed by Lemelson et al. [13], for each of these methods the positioning error is estimated as σ_{mi} with $i \in \{1, \ldots, 3\}$ and the user is assumed to be on a circle with radius σ_{mi} centered at the position fix l. However, in general, the distribution of errors approximately follows

a bivariate Gaussian as shown above. In order to allow a more realistic estimation of the occuring error, one could approximate the real error distribution under the assumtion that the errors E_x and E_y on both axes are uncorrelated. Given a position fix $l = (l_x, l_y)$, this allows to define two Gaussians $E_x \sim \mathcal{N}(l_x, \sigma_{mi})$ and $E_y \sim \mathcal{N}(l_y, \sigma_{mi})$ for each $i \in \{1, \ldots, 3\}$ describing a probability distribution for the ground truth position on each axis. In the following, we employ this methodology for defining an univariate Gaussian as error estimator of SMARTPOS.

For the SMARTPOS system, we propose a new error estimator which also derives an univaiate Gaussian centered at l but in contrast to σ_{mi} with $i \in \{1, \ldots, 3\}$ is not only based on the positions of the k nearest neighbors, but also on their corresponding weight and the position estimate. The basic idea is to capture the closeness of the nearest neighbors to the derived position fix, i.e., to derive an estimate for the precision. Given a measurement m at the ground truth position *gtp*, the only information that is online accessible is the estimated position l and the weight w_i of each fingerprint f_i according to the measurement m. For both axes, the standard deviation σ_{m4} is estimated as the weighted average of the distance of the l_i with $i \in \{1, \ldots, k\}$ to the position estimate l:

$$\sigma_{\text{m4}}(l, l_1, l_2, \ldots, l_k) = \sum_{i=1}^{k} w_i \|l - l_i\|_2 \qquad (6)$$

Again, a cross-validation was performed as described above with an additional computation of the error estimations σ_{mi} for the described estimators $1 - 4$. To evaluate each of these, we propose to standardize the set of observed error distances on each axis with the corresponding σ_{mi}. This allows to compare the standardized samples against the standard normal distribution $\mathcal{N}(0, 1)$. The more likely these samples were drawn from $\mathcal{N}(0, 1)$, the better the given estimator. Thus, for each position fix $l = (l_x, l_y)$ for each axis, the standard score has been computed according to $(gtp_x - l_x)/\sigma_{mi}$ and $(gtp_y - l_y)/\sigma_{mi}$. The fits of the standardized samples to $\mathcal{N}(0, 1)$ have been evaluated using qq-plots, which are depicted in Figure 8 for the x-axis and in Figure 9 for the y-axis. Compared to the straight line $y = x$, the results indicate that the derived Gaussians for σ_{m4} show the best approximation of the real error within the quantiles from -2σ to 2σ for both axes. It can also be seen that the reduction to univariate Gaussians does not prevent the estimator from fitting very well to the real error distribution on both axes. The proposed approach thus returns more accurate error estimations than the methods σ_{m1}, σ_{m2} and σ_{m3}. The few outliers suggest that the derived Gaussians tend to underestimate the probability of large real errors. These underestimations also occur with the other evaluated methods and indicate that large errors follow another not even necessarily Gaussian distribution. Nevertheless, the results show that based on σ_{m4}, the derived probability

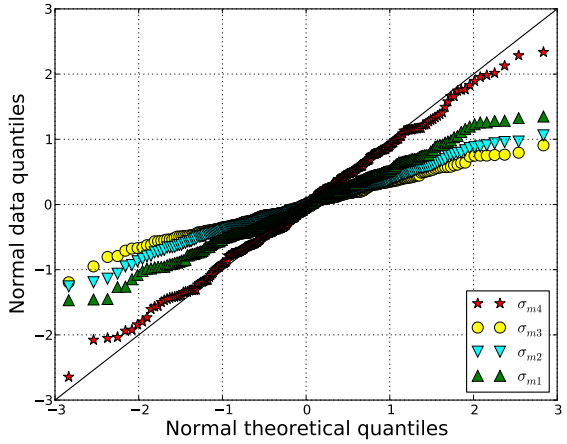


Figure 8: The qq-plot of the generated test data on the x-axis.

density functions for the ground truth position of a position fix correlate with the real error very well. The estimators 1-3

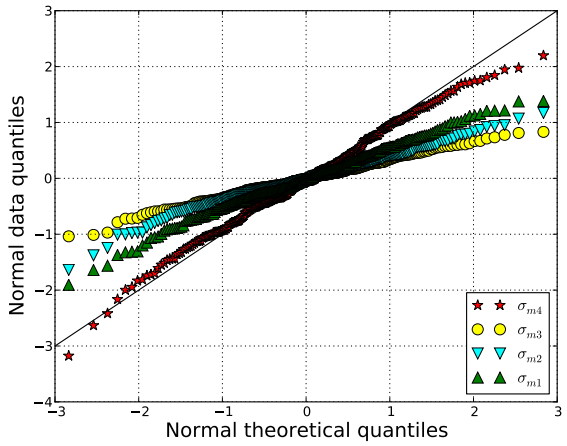


Figure 9: The qq-plot of the generated test data on the y-axis.

tend to underestimate the error with respect to the straight line $x = y$ even within the 2σ quantiles. Clearly, σ_{m3} tends to highly underestimate the errors in nearly all cases. The methods σ_{m2} and σ_{m1} perform slightly better but also tend to underestimate the errors much more than σ_{m4}.

However, we also experimented with the presented error estimators for $k \neq 4$ and observed that still the proposed estimator σ_{m4} has the best fit to $x = y$ but tends to increasingly underestimate the occuring errors with a growing k. Given the obtained results, we suggest that for a given global parameter of k, the best error estimator should be selected using the presented methodology. Compared to existing error estimators, the derived probability density functions in SMARTPOS are expected to yield more robust results in real applications.

V. Density of Radiomaps

As the mean and maximum errors of SMARTPOS are subject to the number of selected nearest neighbors, the density of recorded fingerprints in the underlying radiomap plays an important role. Hence, an interesting aspect is how the mean and maximum errors correlate to this density. To examine this correlation, an experiment has been conducted where these measures have been computed using the online testset against the presented reference database, whose density has been reduced in each iteration by 5%. In each iteration, the 5% of fingerprints to remove have been randomly chosen. If a fingerprint has been picked for removal, the other 3 fingerprints on its location have been removed too. This experiment has been conducted 20 times in sequence. The measured errors for each iteration have been merged and are depicted in Figure 10. The observed

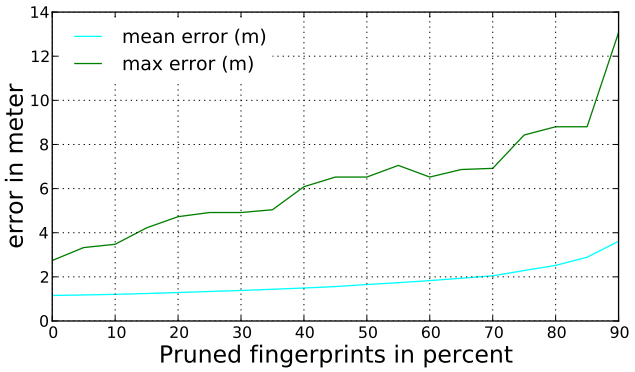


Figure 10: The average positioning error in meters for iteratively reduced fingerprint densities.

results indicate, that the density of the radiomap has much stronger influence on the maximum error as on the mean error. In detail, the maximum error already doubles for a reduction of the reference positions of $35 - 40\%$ while the same holds for the mean error for a reduction of about 80%. However, both, the mean and the maximum error triple for a reduction of 90%. The results give good reason to assume, that a more dense reference database would only show low impact on the mean error, as the mean error seems to be converging for higher densities. However, as the maximum error is increased by approximately 0.5 m for a reduction of only 5%, we expect that the maximum error of SMARTPOS could be even further reduced by a reference database with a higher density.

VI. Dynamically Calculating the Optimal Number of Nearest Neighbors

As our previous experiment has shown, the number k of fingerprints considered for the position estimation in SMARTPOS has a strong impact on the mean and maximum error of the positioning system. However, it may

function *SMARTkNN*(m, *min_k*)
  $l \leftarrow$ *weighted_center_of_mass*(*get_NN*(*1*, m))
  $k \leftarrow$ *min_k*
  while $k <$ |*fingerprints*| **do**
    kNN' $\leftarrow$ *get_NN*(k, m)
    l' $\leftarrow$ *weighted_center_of_mass*(*kNN'*)
    kNN" $\leftarrow$ *get_NN*($k+1$, m)
    l" $\leftarrow$ *weighted_center_of_mass*(*kNN"*)
    if $\|l" - l'\|_2 < \|l' - l\|_2$ **then**
      $l \leftarrow l'$
      $k \leftarrow k+1$
    else
      return l
    end if
  end while
  return l
end function

Figure 11: The proposed *SMARTkNN* algorithm.

depend on the environment and therefore scenarios exist where no analysis of an optimal k has been performed or is even impossible. Additionally, the covered site might have very diverse properties with respect to the density of recorded fingerprints, the number of receivable access points, or building specific singularities, which makes a fixed global k too inflexible. Even for very uniform scenarios, like the SMARTPOS test environment, a dynamic k might decrease the mean error. To test this hypothesis, we propose SMARTkNN as an extension of the SMARTPOS system. Its pseudo-code is shown in Figure 11.

The algorithm works by iteratively increasing k, and computing a position fix for the current k, $k-1$ and $k+1$ based on Formula 2. It iteratively continues up to that value of k after which the position fixes start to diverge. First, the location l of the nearest neighbor is determined and stored in the variable l. The variable k is initialized with *min_k*. Within the loop, the k and $k+1$ nearest neighbors are determined and corresponding position fixes l' and l'' are computed. If the distance of l' to its predecessor l is smaller than the distance to its successor l'', the loop terminates and returns k. This represents the first optimum for the number of nearest neighbors, as for larger values of k the position fixes begin to diverge. In the other case, the position fixes seem to be converging and the search for a larger k is continued. The search also terminates if the value of $k+1$ corresponds to the number of recorded fingerprints.

The SMARTkNN algorithm was evaluated with the online testset against the reference positions for lower bounds *min_k* $\in \{2, 3, 4\}$ and was compared to the $k = 4$ strategy and the *optimal* k strategy. The latter is suitable for evaluating the proposed algorithm as the theoretical optimal k can be used as a reference value. The results are depicted in Table II and as a histogram comparing the

Table II: Evaluation results

Method	Mean. Error	Max. Error	Std. Deviation
fixed $k = 4$	1.16 m	2.74 m	0.65 m
dynamic $k \geq 4$	1.17 m	2.79 m	0.60 m
dynamic $k \geq 3$	1.10 m	2.65 m	0.65 m
dynamic $k \geq 2$	1.31 m	5.14 m	0.83 m
optimal k	0.66 m	2.14 m	0.51 m

number of chosen k in Figure 12. The optimal number for k is distributed over a large interval while SMARTkNN only picked maximally 8 nearest neighbors and thus had only limited overhead compared to a strategy with a fixed value for k. Furthermore, the complexity of kNN lies much more within the distance calculation to every possible fingerprint and the sorting of the results than in the position calculation for given neighbors.

Compared to the SMARTPOS algorithm for a fixed k, SMARTkNN could slightly reduce the mean error from 1.16 m to 1.10 m and the maximum error from 2.74 m to 2.65 m for a lower bound of *min_k* $= 3$ without increasing the standard deviation. A lower bound of 2 or 4 could not improve the results compared to a fixed $k = 4$. Given Figure

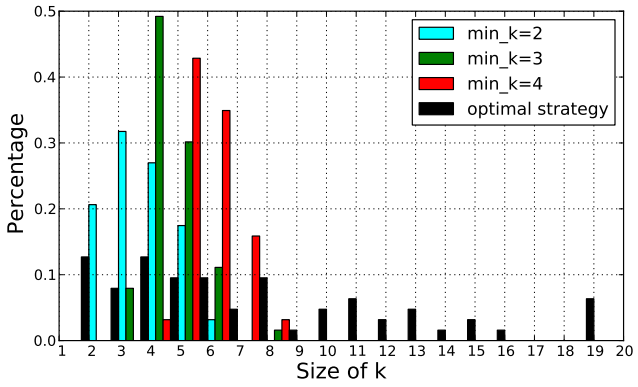


Figure 12: The distribution of the dynamically chosen number of nearest neighbors for SMARTkNN compared to the optimal strategy.

13, the cumulative error distribution of SMARTkNN with $k \geq 3$ has the best fit to the optimal strategy and especially a better fit than SMARTPOS with $k = 4$. An interesting aspect is the mean size of the dynamic k: *min_k* $= 3$ resulted in a mean of 3.5 for k with a standard deviation of 0.90, which subsequently indicates, that values from $k = 3$ to $k = 5$ were preferably selected. This fits quite well to the results observed in Figure 4, where the mean error had its minimum for $k = 4$ with very similar values for $k = 3$ and $k = 5$. For *min_k* $= 2$ and 4, in each case the mean size of k was further away from this minimum with mean values for k of 2.5 and 4.7. The optimal strategy had a mean value of $k = 4.8$ with a deviation of 6.6.

Concluding, the SMARTkNN algorithm showed improved

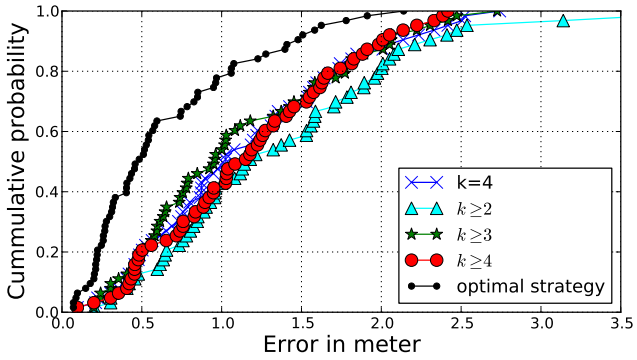


Figure 13: The cumulative error distribution of SMARTkNN compared to SMARTPOS and the optimal strategy.

results compared to SMARTPOS. However, even here we have a strong dependence on the new parameter *min_k*. As the evaluation results indicate, a minimum value of *min_k* $= 3$ should be chosen in the presented scenario. The lower bound of 3 also theoretically accounts for reducing the mean error compared to a lower k as the number of possible candidate points for position fixes is largely increased compared to $k = 2$ or $k = 1$ using the proposed computation over weighted center of mass.

VII. Conclusion and Future Work

In this paper, we presented SMARTPOS, a positioning system on a smartphone based on deterministic WLAN fingerprinting and a digital compass. SMARTPOS utilizes a weighted kNN approach with $k = 4$ and with a distance metric in signal strength space, which ignores RSSI values from access points visible only at one fingerprint. Furthermore, we proposed SMARTkNN, an extension for SMARTPOS, which uses a dynamic k instead of a fixed number of nearest neighbors. It iteratively increases k, computes a position fix for the current k, $k - 1$ and $k + 1$ and continues with increasing k until the position fixes start to diverge. In this algorithm, the first position fix is initialized with the position of the nearest neighbor, while the minimum number of nearest neighbors involved in the next fixes is a parameter of SMARTkNN.

To give an impression of the system's performance, we analyzed the impact of several parameters on SMARTPOS. We conclude that a weighted approach results in more accurate and precise results than a non-weighted approach. Ignoring missing RSSI values provides better results than assigning a minimal value, at least for higher values of k. In our setting, this was the case for $k > 3$ in the oriented approach and for $k > 7$ in the approach without the user's orientation. With adding the user's orientation, SMARTPOS is able to reduce the mean positioning error to 1.16 m and the variance to 0.66 m. The maximal error in this case is 2.74 m, which is 55 cm smaller and therefore

much better than the minimal maximum error of 3.29 m in all experiments without the orientation information. We therefore conclude that the user's orientation should be considered in deterministic 802.11 fingerprinting. The error was reduced even more by introducing SMARTkNN: The mean error was reduced to 1.10 m and the maximum error to 2.65 m without increasing the variance. The cumulative error distributions for several minimum values of k were compared to the optimal strategy and we found that the best results could be obtained for the strategy with $k \geq 3$. However, compared to a fixed $k = 4$, the improvement was quite small but we assume that the main advantage of SMARTkNN unfolds in very diverse scenarios, where a fixed k yields too much inflexibility.

Furtheremore, we examined the reduction of the density of the underlying fingerprint database. We found out that the reduction has a large negative effect on the maximum error but a much smaller influence on the mean error. The reduction of about $35 - 40\%$ doubled the maximum error while the density has a weaker impact on the mean error which doubles for a reduction of 80%.

Finally, the error distribution for SMARTPOS was evaluated and found to be normally distributed on each axis. Given this information, a novel online error estimator was defined which employs the weighted average distance of the nearest neighbors to the position fix to derive a Gaussian probability distribution. The derived distributions are univariate Gaussians centered at the position estimate. An evaluation of several existing estimators showed that our approach gives the best approximation of the real errors.

REFERENCES

[1] M. Kessel and M. Werner, "SMARTPOS: Accurate and Precise Indoor Positioning on Mobile Phones," in *Proceedings of the International Conference on Mobile Services, Resources, and Users (MOBILITY'11)*, 2011, pp. 158–163.

[2] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of wireless indoor positioning techniques and systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 37, no. 6, pp. 1067–1080, 2007.

[3] P. Bahl and V. N. Padmanabhan, "RADAR: an in-building RF-based user location and tracking system," in *19th Annual Joint Conference of the IEEE Computer and Communications Societies*, ser. INFOCOM 2000, vol. 2, pp. 775–784.

[4] M. Youssef and A. Agrawala, "The horus wlan location determination system," in *3rd International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys 2005, pp. 205–218.

[5] T. King, S. Kopf, T. Haenselmann, C. Lubberger, and W. Effelsberg, "Compass: A probabilistic indoor positioning system based on 802.11 and digital compasses," in *1st International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization*, ser. WiNTECH 2006, pp. 34–40.

[6] E. Martin, O. Vinyals, G. Friedland, and R. Bajcsy, "Precise indoor localization using smart phones," in *International Conference on Multimedia*, ser. MM 2010, pp. 787–790.

[7] K. Kaemarungsi and P. Krishnamurthy, "Properties of indoor received signal strength for wlan location fingerprinting," in *1st Annual International Conference on Mobile and Ubiquitous Systems*, ser. MobiQuitous 2004, pp. 14–23.

[8] E. C. L. Chan, G. Baciu, and S. C. Mak, "Orientation-based wi-fi positioning on the google nexus one," in *6th International Conference on Wireless and Mobile Computing, Networking and Communications*, ser. WiMob 2010, pp. 392–397.

[9] F. Evennou and F. Marx, "Advanced integration of wifi and inertial navigation systems for indoor mobile positioning," *EURASIP Journal on Advances in Signal Processing*, vol. 2006, pp. 1–11, 2006.

[10] M. Kessel, M. Werner, and C. Linnhoff-Popien, "Compass and wlan integration for indoor tracking on mobile phones," in *The Sixth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM'12)*, 2012, pp. 1–7.

[11] Widyawan, "Learning data fusion for indoor localisation," Ph.D. dissertation, Cork Institute of Technology, 2009.

[12] O. Woodman and R. Harle, "RF-based Initialisation for Inertial Pedestrian Tracking," in *Proceedings of the 7th International Conference on Pervasive Computing (Pervasive 2009)*, 2009, pp. 238–255.

[13] H. Lemelson, M. Kjaergaard, R. Hansen, and T. King, "Error estimation for indoor 802.11 location fingerprinting," in *4th International Symposium on Location and Context Awareness*, ser. LoCA 2009, pp. 138–155.

[14] C. Beder, A. McGibney, and M. Klepal, "Predicting the expected accuracy for fingerprinting based wifi localisation systems," in *Proceedings of the 2nd International Conference on Indoor Positioning and Indoor Navigation (IPIN 2011)*, 2011.

[15] M. Roshanaei and M. Maleki, "Dynamic-knn: A novel locating method in wlan based on angle of arrival," in *Proceedings of the 2009 IEEE Symposium on Industrial Electronics and Applications (ISIEA 2009)*, 2009, pp. 722–726.

[16] B. Altintas and T. Serif, "Improving rss-based indoor positioning algorithm via k-means clustering," in *Proceedings of the 11th European Wireless Conference*, 2011, pp. 681–685.

[17] B. Shin, J. Lee, T. Lee, and H. Kim, "Enhancing weighted k-nearest neighbor algorithm for indoor wi-fi positioning systems," in *Proceedings of the 8th International Conference on Computing Technology and Information Management (ICCM 2012)*, 2012, pp. 574–577.

[18] P. Marcus, M. Kessel, and C. Linnhoff-Popien, "Securing mobile device-based machine interactions by employing user location histories," in *Security and Privacy in Mobile Information and Communication Systems Social Informatics and Telecommunications Engineering (MOBISEC'12)*, 2012, pp. 81–92.

Supporting Adaptive Flexibility with Communications Middleware

Dirk van der Linden, Georg Neugschwandtner,
Maarten Reekmans, Herbert Peremans
University of Antwerp
Belgium
{*dirk.vanderlinden, georg.neugschwandtner,*
maarten.reekmans, herbert.peremans}*@uantwerpen.be*

Wolfgang Kastner
Automation Systems Group
Vienna University of Technology
Austria
k@auto.tuwien.ac.at

***Abstract*—Automation systems are continuously growing in scope and size. To keep them maintainable (despite their ever-increasing complexity), structures, methodologies and technologies with the capability of responding to diverse and changing requirements are required – a quality we call adaptive flexibility. After presenting highlights from a survey illustrating the variety of requirements, this paper discusses two approaches to supporting adaptive flexibility as well as the relationship between them. The first is OPC Unified Architecture (UA), a communications middleware standard. The second is the Normalized Systems Theory, a formal approach to ensuring systems evolvability. The paper intends to bridge the gap between theory and practice by highlighting several aspects of OPC UA that support adaptive flexibility, with this analysis being based on the concepts of Normalized Systems as far as possible.**

Keywords-Automation; OPC UA; Profiles; Scalability; Diversity; Modularity; Reusability; Evolvability; Normalized Systems.

I. Introduction

Industrial communication has become a key technology in modern industry. A continually growing number of manufacturing companies desire, even require, totally integrated systems. This integration should cover electronic automation devices such as Programmable Logic Controllers (PLCs) and microcontrollers as well as Human Machine Interfaces (HMI) and supervision, trending, and alarm software applications, e.g., Supervisory Control and Data Acquisition (SCADA) and Manufacturing Execution Systems (MES). Industrial communication encompasses the entire range from field device and controller to manufacturing operations management and Enterprise Resource Planning (ERP) applications.

Likewise, the past decade has seen a push towards the integration of building services and building management. Total integration in this field should not only cover Direct Digital Control (DDC) and SCADA/Building Management Systems (BMS), but also Computer Aided Facility Management (CAFM) applications and HMI ranging from dedicated panels and visitor guidance systems to webbased solutions on tablets and smartphones.

Such "totally integrated systems" are not monolithic or developed from scratch, but consist of multiple (sub)systems – such as those just mentioned – connected to form a (more or less) coherent whole. Connecting independent subsystems, which were developed independently, can be a veritable challenge. On the other hand, exactly this separation into independent subsystems is one of the best ways to deal with the high overall complexity of an integrated system. Thus, how a large system can be split into subsystems or modules on the one hand and how these can be connected on the other hand are topics worth exploring.

Modularity is the foundation for several desirable properties, including reusability as well as:

Scalability – the possibility to adapt the configuration of a system in order to fulfil demanding requirements but not be oversized for less demanding requirements,

Diversity – the freedom to choose between (and/or accommodate) different implementations of a particular function, and

Evolvability – the ability of a system to follow as requirements change with time, and stay maintainable.

These aspects are interrelated. For example, a system which supports diversity with regard to a particular subfunction will evolve more gracefully when another, independently implemented, instance of this subfunction needs to be merged into the system (consider, for example, adding a second printer to a PC). In the following, we will refer to all these aspects using the umbrella term *adaptive flexibility* – the ability of a system to adapt to (changing or diverse) requirements.

Adaptive flexibility is an essential quality since, generally speaking, "one size fits all" solutions do not exist – or do not really fit. There is a reason why so many specialized kinds of systems have developed: in the world of industry (and beyond), companies specialize in different tasks. These different tasks come with specific technical requirements, and companies approach them with different solutions. This variety of requirements and approaches is illustrated by the results of a survey we performed [1]. Requirements and preferred approaches are also changing over time. Thus, there is clearly a need for structures, methodologies and

technologies that are capable of supporting customization and change. Modularization is a key concept in this regard. For maximum benefit in terms of adaptive flexibility, modules (or subsystems) must be decoupled as thoroughly as possible. On the other hand, they must interoperate properly. Both are considerable and well known challenges.

Adaptive flexibility and interoperability are commonly considered valuable goals in software engineering practice. Designers of methodologies and standards typically use an intuitive approach to support these goals. OPC UA is a recent communications middleware standard, which – as will be shown below – incorporates several related design choices. OPC initially stood for "OLE for Process Control", but the current OPC UA (Unified Architecture) specifications are no longer based on OLE. OPC UA is also popular in practice: two European developer and user conferences in 2012 and 2013 gathered around 150 attendees each. Taking another angle, the Normalized Systems Theory (NST) [2] is an example for a formal approach to support adaptive flexibility, more precisely, evolvability. Its goal is to provide formal rules on how to construct evolvable software programs, instead of relying on heuristic knowledge for this purpose. This theory was developed with software architectures for business applications in mind, but has been successfully applied to other domains such as industrial control and business processes [3], [4].

This paper intends to bridge the gap between theory and practice. It examines several aspects of OPC UA which support adaptive flexibility. As far as possible, this analysis is based on the concepts laid forth by NST. This makes our evaluation of OPC UA more stringent by providing a theoretical foundation. In addition, it illustrates NST concepts by putting them in the context of a concrete implementation. Given the substantial size of the OPC UA specifications (over a thousand pages in total), this endeavour has to be explorative in nature and limited to highlighting selected aspects. The paper extends previous work which focused on the above mentioned survey and on a recommender tool relating OPC UA specification feature sets to requirements [1]. Other previous work considered the application of NST to automation systems from various angles: with respect to couplings and dependencies between subsystems [5], the design of evolvable, modular PLC programs [6], and the separation of input/output and control functions in such programs [7]. Together with insights on how some design qualities heralded by NST were intuitively built into web technology [8], this inspired the expanded discussion of the relationship between OPC UA and NST concepts which can be found in this paper.

First, OPC UA is introduced, including the profile mechanism to support scalability. Then, NST and its goals are summarized. Section IV discusses the results of our worldwide survey, showing the wide variety in application requirements and technologies. Section V considers how connecting applications via OPC UA middleware can increase adaptive flexibility. NST makes certain stipulations regarding the interface between modules. This section presents examples of why some of these stipulations are satisfied, and how others can be satisfied, when this interface is based on OPC UA. Section VI examines which internal mechanisms of OPC UA support adaptive flexibility, looking at these mechanisms from a black box perspective. Finally, Section VII uses the OPC UA stack and services as a backdrop and concrete example to illustrate how finely system implementations must be divided into modules according to NST. Section VIII concludes the paper.

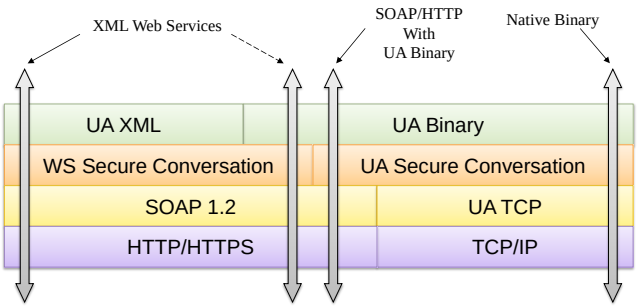


Figure 1. OPC UA transport [11]

II. OPC Unified Architecture

The OPC Foundation started in the mid-1990s to promote cross-vendor interoperability for automation projects. Initially, the OPC specifications were based on Distributed Component Object Model (DCOM) as a communication technology. DCOM is Microsoft's proprietary technology used for communication between software modules distributed across networked computers. The more recent standard family, OPC Unified Architecture (UA), is designed to be more generic, abstract, technology independent and platform agnostic [9], [10]. OPC UA is based on a cross-platform Service Oriented Architecture (SOA) and includes security mechanisms. Its two fundamental components are mechanisms for data transport on one hand and data modelling on the other hand.

The OPC UA specification contains abstract definitions of OPC UA services for data communication on the application level. Mapping these services to a concrete technology, the transport mechanisms tackle platform independent communication while still allowing optimisation with regard to the involved systems. Currently, OPC UA defines two transport mappings that are used for establishing a connection between an OPC UA client and server on the network level. UA/TCP is fast and simple and SOAP/HTTP is firewall-friendly and uses Web Services (WS). While communication between industrial controllers or embedded systems may require high performance and low overhead, business management applications may need an easily parsed data format. As a

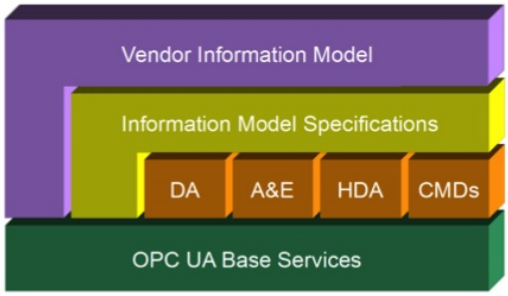


Figure 2. OPC UA information model domains [12]

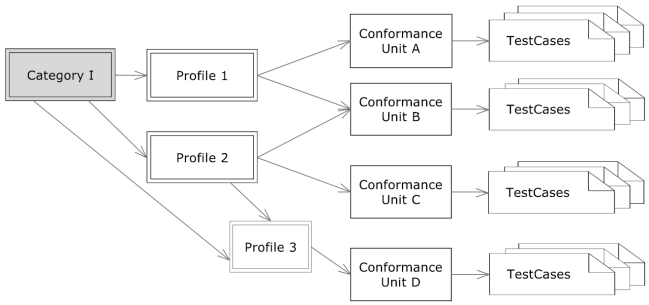


Figure 3. OPC UA Profiles and ConformanceUnits [1]

consequence, two data encoding schemes are defined, called OPC UA Binary and OPC UA XML. Different compromises are possible to find a good balance between security and performance, depending on the application (Figure 1).

The objective of an OPC UA server is to present information of an underlying (automation) process so that it can be used to seamlessly integrate with other processes and management systems. The exposed information represents the current, and possibly the historic state and behaviour of the underlying process. OPC UA defines rules and basic building blocks to expose such an information model. Basically, an OPC UA information model is made up of nodes and references that represent the relationship between nodes. Nodes can contain both online data (instances) and meta data (classes). OPC UA clients can browse through the nodes of an OPC UA server via the references, and gather data from and information about the underlying system.

The OPC Foundation provides dedicated OPC UA information models to structure the legacy OPC specifications (Figure 2). These information models support common tasks of legacy OPC interfaces [13]. These legacy interfaces are data access (DA), alarms and events (A&E), historical data access (HDA) and commands (CMDs). By modelling them with OPC UA, the transition from legacy systems to the new OPC UA communication standard is made easier. In addition, the OPC Foundation encourages definitions of complex data based on related industrial standards. Examples are IEC 61131-3 (PLC programming languages [14]), FDI (Field Device Integration) with EDDL (Electronic Device Description Language) [15] and ISA 95 (integration of enterprise and factory automation and control systems) [16]. Client software can be conveniently written against these complex data types. They also increase the potential of code re-use.

OPC UA is designed in a way that individual implementations do not need to support all features, but can be downscaled to a limited scope if desired. At the same time, advanced products which allow a high degree of freedom will require the support of more sophisticated features. A service based OPC UA implementation can be tailored to be just as complex as needed for the underlying application.

Hence, what is needed is a way to describe (and test) which features are supported by an OPC UA compliant product. A specific set of features (e.g., a set of services or a part of an information model) that can be tested as a single entity is referred to as a *ConformanceUnit*. An example of a ConformanceUnit is "Method Call", containing the call service that is used to call a method on an OPC UA server. ConformanceUnits are further combined into *Profiles*. An application (client or server) shall implement all of the ConformanceUnits in a profile to be compliant with it. Some profiles may contain optional ConformanceUnits, which in turn may exist in more than one profile (Figure 3). Software certificates contain information about the supported profiles. OPC UA Clients and Servers can exchange these certificates via services.

Up to now, more than 60 OPC UA Profiles have been released [17]. The number of released profiles is continuously being extended by OPC Foundation working groups. It is expected that, over time, also other organisations will take part in this activity.

III. Normalized Systems

In general, software gradually becomes unmaintainable as features are added over time. The theory of Normalized Systems (NST) [18] proposes an approach to counter this effect. According to [2],

> *...Normalized Systems are (information) systems that are stable with respect to a defined set of anticipated changes, which requires that a bounded set of those changes results in a bounded amount of impacts to system primitives.*

A. Software is aging

Let us consider the effort necessary to modify system S according to a change requirement (e.g., to add a feature). This requires an effort that depends on the change required (Figure 4).

Following Parnas, software is aging [19]. There are two, quite distinct, types of software aging. The first is caused by the failure of the product's owners to modify it to meet

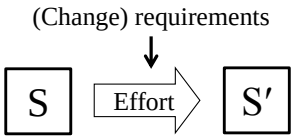


Figure 4. System evolution

changing needs; the second is the result of the changes that are made. Lehman stated that this second cause of aging is a decay of structure and formulated the law of increasing complexity [20]:

> *As an evolving program is continually changed, its complexity, reflecting deteriorating structure, increases unless work is done to maintain or reduce it.*

Thus, the effort that must be spent on a change does not only depend on the change required; in addition, it increases with time. The authors of the Normalized Systems Theory (NST) combine Lehman's law of increasing complexity with the assumption of unlimited systems evolution [3]:

> *The system evolves for an infinite amount of time, and consequently the total number of requirements and their dependencies will become unbounded.*

They admit that, in practice, this assumption is an overstatement for most commercial applications. However, it provides a theoretic view on the evolvability issue, which is independent of time. If we combine this assumption with the law of Lehman, we see that, over time, the impact of required changes will become unbounded in terms of the effort to implement them.

It is challenging to determine the detailed cause of this deterioration. Which new parts of the system contribute to the effect described by this law? In other words, why does adding a feature to the code cause more costs in the *mature* stage of the lifecycle of a system than adding exactly the same feature in the *beginning* stage of the project?

The challenge the authors of NST want to take is to keep the impact of a change dependent on the nature of the change itself, not on the size (or amount of changed or added requirements) of the system. In other words, they want to keep this impact *bounded.* The rather vague questions like "Is this change causing more troubles than another?" should be replaced by the fundamental question: "Is this change causing an unbounded effect?". The authors of NST want to provide a deterministic and unambiguous yes/no answer to this question, by evaluation whether one of the NST theorems is violated or not.

Conversely to a change with bounded impact, changes causing impacts that are dependent on the size of the system are called *combinatorial* effects in NST. Systems where changes do not cause combinatorial effects are called 'stable.' *Normalized Systems* are stable in this sense. Stability can be seen as the requirement of a linear relation between the cumulative changes and the growing size of the system over time. Combinatorial effects or instabilities cause this relation to become exponential (Figure 5). By eliminating combinatorial effects, this relation can be kept linear for an unlimited period of time, and an unlimited amount of changes to the system. In other words, to achieve stability, combinatorial effects must be removed from the system.

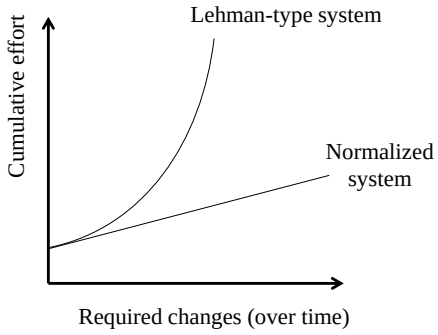


Figure 5. Cumulative change effort over time

The shape of the Lehman curve in Figure 5 is a function of the amount of combinatorial effects in the system, which again depends on the tacit knowledge of the developers and/or software architects. Since tacit knowledge cannot be measured in exact numbers by definition, it is not possible to give a mathematical definition of the shape of the Lehman curve. However, the curve surely becomes flatter when the experience or tacit knowledge of the developer rises. Indeed, a well-performed maintenance activity or 're-write' will reduce the combinatorial effects within the system or subsystem (visible in Figure 6 as discontinuities along the y-axis). In such a 're-write' activity, no extra functional requirements is added. Rather, the structure is improved in a heuristic way, involving tacit knowledge.

There is a limit to improving the shape of the Lehman curve by applying tacit knowledge. While the concepts of NST are not completely new, they make existing heuristic, "tacit" knowledge explicit. This way, it becomes possible to group and apply the (formerly) tacit knowledge of several experts, with the eventual result of reaching the goal of bounded impact.

B. Anticipated changes

Listing up all functional requirements, including those already present but not yet uncovered and those which may come up in the future, is an overly ambitious endeavour. Indeed, numerous system analysts have found that this task is impossible in an ever changing technical and economical environment. The authors of NST do not state they can do better. Instead, they propose to make use of *anticipated*

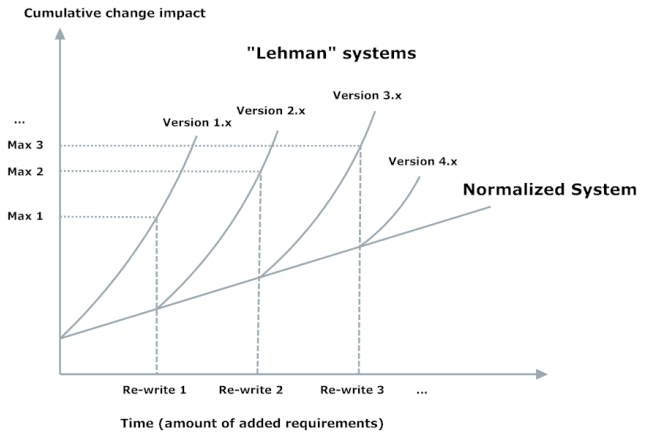


Figure 6. Reduction of cumulative effort by way of a rewrite [8]

changes. These changes are not directly associated to change or feature requests expressed by customers or managers. Instead, anticipated changes focus on *elementary* changes to software primitives: action entities, which are modules containing functionality, and data entities, which are sets of tags or fields. In this way, NST does not focus on complicated high-level changes as a whole, but on elementary changes performed on data and action entities. Typically, one real-life change corresponds to a number of elementary changes.

The set of anticipated (elementary) changes is as follows:

- A new version of a data entity;
- An additional data entity;
- A new version of an action entity;
- An additional action entity.

System changes to meet "high-level requirements" that are obtained by system analysts by traditional gathering techniques (including interviews and use cases) [18] should be converted to these abstract, elementary anticipated changes. We were able to convert all high level changes to one or more of these abstract anticipated changes in several case studies [21], [6], [7].

As an example of breaking down a high-level change into anticipated changes, consider a compressed air installation whose capacity must be increased. The installation initially consists of a single compressor. The control code for this compressor contains data fields like start and stop commands, run or failure states, or manual/automatic states and commands – a data entity – and the necessary logic to implement the appropriate behaviour associated with these data fields – an action entity.

- The change requires the addition of a second compressor. This second compressor also requires control code like the first one: this implies an instance of the anticipated change "an additional action entity" and an instance of the anticipated change "an additional data entity."
- When the new compressed air installation does not reach the desired pressure after a configurable amount of time with the original compressor only, the second shall be started in cascade. Adding this cascade logic implies an instance of the anticipated change "an additional action entity." To make sure the wear and tear is equally divided between both compressors, a permutation algorithm is implemented: at the first run, the first compressor starts, and after the next downtime the second compressor starts first – and vice versa. This permutation logic again implies an instance of the anticipated change "an additional action entity," but it also implies "an additional data entity," because the compressor which was started last has to be remembered.
- A new version of the code module that calls the original control module of the original compressor must be created to ensure that both compressors, cascade and permutation logic are encapsulated. This implies an instance of the anticipated change "a new version of an action entity" as well as an instance of the anticipated change "a new version of a data entity", because the underlying new data has to be encapsulated as well.
- Finally, we need to ensure version transparency, which means that both the old and the new version have to be supported. Indeed, this logic might be used for several compressed air installations, from which several coexisting versions need to be maintained. This implies an instance of the anticipated change "a new version of an action entity".

C. Design theorems for Normalized Systems

The design theorems or principles of Normalized Systems, i.e., systems that are stable with respect to the above set of elementary changes, are:

1) Separation of concerns: An action entity can only contain a single task in Normalized Systems.

This principle is focusing on how tasks are implemented within processing functions. Every concern or task has to be separated from other concerns – a concept with a long tradition, including Dijkstra using the term in an essay he wrote in 1974 [22]. In this way, one can focus on one aspect at a time. Tasks are identified based on change drivers: a task is something which is subject to an independent change. Whenever two or more pieces of functionality in a module can be anticipated to change independently of each other, they must be reassigned to separate modules. Section VII contains an example for a change driver to illustrate the concept.

2) Data version transparency: Data entities that are received as input or produced as output by action entities need to exhibit version transparency in Normalized Systems.

Reducing component incompatibilities between current and previous versions is a relevant research topic in software engineering (see, e.g., [23]). The version transparency theorems contribute towards addressing this challenge. Data version transparency is related to how data structures are passed to processing functions. The requirement of data version transparency is fulfilled when data entities can exist in multiple versions, without affecting the processing functions that consume or produce them. In other words, an old data entity should contain a version number, so that any functionality in a module can recognize its 'age' and tolerate that newer data fields are missing. Conversely, a new data entity should keep the fields from older versions, so that older action entities do not need to be aware of the newer fields.

3) Action version transparency: Action entities that are called by other action entities need to exhibit version transparency in Normalized Systems.

This principle is focusing on how processing functions are called by other processing functions. Action version transparency is the property that action entities can have multiple versions without affecting any of the other processing functions which call this processing function. In other words, when an older action entity calls a younger one, the younger action entity should process the call as if it would be as old as the calling action entity. Conversely, when a younger action entity calls a older module, the younger entity should expect a response corresponding to the older version.

4) Separation of states: The calling of an action entity by another action entity needs to exhibit state keeping in Normalized Systems.

This principle is focusing on how calls between processing functions are handled. The contribution of state keeping to stability is based on the removal of coupling between modules that is due to errors or exceptions. Per call, the caller should maintain a separate data entity to track the state of this call. When an action entity calls another action entity, it should not block to wait for the response of the called module, or even worse, block forever if the response is not like expected. For example, when the response message is of a newer version, which is unknown to an older calling module, the calling module should treat the response as 'unknown', rather than being blocked until the expected response arrives.

D. Versions vs. variants

The above discussion considers versions in the context of "older" and "younger" data and action entities. However, this principle can also be used to achieve a related property of evolvability: support for diversity. Here again, modules are related from a functional perspective; again, they are different versions of the same core task. However, in the case of diversity, these versions do not have a consecutive character. Instead, they are more like alternative implementations of the same task. As an example, consider different brands of variable frequency drives having different tag names, tag data types or slightly different functionality. In this context of diversity, we suggest to consider "versions" as "variants" instead.

E. Encapsulation of software entities

In Normalized Systems, the elementary action and data entities are very small. On the level of applications, there is a need for larger elements with more comprehensive functionality. To achive this, the elementary entities can be encapsulated. Different types of encapsulation are suggested [21], including:

- Action Encapsulation: It is important to be aware of the core functionality of a module. Following the separation of concerns principle, this core functionality should be separated from supporting functionality, because the supporting functionality can evolve independently from the core functionality and vice versa. Action encapsulation ensures that the core functionality and the supporting functionality are kept together, without hampering the independent evolution of the composed entities. One entity for the core task (core action entity) is surrounded by entities for supporting tasks (supporting action entities). Together, they can form an action element.
- Data Encapsulation: The arguments of the individual action entities within an action element, i.e., a number of data entities, can be encapsulated as a data element for use by the action element. The data corresponding to the functionality of the action element must be structured with regard to the action entities which the action element contains. Any action entity can read all data of all the other action entities, but, for each specific set of data, there is only one action entity which is allowed to modify it. In other words, an individual data entity corresponds to an individual action entity with regard to the permission to modify or manipulate the data. All other data entities are, for this particular action entity, read-only. In other words, the data containing a data element is composed of data entities, which are separated with regard to the permission to modify or manipulate; still, all information relevant for the entire action element is kept together in one place for reading.

Encapsulation can also be recursive: elements may in turn contain elements. In addition to action entities and data entities, NST defines flow entities, trigger entities and connection entities. Encapsulation also applies here: based on these entities, flow elements, trigger elements and connection elements can be created. Flow entities combine action entities into a sequence and act as a container for the execution states of these action entities. Depending on these states, trigger entities decide whether an action element

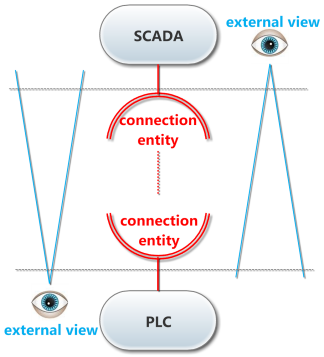


Figure 7. PLC and SCADA de-coupled by a connection entity

has to be triggered. Connection entities and elements are especially relevant in this paper and will be discussed below.

F. Connection encapsulation and migration

The separation of concerns principle imposes that the use of an external technology in an action entity implies an extra, separated task or construct, as it is possible that the external technology will evolve differently from the background technology environment of the action entity.

The concept of *connection encapsulation* allows the representation of external systems in several co-existing versions, or even alternative technologies, without affecting the Normalized System. As soon as there is no control about the evolution of an external system from the view of a Normalized System, such an external system has to be treated as a separate concern. The connection between the external system and the Normalized System is made by way of a *connection entity*. In case of an update of the external system, a new eonnection entity corresponding to the new version is added. A connection element encapsulates these connection entities and selects the one which represents the appropriate version.

As an example, Figure 7 shows a connection entity placed between a PLC and a SCADA system. De-coupling subsystems by using connection entities is an essential step to applying NST in practice: The Normalized Systems Theory promotes a high granularity of (sub)systems. It is certainly a challenging task to make existing systems, which do not have such a high granularity, comply with NST. For these systems, there is a need for a migration path towards Normalized Systems like we recognized in earlier work [8].

Modularity is in general accepted as a good engineering practice [24]. Consequently, Lehman systems typically are implemented respecting a form of modularity, albeit not granular enough to comply with the separation of concerns principle. Nevertheless, we emphasize that having a modular (Lehman) system, constructed based on rather large modules or subsystems, might be a valuable start of a migration path towards the achievement of a NST. In such a scenario,

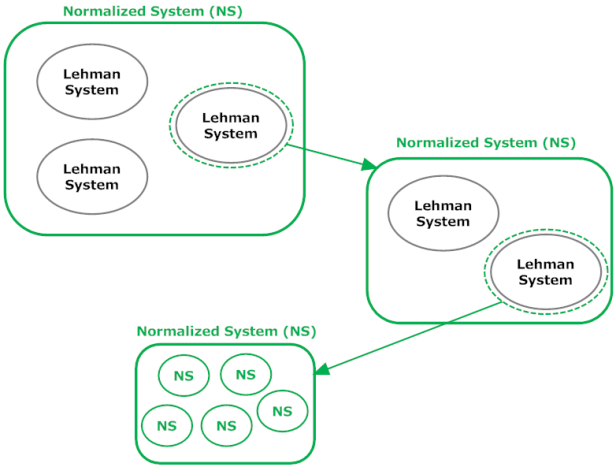


Figure 8. Migration from Lehman to Normalized subsystems [8]

one can concentrate on one single subsystem, which has an amount of couplings with the other subsystems. The first step in the migration scenario is to isolate this subsystem by inserting connection entities into each individual coupling (as illustrated in Figure 7). After isolating this subsystem, one can independently further rewrite the subsystem step-by-step towards removing all internal combinatorial effects, while the connection entities prevent these internal combinatorial effects to have an impact on the other subsystems 'outside'. Figure 8 visualizes this stepwise "fencing off" of Lehman systems.

IV. Requirements Translation

In order to understand where and how to apply adaptive flexibility, it is important to understand the type of industrial applications and how they evolve. We also need an indication of the scalability requirements of those applications and the diversity of the components used in modern day industrial applications.

OPC UA profiles can be considered modular building blocks from which OPC UA servers and clients can be constructed. Typically certain OPC UA profiles, the ones that provide the core functionality, will be used in almost all servers and clients. Other profiles will only be used for applications that require specific functionality provided by these particular profiles. Knowing which profiles are used for a certain type of application provides valuable information regarding the situations in which adaptive flexibility can be applied successfully. Currently, the choice of OPC UA profiles is pragmatically made according to the implementor's knowledge of the OPC UA specifications and the perceived requirements of the application. New concepts of OPC UA, for example information modelling, redundancy or events tend to be skipped by implementors due to a lack of awareness of these profiles.

Knowing the diversity, scalability and evolvability of industrial applications is relevant to our research, but on top of that it would be valuable to determine which OPC UA profiles can be recommended for certain branches of industry. Such a recommendation would give the application architect and project manager a guideline to deciding which profiles are most relevant to implement. This approach follows the assumption that each industry sector requires specific automation applications, resulting in a typical set of automation technologies being used and, likewise, having typical requirements on data communication within and between these technologies. Knowing which OPC UA profile (or combination thereof) is designed to fulfil given communication requirements, it should be possible to recommend a set of profiles based on the industry sector. For example, the redundancy profiles can be recommended for sectors like chemical industry, where high availability is important. Traceability is important for the pharmaceutical sector, so the Auditing profiles would be included in the recommended profiles list.

A. Worldwide survey

We designed a survey [1] to validate this assumption. The survey did not assume any detailed knowledge of OPC UA profiles on the part of the respondents, but focused on generalised questions regarding communication requirements that would allow drawing conclusions about required profiles. To make sure that these questions reflect the capabilities of the available OPC UA profiles well, we consulted one of the lead authors of the Profiles part of the OPC UA specifications for expert advice.

By analysing the results of this survey we can distill valuable information about the diversity of the applications used in modern day industrial systems. The survey should also give an indication of the requirements of scalability in these types of applications. How the requirements evolve over time cannot be seen from the survey results.

To address a representative number and kind of stakeholders, the survey was distributed to OPC Foundation members as well as companies that figure on the Foundation's regular mailing list and several other industry specific mailing lists containing a wide variety of respondents in addition.

About 25,000 questionnaires were sent out, and a total of 719 responses were collected. The geographical distribution of all respondents is shown in Figure 9. It largely matches the geographical distribution of the OPC Foundation members.

Respondents were asked to specify the industrial sectors they are active in. Multiple answers were allowed. The sectors which yielded 15% or more of responses are listed in Table I. A significant number (10–15%) of respondents reported activity in up to 8 different industrial sectors, and still 5–10% are represented in up to 5 areas.

Table I
INDUSTRY SECTORS MOST RELEVANT TO RESPONDENTS [1]

Oil & Gas Production	18%
Oil & Gas Distribution	15%
Chemical	18%
Food & Beverage	16%
Power Distribution	16%
Power Generation	20%
Building Automation	19%
Automotive Industry	16%
Industrial Automation	38%
Process Automation	30%
IT	19%

Table II
TECHNOLOGIES IN USE BY RESPONDENTS [1]

ERP	MES	SCADA	PLC	PAC	DCS	TFM	BMS	DDC
30%	24%	66%	72%	29%	43%	9%	18%	13%

When looking at the technologies reported to be used by respondents (Table II; DCS = Distributed Control System, PAC = Programmable Automation Controller, TFM = Technical Facility Management), we see that respondents clearly indicate PLC and SCADA as dominant automation technologies. Other technologies are also reported to be used extensively in combination with the dominant technologies. This is an indication that a wide variety of systems and subsystems are present in the companies questioned in this survey. Also, quite general management tasks such as alarm, event and user logging received high importance rankings among respondents.

Thus, while there is apparently the need for support of a diverse range of different systems, initial implementation effort can be significantly reduced by focusing on these technologies. Considering that many applications follow a very basic pattern, many implementers will only need to provide the so called *Core Server* profile, in combination with one *Transport* profile.

We found some key trends and assumptions behind the OPC UA technology that can be confirmed by the survey results. For example, 432 interviewees stated to be manu-

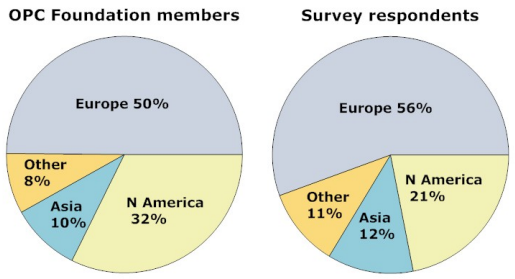


Figure 9. OPC Foundation members and respondents by region [1]

facturer of systems or products that use a communication network. 59% use a field device network, and 37% use the control network in a shared network set-up with the standard computer network. This illustrates the high importance of industrial data communications in general as well as the drive towards combined communication networks and totally integrated systems.

As far as the speed of communication is concerned, communication within less than one second is required by the majority (165/355) of PLC/PAC/DCS users. Also, the time frame for delivering data in the control network is typically short (15% say less than 1 ms; 55% say less than one second). However, a substantial percentage of PLC/PAC/DCS users (81/355) are satisfied with a delivery of data/messages within less than one minute.

This shows that on one hand, demand for fast and efficient transport as provided by *UA-TCP UA-SC UA Binary* transport profile is significant. On the other hand, a market segment exists where lower speed may well be acceptable if compensated by other desirable properties such as firewall friendly communication and an easily parsed data format, which would for example be a key property of the *SOAP-HTTP WS-SC XML* transport profile.

There is also a strong demand for security and robustness. The top three security related issues among respondents are authentication, restricted access and confidentiality of transferred data; for availability, utilizing redundant servers is seen as more relevant than deploying redundant clients.

Regarding operating systems and programming languages in use by the respondents, a technology shift begins to show. Though Windows is still the leading operating system being deployed, a trend towards Linux can be observed. Relevant programming languages are, in decreasing order of importance, C/C++, C#.NET, VB.NET, and Java. The rise of .NET indicates that DCOM is becoming a legacy technology. The use of C#.NET and C/C++ is significantly higher than the other languages ($p < 0.001$). Differences concerning the use of the programming languages in different regions are not significant (at a p-value of 0.05), which leads us to conclude that the technology shift is happening worldwide.

To confirm the suspected dependencies between industry sectors, automation technologies and communication requirements, we applied logistic regression analysis [25] to the survey results. In such an analysis, the estimates of the weight of variables with regards to a specific use provides an idea of the relevance of these variables. The results of our analysis are described in the following.

We found the use of MES to be very high in the food and beverage industry. PLC systems are being used nearly everywhere except in power distribution and IT (with negative estimates of -0.54 and -0.89, respectively). The use of DCS systems is also very diverse, except in the automotive industry, which instead shows a significant use of PAC (at an estimate of 0.70). SCADA is present in power generation, industrial automation, food/beverage and oil production, with a negative estimate for the IT sector (-0.64). Overall, PLC and SCADA are quite correlated (0.55).

Again, using logistic regression analysis, we found differences of preferences of programming languages with regard to the type of automation technology in use. The majority of Java users can be found among ERP, MES, SCADA and TFM users (as confirmed by the Hosmer and Lemeshow Goodness-of-Fit test). The majority of C#.NET users work, in decreasing order, with MES, SCADA and ERP systems. The majority of C users focus on SCADA, DDC and BMS systems. The diversity of VB users is the biggest, they work with PLC, SCADA, MES, DCS and ERP systems. The selection of these technologies is based on the analysis of maximum likelihood estimates of a simplified model with an entry cutoff value of 0.15 and a stay cutoff value of 0.15.

Concerning the most common security issues, we found a good fitting logistic regression model showing that ERP users value rogue system detection, auditability of actions, confidentiality of proprietary data and network intrusion avoidance. PLC users have different priorities, with a focus on auditability of actions, availability of systems, restricted external access to proprietary data and network intrusion avoidance. PAC users place a similar (but lower) priority on network intrusion avoidance, availability of systems and restricted external access. MES users assign high importance to preventing the alteration of proprietary data, auditability of actions, network intrusion detection and authentication of users.

The users who need a very short time frame (less than 1 ms) for delivering data/messages via the control network are mostly MES and PLC users. Those who need the fastest message exchange via the computer network (less than one second) are mostly PLC, MES and SCADA users.

We see that many companies use a lot of different target technologies and have very diverse communication requirements. There is no straightforward correlation between the industry sectors when looking at these communication requirements and technologies. A simple static set of recommended profiles according to industry branches is therefore not apparent. So, we abandoned the intention of making an industry specific, static set of OPC UA profiles.

We can see that the activities of each company are very diverse, almost independent of the industry sector. Each company uses its own approach and mix of technologies, tailored to its very diverse requirements. We focused on finding new ways to provide a clear recommendation of OPC UA profiles, taking into account the individual diverse requirements of each company.

B. Role-based optimization strategy

After processing the survey and analysing the results, we opted to approach the problem of translating requirements into profiles in another way: an online tool to dynamically

produce recommended sets of profiles on an individual, user-by-user basis. The tool is based on the generalized questions regarding communication requirements that reflect the capabilities of the various OPC UA profiles that were created for the survey. It is designed to easily accept new or updated questions to reflect newly released profiles. This *agility* is an additional advantage, as the definition of OPC UA profiles by the OPC Foundation working groups is an ongoing process.

Considering our initial goal of identifying which specific sets of profiles would add the most value for a specific company, we wanted to have the tool take into consideration the economic dimension in addition to the technical one. Each vendor has its own target market, with an assorted set of customers and specific fields of application. While many profiles might make sense from a technical point of view (and thus may well all be requested by customers), implementing some profiles will provide more commercial benefits than implementing others. Vendors must meet the challenge to find the balance between satisfying customer requirements and return on investment for implementing these profiles. To best support this decision, our tool should therefore assign a priority to each recommended profile. Also, it should be capable of linking an estimate of commercial benefits based on development time and budget to this prioritized list of recommended profiles. With this information, end-users of the tool can more accurately envision the development planning of a product even without detailed knowledge of OPC UA technologies, as this knowledge is embedded in the tool.

Thus, for getting the most relevant results, the implementation of the decision support tool takes into account normative constraints (i.e., it shall produce output that is consistent with the OPC UA specifications), budget constraints and maximum commercial benefit.

The decision support tool takes its input from three sources, each representing a particular competence or role. These inputs provide the functional parameters for the decision support tool. When the experts have entered these parameters, the end-user, who typically has little knowledge of OPC UA profiles, can use the tool to help determine the list of recommended profiles for their company and application.

The first role is that of an OPC UA expert who is determining the normative constraints. The main task of the UA expert is to input a set of survey questions and possible answers. Each answer is then linked to one or more profiles. Using these relations a profile is produced according to the answer given by a respondent to the respective question. Besides, what we call *static* normative constraints have been hardcoded into the software. Some examples of these static normative constraints are that no product can be built with only one profile and that an application must at least support one of the core profiles, one security profile and one transport profile. Another example of a static normative constraint relates to nested profiles: the basic profile must be implemented before an enhanced profile can be implemented (e.g., *Core Server* can only be implemented when *SecurityPolicy - None* has already been implemented).

The second role is that of a software architect who provides input regarding the development time required for implementing a specific profile. The software architect must have detailed knowledge of OPC UA profiles to do this. The development time is put into the tool once per profile. The end-user has to provide some additional parameters like the cost of programming labour in their company, the preferred programming language and an indication of the complexity of the application behind the OPC UA interface to get the total cost of implementation of a specific profile.

Third, the role of technical-commercial manager (sales / business) is to estimate the commercial benefit of implementing a specific profile. This commercial benefit can be estimated and used as a parameter to manage the development priority. Some of the commercial benefits can be estimated by the results of our technology survey. As mentioned, it should be noted that typically the technical-commercial role does not have enough OPC UA knowledge to estimate the benefit of a profile directly, which means that they especially profit from decision support as described in this section.

The tool was implemented and put online on a private page. We contacted one of the main contributors of the OPC UA specification part that defines the different profiles. He would take on the role of OPC UA expert. We ourselves also took on this role to come up with questions to which the answer reflects what profiles are relevant. For some basic profiles, the input of the tool is straightforward. For example, a question about redundancy needs in a company:

Does your company use any of the following? (select the ones that apply)

- Redundant servers
- Redundant clients
- Redundant communication devices
- Fault tolerant systems
- I don't know

This question polls how much need there is for the following profiles:

- Redundancy Switch Client Facet
- Redundant Client Facet
- Redundancy Transparent Server Facet
- Redundancy Visible Server Facet

With straightforward profiles questions, the answers given by the respondent can be clearly linked to certain profiles. The translation of basic requirements for communication into a list of profiles that are recommended to be implemented for the respondent was achieved by inputting these

questions and answers into the tool. There is also a mechanism implemented to assign priorities to the recommended profiles. The profiles with a higher priority are considered to be more important and more relevant to implement. So, the first important goal that we set for the tool, delivering a list of recommended profiles, is met.

Obtaining the cost of implementing a profile is one of the aspects of the tool that proved very difficult. Implementation is something specific to each company and a diverse set of parameters determines that cost. For example, in a Lehman system, adding a profile to a very simple application will require less effort than adding a profile to a complex application. Also, the programming language and possible libraries used and the experience of the developers are parameters that determine the economic impact of adding a certain profile. The cost estimation feature was not added in this version of the tool. For Lehman systems, it is by definition not possible to predict the actual costs of adding a component. So, the cost of adding a profile to an existing application, not written according to NST, can only be an estimation. We decided that the input provided by a software architect cannot result in a cost estimation that is accurate enough to be valuable information for the users of the tool.

While evaluating the usefulness of the tool, we approached several key people from the OPC Foundation and several developers, integrators and managers. The first testers of our tool reported a positive experience and saw potential in its outcome. However, the fact that not all profiles were included was reported as a problem. When validating the tool further, it did not seem to have the anticipated resonance and adoption in the OPC community. The tool can provide valuable information about basic functionality, but because of the diversity of the applications this version of the tool can not provide conclusive results.

As an outcome of the work done analysing the results of the survey and creating the tool, we can say that a large diversity of components and sub-components can be seen in industrial applications. We can also conclude that the technologies used in the industry typically have to scale well. To approach the problem of keeping these diverse applications maintainable and scalable means focusing on fundamental concepts and methodologies like adaptive flexibility. Adaptive flexibility can be achieved by applying the concepts known from NST, which have already been successfully used in business software. For creating industrial applications according to the principles of NST, OPC UA technology can provide significant support.

V. OPC UA FOR CONNECTION ENCAPSULATION

OPC UA is dedicated to providing interoperable – industrial – communication. The real production or business application functionality remains a subject of customized implementation. Implementers of this functionality can use OPC UA as an enabler to interoperate with other applications. From the perspective of these applications, OPC UA can be seen as an external technology itself; also, it provides access to other external vendor-specific or platform-specific technologies. In other words, OPC UA is an excellent match for the idea of separating technologies by way of separated tasks, as indicated in the separation of concerns principle. Again, consider Figure 7: OPC UA is excellently suited to acting as the connection entity. This subsection focuses less on the internal structure of OPC UA, but rather on the way how an OPC UA interface can block combinatorial effects and represent continuously evolving applications and/or data sources.

Following Lehman's law of increasing complexity, complexity increases and maintainability decreases when the size of a system grows. This is visualized in Figure 5. System integrators typically do not build all components of applications from scratch, but they use available commercial products as a part of their solutions. They rarely have access to the source code of these components, neither do they have the resources available to rewrite them. Therefore, it is not possible for them to make the entire solution comply with the NST theorems. However, if they manage to isolate the components from each other – or block combinatorial effects from propagating between them –, they have a better chance of staying close to the origin on the Lehman curve of subsystems (Figure 5). In other words, they can reduce the impact of combinatorial effects by keeping the size of the subsystems small. This situation is similar to our proposed migration scenario in Figure 8.

While OPC UA middleware could be implemented as a Normalized System, this is not likely to occur soon. Despite the fact that NST is derived from heuristic knowledge, it is very unlikely that developers who are not aware and familiar with it can implement a fully evolvable system. However, OPC UA does separate applications, and block combinatorial effects. Note that a distributed system, composed of a diversity of applications, interconnected with OPC UA, is actually not in conformance with the NST theorems, at least not if we regard this distributed system as a whole – it is not "fully normalized". Still, OPC UA can be a valuable tool to separate the subsystems. In other words, an OPC UA layer can separate Lehman systems, while being a Lehman system itself (and, thus, not complying with the NST theorems).

Consider, for example, an evolution scenario where two smaller systems should be interconnected to form a larger system. The larger system should support the diversity introduced by integrating these two systems, while allowing each of the original two systems to continue evolving independently. Expanding on this example, consider that a module of the first system should be reused in the second system – it will be challenging to keep this module compatible with both systems while they continue to evolve independently. We think it is possible to meet this challenge when (sub)systems

are based on OPC UA.

A. OPC UA as an integration bus

In the early nineties, the development efforts to access automation data in devices increased while the market was becoming more and more global. The number of different bus systems, protocols and interfaces used to access automation data was called 'uncountable' [26]. In that period, this access was typically based on so-called product-specific drivers. In [27], a product-specific driver is defined as follows:

> *Product-specific drivers describe software components that have been developed for a specific product. They are linked to this product so that they cannot be used with products of other manufacturers. These drivers make available data in a manufacturing-specific form.*

When building solutions in a cross-vendor environment, the 'uncountable' number of product-specific drivers becomes problematic. Addressing this problem was the main motivation of the OPC Foundation task force (1994) to develop a connectivity standard, which became the (classic) OPC standard.

In [2], the use of a messaging or integration bus is stated to be a manifestation of the separation of concerns principle. Replacing product-specific drivers with an OPC interface fulfils the purpose of an integration bus. In a cross-vendor environment, accessing a vendor's product through a coupling using a product-specific driver can be seen as a violation of the separation of concerns principle. Indeed, when an amount of SCADA products are available on the world market, the introduction of any new type of PLC (or PLC access protocol) requires all these SCADA products to be updated in order to be able to support this new type. As the amount of available products increases, the impact of this combinatorial effect becomes worse.

B. Version transparency in the address space

When two (sub)systems are connected via OPC UA, the server arranges data related to its application functionality in the OPC UA address space, where the client can access it. Generally speaking, the OPC UA address space is a collection of nodes and references between those nodes; in simple cases, it takes the form of variable values arranged in a tree structure. However, OPC UA supports advanced data modelling in the address space, including complex objects, method invocation, and type hierarchies.

A change in the server system does not affect how the OPC UA interface works. Neither does it necessarily affect where the client can find data from the server system on this interface: the server need not reflect every change in its address space. As long as the change is not relevant to the part of the system exposed by the server, it will be completely hidden from the client. This does not only apply to changes due to system evolution over time, but also to differences due to diversity. The standard information models (for example, [14], [16]) further strengthen this concept, making the representation of certain data vendor independent.

The structure of an OPC UA address space is very flexible, which allows various ways of supporting evolvability. We can obtain version transparency by simply limiting the changes we allow a server to make to its address space over time: no nodes should be modified or deleted. Rather, the evolution of an address space should be based on additions only. This way, a client which is not aware of the change can continue accessing older existing properties, attributes and references and ignore the new information.

If necessary, the server could also place a new version of its data in a new branch of its address space, while continuing to maintain the information in the original branch for non-agile clients, which will in turn continue to access the data in the way they were developed. If the client is more recent than the server in such a scenario, the "older" server will inform the client that the branch holding the new version does not exist in its address space by answering the client request with the StatusCode BadInvalidArgument. In addition, the server could also place information about the versions it supports at a well-known place in its address space, taking advantage of the flexible structure of the OPC UA address space. Either way, both versions can coexist, whether in-place or in separate parts of the address space.

The previous discussion assumes that the client disposes of pre-configured information about which data to expect where in the server address space. In use cases where this does not apply, the standardized meta information in the OPC UA information model can provide considerable benefit. Object types allow a client to recognize instances of entities in previously unknown places. In such a scenario, type hierarchies enable diversity and version transparency. For example, a client can interact with an unknown motor controller as long as it conforms to a known supertype (representing a basic motor controller interface). The client can identify the instance in the address space by following the HasTypeDefinition and HasSubtype references.

OPC UA also standardizes metadata related to address space versions which a server has the option to expose. The NodeVersion attribute of a Node changes when a node or reference is added or deleted (for example, when a variable node is added below an object node) or when the DataType attribute of a variable or VariableType changes. Clients should be aware of this change: in case of a deletion or data type modification, they may otherwise obtain wrong data. This can be prevented by continually checking the NodeVersion for changes; however, such an approach is not efficient in terms of communication. Therefore, OPC UA specifies the ModelChangeEvent to inform clients that they should expire their node cache and rebrowse the relevant

part of the address space. Related to this mechanism are the DataTypeVersion attribute, which gives information about changes to the definition of data types with custom encoding, and the SemanticChangeEvent for changes to Value Attributes of Properties. These changes also need to be monitored by the client, but are not covered by NodeVersion and ModelChangeEvent, respectively.

Note that if the address space evolves in a version transparent way, following the theorems of Normalized Systems, the mechanisms outlined in the previous paragraph are not required: since previous versions remain accessible to the client, there is no need to be aware of model changes. As a practical consideration, however, a process similar to garbage collection will be needed in the long run in case one wants to delete obsolete nodes, properties or attributes. This process would need to be based on warranties that these entities will no longer be used by any client.

Finally, OPC UA also allows accessing historical address space. This is implemented on top of the Views concept, which allows to present clients with subsets of the address space tailored to a specific purpose (e.g., access to maintenance relevant data only). The ViewVersion parameter in the Query services enables a client to browse previous versions of the address space, including nodes which have in the meantime been deleted. This allows accessing historical variable data attached to such nodes, which would otherwise no longer be reachable. While this mechanism does not appear to be intended to enable access to current data via an older interface (i.e., View) version, nothing would prevent a server to update Variable Nodes in the historical address space with current values and thus use this mechanism for the purpose of version transparency.

C. Profiles and dependencies

The concept of modularity is most commonly associated with the process of subdividing a system into several subsystems [28]. This decomposition allows modifications at the level of a single subsystem instead of having to adapt the whole system at once [24]. To achieve the ideal form of modularity of a system, the underlying subsystems should be loosely coupled and independent [29]. If some dependencies are inevitable, they should be described and made explicit for the user of the subsystem. This enables the user to decide whether to satisfy the dependency or to not use the subsystem.

As an example, consider a program which requires a particular runtime library: you may decide to add the library to your system, or you may rather use another program if the required library would be in conflict with a library version you have in use (and the library management in your system cannot handle this situation). Another example, but from a different application domain, would be a multiplexer component to be placed on a new integrated circuit (IC): for providing its logic function, it depends on the particular IC manufacturing process it was designed for.

A subsystem behind an OPC UA interface is accessed in the form of services provided over a network. The dependencies, therefore, are:

1) A network stack (in the current specifications, TCP/IP).
2) A library implementing the OPC UA protocol. With OPC UA, this is usually comprised of a “stack” for lower level functionality and an “SDK” (Software Development Kit) between this stack and the application.
3) Application logic to interact with the address space (via the SDK: reading/writing values, update notifications, ...).
4) Application logic to map application data structures into the address space.

The network connection between the OPC UA server and client may be a machine-local loopback interface; a physical network interface is required when server and client run on different computers.

As already discussed in previous sections, the OPC UA protocol is comprehensive and complex. The Stack, SDK and application are free to only use a subset of it. Dependencies 2 and 3 in the list above therefore scale to the requirements of the application. The OPC UA specification groups related functionality into profiles. Profile implementations can be modules – units of functionality. Since it is possible to select and implement only a subset of them, this is a manifestation of the separation of concerns principle: if they would not address separate concerns, one would be forced to implement all of them.

It is possible for an OPC UA client and server to have different sets of profiles implemented, but still be able to communicate. This also means that a client or server can have various communication partners that each support a different profile subset. OPC UA contains mechanisms to enable meaningful communication by selecting matching profile subsets for both communication partners. The mechanisms allow one subsystem (the client) to select the best way to interact with another subsystem (the server), based on which profiles each counterpart supports. This is a manifestation of version transparency at the level of communicating applications. It is also a way of exhibiting diversity – supporting different versions at the same time.

One group of profiles concerns transport functionality, with the ability to choose between UA/TCP or SOAP/HTTP, OPC UA Binary or OPC UA XML, and various levels of communication security (security policies). The client can query the server for available endpoints before attempting to establish a communication channel. Each of these endpoints corresponds to a transport configuration; they will depend on the capabilities of the server stack and SDK as well as the server configuration (it may choose to not offer, for example, insecure communication for policy reasons

although stack and SDK would have the ability). The client then checks which of the endpoints matches its capabilities and preferences best and uses this endpoint to establish a secure channel and session. This mechanism enables the server to clearly document its requirements – for example, in terms of acceptable connection security; the client can decide to satisfy the resulting dependency by making the necessary functionality available on its side (for example, via an appropriate library) or decide to not use the server.

Once the session is established, the client has various options how to interact with the server address space. The OPC UA specifications require that a server always supports the most basic set of operations, which are sufficient to retrieve a list of related profiles implemented by the server from a well-known place in its address space. Each of these profiles corresponds to a set of services supported by the server, for example, whether it supports alarms and conditions or if it allows a client to modify the address space. The profile documents which services can be invoked on the server, and in which ways they can be invoked.

The client is not forced to use any of this "special" functionality unless it chooses to. In case it does, matching client profiles are described in part 7 of the OPC UA specifications, describing the dependencies in terms of protocol functionality which the client must provide on its side to take advantage of these server features. In addition, the client can determine from the list of profiles supported by the server which services it may invoke without receiving an error message. To ensure that client and server stacks and SDKs from different developers will communicate flawlessly as long as they implement a compatible set of client and server profiles, the OPC Foundation has established a compliance and certification program for interoperability testing.

Profiles are identified by unique Uniform Resource Identifiers (URI). If the functionality associated with a profile is modified in the future, the updated version will again receive a unique URI, reflecting the new version. This means that profile versions can coexist.

Thanks to the mechanisms and documentation described, an OPC UA client can gather comprehensive information about the dependencies which are caused by its wish to communicate (or communicate efficiently) with a server. No dependencies are hidden; the OPC UA server is a true "black box" in the sense of [29]. It is fully specified in terms of an outside view on its functionality; a client should never have to examine how the server is implemented to be able to use its services. The OPC Foundation also maintains a "profile reporting" website [17] which is documenting the functionality associated with profiles (again, in terms of OPC UA services, not their implementation).

However, these mechanisms and documentation only cover functionality related to the OPC UA interface. By design, OPC UA profiles do not describe anything related to the actual application functionality "behind" the OPC UA interface (although standardized information models such as [14] or [16] blur this distinction somewhat). An example for such a dependency may be that the subsystem requires periodic license payments to continue working.

As an example from another domain, [5] mentions manufacturer and type codes printed on ICs and calls for a website to provide standardized information about well defined dependencies – much in the way that it is possible to find a datasheet describing an IC based on this type code. Such a website could be similar in concept to the OPC Foundation profile reporting website, but focus on functionality outside the area of OPC UA; the OPC UA address space could easily accommodate URIs pointing to "datasheets" about modules.

OPC UA leaves developers entirely free to handle dependencies which may be introduced by certain data presented in the address space. It also leaves them free to decide how to add information about such dependencies to the address space. However, it provides structures which can prove useful in this context, such as the advanced data modelling features already mentioned in the beginning of this section. For example, consider the case that a server (subsystem) is updated to expose a new widget. A client (subsystem) can immediately access the input and output data of this widget, but to use it efficiently, it must know how these inputs and outputs work together – the widget function. If the server adds type information to the widget object, the client can recognize it as the kind of meta-data which describes this function, and can decide whether to bring in complementary functionality on its side to make use of it. Still, the client will always remain in control of the choice whether to satisfy the dependency or not use the new widget.

VI. Adaptive flexibility in OPC UA

While the previous section discussed the role of OPC UA for de-coupling subsystems from the perspective of the server and client applications, this section focuses on the "internal mechanisms" of OPC UA. How do these mechanisms, which are usually hidden from the server or client application by the OPC UA SDK application programming interface (API), support adaptive flexibility? Again, the view taken is as far as possible a "black-box", functional one, independent of how the mechanisms described in the OPC UA specifications are implemented.

A. Message passing

When (sub)systems are connected via OPC UA, all communication between them uses message passing following the request-response paradigm. Message passing requires that a request is stored in a separated (message) memory. Sender and receiver do not share a memory space; all passing of values has to be by copy, not reference. For this reason, message passing systems are also referred to as "shared nothing" systems. This greatly reduces the possibility for

undesired coupling to occur. Consider, for example, the situation that a software module is linked with another module which, by accident, contains a global variable of the same name, but different meaning or purpose. Such a name clash is impossible when these modules communicate via OPC UA, because the two variables will necessarily exist in separated memory spaces in this case. Moreover, the request-response message paradigm is naturally asynchronous in character. Unlike with a subroutine call within the same memory space, the module invoking the service always remains in control of its own program flow.

If implemented right, this basic principle ensures separation of states between the two subsystems and will block combinatorial effects from propagating between them. To this end, both the OPC UA middleware implementation (stack and SDK) and the application module which is using it for communication must react to these messages in a robust manner: most importantly, they must not count on the expected response to arrive promptly.

Let us consider this rule in further detail. First, the response may not arrive promptly, but with a significant delay, or it may not arrive at all. The subsystem must always take this possibility into account, it must not block unconditionally. Within OPC UA, timeouts for services and lifetimes for shared state between client and server (e.g., a logical channel or session) are specified. These timeouts can be negotiated between the communication partners to address varying response time requirements.

Second, if a response arrives, it may not be the expected one. It can also be an error message or an unknown response – unknown maybe because the communication partner was updated in the meantime. Such a situation must be handled as gracefully as possible. The subsystem must not hang or crash as a result.

Version transparency is a key quality in this context. If version transparency is implemented, response messages cannot be unexpected or unexpectedly absent – provided that the communication channel is reliable and that the entity on the other side is not malfunctioning. This is because with version transparency, it is the responsibility of the caller to only invoke the callee in ways that the callee can understand. In other words, in a scenario of controlled evolution, the caller must respect the version of the called entity. This is a logical consequence of the fact that when a fundamentally new requirement is added during the course of evolution of a system, older entities by nature will not have the capability to satisfy it. For example, a system designed to heat will not necessarily be able to cool as well, and a labeler for square boxes cannot label a round box.

However, an external system (such as one behind a connection element based on an OPC UA interface) may evolve in an uncontrolled way. It need not respect the version transparency theorems; the change in the external system may require our own system to adapt. This would cause a combinatorial effect. The connection element must block this combinatorial effect as far as possible, allowing us the choice of not adapting our own system; instead, we may want to display or log an error message saying the external system is not responding like it should – without causing our own system to crash or malfunction. The connection element must therefore catch unexpected responses to achieve version transparency. This is greatly facilitated by a mechanism which allows the older communication partner to declare that it does not understand or support a request, and give this response in a way that the younger can recognize it as such. For this purpose, OPC UA defines standardized response StatusCodes and gives the option to add free-form DiagnosticsInformation to further describe the reason of an error status.

B. Technology mappings

Already back in 1972, Dijkstra recognized difference in scale as a major source of our difficulties in programming [30]. Moreover, a widespread underestimation of the specific difficulties of size seems to be one of the major underlying causes of software failure.

An important design goal for OPC UA was scalability in terms of communication requirements. Small embedded devices should be allowed to contain a very basic OPC UA interface, while more powerful platforms (such as PC-based systems) should be able to provide more complex functionality. Considering their communication requirements, different levels in the automation pyramid are best served by different transport technologies. Communication on the level of ERP applications requires intermittent transport of large amounts of data at high data rates, while applications hosted in small embedded devices typically require continuous transport of small amounts of data with low latency (around 10 ms down to 10 μs).

OPC UA is defined independent of a particular network protocol and low-level encoding. It can be mapped to the most appropriate transport corresponding to the needs of an application. While this does not make it a full alternative to fieldbus systems (as no special provisions are made for time-deterministic communication), it certainly allows OPC UA to scale over a large range of requirements.

Being able to choose between transports does not only improve scalability. It also supports diversity: multiple transports may be able to fulfil a particular set of requirements. And with requirements changing and solutions improving over time, such an option to choose also introduces new possibilities with regard to the property of evolvability.

In 1994, DCOM was a good base technology for OPC. Over time, it became a limiting factor. DCOM was a vendor dependent technology and restricted the choice of operating system and programming language. Choosing a network protocol (and message encoding) as the basis already supports

multi-platform and cross-programming-environment communication better, since the concerns of sending a message to the communication partner and acting upon this message are separated. However, this still would not have allowed diversity with respect to transports. Therefore, a further step was taken: OPC UA was based on an abstract service oriented architecture. Part 4 of the specifications specifies the services in abstract terms, and Part 6 specifies the current transport technology mappings. The API is not standardized, giving the freedom to provide the modest appropriate solution for any programming language or framework.

The technology mappings do not only apply to communication protocols and message encodings, but also to security algorithms for encryption and authentication. By decoupling the specification of the OPC UA services from these implementation choices, as well as decoupling them from the programming environment, the "what" and "how" are very clearly separated. This manifestation of the separation of concerns principle is a very visible improvement in the process from classic OPC towards OPC UA.

Thanks to it being based on abstract service definitions, the stable core purpose of OPC UA – providing access to values and events in another subsystem – is decoupled from its more rapidly envolving technology environment. The choice to standardize a number of technology mappings stems from the necessity to prevent a completely fragmented landscape of implementations, which would not be interoperable with each other. Within these standardized technology mappings, profiles serve the purpose of selecting compatible, complementary subsets of diverse functionality. Finally, interoperability testing addresses the possibility of different interpretations of the specifications.

C. Further improvements over OPC Classic

In comparison with the classic OPC specification, several further improvements towards conformance with the principles of separation of concerns and version transparency can be found in OPC UA. Some of them are highlighted in the following.

First, one of the reasons why the authors of OPC UA used the word 'unified' in the name of the standard is that they unified previously different ways of accessing information. Classic OPC defined independent specifications, each covering a part of the functionality now provided by OPC UA. These specifications each had slightly different ways of doing the same thing, such as browsing the available data tags. By offering a unified way of access to this information, OPC UA provides increased reusability.

OPC UA clearly separates actions and objects. For example, OPC Classic defined a special GetStatus method to obtain status information about a server. In OPC UA, this information is modeled as the server status variable, is placed at a defined position in the server namespace, and can be accessed with the read service or monitored for changes just like any other variable node. This is clearly a manifestation of separation of concerns.

Layering is explicitly mentioned in [2] as a manifestation of the Separation of Concerns principle. In OPC UA, layering can be found in many places. For example, subscriptions (to notifications for changing values in the address space) are separated from an underlying session, and a session is separated from an underlying secure communication channel. These entities maintain their own state, which is independent of the state of the layers below them: subscriptions can be transferred between sessions (which supports smooth transfers between redundant clients or servers), and a session can switch over to another secure channel without being terminated (which enables transparent use of redundant network links).

Another example of layering can be found in the way how most OPC UA middleware is split from an implementation point of view. Basic protocol functions are covered by the OPC UA Stacks, which are made available by the OPC Foundation for its members. On top of such a Stack, commercial SDKs (Software Development Kits) are developed and marketed by OPC UA expert companies. Using these SDKs as a basis, developers create OPC UA client and/or server applications.

Finally, the current transport mappings for OPC UA contain a number of examples for action version transparency support. The OPC UA TCP Hello and Ack messages, which open every conversation between a client and a server using this protocol, contain ProtocolVersion parameters, and the server is required to accept newer protocol versions. The OPC UA Secure Conversation OpenSecureChannel service messages also include ClientProtocolVersion and ServerProtocolVersion parameters. If the web service mappings are used, SOAP as well as the WS-* standards encode the protocol version in every message by way of specifiying the XML namespace URI, which uniquely identifies the version. HTTP headers also contains a version field. Using this version information, the server can correctly interpret messages from an older client, or reject the request in a controlled manner if it has an unknown, more recent version. HTTP/1.1 [31] also specifies the Upgrade header, which a client can use to indicate to the server that it supports another version of the protocol (or other protocols in general) which it would like to use if the server finds this appropriate.

VII. Normalized modules within OPC UA middleware

As discussed in Section IV.B, we found it difficult to obtain cost estimates for implementing individual OPC UA profiles in an application program. Apparently, it is a problem to obtain useful estimates of the development effort, with or without the use of existing tools, in any programming language or development environment. Why could none of the experts, who we truly consider having acknowledged

excellence in their domain, provide concrete answers to our question?

The initial idea of the decision support tool included the assumption that it is possible to estimate the development effort or cost of implementing an individual OPC UA profile in a reliable way – without considering the specific context of such an effort. Estimating the cost of a software project is already a difficult task when context information (such as the current state of the software or the skills of the available developers) is known, but we sought a generic answer, independent of which other OPC UA profiles may possibly already exist in a considered implementation, and also independent of the size of the (existing) system.

This expectation is similar to what was expected from system analysts and software architects for over decades, i.e., listing up all the old, recent and future functional requirements of a system. Doing so seems to include very uncertain assumptions: in particular, the assumption that all dependencies of a newly added or changed feature are under control, are explicitly known, can be measured and estimated. The exponential character of the curve in Figure 5 illustrates that this assumption cannot be defended. The precise slope of this curve for a system is impossible to determine; therefore, it is impossible to give a precise estimate for any given system size. It is for this reason that the Normalized Systems Theory focuses on elementary, anticipated changes instead (as introduced in Section III-B).

It can be safely assumed that the experts' experience only concerned Lehman-type systems of some kind. In such a system, the development effort or cost of an OPC UA profile is not only a function of the functional requirements of the OPC UA profile itself, but also of the size of the (existing) system, including dependencies and potentially existing combinatorial effects. For a Normalized System, a reliable cost estimate would be much easier to give.

In general, the Normalized Systems Theory calls for radical separation of concerns and thus promotes a high granularity of modular systems. Even if this goal cannot be immediately attained, it will typically be a benefit to keep modules small in order to keep the impact of changes low. For purposes of illustration, the remainder of this section therefore explores how finely an implementation of OPC UA should be divided into modules to achieve separation of concerns. Given that the actual application (and all program logic to connect the OPC UA interface with it) is a separate concern for sure, we will focus on units of functionality defined in the OPC UA standard, which will usually be implemented by a Stack and/or SDK.

When looking at the OPC UA profiles specifications, we find four types of entities: profile categories, profiles, facets, and conformance units. Profile categories have no meaning in terms of software constructs, but only exist to structure the specifications for the purpose of readability. They provide major functional groupings, such as all profiles necessary to implement a complete server or client. Likewise, the distinction between facets and profiles has organizational purposes: a facet refers to a profile which is expected to make part of a larger profile. For the purposes of this discussion, facets and profiles can be treated the same.

Profiles describe features of applications. We have already identified profiles as a manifestation of the separation of concerns principle in Section V.C. In other words, when building an OPC UA application, it is possible to select and implement (or not) each individual profile. The features represented by OPC UA profiles are well separated and the profiles make explicit for a communication partner which features are supported and which are not. Profiles are the smallest unit of coherent functionality from the perspective of an OPC UA communication partner.

Profiles are comprised of ConformanceUnits. Conformance units are the foundation of the OPC compliance and certification program They are defined as a specific set of features that can be tested as a single entity. OPC UA Compliance testing activities include functionality testing, behavior testing, interoperability testing, load testing and performance testing. Conformance units are the smallest entities in the OPC UA standard from the perspective of compliance testing.

However, a ConformanceUnit can cover a group of services, portions of services or information models by definition. And, actually, services are the smallest entities which can be invoked by an OPC UA communication partner; with profiles as the smallest unit of coherent functionality from this perspective, they could be considered the smallest unit of individual functionality from the perspective of an OPC UA communication partner.

When examining how finely an implementation of OPC UA should be divided into modules to achieve separation of concerns, we must however take an "inside" instead of an "outside" view. Up to now, we have been taking a functional, or "black box" perspective. From this perspective, it is described what a module does, i.e., what its function is. Actually, functionality testing, which we mentioned in the context of ConformanceUnits, is also called black box testing: the test candidate is seen as one single black box – from the "outside". On the other hand, the constructional (or white box) perspective describes the subsystems of which a system consists, as well as the way in which these subsystems collaborate in order to bring forth the function as described in the black box perspective [29]. The Normalized Systems Theory contains rules with relation to the functional content of a module, the interaction between data and action entities, and the mutual interaction of action entities. In other words, it is concerned with the constructional perspective – the "inside" of a module.

If we consider an OPC UA service from the constructional perspective, i.e., from a developer's point of view, we realize that it requires some lower-level functionality. For example,

the network packet containing a service request or response has to be assembled (or disassembled), taking the correct message encoding into account. The basic encoding structure is the same for all services. For example, all messages contain a header identifying it as an OPC UA message. This common header is an example for a change driver as introduced in Section III.C. When it changes (for example, to reflect a new version of the OPC UA protocol), this affects all service implementations. It must therefore be reassigned to a separate module.

This example illustrates that it is a challenging goal to reach the high granularity required for Normalized Systems. A migration strategy focusing on selected interfaces and the encapsulation of external systems offers a transitional path.

VIII. Conclusion and outlook

Adaptive flexibility is an essential quality in modern information and communication systems. In our survey, we saw a great variety in requirements regarding industrial communication all over industry. While we were able to correlate OPC UA feature sets (i.e., profiles) with application requirements easily, the survey did not yield a conclusive set of profiles associated with any single field of application.

We consider OPC UA to be a highly valuable tool for supporting scalability, diversity and evolvability – that is, adaptive flexibility – in the domain of industrial communication and, in general, everywhere it can be used to de-couple subsystems with compatible communication requirements. When gauging OPC UA against the principles of Normalized Systems Theory, this becomes even more evident.

We find that the considerable tacit heuristic knowledge of the authors of OPC UA has lead to an amount of manifestations of several fundamental principles of Normalized Systems, such as separation of concerns. The user of OPC UA takes advantage of these manifestations without even being aware of them. OPC UA also does not prevent the separation of states between subsystems it connects.

For the same reason, some of the manifestations discussed may appear to be nothing but "established good programming practice" to seasoned software developers, who possess a significant amount of relevant tacit knowledge. The goal of NST is to make this knowledge explicit, and the discussion in this paper should also serve to illustrate the connection between theorems and established practice. In addition, it should be considered that embedded systems, including industrial automation systems such as PLCs, trail behind the forefront of software development to a varying, but significant extent: in some of these environments, name conflicts between global variables in modules are still an everyday problem. In this sense, the drive of advanced technologies such as OPC UA towards the embedded level can certainly be considered a welcome evolution. First SDKs for OPC UA on embedded systems are already commercially available, but remain a relevant target of ongoing work.

We also identified some features within the OPC UA specifications which can support adaptive flexibility and system-wide evolvability – both through their originally intended use as well as for applying NST concepts. According to personal communication, some of these features are still rarely used by practicioners. In particular, this seems to be true for the version attributes of the OPC UA objects. We feel that raising awareness of these features among practitioners, users and developers alike, could be beneficial for the industry.

After all, towards a truly Normalized System, the principles of the theory must also be followed in any application built on top of OPC UA. While OPC UA encourages applying these principles by way of its design, it does not enforce them. OPC UA stack, SDK and application programmers must still do their part to prevent the system from "hanging".

Certainly, no stack, SDK or application program can be expected to become a fully Normalized System soon. Among other reasons, one faces the challenge of introducing new paradigms to practitioners when deploying Normalized Systems. Those who are not aware of NST theorems will find it harder to read and debug "normalized" code. However, an OPC UA layer can separate Lehman systems while being a Lehman system itself. OPC UA is an excellent foundation for connection elements, which again can be the base for a stepwise migration strategy towards Normalized Systems.

Due to the complexity of the OPC UA specifications, their evaluation in this single paper could not be complete. We believe there is ample room for future work to further investigate and elaborate them against the light of NST. Also, a deeper investigation of potential combinatorial effects could contribute to improving the quality of OPC UA applications.

Acknowledgment

This paper is part of the project IMA, funded by the Flemish Agency for Innovation by Science and Technology (IWT). The authors wish to thank all respondents of the survey and the OPC Foundation for their constructive collaboration. In particular, the authors would like to thank Paul Hunkar, one of the lead authors of the OPC UA profiles specification, who has provided valuable consulting expertise.

Acronyms

A&E	Alarms and Events
API	Application Programming Interface
BMS	Building Management System
CMD	Commands
DA	Data Access
DCOM	Distributed Component Object Model
DCS	Distributed Control System

EDDL	Electronic Device Description Language
ERP	Enterprise Resource Planning
FDI	Field Device Integration
HDA	Historical Data Access
HMI	Human Machine Interface
HTTP	Hyper Text Transfer Protocol
IC	Integrated Circtuit
IP	Internet Protocol
MES	Manufacturing Execution System
NST	Normalized Systems Theory
OPC	Open Productivity and Connectivity
PAC	Programmable Automation Controller
PLC	Programmable Logic Controller
SCADA	Supervisory Control and Data Acquisition
SDK	Software Development Kit
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
TCP	Transmission Control Protocol
TFM	Technical Facility Management
UA	Unified Architecture
URI	Uniform Resource Identifier
WS	Web Service
XML	eXtensible Markup Language

REFERENCES

[1] D. van der Linden, M. Reekmans, W. Kastner, and H. Peremans, "Towards agile role-based decision support for OPC UA profiles," in *Proc. 7^{th} International Conference on Internet and Web Applications and Services (ICIW 2012)*, 2012, pp. 40–45.

[2] H. Mannaert and J. Verelst, *Normalized Systems. Re-creating Information Technology Based on Laws for Software Evolvability*. Koppa, 2009.

[3] D. van der Linden and H. Mannaert, "In search of rules for evolvable and stateful run-time deployment of controllers in industrial automation systems," in *Proc. 7^{th} International Conference on Systems (ICONS 2012)*, 2012, pp. 67–72.

[4] D. Van Nuffel, "Towards designing modular and evolvable business processes," Ph.D. dissertation, University of Antwerp, 2011.

[5] D. van der Linden, H. Mannaert, and P. De Bruyn, "Towards the explicitation of hidden dependencies in the module interface," in *Proc. 7^{th} International Conference on Systems (ICONS 2012)*, 2012, pp. 73–78.

[6] D. van der Linden, H. Mannaert, and J. de Laet, "Towards evolvable control modules in an industrial production process," in *Proc. 6^{th} International Conference on Internet and Web Applications and Services (ICIW 2011)*, 2011, pp. 112–117.

[7] D. van der Linden, H. Mannaert, W. Kastner, and H. Peremans, "Towards normalized connection elements in industrial automation," *International Journal On Advances in Internet Technology*, vol. 4, no. 3&4, pp. 133–146, 2011.

[8] D. van der Linden, G. Neugschwandtner, and H. Mannaert, "Industrial automation software: Using the web as a design guide," in *Proc. 7^{th} International Conference on Internet and Web Applications and Services (ICIW 2012)*, pp. 67–72.

[9] *OPC Unified Architecture Specification*, OPC Foundation Std., Release 1.02, 2012–2013.

[10] *OPC unified architecture*, International Electrotechnical Commission Std. IEC 62 541, 2010–2012.

[11] R. Armstrong, "Implementing UA," OPC Unified Architecture Developers' Conference and Workshop 2008, Oct. 2008.

[12] J. Luth, "OPC UA Architecture," OPC Unified Architecture Developers' Conference and Workshop 2008, Oct. 2008.

[13] T. Hannelius, M. Salmenpera, and S. Kuikka, "Roadmap to adopting OPC UA," in *Proc. 6^{th} IEEE International Conference on Industrial Informatics (INDIN 2008)*, 2008, pp. 756–761.

[14] PLCopen and OPC Foundation, *OPC UA Information Model for IEC 61131-3, Release 1.00*, 2010.

[15] D. Grossmann, K. Bender, and B. Danzer, "OPC UA based field device integration," in *Proc. SICE Annual Conference*, 2008, pp. 933–938.

[16] OPC UA ISA-95 Working Group, *OPC UA for ISA-95 Common Object Model, Release Candidate 1.01.00*, 2013.

[17] "OPC UA Profiles," http://www.opcfoundation.org/profilereporting/, last accessed June 2013.

[18] H. Mannaert, J. Verelst, and K. Ven, "The transformation of requirements into software primitives: Studying evolvability based on systems theoretic stability," *Science of Computer Programming*, vol. 76, no. 12, pp. 1210–1222, 2011.

[19] D. Parnas, "Software aging," in *Proc. 16^{th} International Conference on Software Engineering (ICSE-16)*, 1994, pp. 279–287.

[20] M. Lehman, "Programs, life cycles, and laws of software evolution," *Proceedings of the IEEE*, vol. 68, pp. 1060–1076, 1980.

[21] H. Mannaert, J. Verelst, and K. Ven, "Towards evolvable software architectures based on systems theoretic stability," *Software: Practice and Experience*, vol. 42, no. 1, pp. 89–116, 2012.

[22] E. W. Dijkstra, "On the role of scientific thought," in *Selected writings on computing: a personal perspective*. Springer, 1982, pp. 60–66.

[23] I. Yoon, A. Sussman, A. Memon, and A. Porter, "Testing component compatibility in evolving configurations," *Information and Software Technology*, vol. 55, no. 2, pp. 445–458, 2013.

[24] C. Y. Baldwin and K. B. Clark, *Design rules: the power of modularity*. MIT Press, 2000.

[25] S. Sharma, *Applied Multivariate Techniques*. John Wiley & Sons, 1996.

[26] W. Mahnke, S. H. Leitner, and M. Damm, *OPC Unified Architecture*. Springer, 2009.

[27] J. Lange, F. Iwanitz, and T. Burke, *OPC – From Data Access to Unified Architecture*. VDE-Verlag, 2010.

[28] D. Campagnolo and A. Camuffo, "The concept of modularity within the management studies: a literature review," *International Journal of Management Reviews*, vol. 12, no. 3, pp. 259–283, 2009.

[29] P. De Bruyn and H. Mannaert, "Towards applying Normalized Systems concepts to modularity and the systems engineering process," in *Proc. 7^{th} International Conference on Systems (ICONS 2012)*, 2012, pp. 59–66.

[30] E. W. Dijkstra, "Notes on structured programming," in *Structured programming*, O. J. Dahl, E. W. Dijkstra, and C. A. R. Hoare, Eds. Academic Press Ltd., 1972, pp. 1–82.

[31] IETF, "Hypertext transfer protocol – HTTP/1.1," Jun. 1999, RFC 2616.

Evaluation of an Architecture for Providing Mobile Web Services

Marc Jansen
Computer Science Institute
University of Applied Sciences Ruhr West
Bottrop, Germany
marc.jansen@hs-ruhrwest.de

***Abstract*—As the role of mobile devices as Web Service consumers is widely accepted, already today a large number of mobile applications consume Web Services in order to fulfill their task. Still, no reasonable approach exists as yet, to allow deploying Web Services on mobile devices and thus use these kinds of devices as Web Service providers. In this paper, our approach is presented that allows deploying Web Services on mobile devices by the usage of well-known protocols and standards. In order to achieve this, the presented approach overcomes problems that usually occur when mobile devices are used as service providers. Here, the description of an implementation is presented, along with first performance tests and an evaluation of the battery consumption that results in using the presented approach. The performance test shows that the described approach provides a reasonable way to introduce Web Service provisioning for mobile devices, but the results for the battery consumption provide some challenges that need to be met, e.g., the determination and evaluation of scenarios that benefit from using mobile Web Services. Last but not least, this paper provides first ideas how complex mobile scenarios can be evaluated in order to decide whether they benefit from using mobile Web Services.**

Keywords - mobile devices; Web Services; mobile Web Service provider, battery consumption, scenario development.

I. INTRODUCTION

As already explained in [1], a need for a technology that allows deploying Web Services on mobile devices is necessary. In recent years, the number of reasonably powerful mobile devices has increased dramatically. According to [2], 216.2 million smartphones where just so sold in Q1 2013 worldwide.

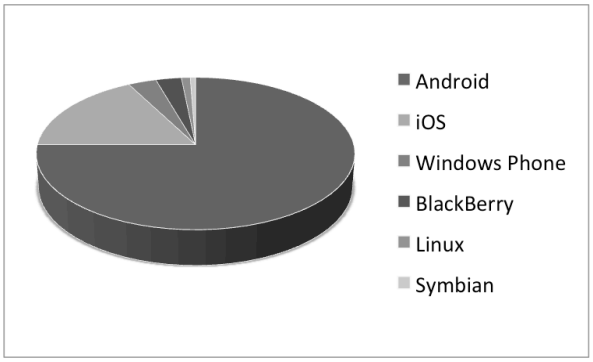


Figure 1. Distribution of different operating systems for smartphones in 2013.

On the other hand, this huge number of smartphones represents a large number of heterogeneous devices with respect to the operating systems smartphones are currently using. According to [3], there were at least five different operating systems for smartphones available on the market in 2010, and their distribution is shown in Figure 1. It thus seems to be necessary to have a platform-independent mechanism for the communication with services provided by smartphones in order to not re-implement each service for each of the mentioned operating systems.

Usually, Web Services are used in order to provide a standardized and widely used methodology that is capable of achieving a platform-independent way to provide services. Unfortunately, in contrast to consuming Web Services on mobile devices, providing Web Services on mobile devices is not yet standardized due to several problems that occur when a service runs on a mobile device. To change this was one of the major motivations for the work described in this paper. Providing a framework that allows to deploy standardized Web Services on mobile devices provides big advantages for a number of different mobile technologies, e.g., in order to contextualize mobile users.

Therefore, this paper presents the description of a framework that allows providing Web Services on mobile devices. The outline of the paper is as follows: the next section provides an overview of related work and the motivation for the development of the described approach, after which the scenario - together with the problems that usually occur if Web Services should be provided by a mobile device - is explained. The following section explains the implementation of the framework in detail and the results of a first performance test are presented. Afterwards, the power consumption of the presented approach is evaluated and discussed with respect to user acceptance and possible types of scenarios that benefit from consuming Web Services deployed to mobile devices. Furthermore, another section provides first ideas about how different scenarios can be determined and evaluated with respect to the question if these scenarios would benefit by consuming Web Services deployed to mobile devices. The paper is closed by a conclusion and an outlook for further research questions.

II. MOTIVATION

The idea of providing Web Services on mobile devices was probably presented first by IBM [4]. This work presents a solution for a specific scenario where Web Services are hosted on mobile devices. More general approaches for providing Web Services on mobile devices are presented in

[5] and [6]. In [7], another approach, focusing on the optimization of the HTTP protocol for mobile Web Services provisioning, is presented. Importantly, none of the mentioned approaches manages to overcome certain limitations of mobile devices, as demonstrated in the next section.

The major difference between previous research and the approach presented in this paper is that, to the best of our knowledge, previous research focused very much on bringing Web Services to mobile devices by implementing server side functionality to the mobile device in question. The approach presented here follows a different line: from a technical and communication point of view, the mobile Web Service provider communicates as a Web Service client with a dynamically generated Web Service proxy. This approach provides an advantage for overcoming certain problems with mobile Web Services as described in the next section. Furthermore, this approach does not rely on an efficient server side implementation of Web Services on the mobile device, and thus allows to implement a very lightweight substitution to a common application server where a common Web Service is running.

Since nothing comes for free, this approach has some drawbacks as well, e.g., it implements a polling mechanism that permanently polls for new service requests. Therefore, this approach produces an overhead with respect to the network communication and the computational power of the mobile device. The computational overhead, though, can be dramatically reduced by adjusting the priority of the polling mechanism according to the priority of the provided Web Service.

Another drawback of the presented approach is that it relies on a publicly available proxy infrastructure for the part of the framework that dynamically generates the Web Service proxies. This drawback can be overcome if, for example, mobile telecommunication companies provide this kind of infrastructure centrally.

In contrast to the aforementioned approaches, the approach presented in this paper differs with respect to one major aspect: from a network technical point of view there is no server instance installed on the mobile device. Therefore, a certain Web Service client does not call the Web Service on the mobile device directly but calls a centrally deployed proxy. The Web Service running on the mobile device polls in regular intervals for any new message requests of interest. The sequence of the Web Service request from the client point of view and from the Web Service point of view is shown in the sequence diagram in Figure 2.

The exact sequence of the different messages and events will be described in more detail later. Since especially polling mechanisms cause certain drawbacks, one of the major questions concerning the presented approach is the question of benefits and drawbacks of the polling mechanism and, in particular, whether the benefits justify the drawbacks.

One of the major problems of dealing with Web Services on mobile devices is the fact that mobile devices often switch between networks. Therefore, the Web Service running on a mobile device is usually not available under a fixed address, a fact that leads to a number of problems for the consumer of a mobile Web Service: Besides the usual network switch, the fact that mobile devices are usually not meant to provide 24/7 availability, but are designed towards providing the user with the possibility to exploit certain services, e.g., phone calls, short messages, writing and receiving emails, etc., yields the problem that mobile devices might get switched off by the user. Hence, not only that the provided Web Service might be available under different network addresses, but it might not be available at all.

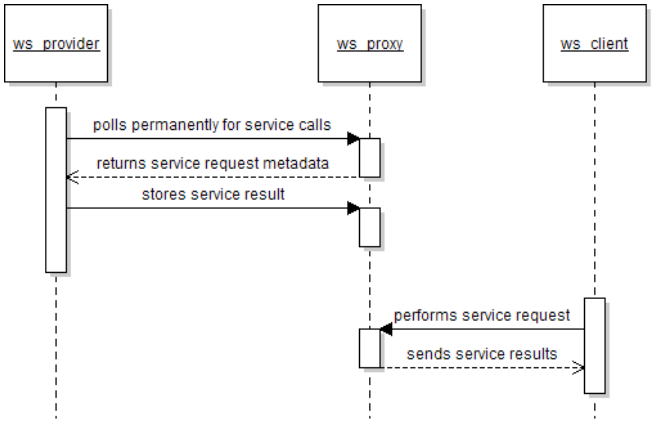


Figure 2. Sequence diagram of the Web Service requests in the presented approach.

All these drawbacks can be solved by using the approach presented here. By using the central proxy, the service requests of a certain Web Service client can be stored and if the mobile Web Service is running, it can pull for service requests that are of interest to it. Since from a technical point of view the Web Service provider only acts as a client to the Web Service proxy, the potentially changing network addresses of the mobile device does not pose a problem at all.

In addition, one of the major drawbacks of the described polling mechanism can be limited by adjusting the priority of the Web Service running on the mobile device, resulting in a lower frequency of the polling for the service request.

To conclude, in our opinion, the advantages of the described mechanism justify the drawbacks that are inherent to the approach.

III. Scenario Description

The major idea behind the implementation of the middleware is to provide a Web Service proxy, according to the proxy design pattern [8], in order to overcome certain problems in mobile scenarios as described by [9]. One major problem here is that mobile devices often switch networks, e.g., at home the mobile phone might be connected to a Wi-Fi network, at work the connection might be established through another Wi-Fi network and on the way home from work the mobile phone might be connected to a GPRS/UMTS-network. Each of these different networks provides different IP addresses and possibly different

network scenarios. For example, it can be private IP addresses with network address translation (NAT), where the Web Services running on the device are not directly accessible from the internet, or public IP addresses.

Frequently switching between IP addresses, and therefore frequently changing IP addresses (as occurs especially with mobile devices), might raise certain problems for the provision of Web Services, since the client of a certain service always needs to know the actual IP address at which the service can be reached. More than that, within a private network the provided Web Services are usually not reachable at all from the internet. Therefore, the problem, from the client point of view, is that the service is not always accessible under the same (and constant) IP address. The presented approach provides a solution to overcome this problem, with the exception of the case when a device is completely switched off. The switch off problem can be overcome as well, in which case slight modifications to the presented approach, together with an asynchronous call of the Web Service, are necessary.

The approach presented here suggests solving these problems by implementing a Web Service proxy that dynamically creates a proxy for each Web Service that gets deployed on a mobile device. The created proxy allows receiving service requests as a representative to the actual service and storing a service request along with the necessary data. In the next step, the mobile Web Service provider continuously polls for requests to its services, performs the services and sends the result back to the dynamically generated Web Service proxy. Receiving the result, the Web Service proxy can send the result back to the client that originally performed the service request.

IV. IMPLEMENTATION

The major goal of the work presented here is to provide a solution to the described scenario. Therefore, we implemented a middleware that allows the provision of Web Services on mobile devices. Here, the standard protocols, e.g., WSDL for the description of the Web Service interface, SOAP/REST as the standard network protocol and http as the usual transport protocol, are used such that there is no additional effort on the client side for requesting a mobile Web Service.

The following three sections provide a short introduction to the services offered by the middleware, followed by a description of the communication between the mobile Web Service provider and the Web Service client/consumer. Last but not least some details are presented about the Java based implementation for the test scenario.

A. Use-Case Analysis

In order to achieve the goal of implementing a Web Service proxy, an analysis of use-cases that this proxy will have to support has been performed. The result of this analysis is shown in Figure 3.

Relations in this Use-Case diagram reflect the interaction between the different use cases or an actor and the use case. From a technology point of view four different actors participate in the scenario.

A.1 The Web Service Provider

Obviously, a provider for the mobile Web Service is necessary. This is a piece of software running on the mobile device that provides the Web Service itself. This piece of software can best be compared with an application server hosting a Web Service in a scenario where the Web Service is provided by a common server system

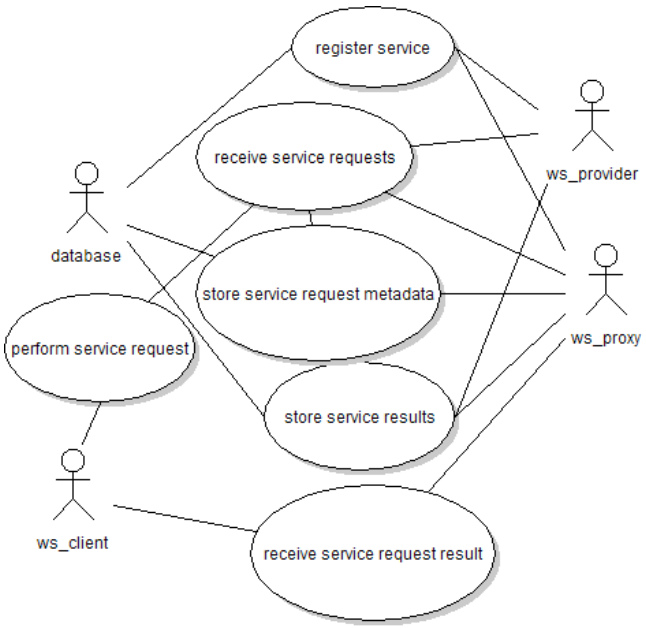


Figure 3. Use-Case description of the developed middleware.

A.2 The Web Service Client

The second quite obvious actor is the consumer of the Web Service: the Web Service client. This is a piece of software running on the client side, performing requests to the Web Service.

A.3 The Web Service Proxy

As already described, one of the major ideas of the presented approach is to provide a proxy for the Web Services provided by the mobile devices. Therefore, the Web Service proxy is another actor that participates in the scenario. The proxy represents a surrogate of the Web Service provided by the mobile device. The basic function of this proxy is to implement the same interface (same methods with identical parameter lists and return values) as the Web Service itself. Moreover, the methods provided by the proxy (in order to register a service, de-register a service, etc.), should be accessible via the standard network protocols of Web Services and the description of the proxy interface should also be available in WSDL (in the implementation here the SOAP protocol was chosen). The proxys' major task is to receive client requests, store them in a database and wait for the mobile Web Service to provide the result of the service request. While in the traditional proxy pattern the proxy would directly forward (push) the incoming service requests to the Web Service, we have decided to just store the requests in a database in order to allow the mobile Web Services to pull the requests from the proxy. This change to

the traditional proxy pattern basically allows handling constantly changing network connections (as explained before), since within this approach neither the Web Service proxy nor the Web Service client need to know the actual IP address of the mobile device that provides the actual Web Service.

A.4 The Database

Fourth and last, the database is taken to be an actor of the middleware. Usually, the database would more likely be modeled as a system (and not as an actor), but for the sake of clarity and consistency, we decided to model the database also as an actor in the system. The major task of the database is to store the necessary information about the service request in order to allow the Web Service running on the mobile device to perform the requested task, and to later-on store the return values of the service request as well. By storing also the return value, the Web Service proxy is able to send the result back to the client that made the request. This is necessary since the usage of the proxy is transparent to the client, in the sense that the client is not aware that the actual service request is not answered by the proxy, but by the Web Service running on the mobile device. Therefore, the Web Service proxy needs to send the result of the service to the Web Service client, and not the mobile Web Service itself.

Besides the four actors, a number of use-cases need to be implemented in order to fully run the described scenario.

A.5 Service Registration

First of all, a mobile Web Service provider needs to be able to register a service to be provided. Besides the Web Service provider, the Web Service proxy and the database are interacting within this use-case, too. The Web Service proxy needs to dynamically implement the interface of the mobile Web Service and the storage of the metadata (basically the name of the method that should be called and its parameter values) of the service requests. The database needs to provide certain storage for the parameter values of each method (in case of a relational database: a table) and the according return values of the mobile Web Service.

A.6 Receive Service Requests

The second, quite obvious, use-case is that the mobile Web Service provider needs to be able to receive service requests. Besides the mobile Web Service provider, the Web Service proxy participates in this use-case also, since this is the instance that directly receives the requests from the Web Service client and stores the necessary information in the database.

A.7 Perform and Receive Service Requests

Two additional use-cases, namely, perform service requests and receive service request results, participate in the store service request metadata use-case.

A.8 Storing Service Metadata and Handling of Return Values

Additionally, we have identified two other use-cases that are necessary for the handling of the service request metadata (store service request metadata) and the handling of the return values (store service result). The first of these two use-cases interacts with two actors: the Web Service proxy and the database; the second one additionally interacts with the Web Service Provider.

Beside the fact that the provision of these use-cases allows the implementation of the described scenario, one of the major advantages of this approach is that the Web Service client only interacts with the performed service request and receives corresponding answers from the service request result use-case. Therefore, from a client point of view, the request to a mobile Web Service is no more than a usual service request. No additional effort is necessary on the client side in order to receive results from a Web Service running on a mobile device.

B. Communication between the mobile Web Service and its clients

In order to explain the necessary communication for a service request from the Web Service client to the mobile Web Service provider, we modeled the communication flow within the sequence diagram shown in Figure 4.

Within the sequence diagram we have modeled an object life line for each of the actors, to be discussed later. First of all, the mobile Web Service provider needs to register its service with the Web Service proxy. As part of the service registration process the Web Service proxy creates the necessary data structure for storing the service requests in the database.

After the mobile Web Service provider has registered its service, it permanently polls the Web Service proxy for new service requests. The Web Service proxy asks the database if a new service request for the respective mobile Web Service provider is available and if so, returns the request's metadata to the mobile Web Service provider. After receiving the metadata of a new service request, the mobile Web Service provider performs the service and sends the result of the service to the Web Service proxy that directly stores the result in the database.

From a client point of view, the Web Service client simply calls the service provided by the Web Service proxy. While receiving a new service request, the Web Service proxy stores the necessary request metadata in the database. Afterwards the Web Service proxy directly starts to poll the database periodically for the result of the respective service request. Once the mobile Web Service provider has finished performing the request and has stored the result (via the Web Service proxy) in the database, the Web Service provider is able to send the result of the service request back to the client.

C. A sample implementation

In order to test the described approach with respect to its performance, we implemented the Web Service proxy in Java. Additionally, the mobile Web Service provider was implemented for Android. Here, we focused on an intuitive and easy way for the implementation of the Web Service, and have therefore, oriented ourselves by the JAX-WS (Java API for XML-Based Web Services), as described in the Java Specification Request 224 (JSR 224). The major idea, adapted from JAX-WS, was that a Web Service can easily be

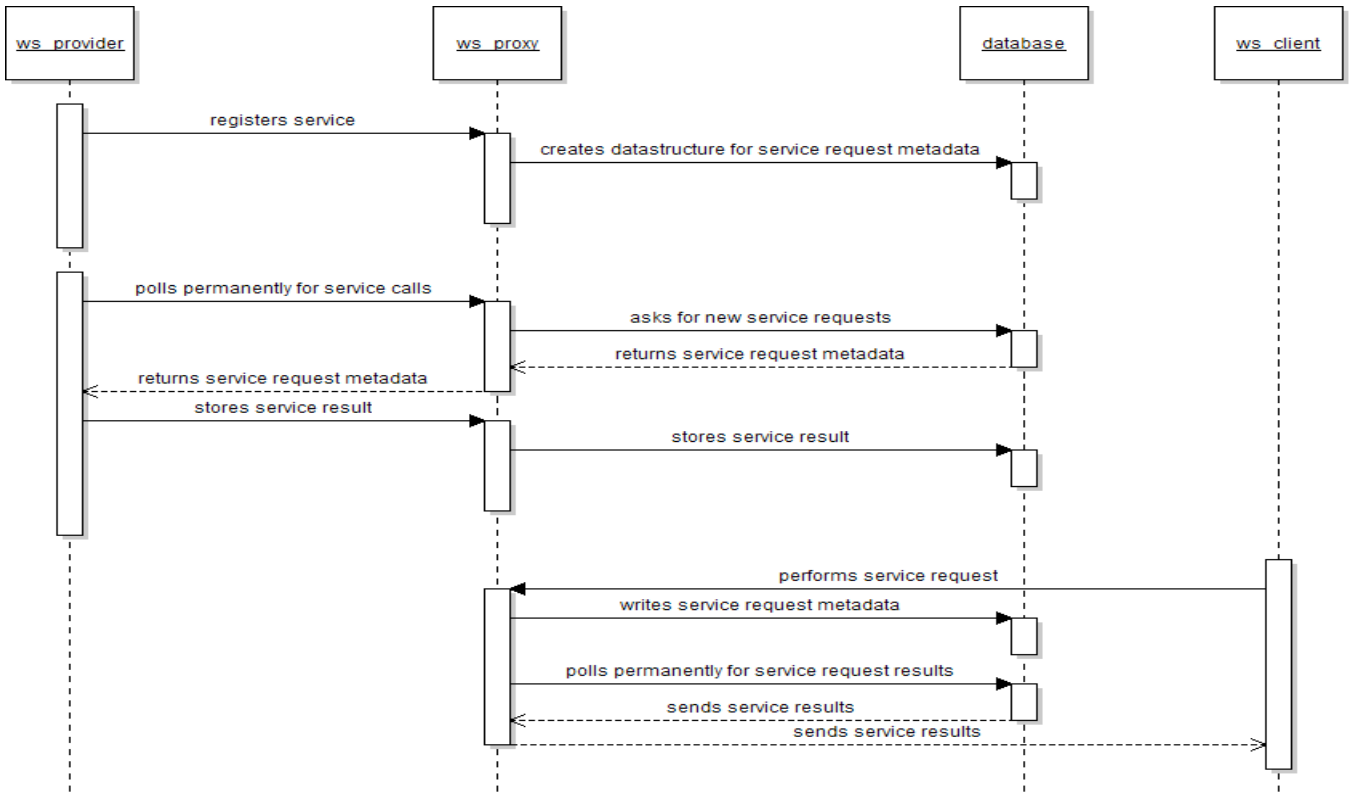


Figure 4. The UML sequence diagram for the communication between a mobile Web Service and its client.

implemented by the use of two different annotations: the @MobileWebService annotation marks a class as a Web Services and methods within this class can be marked as method available through the mobile Web Service with the @MobileWebMethod annotation.

With the help of these two annotations a simple mobile Web Service, which only calculates given integer values, can be implemented as follows:

```
@MobileWebService
public class TestService {

	@MobileWebMethod
	public int add(int a, int b) {
		return a + b;
	}

}
```

The basic relationships between the major classes of the sample implementation are shown in Figure 5. For the sake of simplicity and transparency, less important classes (and methods of each class) have not been modeled. Basically, the implementation consists of two packages. Package one is the proxy package which is usually deployed on a server that is reachable from the internet via a public IP address. Here, we find one class that implements the necessary methods for the registration of a new mobile Web Service, the permanent polling from the mobile Web Service for the service request metadata and the method that allows storing the result of the service request in the database. All these methods are reachable as Web Services themselves, so that the communication between the instance running the mobile Web Service and the Web Service proxy is completely Web Service-based.

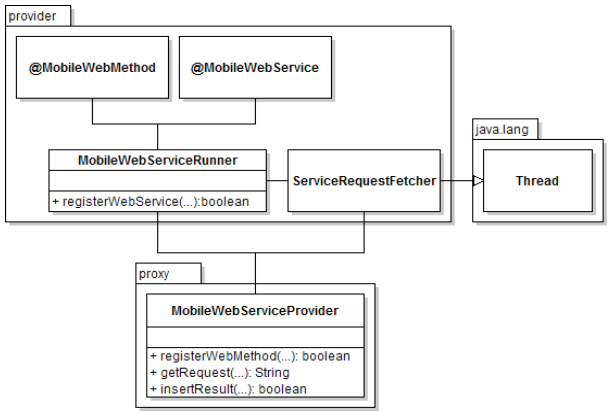


Figure 5. UML class diagram of major parts of the sample implementation.

In the provider package we find, as one of the major classes, the MobileWebServiceRunner class to which the mobile Web Service gets deployed. This class is basically comparable to an application server in a common Web Service environment, but with a dramatically lower footprint. This lower footprint is extremely important to mobile devices due to their usually limited resources. Additionally, this package also provides the two formerly mentioned

annotations that allow an easy marking of a class as a mobile Web Service and, accordingly, a certain method of such a class as a mobile Web Method. Last but not least, this package also implements the ServiceRequestFetcher class. This class inherits the java.lang.Thread class since its responsibility is to permanently poll the Web Service provider for new service requests.

V. PERFORMANCE TESTS

Since the communication is a little bit more complicated, in comparison to a common Web Service call, one concern of this approach is the question of its performance. In order to get a first idea of how good or bad this implementation behaves with respect to performance issues, we implemented a simple performance test. Here, we focused only on the evaluation of the transmission delay, since this seemed to be the most critical parameter. Other parameters like the number of lost packets, etc. were not taken into account yet.

A. Description of the test scenario

For the performance test and the sake of simplicity we implemented a very simple mobile Web Service. This service only calculates the sum of two given integers and returns the respective value as the result. The major advantage of such a simple mobile Web Service is that almost the entire duration of the mobile Web Service call is dedicated to the communication, and almost no amount of the round-trip time is used for the calculation itself. Since the communication is the complex part of the presented approach, we assume that this method of performance testing would provide the best overview about the communication performance of the presented approach. In the test scenario a common client (running on a common PC) had to put a number of service requests to the mobile Web Service.

In order to compare the results against the performance of common Web Service requests, we implemented the test scenario also the other way around: we implemented a common Web Service (running on a common server) and called this Web Service from a mobile device. Here, the basic idea was to use the same hard- and software-environment with minimal changes and also to maintain the same network environments in all of the tests.

In addition, we were interested in the communication performance in different network settings. Therefore, we performed the same tests in four different network settings. For each of the tests the (mobile) Web Service and its consumer where running:

- … in the same (Wi-Fi) network,
- … different networks, and the mobile device was connected via Wi-Fi,
- … different networks, and the mobile device was connected via UMTS
- … different networks, and the mobile device was connected via GPRS

We conducted eight different test cases: four for the different network scenarios with a mobile Web Service running on a mobile device and a Web Service client running on a common PC, and four test cases where the Web Service was running on a common Server and the client was running on a mobile device. The test cases, in which the (mobile) Web Service and the consumer were both connected to the same network, were only conducted in order to receive results with minimal latency.

In the test cases where the (mobile) Web Service provider and the client were not connected to the same network, the central components have been deployed to a server running via Amazon Web Services (AWS), as a Cloud Computing provider.

B. Test results

Within each of these eight test cases, one hundred service calls were performed and the duration of each call was measured.

The results for the mobile Web Service in the different network scenarios are shown in Figure 6.

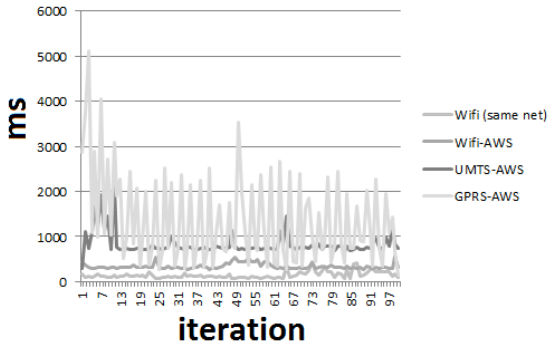


Figure 6. Results for the mobile Web Service in different network scenarios.

As expected the performance for the mobile Web Service requests are pretty good and pretty constant if the mobile device is connected with a Wi-Fi network. The average time if both, the mobile Web Service provider and the client, are connected to the same Wi-Fi network, was M = 147.69ms (SD = 76.00ms). Having the mobile Web Service provider connected to a different, still Wi-Fi, network, the average time for one service call calculates to M = 339.04ms (SD = 61.71ms).

Of course, we measured less performance of the service calls when the mobile Web Service provider was connected to a mobile network, the performance of the service calls was lower. The results for the UMTS based network connection of the mobile Web Service show an average of M = 827.55ms (SD = 250.35ms) for each service call, while the results for the GPRS based network are even worse. Here, the average for a single service call is M = 1355.96ms (SD = 986.38ms). As can be seen from the values for the standard deviation, the performance of single service calls differs dramatically as well, e.g., the minimum duration measured within the UMTS scenario was MIN = 283ms and the maximum was MAX = 2169ms. The results for the GPRS based scenario are even worse, with a MIN = 142ms and MAX = 5123ms.

The task of the second step of the test was to compare the performance results with the performance of a common Web Service call. For that purpose we conducted the same test, but this time the Web Service was not running on a mobile device but on a common server, while the Web Service client was running on a mobile device - again in the four different network settings. The results of these tests are shown in Figure 7. As demonstrated, the results are better from both perspectives - the overall performance and the standard deviation in the different network settings. A common Web Service call, if the Web Service provider and the mobile Web Service consumer are connected to the same Wi-Fi network, has an average round-trip duration of M = 61.16ms (SD = 301.36ms). When the Web Service client was connected to a different (still Wi-Fi) network the average performance was M = 156.71ms (SD = 15.24ms).

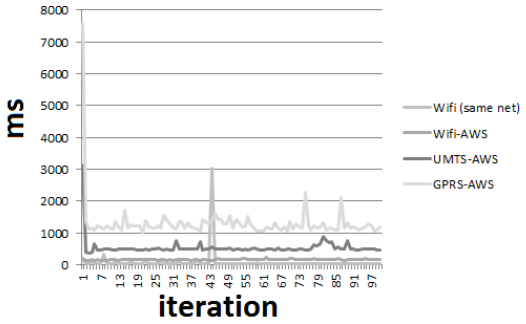


Figure 7. Results for the usual Web Service requests in the different network scenarios.

Here, again, the values for the Web Service client connected to a mobile network are somewhat lower. In the case of the UMTS network, the average service call showed a performance of M = 528.55ms (SD = 273.34ms), and the results for the GPRS based network were even worse with an average for each of the service calls of M = 1299.10ms (SD = 658.75ms).

The next step was to compare the different results. The major goal of this comparison was to get an idea of how good the performance of the presented approach for mobile Web Service requests is, in comparison to common Web Service requests. Therefore, we calculated the difference in the average performance of a single Web Service call in the different scenarios first, and as a second step we calculated the percentage of the performance difference in the different scenarios. The results are shown in Table 1.

TABLE 1: COMPARISON OF THE COMMON WEB SERVICE REQUESTS AND THE MOBILE WEB SERVICE REQUESTS IN THE DIFFERENT NETWORK SCENARIOS

	WiFi (same net)	WiFi-AWS	UMTS-AWS	GPRS-AWS
difference	85.53ms	182.33ms	299.00ms	56.86ms
percentage	137.60%	116.35%	56.57%	4.38%

The table shows that, in comparison to common Web Service requests, the performance of the presented approach was not too good when the mobile Web Service was connected to a Wi-Fi network. The results for the mobile Web Service provider and the client connected to the same network showed a performance overhead of 137.60 per cent, and when the mobile Web Service was provided within a different Wi-Fi network the performance overhead was about 116.35 per cent. But, if the mobile Web Service was connected to a mobile network, the performance overhead was not that dramatic anymore. In the case of the UMTS network the overhead was limited to 56.57 per cent, and for the GPRS based network the overhead was even lower at 4.38 per cent. Therefore, on the basis of our test results, it can be said that the performance degredation seems to decrease with the presented approach for mobile Web Services (in comparison to common Web Services) in lower quality networks, e.g. networks with lower bandwidth. This could best be seen by the results for the GPRS based network, where the actual overhead for the presented approach was below 5 per cent.

VI. TESTS FOR BATTERY CONSUMPTION

Beside the technical performance of the described solution, another very important aspect of the technical implementation is the impact of the implementation for the battery consumption of the mobile device. Already a lot research in the area of energy consumption for Android based devices has been conducted, e.g. [10], [11]. In general, the battery consumption is still one of the critical aspects for modern mobile devices. Users are still complaining about devices that need to be recharged daily. Therefore, users are for sure not interested in technical solutions that unnecessarily exploit the battery life of their mobile devices.

In order to investigate the battery consumption of the described technical implementation, a small testing scenario, based on the ideas described in [12], was implemented. With a set of five equal mobile devices (Android based mobile phones) the following test procedure was implemented.

As described in [13] each and every service running on the device might be the reason for a significantly higher amount of power, and therefore battery consumption. In order to test the effect that the described approach has on the battery consumption, the following steps were conducted.

For the first step the battery of each device was completely loaded and a little software was implemented that measured the actual status of the battery each ten minutes. This software allowed the measurement of the battery consumption for each single device. Within the first step, no other application (beside the usual operating system services) was running on the devices and the device was connected to the local wireless LAN.

In the second phase of the experiment, the described solution for the provision of mobile Web Service was deployed to the same devices and the battery of the devices was completely loaded again. Still, the devices where connected to the local wireless LAN. The proxy architecture for the mobile Web Services was deployed to a server running at the Amazon AWS Cloud system. A very simple

mobile Web Service was deployed on each of the devices, and the service polled every second for new service requests. Here, the decision for a simple Web Service, that only calculates the sum of two numbers, was taken, since this test was designed in order to provide a first inside of the battery consumption of the technology itself. Of course, the more complex the deployed mobile Web Service gets, the more battery it will consume. Therefore, the concentration on a very simple service seemed reasonable for the results that the performed test should bring. Again, with the help of the software that allows the measurement of the battery status of the mobile devices, the battery consumption was measured every ten minutes.

The results show that the battery consumption differs of course a little bit from device to device. Anyway, in average the five mobile devices still had about 91.2% of their battery capacity available after a twelve hours test run and the consumption took, as it could be assumed also beforehand, an almost linear degression.

The results for the second phase, in which the mobile Web Service was deployed to each of the mobile devices show also an almost linear degression of the battery power, but in this case the devices had only 85.2% of their battery load left after another twelve hours test run. The difference becomes clearer by comparing only the average values, as shown in Figure 8.

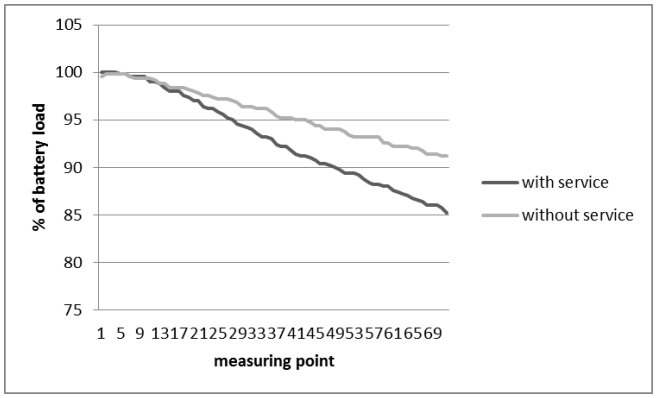


Figure 8. Comparison of the average values for the two different phases.

Here it can be seen that the average power consumption of the devices running the mobile Web Service with the described approach, is bigger than for the devices that are not running the mobile Web Service.

In order to check whether the difference between the power consumption is statistically significant, a simple t-test was conducted over the data provided by the experiment. Therefore, the hypothesis was:

H_0: Statistically the amount of power consumed by the mobile Web Service deployed to each mobile device is different from zero.

In order to test this hypothesis the difference between the measured battery status for the experiment with and without the mobile Web Service running on the mobile device was calculated for every measuring point. Afterwards, with the help of the average of the calculated differences, the standard deviation, the number of measurements and the control value (which is actually zero in this case, since the hypothesis was chosen that the amount of consumed power is different from zero), the according t-value was calculated. The results of this test show significant values for $n = 72$ (the number of measurements) and $\alpha = 0.01$. Therefore, the hypothesis H_0 can be seen as correct and we can assume that the deployed mobile Web Service is using around 6% of additional energy.

Having in mind that battery consumption is still a critical issue for owners of mobile devices, the consumption of at least additional 6% of their battery for a simple service with a polling interval of about a second, does not seem to be feasible.

On the other hand, the measured battery consumption might also lead to the question what kind of scenarios can be supported with the help of the described technology. Since the results of the performed test show that there is a significant amount of the battery consumed by the presented technology, also if the deployed mobile Web Service is itself not at all complex and the polling interval is just about each second, the solution might probably be to identify scenarios in which the polling interval for the provided services on the mobile device is significantly longer than one second.

VII. Development of Scenarios that Benefit from Mobile Web Services

As already indicated in the previous section, beside the technical feasibility of the described technical solution as explained in this paper, the development of scenarios that benefit from Web Services deployed to mobile devices is a critical issue in order to make this technology a success.

Usually, Web Services are deployed on large servers in data center environments in order to provide at least one of the following three different benefits to the consumer of such a Web Service:

- Access to large computing resources, e.g. computational power or memory
- Access to large databases that cannot be stored locally
- Access to data and/or procedures that are not available locally

Obviously, the provision of Web Services on mobile devices is not interesting for the first two scenarios, since mobile devices do usually not provide enough power neither with respect to computational power nor with respect to the amount of the provided memory. Also, large databases are usually not installed on mobile devices for similar reasons.

Therefore, the only possibility for reasonable approaches in which scenarios might benefit from Web Services deployed to mobile devices is the last one, in which these

mobile Web Services either provide access to certain data and/or procedures especially available on mobile devices.

Following this idea, one of the major advantages of todays modern mobile devices is that they are more of a set of sensors rather than a single device: looking at a modern mobile phone, these devices usually encapsulate a GPS sensor, a digital compass, an acceleration sensor, … Most of these sensors allow to easily contextualize the user in his/her current situation, e.g., the GPS sensor can be used in order to determine the actual position of the user of the mobile device, additionally the digital compass provides the direction in which the user is probably looking currently.

Therefore, scenarios that on the one hand need to contextualize a single user, e.g., supporting the user in finding the fastest way to work or providing commercials for goods the user can buy in a store close to his/her current position and in his/her current viewpoint. On the other hand these kinds of scenarios typically do not need poll permanently for actual information, e.g., determine the actual temperature at the current position of the user, are ideal candidates to consume services provided by mobile devices.

Another type of scenarios in which the usage of mobile Web Services seem reasonable, are scenarios that concentrate more on procedures where the user of a mobile device is actively integrated. Here, scenarios that need fast feedback from users might benefit from reaching users that are currently mobile, e.g., crowd sourcing [14]. Also a combination of both ideas, like a short survey (consisting of a very limited number of questions) send to a customer that leaves a certain store about how he/she felt during his/her stay in the store might provide a reasonable scenario for mobile Web Services.

In order to determine and evaluate different scenarios that might benefit from consuming mobile Web Services, the sensitivity model, as described in [15], might provide a reasonable approach in order to evaluate whether a certain scenario benefits from using mobile Web Services. This cybernetic based approach allows to evaluate different parameters with respect to their effectiveness in order to reach a certain goal, also if these parameters do interact with each other. Since usually the different parameters that are important for the success of a mobile application/service interact with each other, this approach seems to provide a good starting point for the evaluation of mobile Web Services.

The basic steps that need to be performed in order to provide a sensitivity model are:

- Description of the system: Here, the system itself in which the services are provided, has to be described.
- Determination of different variables: In this step different variables of the system (with respect to the currently evaluated service) are determined.
- Evaluation of relevance of the variables: Since so far the different variables are only determined, their relevance for the system has to be evaluated in a separate step.
- Determination of the interaction of the different variables: Here, for each of the variables a value has to be determined, how much this variable interacts with any other variable in the system.
- Clarification of the role of each variable: In this step, a role is attached to each variable that reflects e.g., how active and how critical this variable is in the system.

With the help of this information, certain tests and simulation can be run against the set of variables that reflect their behavior in the system.

VIII. Conclusions and Future Work

As demonstrated in this paper, todays' modern and powerful mobile devices can be used as Web Service providers by using well-known and accepted standards and protocols. The presented approach is capable of solving some of the problems that usually occur while providing Web Services on mobile devices, e.g., the problem of constantly changing IP addresses. Furthermore, the overhead that is inherent (resulting in a transmission delay) in the presented approach does not seem to be a show stopper. As shown, the performance in commonly available mobile networks, like UMTS or GPRS, is comparable to common Web Service requests.

It can, therefore, be concluded that the presented approach provides an interesting alternative to the common Web Service provisioning by using mobile devices that act as a server also from a technical point of view. It eliminates certain problems that usually occur if mobile devices provide Web Service provider infrastructures, and the resulting drawbacks from the performance point of view are acceptable.

Having in mind the power that the presented approach would provide for new approaches and scenarios, it could be asserted that bringing Web Services to mobile devices will probably become more important in the future and that we will most likely see an increasing number of applications making use of that kind of technology.

Anyway, as shown by the test of the battery consumption of the presented approach, the provisioning of mobile Web Services also provides a number of drawbacks. Here, it will be important in the future to develop scenarios that on the one hand actually benefit from using mobile Web Services and on the other hand try to decrease the battery consumption of the presented approach by lowering the polling interval accordingly. Therefore, a complete new understanding for Web Services needs to be established, with respect to mobile Web Services. As discussed, mobile Web Services do usually not provide additional computational power, access to more memory or access to large databases, but may provide access to data from personal sensors, e.g., for the contextualization of the user.

On the other hand, other technologies, like C2DM (Cloud-to-Device-Management), an Android based technology that allows to send activation commands to a certain mobile device might help to further decrease the

battery consumption for the provisioning of mobile Web Services.

As also discussed in the last section, the development of scenarios that benefit from consuming mobile Web Services also need a new approach that on the one hand reflect the complexity for the evaluation of mobile scenarios in general and on the other hand the different view on mobile Web Services (in contrast to usual Web Services) as described above.

The last two points, using other technologies like C2DM and the development and evaluation of scenarios with respect to using mobile Web Services, will be part of future investigations and research.

ACKNOWLEDGMENT

This work was partly supported by an Amazon AWS research grant.

REFERENCES

[1] M. Jansen, "About an Architecture That Allows to Become a Mobile Web Service Provider", In: Proceedings of the 7th International Conference on Internet and Web Applications and Services (ICIW 2012), pp. 90-96.

[2] IDC Worldwide Quarterly Mobile Phone Tracker (May 2013)

[3] B. Tudor, C. Pettey, "Gartner Says Worldwide Mobile Phone Sales Grew 35 Percent in Third Quarter 2010", Smartphone Sales Increased 96 Percent, Gartner, http://www.gartner.com/it/page.jsp?id=1466313, last visited 19.11.2011

[4] S. McFaddin, C. Narayanaswami, M. Raghunath, "Web Services on Mobile Devices – Implementation and Experience", In: Proceedings of the 5th IEEE Workshop on Mobile Computing Systems & Applications (WMCSA'03), pp. 100-109, Monterey, CA

[5] S. Srirama, M. Jarke, W. Prinz, "Mobile Web Service Provisioning", In: Proceedings of the Advanced International Conference on Telecommunications and International Conference on Internet and Web Applications and Services (AICT/ICIW 2006), p. 120, Guadeloupe, French Caribbean

[6] F. AlShahwan, K. Moessner, "Providing SOAP Web Services and REST Web Services from Mobile Hosts", In: Fifth International Conference on Internet and Web Applications and Services (ICIW 2010), pp. 174-179.

[7] L. Li, W. Chou, "COFOCUS – Compact and Expanded Restful Services for Mobile Environments", In: Proceedings of the 7th International Conference on Web Information Systems and Technologies, pp. 51-60, Noordwijkerhout, The Netherlands

[8] E. Gamma, R. Helm, R. Johnson, J. Vlissides, "Design Pattern – Elements of Reusable Object-Oriented Software", pp. 185-195, Addison-Wesley.

[9] D. Svensson, "Assemblies of Pervasive Services. Dept. of Computer Science," Institutional Repository – Lund University.

[10] T. Kundu, K. Paul, "Android on Mobile Devices – An Energy Perspective", In: Proceedings of the 10th IEEE International Conference on Computer and Information Technology (CIT 2010)

[11] A. N. Moldovan, O. Ormond, G.-M. Muntean, "Energy consumption analysis of video streaming to Android mobile devices", In: Proceedings of Network Operations and Management Symposium (NOMS), IEEE

[12] F. Ding, F. Xia, W. Zhang, X. Zhao, C. Ma, "Monitoring Energy Consumption of Smartphones", In: Proceedings of the 1st International Workshop on Sensing, Networking, and Computing with Smartphones

[13] A. Caroll, G. Heiser, "An analysis of power consumption in a smartphone"; In: Proceedings of the 2010 Annual Technical Conference on USENIX

[14] S. Roth, "The Diaspora as a Nation's Capital: Crowdsourcing Strategies for the Caucasus", International Journal of Transition and Innovation Systems 1(1), pp. 44-58.

[15] F. Vester, "The Art of interconnected thinking: Tools and concepts for a new approach to tackling complexity", pp. 149-256, McB

Formal Approach to Design and Automatic Verification of Cooperation-Based Networks

Alessandro Aldini
University of Urbino "Carlo Bo"
Urbino, Italy
email: alessandro.aldini@uniurb.it

***Abstract*—The efficacy and efficiency of cooperation incentives in user-centric networks is a challenging issue that involves tradeoff among trust, social, and economic aspects. Two well-established approaches to stimulate resource sharing and cooperation rely on reputation and remuneration, the complementary functioning of which shall increase users' motivation and discourage mistrust and selfishness. In order to verify the benefits of the joint application of these mechanisms, we specify and analyze formally a recently proposed cooperation model by employing tool-supported probabilistic model checking techniques.**

***Keywords*—model checking, cooperation incentives, trust, remuneration, user-centric networks.**

I. INTRODUCTION

User-centric networks (UCNs, for short) emerged among the fastest growing community-scale places of the Internet. They include social networks, online games, auction systems, and peer-to-peer environments. User centricity entails interaction among users, which are expected to share some form of prosocial attitude to cooperation deriving from, e.g., synergy and sense of community. In order to strengthen such an attitude and to ensure the community grows healthily, explicit incentive mechanisms in the form of rewards are used to provide motivations and to contrast typical obstacles of cooperation systems, like mistrust and selfishness. Rewards can be either *indirect*, as in the case of automatically computed notions of trust used to regulate the access to services and resources, or *direct*, e.g., in the form of remuneration, which can be based on fiat money or virtual currency.

Trust systems provide explicit metrics estimating the subjective reliance on the character, integrity, ability, and honesty of each user, thus providing the means for setting up a reputation infrastructure. The aim of reputation is not only giving a perception of the public trustworthiness of each user, but also providing the enabling conditions for exchanging services. On the other hand, virtual currency supports monetization of services, which provides additional motivations to sharing whenever barter, sense of community, and reputation do not suffice.

The application of incentive mechanisms is more critical in wireless and mobile environments. In fact, in these systems, even the underlying communication infrastructure could be dynamically built by users sharing Wi-Fi connections. However, joining the community for short periods of time, thus hindering long-term relationships, may keep such users from adopting prosocial behaviors. Similarly, the inherent limitations of mobile devices (e.g., battery and bandwidth) usually restrict the application of pervasive controls, like assurance of payment or service delivery, thus exposing the system to dishonest behaviors that, however, must be contrasted by the adoption of incentive mechanisms.

Hence, several orthogonal aspects come into play to establish to what extent cooperation incentives can deal successfully with mistrust, selfishness, cheats, and limited resources. In the light of these considerations, the objective of this paper is twofold. On one hand, in order to consider properly the analysis of all the issues above, we advocate the use of tool-supported, model checking based, formal techniques. On the other hand, we apply such techniques for the design and verification of a specific cooperation model for wireless UCNs. This work is a revised and extended version of [1], which is a paper presented at AFIN 2012. The first proposal of the cooperation model under consideration can be found in [2], while its formal design and verification are supported by probabilistic model checking and the related software tool PRISM (see, e.g., [3], [4], [5] for a survey).

The exhaustive analysis conducted in this paper takes into account security, trust, social, and economic aspects in order to verify whether users requiring services are motivated to behave honestly, while users offering services are encouraged to share resources. The formal specification is given in the modeling language of PRISM, which is a state-based mathematical formalism based on the Reactive Modules of [6], from which different types of probabilistic models can be derived, including discrete-time Markov chains (DTMCs, for short) and Markov decision processes (MDPs, for short), see, e.g., [7], [8]. By following the lines of [9], performance properties are expressed in a temporal logic – subsuming both Probabilistic Computation Tree Logic (PCTL) and Linear Time Logic (LTL) – which is expressive enough to specify state-based and path-based properties, and including both probabilistic and reward operators. Thanks to these capabilities, we can describe properties including probabilistic and temporal information, as well as expressing social and economic aspects.

In the rest of the paper, we first discuss some comparison with related work. Then, we recall the cooperation model of [2] and related modeling assumptions (Section II), we discuss the formal specification of such a model (Section III), we present the results of the model checking analysis (Section IV), and we finally draw some conclusions (Section V).

A. Related Work

Sustaining a secure, reliable and efficient environment for the sharing of services and resources in highly mobile communities represents a well-studied problem in the literature, see, e.g., [10], [11], [12], [13]. The application of formal approaches to the analysis of this problem is not novel, refer, e.g., to [14], [15], [16], [17], [18]. In particular, game theory is the most applied approach to networking, see, e.g., [19].

Whenever the intrinsic attitude to prosocial behaviors is not enough, as emphasized in [20], a suitable support to more explicit and extrinsic motivations is given by the joint application of *trust management* (see, e.g., [21]) and *virtual currency* (see, e.g., [22]). Trust management and virtual currency implement the so-called *soft security*, which is characterized by relaxation of the security policies and enforcement of common ethical norms for the community, see, e.g., [23].

Recently, it has been proved that intertwining indirect and direct rewards maximizes the effect of the incentives to cooperate, as shown in [1], [24], [14], [25]. In particular, in [14] game theory is employed to study the balanced tradeoff between reputation-based and price-based cooperation strategies. The obtained analytical results are consolidated by a simulation analysis showing the fast convergence towards cooperative behaviors in the case of mixed incentive strategies. Analogous results are achieved in our approach, which, however, provides a unifying formal framework allowing for the evaluation of all the quantitative properties of interest without requiring simulation analysis. Similarly, the utility-based decision making framework of [24] is used to verify a QoS-based incentive mechanism in which, however, only some of the aspects considered in our approach are taken into account.

II. Cooperation Model

This section briefly outlines the cooperation model introduced in [2], by illustrating the way in which indirect and direct rewards are combined. We then specify the modeling assumptions adopted for analysis purposes.

Cooperation involves users providing services, hereafter called *requestees*, and recipients of such services, hereafter called *requesters*. The cooperation process entails the following four phases:

1) discovery and request;
2) negotiation;
3) transaction and payment;
4) evaluation and feedback.

As we will see, the four phases rely on trust management and virtual currency. In the first phase, the requester searches for a requestee offering the required service. Reputation of the requestee is a parameter guiding the selection. If the requester is trustworthy enough to access the required service, then the issued request can be accepted. However, it may be also refused because of, e.g., lack of willingness to cooperate. In the second phase, requester and requestee establish service parameters and reward, possibly taking into account the trust of the requestee on the requester. In the third phase, service is delivered and the related payment is provided. Finally, in the fourth phase, the transaction results are used to adjust, if necessary, reputation of the involved parties.

A. Reputation System

As usual in several trust-based systems, see [23], we model trust as a discrete metric. Then, given a user i and another user j, the computation of the trust value of i towards j is obtained by mixing direct experience and indirect recommendations:

$$T_{ij} = \alpha \cdot trust_{ij} + (1 - \alpha) \cdot recs_{ij} \quad (1)$$

Parameter $\alpha \in [0, 1]$ represents a risk factor. The trust metric $trust_{ij}$ is the result of previous direct interactions of i with j. In absence of previous experience, the value of $trust_{ij}$ is set to the dispositional trust of i, dt_i, which represents the initial willingness to trust unknown users. Finally, $recs_{ij}$ is the average of the trust metrics towards j of other users (different from i) that in the past negotiated directly with j.

B. Virtual Currency System

Indirect and direct rewards are combined by including the trust value T of the requestee towards the requester as a parameter affecting the cost of the negotiated service. The other parameters are C_{min}, which is the minimum price asked by the requestee regardless of the trust on the requester, C_{max}, which is the maximum price asked to serve untrusted users, and T', which is the trust threshold above which the minimum price is applied to the requester. Then, the cost function C proposed in [2] is defined as follows:

$$C(T) = \begin{cases} C_{min} + \frac{C_{max} - C_{min}}{T'} \cdot (T' - T) & T < T' \\ C_{min} & T \geq T' \end{cases} \quad (2)$$

An alternative simple formula for expressing the service cost is given by the following four-step function:

$$C(T) = \begin{cases} C_{min} & T \geq T_3 \\ C_{med} & T_2 \leq T < T_3 \\ C_{med'} & T_1 \leq T < T_2 \\ C_{max} & T < T_1 \end{cases} \quad (3)$$

where C_{med} and $C_{med'}$ are two intermediate prices, while each T_i, with $1 \leq i \leq 3$, represents a trust threshold, such that $T_i > T_j$ if $i > j$.

C. Modeling Assumptions

For modeling purposes, we distinguish between users playing the role of requester and users playing the role of requestee. Moreover, we consider a unique type of service that is offered by each requestee in the community. Trust values range in the interval $[0, 10]$, such that $null = 0$, $low = 2$, $med = 5$, $high = 8$, and $top = 10$. Based on the system described above, the modeling assumptions concerning the four-phase cooperation process are as follows.

As far as the first phase is concerned, we consider three alternative choice policies adopted by the requester to select a requestee:

- Nondeterministic.

- Prioritized on the basis of requestee's reputation, i.e., the trust value of the requester towards each available requestee is used to govern the selection. The choice among requestees with the same reputation is random.
- Probabilistic, in which case requestee's reputation is used as a probabilistic weight.

The initial reputation is *low* for every requestee. Moreover, by default, requestee i is not available to accept a request by requester j if and only if $T_{ij} < st_i$, where the service trust level st_i represents a trust threshold below which the service offered by i is not accessible. A refused request is sent by the requester to one of the remaining requestees according with the selection policy.

As far as the second phase is concerned, we assume that the agreement is successful. By default, the cost is determined through the application of Equation (2) and is accepted by the requester without any further negotiation. The default values are $C_{min} = 0$, $C_{max} = 10$, and $T' = high$.

As far as the third phase is concerned, by default the service is delivered with success. Then, the requester decides whether to pay or not, either nondeterministically or probabilistically with parameter $p \in [0, 1]$, that is the payment is honored with probability p.

As far as the last phase is concerned, since the service is satisfactory, the reputation of requestee i as perceived by requester j is increased by 1. On the other hand, the trust of i towards j increases (resp., decreases) by 1 (resp., by a factor k) in the case j pays (resp., does not pay) the service. Feedback is provided by i to the other requestees.

III. Specifying Formally the Cooperation Model

In order to illustrate briefly the PRISM specification of the cooperation model, in this section we describe three basic versions of the requester and one basic version of the requestee in a scenario with one requester and two requestees. The reader interested in the evaluation results may skip this part.

Let us start with a system specification with MDP-based semantics, i.e., choices can be nondeterministic. A system component is represented by a module specifying its local variables and its behavior. The requester is defined as follows:

```
module Requester
 x : [0..n] init 0;
 ns : [0..N] init 0;
 ...
```

The local variable `x` denotes the local state of the requester, such that `x=0` represents the initial state and the integer constant `n` is the number of possible local states. The local variable `ns` represents the number of requested services, the maximum value of which is given by the constant integer `N`.

The behavior is given by a set of guarded commands specifying variable updates. In our example, the requester chooses nondeterministically one of the two requestees:

```
[] x=0 & ns<N -> (x'=11);
[] x=0 & ns<N -> (x'=21);
[] x=0 & ns=N -> (x'=1);
[] x=1 -> true;
```

A pair of brackets represents the start of the command, while the symbol `->` is preceded by the boolean guards to satisfy in order to enable the following variable updates. The primed name `x'` denotes the next value that `x` assumes by virtue of the update. If not specified explicitly, the other local variables remain unchanged (`true` stands for no changes). In our example, if `x=0` and `ns<N` then two updates are possible: the former refers to the case in which the requester chooses the first requestee, while the latter is specific for the second requestee. Without loss of generality, we concentrate on the former case in which `x` is set to 11.

The first requestee is expected either to accept the request and deliver the service, or to refuse the request. In the case of success, the requester decides nondeterministically to pay or not for the obtained service. The commands expressing such a behavior are as follows:

```
  [accept] x=11 -> (x'=12) & (ns'=ns+1);
  [refuse] x=11 -> (x'=13) & (ns'=ns+1);
  [pay] x=12 -> (x'=0);
  [nopay] x=12 -> (x'=0);
  [] x=13 -> (x'=0);
  ...
endmodule
```

Notice that in some cases the brackets marking the start of the command include a label, which expresses an action name on which the module is expected to synchronize with another module, in the same style of, e.g., [26]. In our example, if `x=11` and the first requestee is ready to execute a command labeled with the action name `accept`, then the updates `x'=12` and `ns'=ns+1` are executed. In this case, the requester decides nondeterministically to synchronize with the first requestee either through action `pay` or through action `nopay`, after which the module goes back to its initial state.

On the other hand, a basic description of the requestee behavior is as follows:

```
module Requestee
  y : [0..m] init 0;
  ...
  [accept] (y=0) & (Teq>=st) -> (y'=1);
  [refuse] (y=0) & (Teq<st) -> (y'=0);
  [pay] (y=1) -> (y'=0) &
              (t' = (t<top) ? t+1 : top));
  [nopay] (y=1) -> (y'=0) & (t'=null);
endmodule
```

The local variable `y` expresses the local state of the requestee, while other local variables are `st`, modeling the service trust level, and `t`, modeling the trust of the requestee towards the requester. Parameter `Teq` is the result of a formula expressing Equation (1). The command labeled with action `pay` includes a conditional expression that increments `t` by 1 if the value of such a variable is less than `top` and assigns to it value `top` otherwise. Moreover, the update in the command labeled with action `nopay` expresses the most punishing reaction to a dishonest behavior of the requester.

Now, let us consider a more detailed version of the system with semantics based on DTMC, meaning that choices cannot

be nondeterministic and the execution of each command takes one discrete unit of time. In particular, as far as the selection of the requestee is concerned, we assume that the choice is probabilistic and weighted by reputation. Hence, in the requester module we add two more local variables, one for each requestee, storing their reputation values:

```
rep  : [0..top] init low;
rep2 : [0..top] init low;
```

and then we change the selection as follows:

```
[] x=0 -> (rep/totrep):(x'=11) +
          (rep2/totrep):(x'=21);
```

where the syntactic expression $p_1 : (c_1) + p_2 : (c_2)$ indicates that command c_1 is executed with probability p_1, while command c_2 is executed with probability p_2, such that $p_1 + p_2 = 1$. Parameter `totrep` represents the overall reputation of the requestees, which is necessary to compute the relative weights, and is defined as the result of the following expression:

```
formula totrep =
   (rep+rep2) = 0 ? 1 : (rep+rep2);
```

Analogously, the other source of nondeterminism in the requester module, which is the choice related to payment, is changed as follows:

```
[accept] x=11 -> p:(x'=12) +
             (1-p):(x'=13);
[refuse] x=11 -> (x'=14);
[pay] x=12 -> (x'=0) &
              (rep'=min(rep+1,top));
[nopay] x=13 -> (x'=0) &
                (rep'=min(rep+1,top));
[] x=14 -> (x'=0);
```

The real value `p` represents the parameter governing probabilistically the choice between honest and cheating behaviors, while the reputation of the first requestee is increased whenever the service request is accepted.

Alternatively, in order to select a requestee, the requester may follow a prioritized model of choice based on reputation. In such a case, given the following expression:

```
formula maxrep = max(rep,rep2);
```

we change the selection as follows:

```
[] x=0 & rep=maxrep -> (x'=11);
[] x=0 & rep2=maxrep -> (x'=21);
```

If both requestees have the same reputation, then the two alternative commands are given the same weight and the probabilistic choice follows a uniform distribution.

Finally, the properties are specified in an extension of PCTL including reward operators. A reward is a cost associated with state-based and transition-based conditions. Rewards are accumulated each time the related conditions hold, while ad-hoc operators are used to reason about the amount of accumulated rewards, e.g., along specific paths by a given amount of time or at the equilibrium. For instance, the reward structure that is used to calculate the total expected earnings for the first requestee is as follows:

```
rewards "cost1"
[pay] true : f;
endrewards
```

This structure establishes that `pay`-labeled transitions from states enabling the guard `true` acquire a reward equal to the value of formula `f`, which abstractedly denotes, e.g., Equation (2). Based on this structure, as an example we show the property specifying the total expected earnings accumulated until `t` time units have elapsed:

```
R{"cost1"}=? [ C<=t ]
```

In particular, the operator `C<=t` is used to reason about the transient-state behavior of a system.

IV. Model Checking of the Cooperation Model

The analysis of the cooperation process through model checking is divided into two steps. First, we study the vulnerabilities of the trust system with respect to a cheating profile of the requester. The reason is that the soft transaction mechanism does not ensure guarantee of payment. Based on the results of such an analysis, we then verify the efficiency of the mixed cooperation incentives in discouraging selfish behaviors of the requestees and motivating honest behaviors of the requesters.

A. MDP Analysis

The effectiveness of the trust system with respect to cheating requesters is expressed through the following property, which is investigated in a scenario with a single requester and three alternative requestees.

Property 1. What is the maximum number of services (out of N requests) that can be obtained by a requester without honoring the payment?

Formally, the specification of this property is given as a *reachability reward* property:

```
R{"nopayed"}max=? [ F(x=1) ]
```

computing the maximum reward, as defined by the reward structure `nopayed`, accumulated along any path until the condition `(x=1)` holds, which denotes the local state that is reached whenever the maximum amount of requests has been issued. We point out that `F` represents the *eventually* operator of LTL. The reward structure `nopayed` is defined as follows, where the three action names refer to the three requestees:

```
rewards "nopayed"
[nopay1] true : 1;
[nopay2] true : 1;
[nopay3] true : 1;
endrewards
```

With respect to the assumptions of Section II-C, we consider requester's choices to be nondeterministic. Hence, the requester can be viewed as an adversary controlling the way in which the nondeterminism is solved adaptively. The aim of such an adversary is to find out the strategy maximizing the

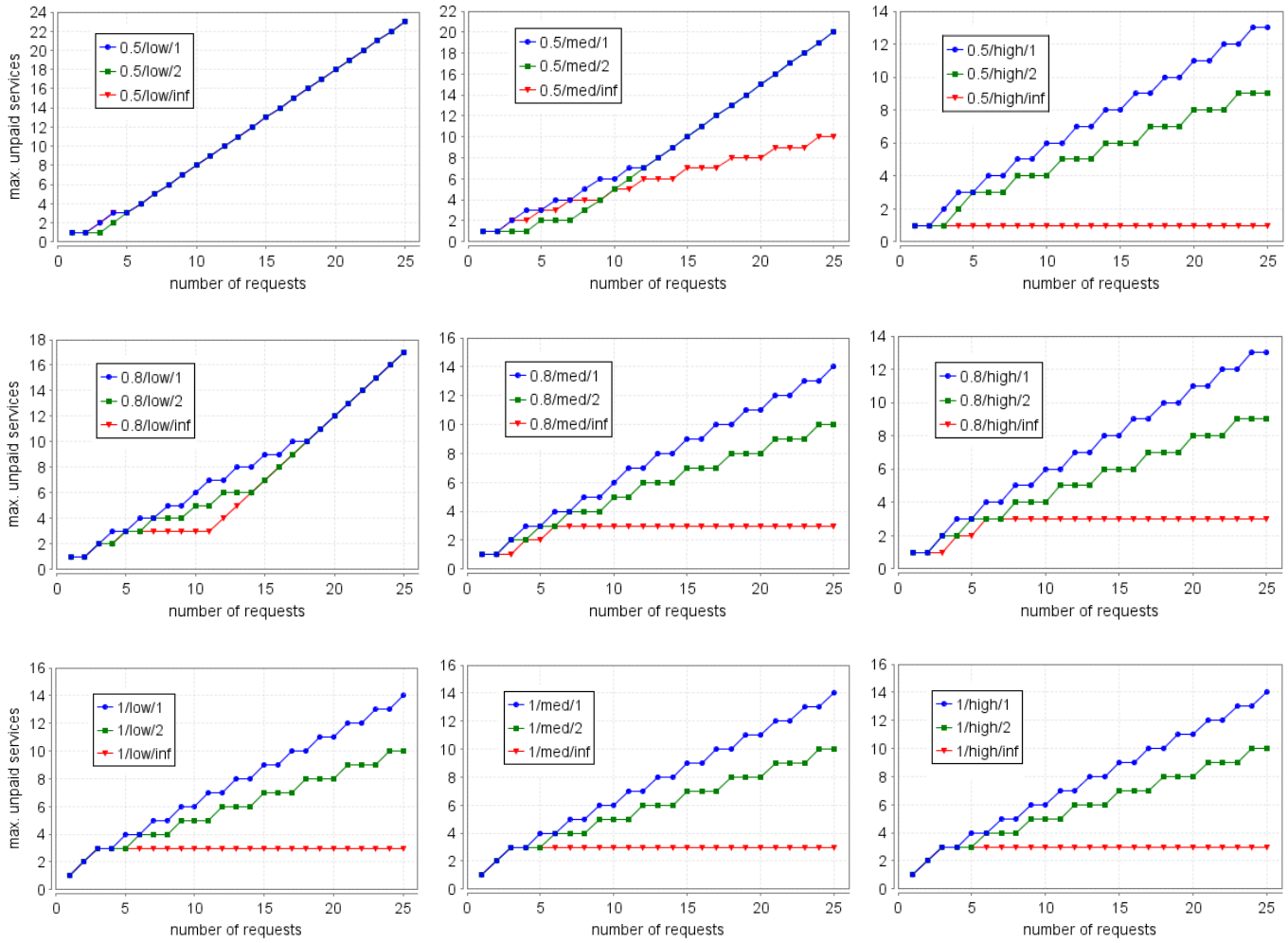


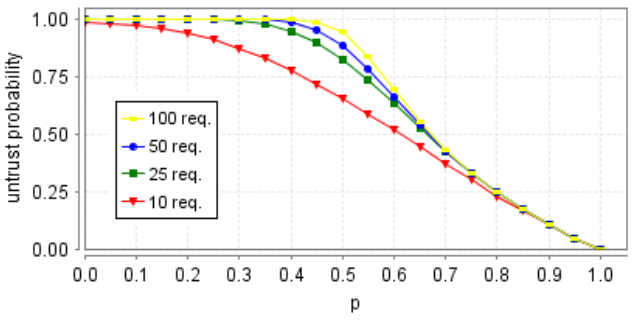
Fig. 1: MDP analysis: verification of Property 1 for 27 combinations of parameters $\alpha/st/k$.

number of unpaid services, thus revealing the worst case from the viewpoint of the requestees.

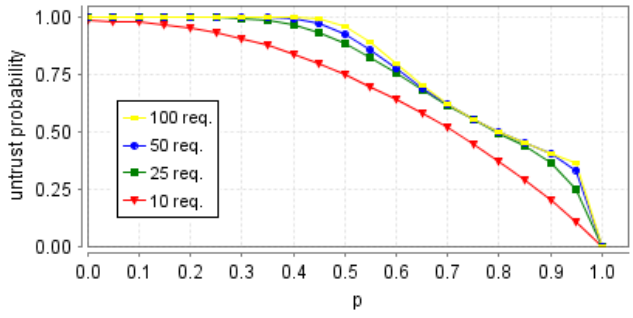
Formally, the semantics of the model is an MDP on which Property 1 is evaluated by solving the nondeterminism in all possible ways. Then, the model checker returns the result for the *best adversary* strategy. Notice that such a strategy corresponds to the most powerful adversary, which can observe the behavior and the configuration parameters of all the requestees.

To conduct the analysis, we assume three equal requestees characterized by the configuration of parameters $\alpha/st/k$, where: $\alpha \in \{0.5, 0.8, 1\}$ is the contribution of direct experience to trust, $st \in \{low, med, high\}$ is the service trust threshold, and $k \in \{1, 2, \infty\}$ denotes the rapidity with which the trust towards a cheating requester is decreased each time a payment is not honored (∞ stands for the immediate assignment of the value *null* to the trust value). The dispositional trust of each requestee is chosen to be equal to the service trust threshold in order to make it possible for a new requester to start negotiating services with the requestees.

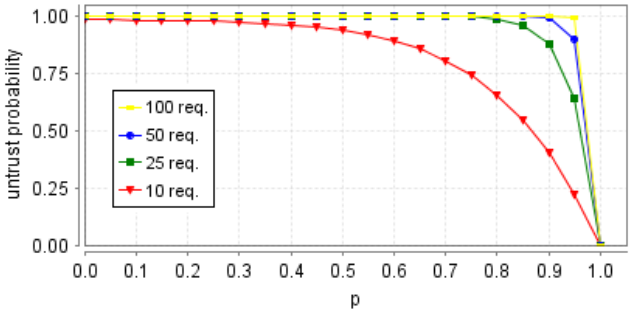
All the 27 combinations of the parameters introduced above are analyzed, as illustrated in Figure 1. The horizontal axis denotes the total number of requests *N*, ranging from 1 to 25, while the vertical axis reports the maximum number of unpaid services. From the analysis, we observe that for each value of α and st the success of the cheating strategy is inversely proportional to the factor k. In practice, the higher the value of k is, the faster the reaction to dishonest behaviors and, therefore, the negative effect upon trust. For the same reason, the higher the service trust level st is, the lower the number of unpaid services. When $\alpha = 1$, however, the service trust level does not affect the results because any decision depends only on previous direct experience. The analysis could be extended to the case $\alpha < 0.5$, obtaining results similar to those related to $0.5/low/\_$, regardless of the value of st. These results reveal a typical attack of a dishonest requester cheating only one requestee, which gives too much weight to the positive



(a) 3 risky requestees.



(b) 1 risky, 1 cautious, and 1 default requestee.



(c) 3 cautious requestees.

Fig. 2: DTMC analysis: verification of Property 2.

recommendations provided by the other requestees.

The results of Figure 1 suggest to categorize the behavior of the requestee according to two limiting profiles:

- *risky* profile, for which the unpaid services increase linearly and most of the served requests are unpaid (see, e.g., configurations $0.5/low/\_$, $0.8/low/\_$, and $\_/\_/1$).
- *cautious* profile, for which the number of unpaid services is essentially constant (see, e.g., configurations $\_/high/\infty$, $0.8/med/\infty$, and $1/\_/\infty$).

B. DTMC Analysis

The two profiles defined above give a clear and precise perception of requestee's attitude to take prosocial decisions in an environment where requesters are possibly cheating. This subsection reports the results of further investigations conducted by considering risky requestees represented by configuration $0.5/low/1$ and cautious requestees represented by configuration $0.8/med/\infty$. Whenever the profile is not specified, configuration $0.8/low/1$ is taken as default.

In order to analyze performance properties, we assume reputation-based prioritized choice of the requestee and payment honored probabilistically with parameter p (see Section II-C). Hence, the semantics of the model turns out to be a DTMC, on which both *steady-state* and *transient-state* analyses can be conducted.

On one hand, the steady-state analysis reveals the success of the cooperation mechanism on the long run. Indeed, at the equilibrium, for each $p < 1$ the requester becomes untrusted with probability 1 by any requestee. On the other hand, the transient-state analysis is important to study the convergence speed towards such a result.

Property 2. What is the probability for a cheating requester of being untrusted by each requestee after N requests?

The specification of this property is given through the operator `P` of PCTL, which is used to reason about the probability of satisfying a given condition. Formally:

```
P=? [ F<=t (x=41) ]
```

returns the probability that the path property expressed between brackets is satisfied by paths starting from the initial state of the system. More precisely, `F<=t` expresses a bounded path property as it imposes the time upper bound `<=t` on the length of the analyzed paths. In our analysis, `t` is chosen to express the constraint upon the number of requests *N*. On the other hand, the state condition `(x=41)` is associated with a local state of the requester module that denotes the case in which no requestees are available to accept the current request.

We evaluate this property by varying parameter p and by assuming $N \in \{10, 25, 50, 100\}$. Moreover, we consider: (i) three risky requestees (see Figure 2a), (ii) three requestees among which one is risky and one is cautious, while the default configuration is adopted for the third one (see Figure 2b), and (iii) three cautious requestees (see Figure 2c). All the curves tend rapidly to 1 for $p < 0.5$ and converge to zero as p tends to 1. In particular, notice that in the case of three cautious requestees, for $N \geq 25$ the curves approximate a step function, meaning that a cheating requester is almost immediately untrusted by each requestee.

Three more properties are tested in order to investigate the economic aspects of the cooperation mechanism:

Property 3. What is the number of requests accepted by each requestee?

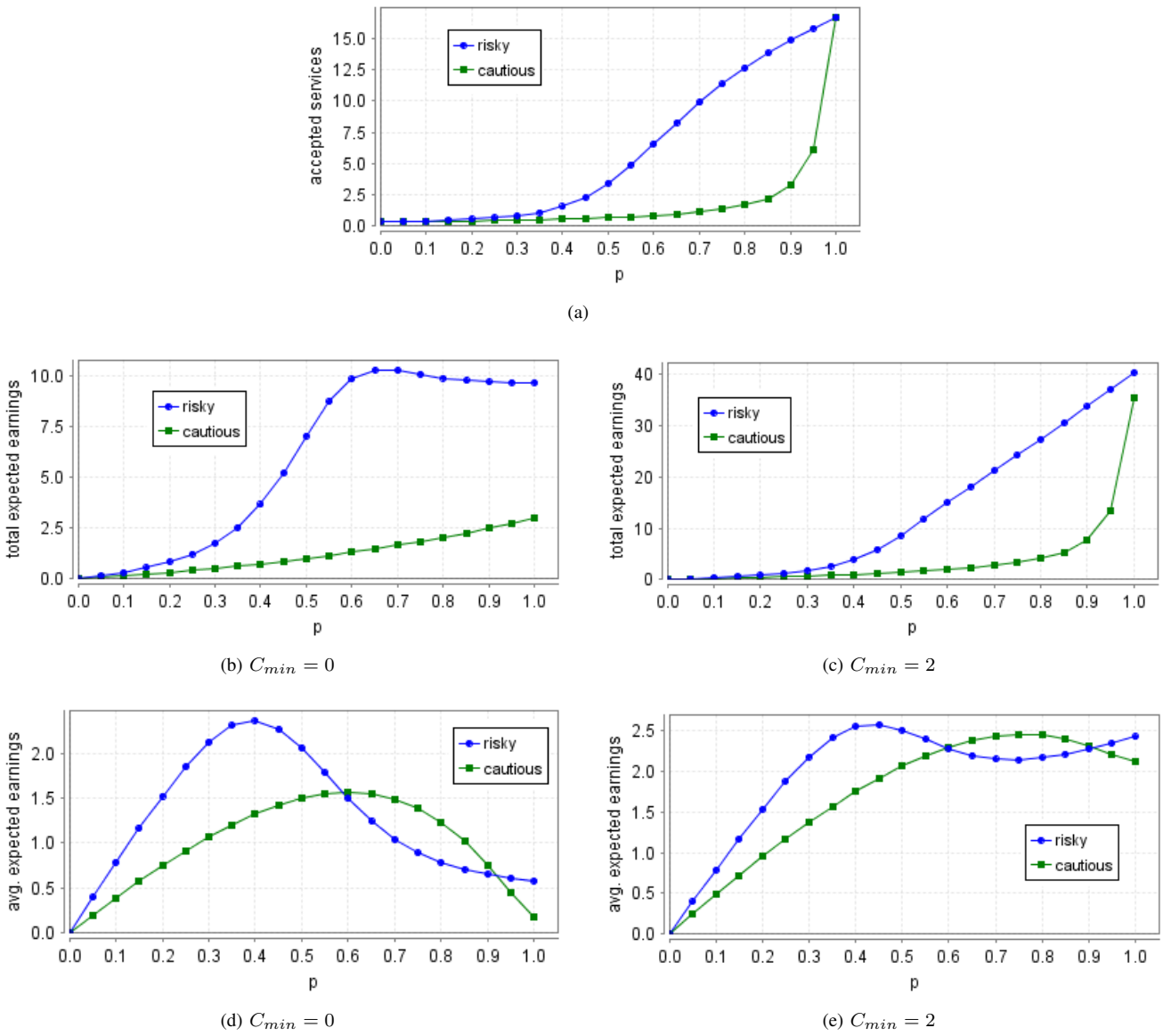
Property 4. What is the total expected earning for each requestee?

Property 5. What is the average earning per accepted request?

For instance, for the first requestee Property 3 is formalized as follows (we can argue analogously for the other properties):

```
R{"acc1"}=? [ C<=t ]
```

where:



(a)

(b) $C_{min} = 0$

(c) $C_{min} = 2$

(d) $C_{min} = 0$

(e) $C_{min} = 2$

Fig. 3: DTMC analysis: verification of Properties 3, 4, and 5.

```
rewards "acc1"
[accept1] true : 1;
endrewards
```

We use these properties to compare the two profiles in a scenario with 50 requests and three requestees like those of Figure 2b. Figure 3 reports the performance of the risky and cautious requestees as a function of parameter p. The curves show the following results.

The number of services accepted by the risky requestee is higher than that related to the cautious requestee, see Figure 3a. The difference is due to the conditions applied by the risky requestee, in particular the assumption $k = 1$, which is much less restrictive with respect to the assumption $k = \infty$ adopted by the cautious requestee. In fact, by setting $k = \infty$ also for the risky requestee, its curve would collapse with that of the cautious requestee. Notice that in case of honest requester (i.e., $p = 1$), the profile of the requestees does not play any role, so that the requests are equally distributed among them, because they are characterized by the same initial reputation.

As p increases, the total expected earnings of the risky requestee become much higher than those of the cautious one, see Figure 3b. The difference can be interpreted as a reward for taking more risk.

Similarly, Figure 3d shows that the average expected earning per service grows with the value of p up to a maximum point beyond which it decreases because of the effect of the trust-based discount applied to trustworthy requesters. Such a

maximum point is reached earlier by the risky requestee, thus motivating the better performance of the cautious requestee for $p \in [0.6; 0.9]$. This result is also confirmed by observing that in such an interval the trust towards the requester becomes stably high from the viewpoint of the risky requestee, as emphasized by the total earnings curve of Figure 3b. For $p \geq 0.95$, the result is better for the risky requestee, because the requester becomes trustworthy also from the viewpoint of the cautious requestee, with a positive impact upon the number of services such a requestee accepts, see Figure 3a.

In general, the combination of remuneration and trust management works as an incentive to adopt a "risky" prosocial behavior. On the other hand, the requester obtains more services at a lower average cost whenever behaving honestly.

C. Sensitivity Analysis

The DTMC analysis of the previous section reveals the tradeoff between indirect and direct rewards, by emphasizing how trust-based mechanisms impact the economic trend of the system. We now investigate more deeply the effect of the various configuration parameters on Properties 3 to 5.

First, we show that the shape of the earnings curves is not purely a side effect of the assumption $C_{min} = 0$. Figs. 3c and 3e report the total and average expected earnings obtained in the case $C_{min} = 2$. The major earnings with respect to the corresponding curves of Figs. 3b and 3d reflect the difference between the minimum costs that are applied in the two experiments.

Second, in order to emphasize the role of parameters k and dt, we tune them for the risky requestee in the same scenario of Figure 3, by showing the related influence upon performance. More precisely, in Figure 4 we vary k in $\{1, 2, \infty\}$, where the case $k = 1$ is taken from Figure 3. Observe that the number of accepted services increases as k decreases. Indeed, as previously shown, k and tolerance to cheating behaviors are inversely proportional. Therefore, decreasing k has the effect of accepting more services, many of which, however, remain unpaid in case of cheating requester. On the other hand, increasing k corresponds to a fast trust decrease and, therefore, higher costs per service. For these reasons, as k decreases, the average expected earnings decrease as well. Also notice that whenever the requester is honest ($p = 1$) and, as a consequence, k is never used, the three curves converge to the same values.

Similarly, we study the effect of tuning the dispositional trust of the risky requestee, by varying dt in $\{low, med, high\}$, where the case $dt = low$ is taken from Figure 3. As shown in Figure 5, increasing the dispositional trust works as an incentive to accept more services as well as to augment the total earnings whenever the requester is rarely honest. The beneficial effect on the total earnings decays as parameter p increases, in which case the most important consequence of increasing dt is a rapid convergence of the service cost towards the minimum cost. Moreover, similarly as observed in the case of parameter k, we have that tolerant behaviors contribute to decrease the average expected earnings.

The consequences of changing the cost function are evaluated in Figure 6, where we propose the same analysis of

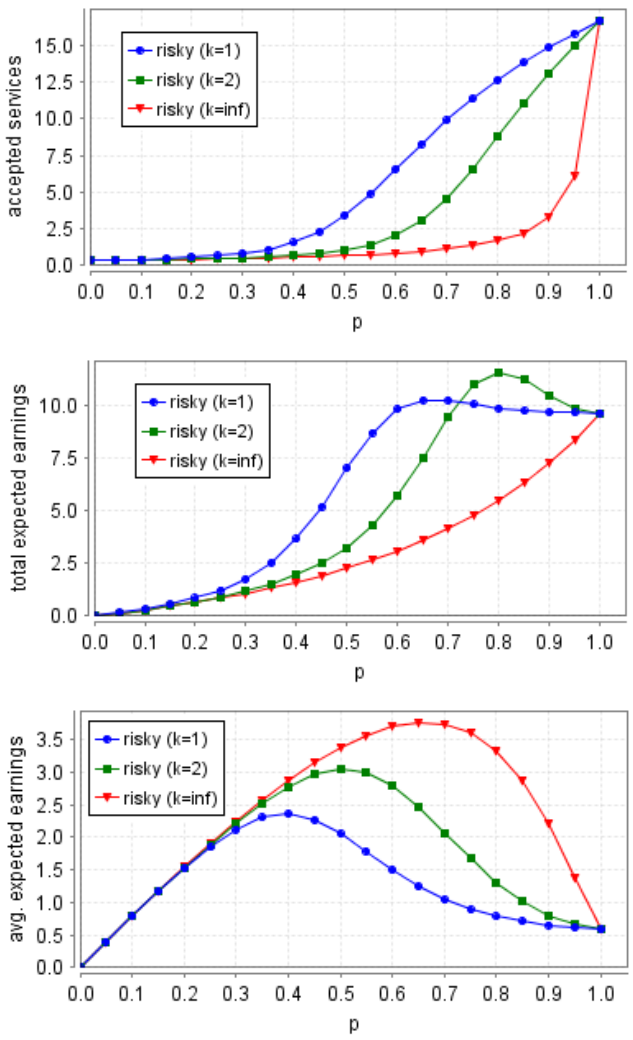


Fig. 4: DTMC analysis: verification of Properties 3 to 5 for the risky requestee by varying parameter k.

Figure 3 by replacing Equation (2) with Equation (3), for which we assume $T_1 = low$, $T_2 = med$, $T_3 = high$, while $C_{min} = 0$, $C_{med} = 4$, $C_{med'} = 7$, and $C_{max} = 10$. By comparing the effects of the two equations, notice that while the values change, the shape of the curves is invariant. Indeed, while at the same conditions Equation (3) ensures higher prices than Equation (2), both functions respect the relation between trust and cost.

Finally, we verify the scalability of results by considering five requestees (four risky and one cautious with the same parameters assumed in the analysis of Figure 3). It is worth comparing the obtained results, see Figure 7, with those of Figs. 3a and 3b. The analogy is emphasized by the fact that the average expected earnings are exactly the same as those of Figure 3d.

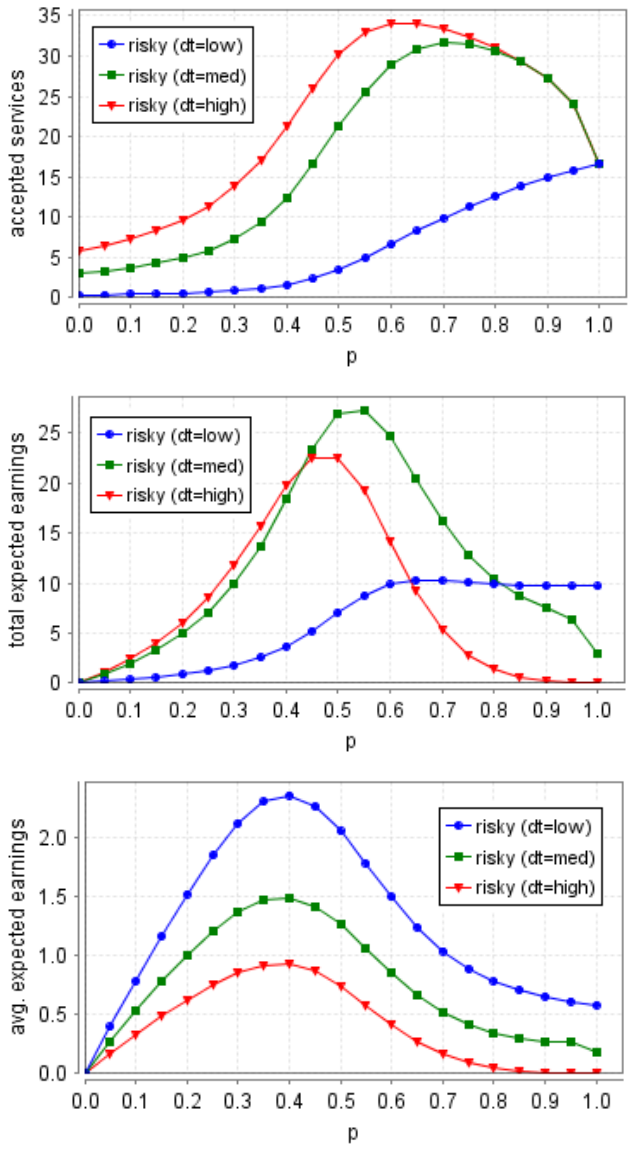


Fig. 5: DTMC analysis: verification of Properties 3 to 5 for the risky requestee by varying parameter dt.

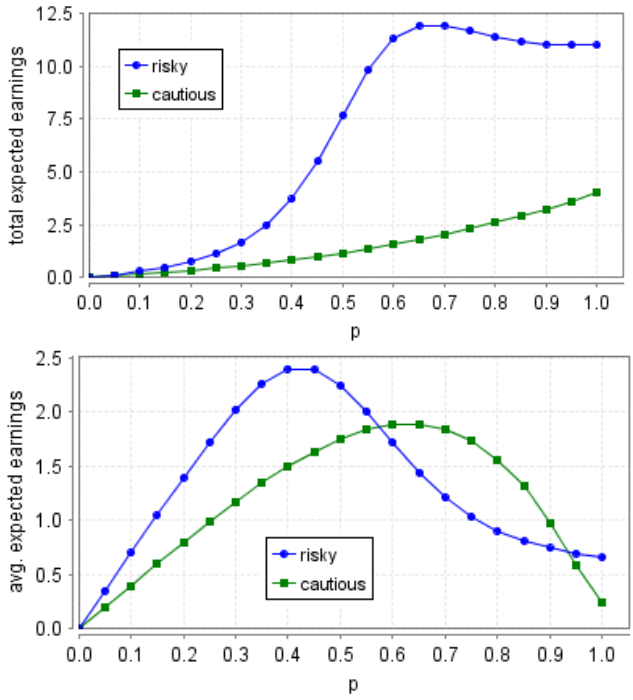


Fig. 6: DTMC analysis: verification of Properties 4 and 5 by using Equation (3).

D. Requester's Choice Policy

While so far we have considered security and economic issues for the basic cooperation process, we now take into account the effect of changing the choice policy adopted by the requester to select a requestee in the first phase, which is one of the most important strategies behind the success of the cooperation incentives. Previous results of the DTMC analysis refer to the reputation-based prioritized choice model, which is the policy with the strongest impact of requestee's reputation upon performance.

By assuming the same scenario of Figure 3, in Figure 8 we analyze Properties 3 to 5 whenever the prioritized choice is governed by best price rather than best reputation. In particular, the performance figures refer to the risky requestee under different values of its dispositional trust, because such a parameter is essential for determining initially the service cost, which depends directly on trust. The results confirm the strong influence of the prioritized mechanism and reveal similarities with the experiment of Figure 5. As dt increases, the risky requestee attracts rapidly most service requests, because the applied service cost is inversely proportional to the dispositional trust. The threshold value affecting the shape of the curves is $dt = med$ – which represents the dispositional trust of the cautious requestee – thus revealing the important role of this parameter in the competition among requestees. The relation between dispositional trust and total expected earnings is strict as well. While increasing dt is beneficial for low values of parameter p, the trend is inverse as p increases. Indeed, a high value of dt allows the trustworthy requester to rapidly attain the maximum trust-based discount. For similar reasons, increasing dt cannot have a positive influence upon the average expected earnings.

The analysis concerning the price-based selection is completed by verifying the effects of breaking the relation between cost and trust. In the same scenario of the previous experiment, for the risky requestee we assume $dt = med$ and a constant cost function $C = z$, where z ranges in $\{3, 4, 5\}$. Then, we concentrate on Property 3, see Figure 9. The obtained result is zero services for $z \geq 5$, while the other curves suggest that in order to be competitive with the corresponding curve of

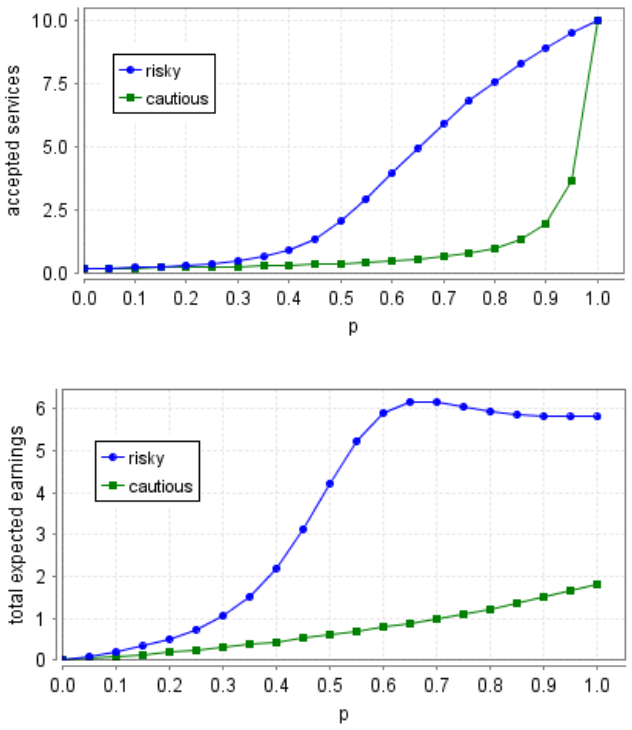


Fig. 7: DTMC analysis: verification of Properties 3 and 4 with 5 requestees.

Figure 8a, it is necessary to set a very low constant price. As a consequence, breaking the relation between cost and trust is not beneficial from the viewpoint of the requestees. On the other hand, from the viewpoint of the requester we have that the average expected cost per service is equal to z independently of the attitude to behave honestly. Instead, by considering the corresponding curve of Figure 8c, we observe that the average expected cost per service converges to a value close to zero as p tends to 1. Therefore, breaking the relation between cost and trust does not work as an incentive to honest behaviors of the requester.

Finally, we replace the reputation-based prioritized selection of the requestee with the reputation-based probabilistic model. In Figure 10, we report the results corresponding to the same scenario of Figure 3. As far as Property 3 is concerned, the difference is negligible, because the choice model adopted by the requester does not affect its trustworthiness as perceived by the requestees. The case of Property 4 and, as a consequence, Property 5, is more interesting. On one hand, in the prioritized model most requests involve the requestee with highest reputation, thus allowing the requester to reach rapidly the minimum service cost as p tends to 1. On the other hand, in the probabilistic model the requests are more equally distributed, thus slowing down the convergence towards the minimum price and justifying the major earnings for the requestees. Therefore, we derive that the prioritized model of choice is more favorable from the viewpoint of the requester.

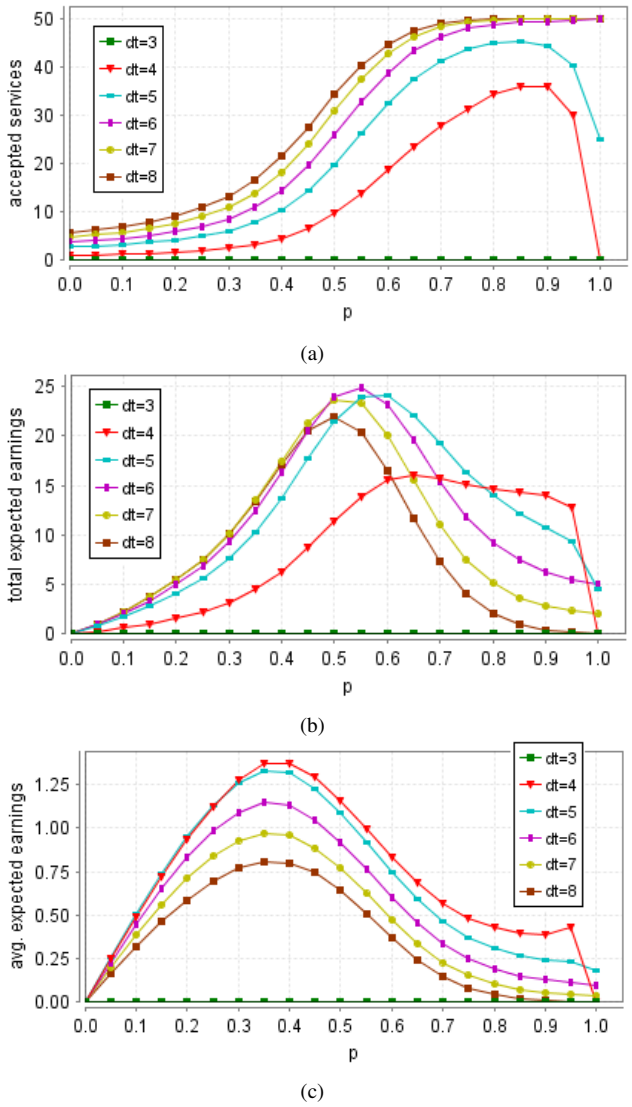


Fig. 8: DTMC analysis: verification of Properties 3 to 5 for the risky requestee with price-based selection of the requestee.

E. Requestee's Reputation

Requestee's reputation is an orthogonal aspect the effects of which are analyzed in Figure 11. The objective is to measure the impact of requestee's reputation with respect to Property 3. In Figure 11a, we consider prioritized choice of the requestee, one risky requestee with reputation *high*, one cautious requestee with reputation *low*, while the reputation of the third requestee (with default profile) is *med*. Regardless of the profile, all the requests are served by the requestee with highest reputation, as imposed by the choice strategy followed by the requester. In fact, an analogous result would be obtained by swapping the reputations of the risky and cautious requestees. Giving less importance to reputation during the discovery phase has the effect of mitigating such a drastic

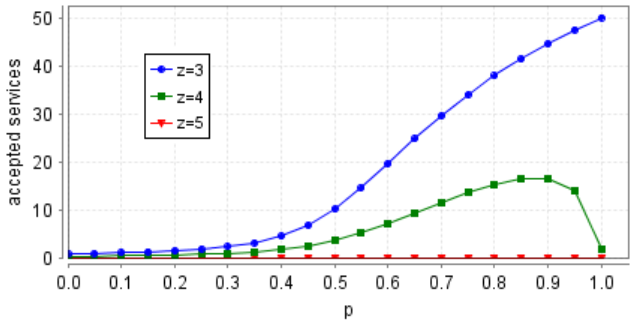


Fig. 9: DTMC analysis: verification of Property 3 for the risky requestee with price-based selection of the requestee and constant cost function.

behavior, as confirmed by the following experiment, in which the prioritized model of choice is replaced by the probabilistic one. The results, shown in Figure 11b, emphasize that also the cautious requestee can obtain some service. However, regardless of the value of p, the cautious requestee is always outperformed by the risky requestee.

The effect of requestee's reputation is investigated also by testing the performance of a paranoid requestee ($\alpha = 0.5$, $dt = low$, $st = med$, $k = \infty$) replacing the cautious requestee in the experiment of Figure 3. In Figure 12a, we evaluate Property 5 for the paranoid requestee in two possible cases depending on its initial reputation. Apparently surprising, a paranoid requestee with reputation *med*, when put in competition with the other requestees (whose reputation is *low*), does not obtain any reward. This result is motivated by the fact that, initially, the paranoid requestee does not accept any request until a sufficiently high number of positive recommendations is received, because its service trust level is higher than its dispositional trust. Moreover, such requests are accepted by the other requestees, which gain reputation, thus causing preemption over the paranoid requestee during the prioritized discovery phase. In order to observe some request served by the paranoid requestee, it is necessary to set its initial reputation to *high*. In this case, we evaluate also Property 3 (see Figure 12b) and Property 4 (see Figure 12c). Notice that the paranoid requestee accepts a very low number of services for $p < 0.9$ and outperforms the risky requestee only for $p = 1$, the reason being that the honest requester becomes trustworthy rapidly enough to overcome the non-cooperative attitude of the paranoid requestee.

F. Impact of Feedback

In this section, we concentrate on the role of feedback for the functioning of the cooperation incentives. On one hand, we analyze the case in which the requester provides a negative feedback. On the other hand, we observe the effects of inaccurate recommendations provided by the requestees.

In a real-world setting, the quality of the delivered service may not match the negotiated parameters. The consequence is

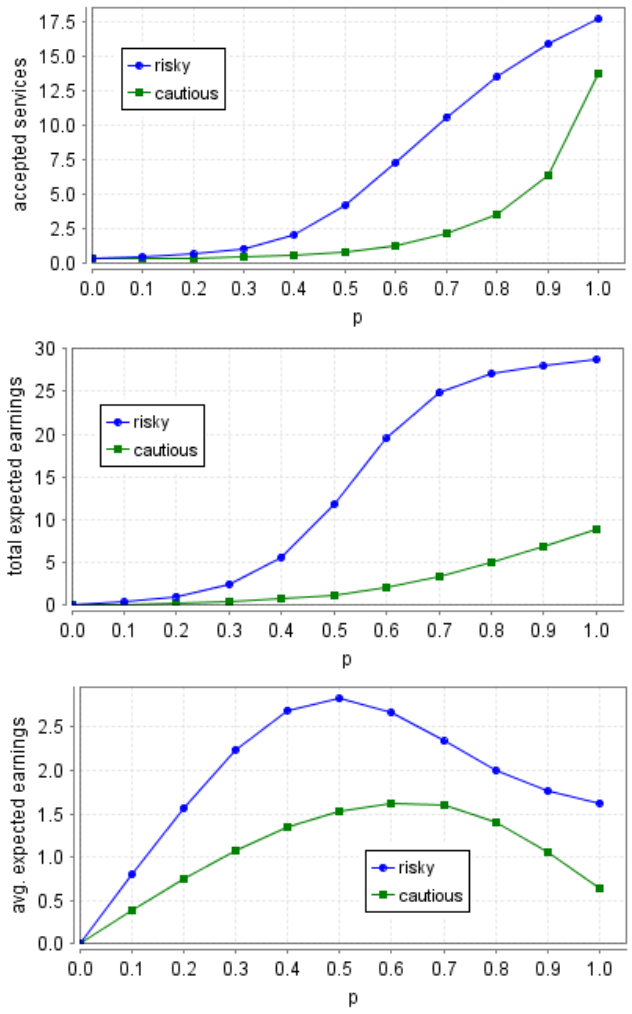
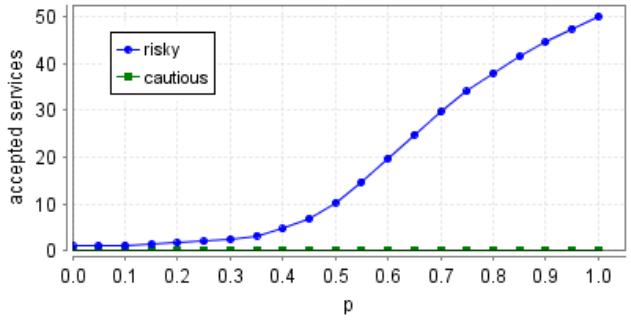


Fig. 10: DTMC analysis: verification of Properties 3 to 5 with probabilistic selection of the requestee.

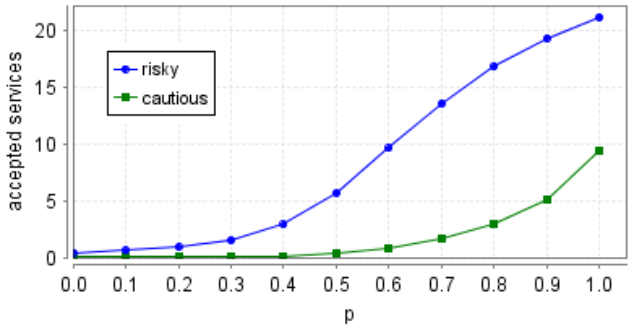
a negative feedback of the requester that impairs the reputation of the requestee. This situation is not captured by the experiments reported so far. Hence, we now represent the (possibly negative) change of requestee's reputation due to requester's evaluations in order to check the following property.

Property 6. How is requestee's reputation related to the number of accepted requests in the case of fallible services?

For design issues, we model probabilistically with parameter $q \in [0, 1]$ the event of a service failure causing a negative evaluation. Notice that in the scenario of the previous experiments, modeling an ideal service provider, it holds that $q = 0$. Hence, we consider two additional situations. In a pessimistic case, upon each served request, requestee's reputation has the same probability of remaining unchanged, being increased by 1, or being decreased by 1 (namely, $q = 0.33$). In an optimistic case, the probabilities of these three events are 0.15, 0.8, and 0.05, respectively (namely, $q = 0.05$) For analysis



(a) Prioritized choice (risky rep. = *high*, cautious rep. = *low*)



(b) Probabilistic choice (risky rep. = *high*, cautious rep. = *low*)

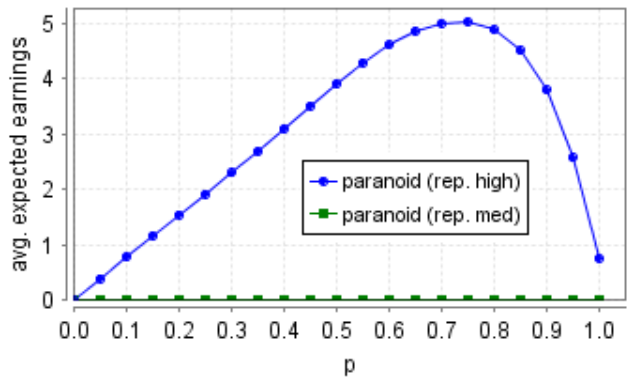
Fig. 11: DTMC analysis: verification of Property 3 with mixed reputations.

purposes, we consider a honest requester using reputation-based prioritized choice, one cautious requestee with reputation *high*, one requestee with default profile and reputation *med*, and one risky requestee. In Figure 13, we evaluate Property 6 for the risky requestee, by varying its initial reputation from 1 to 10. For $q = 0$, a risky requestee with initial reputation less than *high* is always outperformed by the cautious requestee. The two requestees share the same amount of services if the initial reputation of the risky requestee is *high* as well, while the risky requestee takes all the requests in the remaining cases. These results depend on the deterministic trend of reputations, which never decrease. The other curves approximate such a behavior (the lower q is, the closer the approximation becomes) and reveal that the possibly negative feedback provided by the requester affects the performance of the requestees.

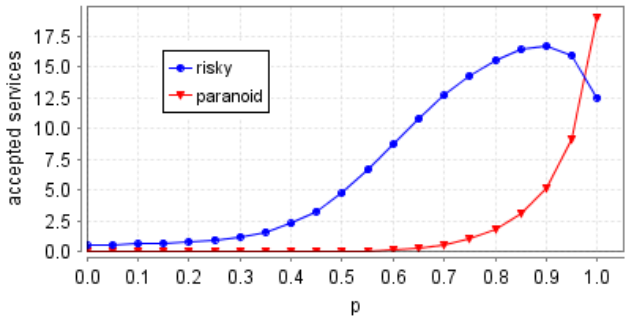
In an orthogonal way with respect to the previous experiment, we now consider the case of non-cooperative requestees, which may refuse a request even if the requester is trustworthy enough to access the service. Hence, the property of interest is as follows.

Property 7. How does requestee's reputation vary in the case of non-cooperative requestees?

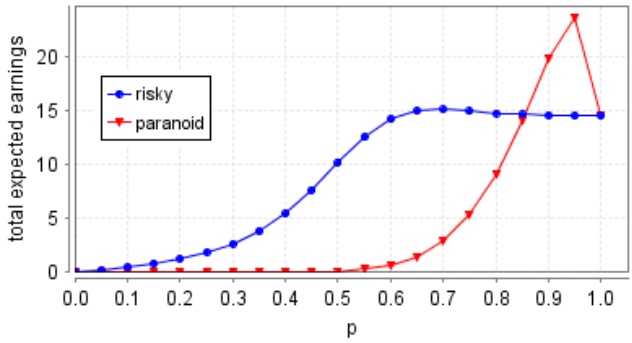
As an abstraction, we model probabilistically with parameter $c_i \in [0, 1]$ the cooperative attitude of requestee i, such that i accepts a trustworthy request with probability c_i and refuses it with probability $(1 - c_i)$. Obviously, refusing a trustworthy request is evaluated with a reputation decrease, as opposite to



(a) risky rep. = *low*



(b) risky rep. = *low*, paranoid rep. = *high*



(c) risky rep. = *low*, paranoid rep. = *high*

Fig. 12: DTMC analysis: verification of Properties 5, 3, and 4 with paranoid requestee.

the reputation increase determined by a satisfactory service.

For analysis purposes, we consider a honest requester using reputation-based prioritized choice, and three risky requestees with initial reputation *low*. In Figure 14a, we evaluate Property 7 for the first requestee by varying parameter c_1. In particular, we report its average relative reputation variation after 50 requests in two different cases, depending on the behavior of the other two requestees. In the first case, they are fully cooperative (i.e., $c_2 = c_3 = 1$), while in the second case they are partially cooperative (i.e., $c_2 = c_3 = 0.5$). In general, we observe that the lack of cooperation attitude has a negative impact upon reputation, which converges towards the

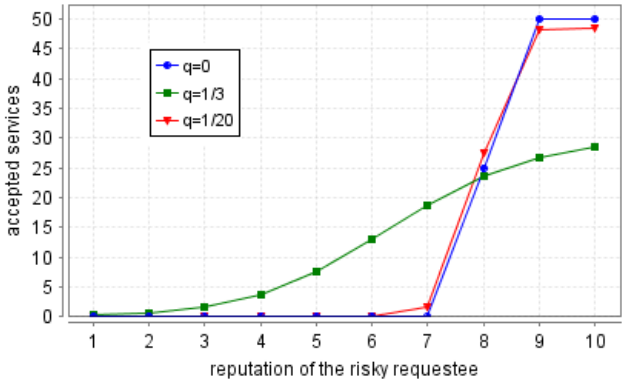


Fig. 13: DTMC analysis: verification of Property 6.

top level as c_1 increases. We also observe that the reputation variation is slower in the first case with respect to the second case. The reason is that in the first case most services are required to the two cooperative requestees, whose reputation increases rapidly thanks to their prosocial behavior. In order to emphasize the benefits of cooperative behaviors, in Figure 14b we evaluate Property 3 for the first requestee in the two cases above. Notice that in the second case the number of services accepted by the first requestee increases dramatically whenever its attitude to cooperate becomes higher (≥ 0.5) than that of the other two requestees.

The accuracy of feedback is a critical aspect of trust-based incentive mechanisms, as emphasized in [27], where additional incentives are proposed to stimulate the honest and active participation in the evaluation and feedback phase. In Figure 15, we evaluate the effects of inaccurate recommendations on Properties 3 to 5 for the same scenario of Figure 10. In particular, we model with parameter $f \in \{-5, 0, 5\}$ the error introduced to alter the correct recommendations to be provided to other users. The results refer to the risky requestee, whose trust formula is the most influenced one by recommendations ($\alpha = 0.5$). As can be noticed, false positive recommendations have a significant impact, especially for $p \leq 0.5$, as they contribute to increase the trust towards a dishonest requester. On the other hand, false negative recommendations impair the performance, especially for $p \geq 0.5$, as they contribute to keep the requester from obtaining the service. For $p = 1$, the influence of altered recommendations is negligible, because a completely honest requester is trusted enough to get always the service. In general, this analysis confirms the importance of motivating the requestees to provide honest recommendations. On the other hand, we also derive that a honest requester is protected from the feedback variability.

G. Discussion

In summary, cooperation incentives work properly for both the requester and the requestee. For instance, a honest behavior of the requester is motivated by a higher number of accepted services at a lower average cost with respect to the results

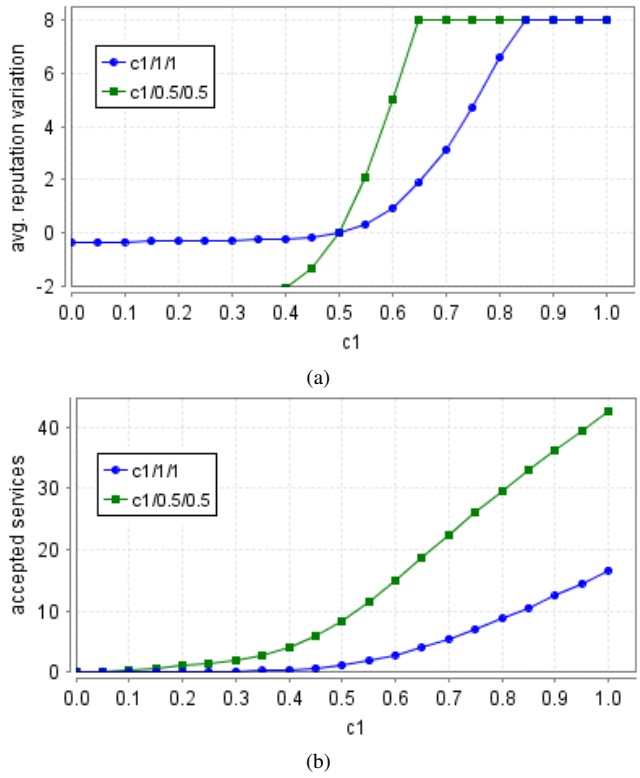


Fig. 14: DTMC analysis: verification of Properties 7 and 3 with non-cooperative requestees.

obtained by a possibly cheating requester. This relation is exacerbated whenever the requester adopts a prioritized model for choosing the requestee during the discovery phase. From the viewpoint of the requestee, both the reputation and the attitude to cooperate affect the amount of delivered services and the related earnings. Moreover, cautious choices for the configuration parameters influencing trust reduce the risk of suffering cheats but impair directly the earning opportunities and indirectly the reputation if in the network cooperative requestees are active.

The sensitivity analysis emphasizes the influence of each policy and configuration parameter chosen by the involved parties. In any case, the results confirm that making cost and trust mutual dependent plays a fundamental role for the success of the cooperation incentives. Similarly, the reliability of trust variations as well as the accuracy of feedback represent important conditions affecting all the performance figures. These relations demonstrate that cooperation incentives provide necessary motivations for the sustainability of collaborative networks.

V. Conclusion

Mixed incentive strategies, combining reputation and price-based mechanisms, have proved to be effective in inducing prosocial behaviors while isolating selfish or cheating nodes,

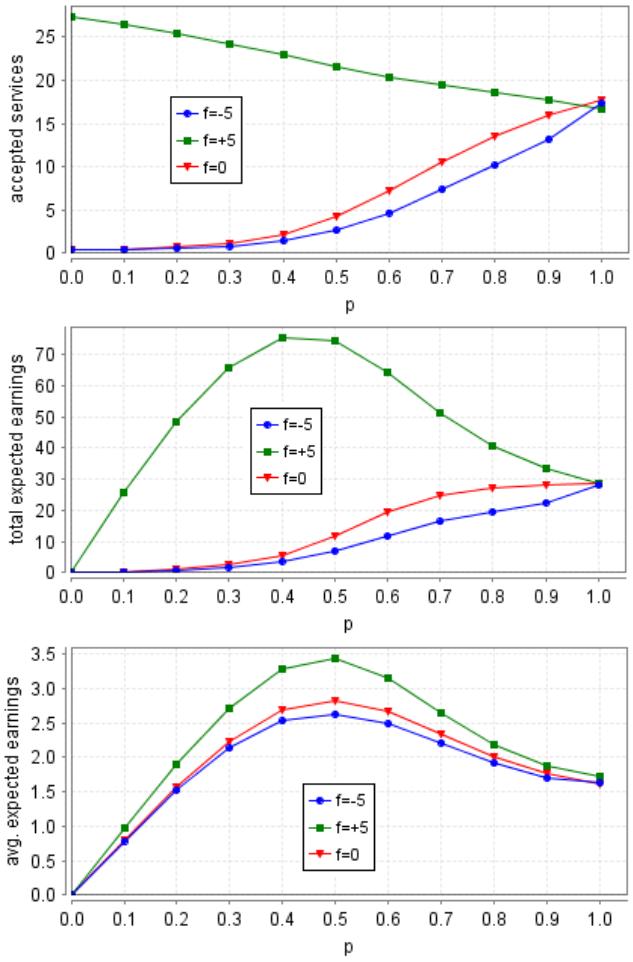


Fig. 15: DTMC analysis: verification of Properties 3 to 5 for the risky requestee with probabilistic selection of the requestee and altered feedback.

as already claimed in [14]. Following such a principle, a cooperation process entailing both trust management and virtual currency has been recently proposed for wireless user-centric networks [2]. This paper has reported the results obtained by applying model checking techniques in order to provide formal evidence of the properties of such a cooperation process.

The same formal approach can be applied to verify the robustness of cooperative networks in more complex environments in which the incentive mechanisms are contrasted by coalition or sybil attacks (see, e.g., [28]). Alternatively, it can be used to evaluate the social, security, and performance effects of the adoption of specific payment systems.

The ideas presented in this work are currently under development in order to build a design tool to be used to assist the design and configuration of mixed incentive strategies in real-world user-centric networks. In particular, the perspectives provided in this paper are under consideration for being adopted by the ULOOP Consortium [29].

ACKNOWLEDGMENT

The research leading to these results has received funding from the EU IST Seventh Framework Programme ([FP7/2007-2013]) under grant agreement number 257418, project ULOOP User-centric Wireless Local Loop.

REFERENCES

[1] A. Aldini and A. Bogliolo, “Model Checking of Trust-Based User-Centric Cooperative Networks,” Proc. 4th Int. Conf. on Advances in Future Internet (AFIN’12), IARIA, 2012, pp. 32–41.

[2] A. Bogliolo, P. Polidori, A. Aldini, W. Moreira, P. Mendes, M. Yildiz, C. Ballester, and J.-M. Seigneur, “Virtual Currency and Reputation-Based Cooperation Incentives in User-Centric Networks,” Proc. 8th Int. Wireless Communications and Mobile Computing Conf. (IWCMC’12), IEEE Press, 2012, pp. 895–900.

[3] V. Forejt, M. Kwiatkowska, G. Norman, and D. Parker, “Automated Verification Techniques for Probabilistic Systems,” in M. Bernardo and V. Issarny, Eds., Formal Methods for Eternal Networked Software Systems (SFM’11), LNCS, vol. 6659, Springer, 2011, pp. 53–113.

[4] M. Kwiatkowska, G. Norman, and D. Parker, “PRISM 4.0: Verification of Probabilistic Real-time Systems,” Proc. 23rd Int. Conf. on Computer Aided Verification (CAV’11), LNCS, vol. 6806, Springer, 2011, pp. 585–591.

[5] M. Kwiatkowska, G. Norman, and D. Parker, “Stochastic Model Checking,” in M. Bernardo and J. Hillston, Eds., Formal Methods for Performance Evaluation (SFM’07), LNCS, vol. 4486, Springer, 2007, pp. 220–270.

[6] R. Alur and T. Henzinger, “Reactive Modules,” Formal Methods in System Design, vol. 15, 1999, pp. 7–48.

[7] W.-J. Stewart, “Introduction to the Numerical Solution of Markov Chains,” Princeton, 1994.

[8] R. Segala, “Modelling and Verification of Randomized Distributed Real Time Systems,” Ph.D. thesis, MIT Press, 1995.

[9] C. Baier and J.-P. Katoen, “Principles of Model Checking,” MIT Press, 2008.

[10] A. Abraham and A.-E. Hassanien (Eds.), “Computational Social Networks: Tools, Perspectives and Applications,” Springer, 2012.

[11] Y. Zhang and M. Guizani (Eds.), “Game Theory for Wireless Communications and Networking,” CRC Press, 2011.

[12] F. Fitzek and M. Katz (Eds.), “Cognitive Wireless Networks,” Springer, 2007.

[13] K. El Defrawy, M. El Zarki, and G. Tsudik, “Incentive-Based Cooperative and Secure Inter-Personal Networking,” Proc. Int. Workshop on Mobile Opportunistic Networking (MobiOpp’07), ACM Press, 2007, pp. 57–61.

[14] Z. Li and H. Shen, “Game-Theoretic Analysis of Cooperation Incentives Strategies in Mobile Ad Hoc Networks,” IEEE Transactions on Mobile Computing, vol. 11(8), 2012, pp. 1287–1303.

[15] U. Golas, K. Hoffmann, H. Ehrig, A. Rein, and J. Padberg, “Functorial Analysis of Algebraic Higher Order Net Systems with Applications to Mobile Ad-Hoc Networks,” Electronic Communication of the European Association of Software Science and Technology, vol. 40, 2010.

[16] S. Nanz and C. Hankin, “A Framework for Security Analysis of Mobile Wireless Networks,” Theoretical Computer Science, vol. 367, 2006, pp. 203–227.

[17] V. Srivastava, J. Neel, A. MacKenzie, R. Menon, L. DaSilva, J. Hicks, J. Reed, and R. Gilles, “Using Game Theory to Analyze Wireless Ad Hoc Networks,” IEEE Communications Surveys and Tutorials, vol. 7(4), 2005, pp. 46–56.

[18] A. Acquaviva, A. Aldini, M. Bernardo, A. Bogliolo, E. Bontà, and E. Lattanzi, "Assessing the Impact of Dynamic Power Management on the Functionality and the Performance of Battery-Powered Appliances," Proc. 5th Int. Conf. on Dependable Systems and Networks (DSN'04), Performance and Dependability Symposium, IEEE Press, 2004, pp. 731–740.

[19] E. Altman, T. Boulogne, R. E. Azouzi, T. Jimenez, and L. Wynter, "A Survey on Networking Games in Telecommunications," Computers and Operations Research, vol. 33(2), 2006, pp. 286–311.

[20] C.H. Declerck, C. Boone, and G. Emonds, "When Do People Cooperate? The Neuroeconomics of Prosocial Decision Making," working paper of the Faculty of Applied Economics, University of Antwerp, 2011.

[21] S. Marsh, "Formalizing Trust as a Computational Concept," Ph.D. thesis, Department of Mathematics and Computer Science, University of Stirling, 1994.

[22] S. Greengard, "Social Games, Virtual Goods," Communications of the ACM, vol. 54(4), 2011, pp. 19–22.

[23] A. Jøsang, "Trust and Reputation Systems," in A. Aldini and R. Gorrieri, Eds., Foundations of Security Analysis and Design IV (FOSAD'07), LNCS, vol. 4677, Springer, 2007, pp. 209–245.

[24] M. Yildiz, M. A. Khan, F. Sivrikaya, and S. Albayrak, "Cooperation Incentives Based Load Balancing in UCN: A Probabilistic Approach," Global Communications Conf. (GLOBECOM'12), IEEE Press, 2012, pp. 2746–2752.

[25] Y. Zhang, L. Lin, and J. Huai, "Balancing Trust and Incentive in Peer-to-Peer Collaborative Systems", Journal of Network Security, vol. 5, 2007, pp. 73–81.

[26] A. Aldini, M. Bernardo, and F. Corradini, "A Process Algebraic Approach to Software Architecture Design," Springer, 2010.

[27] A. Fernandes, E. Kotsovinos, S. Ostring, and B. Dragovic, "Pinocchio: Incentives for Honest Participation in Distributed Trust Management," Proc. 2nd iTrust Conf., LNCS, vol. 2995, Springer, 2004, pp. 63–77.

[28] F. G. Marmol and G. M. Perez, "Security Threats Scenarios in Trust and Reputation Models for Distributed Systems," Computer and Security, vol. 28, 2009, pp. 545–556.

[29] ULOOP, "EU IST FP7 ULOOP: User-Centric Wireless Local Loop," 2013, [accessed 20-June-2013]. [Online]. Available: http://uloop.eu

Securely connecting Electric Vehicles to the Smart Grid

Rainer Falk and Steffen Fries
Corporate Technology
Siemens AG
Munich, Germany
e-mail: [rainer.falk | steffen.fries]@siemens.com

***Abstract*—Rechargeable electric vehicles are receiving increasing attention from different stakeholders: from customers as gas prices are constantly rising, from car manufacturers to address customer, market, and environmental demands, and also from electric energy utilities for integrating them into smart electric grids. While in the first step, the emphasis is placed on electric vehicles as energy consumers, using their battery for storing energy and feeding it back to the energy network will be the consequent next step. Batteries of electric vehicles will realize a distributed energy electric storage for stabilizing the electric power grid. Thus the electric vehicle will participate as a mobile energy node within the smart grid having two types of interfaces, one for electricity and one for data communication for charging and feedback control, information exchange, and for billing. Since IT security in the smart grid is already considered as a major point to be addressed, the enhancement of the smart grid with electric mobility has to address IT security as well. This article describes example interactions of electric vehicles with the charging infrastructure and it shows which security requirements have to be fulfilled in important use cases. Moreover, security considerations of current standardization activities in ISO/IEC and SAE are described.**

Keywords*—*eMobility security; Smart Grid security; charging infrastructure; IEC 61851; IEC 15118

I. INTRODUCTION

The Smart Grid can be roughly characterized as a combination of two infrastructures, the electrical grid carrying the energy, and the information infrastructure used to supervise and control the electrical grid operation. The importance of information security for the power systems communication infrastructure has increased tremendously over the last couple of years. Until recently, automation has mainly targeted the transmission network to address the multilateral exchange of energy from different providers. With the advent of decentralized energy resources like wind parks and solar cells and their interaction with the electric grid there is a higher demand for automation in the distribution network. These energy resources show a high fluctuation depending on the environmental conditions and also go along with the possibility to influence energy demand. This will require supporting demand response services. The introduction of electric vehicles as flexible load, and in the future potentially as decentralized energy resource (power feedback), emphasizes this development (see also [1]).

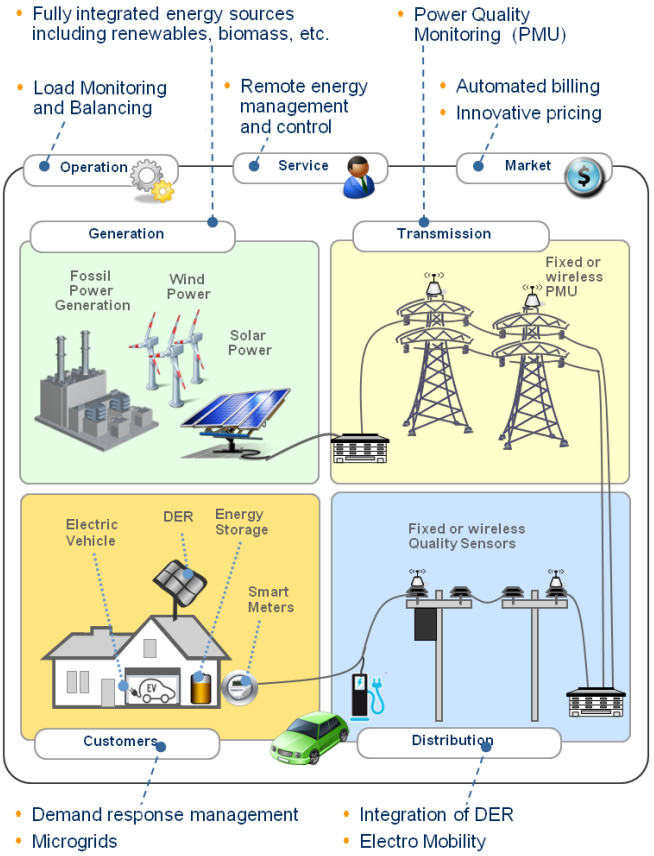


Figure 1. Potential Smart Grid Scenarios.

Figure 1 shows a high level view on typical smart grid scenarios, also targeting the integration of electric vehicles. The four center domains shown are the typical domains, used to describe a smart grid:

- **(Bulk) Energy Generation** is the process of converting non-electrical energy into electricity, and is the first step in the process of delivering power to consumers. Besides classical energy generation like coal- or gas-fired power plants or nuclear power plants, decentralized energy generation using photovoltaic, block heat and power plants, or windmills are getting more and more integrated into the power grid for bulk energy generation.
- **Power Transmission** is the bulk transfer of electric power to substations. A power transmission network connects power plants generating electrical energy with

substations and typically works on high voltage level (e.g., 380 kV).

- **Energy Distribution**: Substations distribute the electrical energy further down to industrial, commercial, or residential consumers in the range of medium voltage (typically covers the range between 20kV to 100kV). Substations provide the transition to the low voltage area (typically around 400V). The energy distribution level is likely to provide connection points for vehicle charging, especially, when high power AC or DC charging spots are used.
- **Customer**: The customer role as consumers of electric energy was typically the endpoint for the energy transfer. Within the Smart Grid, this role may change due to the option to move from pure consumption of energy to producing and storing energy in residential areas. Then the customer would become a so-called prosumer. As visible in Figure 1, electric vehicles may connect to the customer or the distribution domain.

There exists further Smart Grid domains like operation and service of the four domains stated above as well as the market, which enables the interaction between energy generators an energy consumers. The number of electric vehicles as bicycles, motorcycles, and cars is expected to increase significantly. Electric vehicles will be connected with the Smart Grid for charging or even for power feedback. Typically, they connect to the Smart Grid through charging stations or charging points. Charging points in public or corporate places provide the possibility for high power AC or DC charging. Other connection points may be provided by combined service stations, e.g., for parking lots or common home power plugs in residential areas. Closely linked with the pure flow of energy is the management and control of the energy demand for charging electric vehicles. It allows matching the energy demand for the charging process with the energy available at the specific location within the energy grid. A defined part of the vehicle battery's capacity can also be used as energy storage to stabilize the energy grid when needed by feeding back energy from the vehicle to the electrical grid. Besides the control of energy flow there may be a second communication channel for the billing for consumed or provided energy.

The charging infrastructure as a part of the critical infrastructure Smart Grid requires integrated protection against unintentional and intentional attacks. Safety and IT security measures, which are already being part of the Smart Grid core (e.g., defined as standard or realized in proprietary deployments), need to be enhanced to cover also the Smart Grid access infrastructure. This Smart Grid access infrastructure is provided for electric vehicles through the charging infrastructure. While current deployments do not feature an information exchange between the electric vehicle and the charging infrastructure beside a minimum local control of the charging process through pilot signals, upcoming standards and proposed scenarios provide feature rich communication options. The Smart Grid communication and control network of an energy utility is increasingly opened to various nodes not being under control of any energy network operator and thereby exposed to attacks.

Highly dependable management and operations of the information infrastructure are prerequisites for a highly reliable energy network as the power system increasingly relies on the availability of the information infrastructure. Therefore, the information infrastructure must be operated according to the same level of reliability as required for the stability of the power system infrastructure to prevent any type of outage. Especially consumers and utility companies can both benefit from managing this intelligently, and standards anticipating the new environment are emerging from many directions (see [2]). The immediately apparent security needs target the prevention of financial fraud and ensure the reliable operation of the power grid. Both are complex objectives. But surely all of the security ramifications of the charging infrastructure have not been discovered yet. Especially the interaction between new market participants and value added services is currently under investigation. In any case, ensuring privacy, safety, and assuring that the charging service is operating correctly are basic objectives to derive related IT security requirements. Hence, integrated information security is a central part of the charging infrastructure.

The remainder of this paper is structured as follows: Section II describes use cases around the electric vehicle charging infrastructure. Section III discusses information assets derived from the use cases, threats to these assets and also defines first security requirements. Section IV gives an overview about the security standardization for the vehicle to grid interface, while Section V concludes the document.

II. USE CASES

The electrical vehicle charging infrastructure consists of a combination of power services for electric vehicles and value-added services based on the information and communication infrastructure as illustrated in Figure 2.

One main goal of this information and communication infrastructure is to offer customers a choice of service options beneficial to all three, the utility company, the mobility operator as power (service) provider, and the customer. The utility can operate most efficiently when energy demand is fairly constant over time. Price incentives can be offered towards those customers having a flexible vehicle charging schedule with the objective to smooth out energy demand variations. This requires the analysis and consideration of several variables, e.g., schedule, equipment, location, payment options, and additional services.

The variety of peers in a charging infrastructure as depicted in Figure 2 shows the complexity, but also the manifold of possibilities for optimized service offerings.

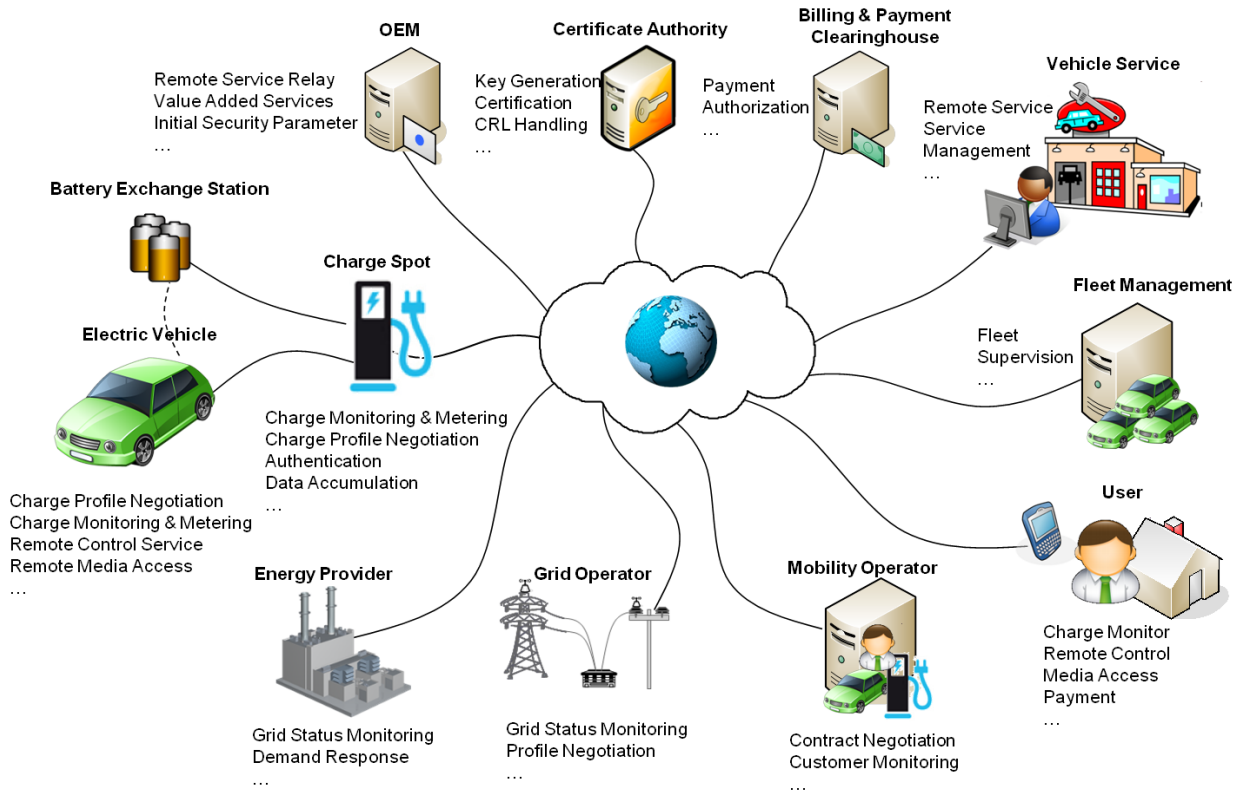


Figure 2. Communication among Actors of an Electric Vehicle Charging Infrastructure.

While not shown in Figure 2, there are different protocol frameworks being used for the communication between the different participants in the scenario.

The following list provides a short overview of potential protocol candidates:

- ISO/IEC 15118 – Communication between electric vehicle and charging spot (cf. [6], [7], and [8])
- IEC 61850 – Communication between charging spot and energy provider (cf. [11])
- OCPP (Open Charge Point Protocol) for the communication between charging spot and mobility operator. Note, as OCPP is not yet a standardized protocol per se, work is currently ongoing to define an infrastrucutre related protocol. It is likely, that this will enhance the exisiting IEC 61850 protocol series.
- OCSP (Online Certificate Status Protocol) between charging spot (or mobility operator) and certification authority.

Further protocols exist, which are not stated here, allowing user interaction with the electric vehicle or the charging infrastructure as well as protocols for the provisioning of value added services. Value added services may be for instance the firmware update of the infotainment system during charging.

The following subsections provide an overview on potential use cases surrounding the charging infrastructure. Each subsection provides potential realization options for the considered use case. Note that the use case discussion stems mainly from standardization work currently done in ISO (International Standardization Organization), IEC (International Electrotechnical Comission), and SAE (Society of Automotive Engineers). but the use cases show the potential of a Smart Grid charging infrastructure to be a flexible platform to realize a variety of known and upcoming service offerings.

A. Control of the Electric Vehicle Charging Environment

Connecting electric vehicles with the charging infrastructure provides flexible control of the charging process through enhanced communication between electric vehicle, charging spot, and the energy provider in the backend, e.g., to adapt the charging to the current energy provisioning situation. It also covers scenarios with limited control of the charging operation through the charging spot or backend. Charging in these scenarios may be controlled completely by the electric vehicle to the limits set by the environment. This is typically the case for AC (alternating current) charging, while in DC (direct current) charging control is being performed by the charging spot.

B. Connecting to the Charging Infrastructure

Connecting a vehicle to the charging infrastructure may use a portable cord set to be provided by either the electric vehicle owner or the charge spot operator. This cord set and the connectors may be different depending whether charging is being done using AC or DC , or depending on the country. An alternative is provided through wireless (inductive)

charging avoiding any power cord to the car. Special consideration of the physical charging environment is necessary here, to ensure safe operation.

C. Billing and Payment for Charging Service

Billing and payment for consumed energy or value added services can be performed through various options:

- At the charging spot, including money, prepaid, credit cards, combination with parking ticket, etc.
- From within the vehicle (e.g., via a contract-related credential stored within the car). This option includes identification of the electric vehicle as well as charging contract verification.

Besides the direct customer interaction, there is also the interaction with clearinghouses that settle accounts between different energy providers. These become necessary when using contract based payment from within a car at a charging spot belonging to a different mobility provider.

D. Negotiated Incentive Rate Plan

Negotiating incentive rate plans may depend on, e.g., the contract between the customer and the mobility provider. Thus different realization options may be:

- **Time of use (TOU)**: The utility provides a price incentive to charge a vehicle at times of lower demand typically based on time of day, day of week, and season of year. Prices are set ahead of time, in an attempt to shift load towards a more favorable time of day.
- **Direct load or price control through utility**: The customer receives a price incentive to give the utility direct control over the charging process. Normally, the customer is given a fixed, reduced price, and the utility has the option to interrupt or delay charging at critical times.
- **Dynamic tariffs**: This is a variation of time of use sometimes called real-time pricing (RTP). Price schedules vary more frequently, usually daily. Once delivered, the prices are firm and the customer, not the utility, controls the load.
- **Critical peak pricing (CPP)**: This is another variation on time of use, in which the utility retains the right to override the price schedule with higher prices on a limited number of days having particularly high demand or other unusual events.
- **Optimized charging**: The customer gives the utility control of the charging load in turn for a price incentive. The utility may, at critical times, reduce or interrupt charging, based in part on the state of charge of the vehicle.

E. Charging Location

The charging location may vary effecting potentially also the provided service and payment options:

- Charging in private environments like the vehicle owner's home or another's home within the same utility's service area or another's home within a different utility's service area. The charging location may not be directly connected with the charging infrastructure in terms of dynamic charging control. Hence, certain options for tariffs or value added services may not always be available.
- Charging at public charge spot can also be distinguished based on the contractual relation of the vehicle owner to the charging spot operator or mobility operator like: charging spot belonging to the same utility as customer contracted, different utility (comparable to "roaming") or charging without a contractual relationship (payment based on money, pre-paid card, credit card, etc.).
- Fleet operator premises may not require a contractual relationship per vehicle directly. They may be based on the fleet operator, providing an energy "flat rate". Control of the charging process may be distinguished as described above.

F. Value Added Services

Connecting the vehicle with a charging spot featuring a communication interface provides the opportunity to leverage this communication connection also for value added services. Examples comprise:

- Software updates for Engine Control Unit (ECU) or infotainment systems
- Remote diagnosis and maintenance
- Multimedia service during charging

G. Electricity Feedback

While in the first place charging is the main service provided for electric vehicles, it is also envisioned to use electric vehicles as dynamic energy storage. The electric vehicle could feed back energy into the Smart Grid upon request. Here, a distinction of the use cases can be done in a similar way as for charging:

- Based on the feedback locations, e.g., for integration within micro grids, to increase their independence from the main grid allowing the local usage of stored energy.
- Based on a local feedback plan, where the customer configures, e.g., a certain amount of energy, which is required as minimum capacity of the vehicle battery.
- Based on backend scheduling / needs.

These use cases show a variety of different services for the electric vehicle charging infrastructure. They illustrate how valuable the transmitted information is for the availability and reliable operation of the services, but also for the safety and privacy of the end user.

III. Information Assets, Potential Threats, and Derived Security Requirements

As just shown in the previous section, various use cases exist in which different peers exchange information to realize a dedicated service. Experience with the existing data communication infrastructure can be leveraged to analyze the charging infrastructure regarding potential threats as well as to determine suitable countermeasures. This may especially comprise security protocols or security mechanisms, which have been proven effective in the current communication infrastructures. Examples comprise security protocols like TLS (Transport Layer Security [4]) and digital signatures.

A. Information Assets in Charging-Related Communication

The information transported over the different connections is the asset that may motivate attacks against the charging infrastructure. The following table summarizes important information assets and their criticality for the system. The majority of these information assets are expected to be transmitted especially over the vehicle-to-grid interface.

TABLE I. INFORMATION ASSETS IN THE ELECTRIC VEHICLE CHARGING INFRASTRUCTURE

Information asset	Description, potential content	Security relation
Customer ID and location data	Customer name, vehicle identification number, charging location, and charging schedule	Affects customer privacy
Meter Data	Meter readings that allow calculation of the quantity of electricity consumed or supplied over a time period. These are generated by the charge spot and may be validated by the vehicle.	Affects system control and billing
Control Commands	Actions requested by one component of other components via control commands. These may also include inquiries, alarms, or Notifications.	Affects system stability and reliability and also safety
Configuration Data	Configuration data (system operational settings and security credentials, also thresholds for alarms, task schedules, policies, grouping information, etc.) influence the behavior of a component and may need to be updated remotely.	Affects system stability and reliability and also safety
Time, Clock Setting	Time is used in records sent to other entities. Phasor measurement directly relates to system control actions. Moreover, time is also needed to use tariff information optimally. It may also be used in certain security protocols.	Affects system control (stability and reliability and also safety) and billing
Access Control Policies	Determination whether a communication peer is entitled to send and receive commands and data. Such policies may consist of lists of permitted communication partners, their credentials, and their roles.	Affects system control system stability, reliability, and also safety
Firmware, Software, and Drivers	Software packages installed in components may be updated remotely. Updates may be provided by the utility (e.g., for charge spot firmware), the car manufacturer, or another OEM. Their correctness is critical for the system reliability.	Affects system stability and reliability and also safety
Tariff Data	Utilities or other energy providers may inform consumers of new or temporary tariffs as a basis for purchase decisions.	Affects customer privacy and competition

B. Potential Threats

Some example threats are described in the following to illustrate the need to integrate security measures into the charging infrastructure right from the beginning. The described threats focus on the specifics of electric vehicle charging and connected communication.

1) Eavesdropping / Interception

Eavesdropping is a passive attack to intercept information, which may compromise privacy or be used to gain more information for additional, active attacks. Eavesdropping requires the adversary to have either physical or logical access to the communication connection. Both the link to the vehicle and to the backend may be intercepted (Figure 3).

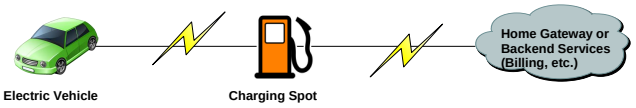


Figure 3. Potential Locations for Eavesdropping.

Communication with the charging spot in general can be done using different technologies, like Wireless or Powerline Communication (PLC). Common to these technologies is that the radiation of the communication transfer (through the frequency used) is high enough that it is sufficient for an adversary to be in closer vicinity to the communication instead of having direct physical access. Missing security measures will enable an adversary to eavesdrop the communication. As shown above, charging related communication may include a variety of information being valuable for an attacker like tariff information, charging status information, or billing relevant information.

2) Man-in-the-Middle Attack

An attacker may intercept communication on the interface between the vehicle and the charging point and modify this information. An example may be tariff options provided by the mobility operator and send via the charging spot to the vehicle. This may be accomplished in the easiest case through a modified charging cable.

Another example is the usage of a faked charge spot as depicted in Figure 4: A potential adversary may use its own (faked) charging spot to which honest customer connect. The adversary's charge spot is connected to an official charge spot and only routes the communication between the honest customer and the original charge spot. The adversary can then consume the charging energy partially, so that the honest customer receives only a fraction of her purchased energy, but pays for the complete consumption by her vehicle plus the adversary's vehicle.

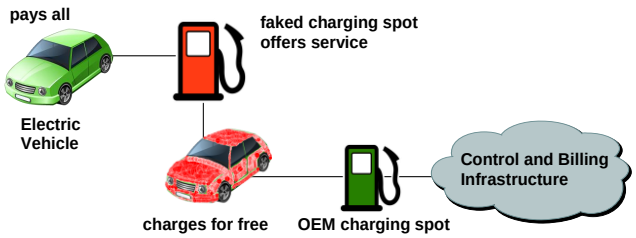


Figure 4. Man-in-the-Middle Attack to steal Energy.

Interesting in this attack is that the adversary actually performs the manipulation on the energy provisioning path and not on the communication path. The latter one is

untouched. This attack shows the need for connecting the flow of energy to the flow of information.

3) Transaction Falsifying or Repudiation

The customer himself may intentionally or unintentionally claim to have received less energy than stated on the billing record. Likewise, the utility may claim to have delivered more energy to the customer.

4) Attack network from within vehicle (and vice versa)

If the electric vehicle is connected to the charging infrastructure, e.g., using a value added service, an adversary (software) may inject or modify application-level traffic intentionally (as an attack) or unintentionally (faulty software component, malware).

5) Tampered or substituted component

A customer may manipulate a component trusted by the utility to provide accurate billing or control information. This affects both components in the charging spot and within the electric vehicle. Examples are pirated or faked replacement parts.

C. First Set of Security Requirements

Basic security requirements of the electric vehicle charging infrastructure have to be addressed. They target the availability and reliable energy provisioning. Moreover, they aim to limit attack effect (geographical and functional), enforce authorized control actions on the smart grid, and correct billing of energy transactions between involved peers (customer, charging spot operator, market, utility).

Based on the stated information assets and depicted threats, the basic security requirements can be addressed more specifically by requiring dedicated cryptographic measures as there are:

- Mutual authentication of end-to-end communicating entities. The authentication may be performed on different layers of the OSI reference model, e.g., on transport layer and on application layer. This is especially useful, if the peer to authenticate against is either a local communication peer or a backend peer, depending on the online state of the charging spot. Hence, end-to-end authentication strongly relates to the related OSI layer and its terminating end points.
- Non-repudiation of billing and tariff information to ensure secure transactions and the connected payment process.
- Protected communication between the electric vehicle and the charging spot, the electric vehicle and backend services, the charging spot and backend services, between backend services.
- Privacy preserving communication between the electric vehicle, the charging spot, and the backend
- Authorization, especially for control of the charging.
- Integrity-protected, authenticated and authorized software updates to avoid malfunctions through software from unauthorized sources
- Logging of security relevant events to enable auditability of the system.
- Security failure and exception handling, to support system reliability, also in case of security breaches.
- In general confidentiality and integrity of sensitive data.
- Support of a secured key management to support all of the requirements above.

These security requirements typically lead to technical and organizational security measures. Hence, to ensure a thorough security approach supporting the interaction of different peers using equipment from different vendors, standardization of an appropriate security approach as part of the overall system approach is necessary.

IV. STANDARDIZATION LANDSCAPE FOR THE CONNECTION TO THE CHARGING INFRASTRUCTURE

This section details the standardization activities focusing on the communication interface between the electric vehicle and the charging spot, but further connections to the backend are also considered. The main focus is placed on standardization activities from the ISO/IEC. An overview about related SAE activities is given as well.

As shown in Figure 5, standardization activities of ISO/IEC and SAE can be divided into four categories: charging connector, charging communication, charging topology, and safety. The following table summarizes more information about relevant standards.

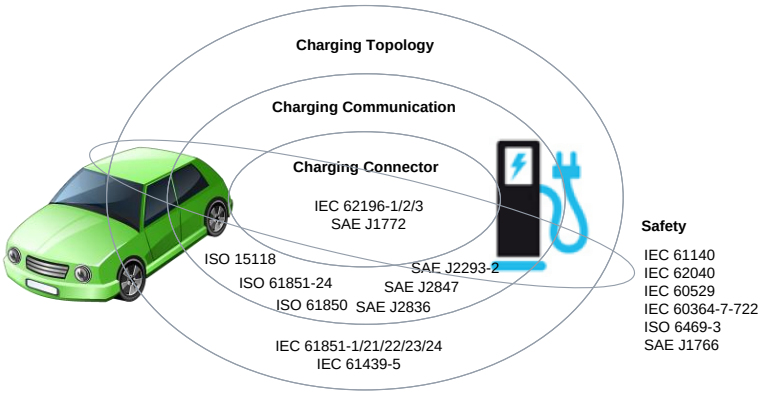


Figure 5. Communication Standards for the Electric Vehicle Charging Infrastructure [1].

TABLE II. COMMUNICATION STANDARDS AND THEIR SCOPE FOR THE ELECTRIC VEHICLE CHARGING INFRASTRUCTURE

Standard	Scope	Content
IEC 62196	Charging Connector	Plugs, socket-outlets, vehicle couplers and vehicle inlets – Conductive charging
SAE J1772	Charging Connector	Electric Vehicle Conductive Charge Coupler
ISO 15118	Charging Communication	Road vehicles - Communication protocol between electric vehicle and grid
SAE J2293	Charging Communication	Energy Transfer System for Electric Vehicles
SAE J2836	Charging Communication	Use Cases for Communication between Plug-in Vehicles and the Utility Grid (-1), Supply Equipment (EVSE) (-2), Utility Grid for Reverse Power Flow (-3)
SAE 2847	Charging Communication	Communication between Plug-in Vehicles and the Utility Grid (-1), Supply Equipment (EVSE) (-2), Utility Grid for Reverse Power Flow (-3)
IEC 61850	Power Systems Communication	Communication networks and systems in substations
IEC 61851	Charging Topology	Electric vehicle conductive charging system
IEC 61439	Charging Topology	Low-voltage switchgear and control gear assemblies

The following sections describe ISO/IEC activities related to charging communication and their IT-security considerations. This overview shows the increasing consideration of IT security requirements in the definition of evolving charging communication protocols. This is especially the case for new protocols like ISO/IEC 15118 targeting the communication for charging control and value added services between electric vehicles and charging spots.

A. *Simple Communication EV/EVSE – IEC 61851*

IEC 61851 (cf. [12][11]) defines a conductive charging system and was standardized in 2001. The standard addresses equipment for charging electric road vehicles at standard AC supply voltages (as per IEC 60038) up to 690 V and at DC voltages up to 1000 V, and for providing electrical power for any additional services on the vehicle if required when connected to the supply network. The standard comprises different parts addressing specific charging options:

- IEC 61851-1: Electric vehicle conductive charging system – General requirements
- IEC 61851-21: Electric vehicle conductive charging system - Electric vehicle requirements for conductive connection to an A.C./D.C. supply
- IEC 61851-22: Electric vehicle conductive charging system - A.C. electric vehicle charging station
- IEC 61851-23: Electric vehicle conductive charging system - D.C electric vehicle charging station
- IEC 61851-24 Electric vehicle conductive charging system - Control communication protocol between off-board D.C. charger and electric vehicle

IEC 61851 targets four different charging modes:

- Mode 1 (AC): slow charging from a standard household-type socket-outlet
- Mode 2 (AC): slow charging from a standard household-type socket-outlet with in-cable protection device
- Mode 3 (AC): slow or fast charging using a specific EV socket-outlet and plug with control and protection function permanently installed
- Mode 4 (DC): fast charging using an external charger

The communication between the vehicle and the charging spot depends on the mode applied. There is no data communication in Mode 1 and Mode 2. In Mode 3 only the control pilot communication exists, while in Mode 4 additional communication functions are available to allow battery management. Common to all modes is that IT-security is not provided. Therefore, there is no protection against any threats discussed in section III.B. Nevertheless, for the vehicle integration into a smart-grid-connected charging infrastructure, (secure) communication is required for tariff exchange, billing, optimization of charge cost and grid load, value added services, etc. To support these functions in the future, ISO/IEC 15118 is currently being specified addressing these communications needs, including an integrated security concept (see next section).

B. *Enhanced Communication EV/EVSE – ISO/IEC 15118*

ISO/IEC 15118 is being standardized in an ISO/IEC joint working group. Its main focus is the interface between an electric vehicle and a charging spot interface. Communication with the backend infrastructure is not directly targeted. The specification is split into different parts, which are all still work in progress:

- ISO 15118-1: General information and use-case definition [6]
- ISO 15118-2: Technical protocol description and Open Systems Interconnections (OSI) layer requirements [7]
- ISO 15118-3: Physical layer and Data Link layer requirements [8]

Security is integral part of the standard and has been considered right from the beginning of the design phase. ISO/IEC 15118-1 contains a security analysis, which investigates in specific threats, which are partly stated in section III above. This security analysis is the base for the security requirements and resulting security measures targeting the specified use cases.

The security measures defined in ISO/IEC 15118-2 build upon existing standards as far as possible. The access media for AC and DC charging will be power line communication in the first step. Support of inductive charging will most likely use wireless communication. As both feature different OSI layer 1 and 2, security measures have been placed on higher layers, to allow an independent solution. Besides the AC and DC profiles, charging options also exists regarding the authentication means. In general, authentication can be performed at the charging spot (External Authentication Means – EAM) or from within the car (plug&charge, or PnC).

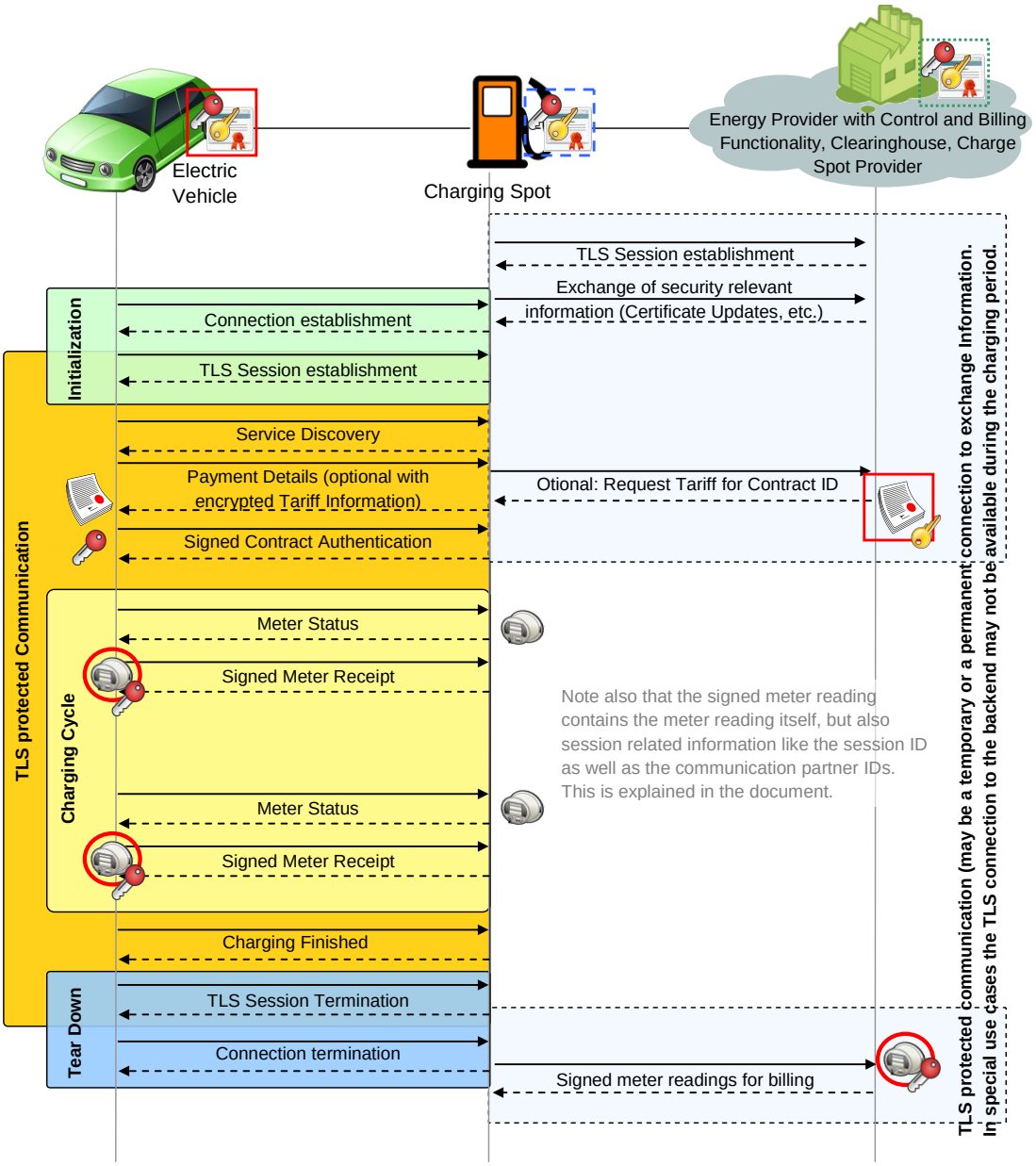


Figure 6. Information Exchange for Electric Vehicle Charging.

While in the first case the user typically may pay directly at the charging spot using either coins or credit cards. Alternatively, authentication can also be done using Near Field Communication (NFC), e.g., an RFID tag (Radio Frequency IDentification) or a mobile phone featuring a NFC interface. In case of PnC, the EV features a security credential allowing it to authenticate itself. While this security credential is typically applied to authenticate towards the charging infrastructure, it may also be used to identify stolen vehicles while charging.

As shown in Figure 6, ISO/IEC 15118-2 applies TCP/IP for the communication between the vehicle and the charging spot. Consequently, security is applied on transport layer using TLS (cf. [4]) ensuring a protected channel between both. Since ISO/IEC 15118 targets the communication between the vehicle and the charging spot, this might be sufficient at the first glimpse. But security measures on application layer have also been defined applying XML security (digital signatures and encryption).

Application layer security became necessary, as the communication also targets billing and payment relevant information, which are exchanged with the backend in contract based payment scenarios. Moreover, to enable contract based payments, the vehicles need authentication means.

To enable secure communication with the backend, the electric vehicle possesses a digital vehicle certificate and a corresponding private key. Here, X.509 certificates [9] are being applied. These security measures go beyond the communication hop between the electric vehicle and the charging spot. The direct data interaction of the electric vehicle with the backend is shown in Figure 6 in the charging cycle loop. Here, charging spot meter readings are signed by the vehicle and forwarded by the charging spot to the backend. They build the base for the billing process later on. Note that the general data exchange in Figure 6 has been simplified and mainly security related exchanges are shown.

The proposed security solution takes the connection state of a charging spot into account to support charging spots that have very limited or even no online connectivity. In general, the charging spot is assumed to be online at least once a day. This online period may coincide with the charging period of an electric vehicle. Therefore, explicit precautions have to be given to the exchanged data, especially, if the backend depends on these.

To enable secure transmission of data from the backend to the vehicle (e.g., updates of credential or of tariff information), a secret needs to be established between the vehicle and the backend allowing an end-to-end encrypted transfer. The vehicle certificate is an ECDSA certificate, where the public key can be considered as static Diffie-Hellman parameters to enable an easy setup of a session based encryption key with a communication peer. Only the backend needs to generate fresh per-session Diffie-Hellman parameters that are used to calculate a fresh Diffie-Hellman secret, which can then be used as session secret. This has the advantage, that the backend can pre-calculate session keys for vehicle communication, once the vehicle's certificate is known at the backend. This approach is known from many of today's web server applications, which use the same technique.

For the normal operation the vehicle certificate will be a contract-based credential. Thus the backend already possesses the certificate information, once the customer enrolled for a contract. For setup operation, the vehicle may possess an OEM credential installed during manufacturing of the car and used for bootstrapping the contact based credential. Notably, the used security mechanisms target elliptic curve cryptography (ECC) for authentication (during key management phases) and for digital signatures. The digital signature standard ECDSA based on ECC provide comparable security to RSA but uses significantly shorter cryptographic key sizes. As the certificates support ECDSA, the Diffie-Hellman key agreement is performed in its elliptic curve variant ECDH. Moreover, elliptic curves can be implemented efficiently in hardware. As ISO/IEC 15118 targets especially electronic control units (ECU) in vehicles and charging sports, memory and calculation constraints are evident and pose further implementation requirements.

The call flow as depicted in Figure 6 is based on the application of unilaterally authenticated TLS, where the electric vehicle implements the client part. Hence, the client is required to check the certificate validity including the issuer. The standard ISO/IEC 15118 requires vehicles to store only a fixed, limited number of root certificates to enable issuer verification. Moreover, it also restricts the number of supported intermediate certification authorities. Besides the validity and issuer, the client also needs to check the certificate revocation status.

One option to avoid the handling of certificate revocation lists is the usage of short term certificates from the server side. Another option is the provisioning of the revocation state by the server itself, e.g., by attaching a fresh Online Certificate Status Protocol (OCSP) response to the certificate during the authentication phase. To keep a balance regarding the implementation and operational effort, the current ISO/IEC 15118 proposal features both, short term certificates for the server side certificates and OCSP responses for intermediate CAs.

As said before, all of the security functionality in ISO/IEC 15118 builds on X.509 certificates and corresponding private keys. Hence, an infrastructure is necessary to manage this key material. It has to be noted, that there are different trust relations for the application and utilization of the key material.

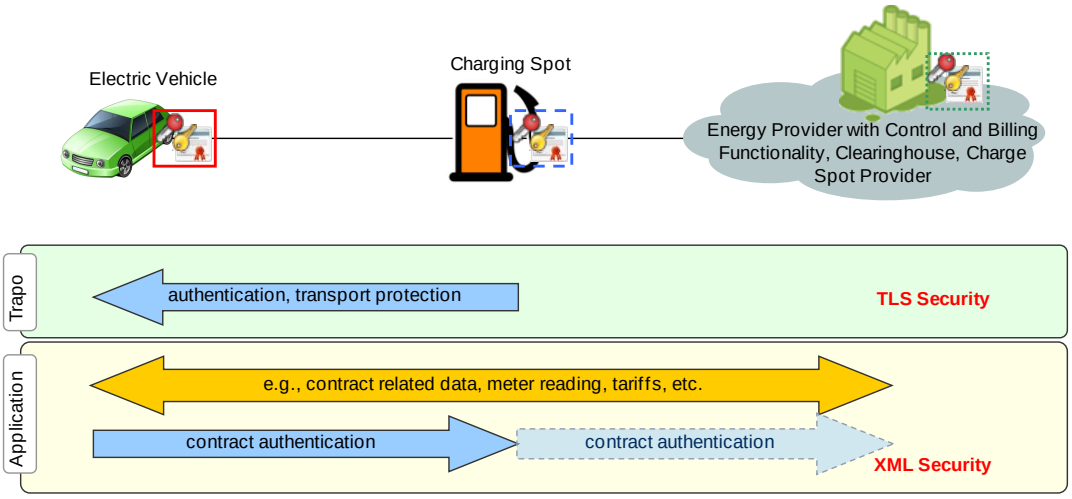


Figure 7. Information Exchange for Electric Vehicle Charging.

As shown in Figure 7 for the transport connection the trust relation exists between the electric vehicle and the charging sport. On application layer there are some messages, which are bound to the communication between the same peers. This applies for instances to the acknowledgement of cyclic meter readings through the electric vehicle by applying a digital signature.

Nevertheless, to get this security on an operational level, electric vehicle and charging spot, both have to get at least the X.509 certificates from a 3rd party. At least, as the generation of public key pairs may be either directly at the component or at the 3rd party. The 3rd party for issuing the certificates may be different. While the electric vehicle will get its contract certificates from a mobility operator, the charging spot will be equipped with a certificate also from potentially another mobility operator. Having different mobility operators relates to the typical situation of having different energy providers depending on the geographic area.

Figure 8 provides an overview of the certificates used by the different actors. Note that this figure reflects the current draft status of ISO/IEC 15118-2. Especially the certification path of the contract certificate may allow also other root certificates as the V2G Root CA in the future. On the vehicle site, the OEM is expected to provide an initial certificate during manufacturing. This certificate is used to enable the secure bootstrapping of operational credentials through the mobility operator. The mobility operator will issue contract based certificates, if the electric vehicle is going to participate in plug&charge scenarios, which allow the payment directly out of the vehicle, without additional identification and authentication at the charging spot. On the infrastructure side, the charging spot needs to possess a certificate and a corresponding private key. The certificate is also issued by the mobility operator, which is not necessarily the same as for the electric vehicle (the mobility operator issuing the contract certificates may be different in roaming scenarios). As the charging spot may be offline during charging and the electric vehicle may not have another communication path to the backend, certificate revocation needs to be addressed in some way. The one depicted in Figure 8 uses short term certificates for the charging spot. Another option is the utilization of multiple OCSP stapling. This approach avoids the handling of short term certificates as an OCSP for both, the charging spot certificated and the issuing sub certification authority certificate can be transmitted to the vehicle.

As described above, digital certificates for the charging spot, and, depending on the use case, also for the electric vehicle, are the basis for protecting the charging control communication. Common to all components for charging control is that the certification path of the certificates applied has a common set of (at east) five root certificates. Five root certificates have been agreed on to address the memory restrictions within an electric vehicle. To enable a smooth operation a dedicated credential management infrastructure (Public Key Infrastructure – PKI, cf. also [9]) handling the initial provisioning, but also the revocation and update of certificates and cryptographic keys is required.

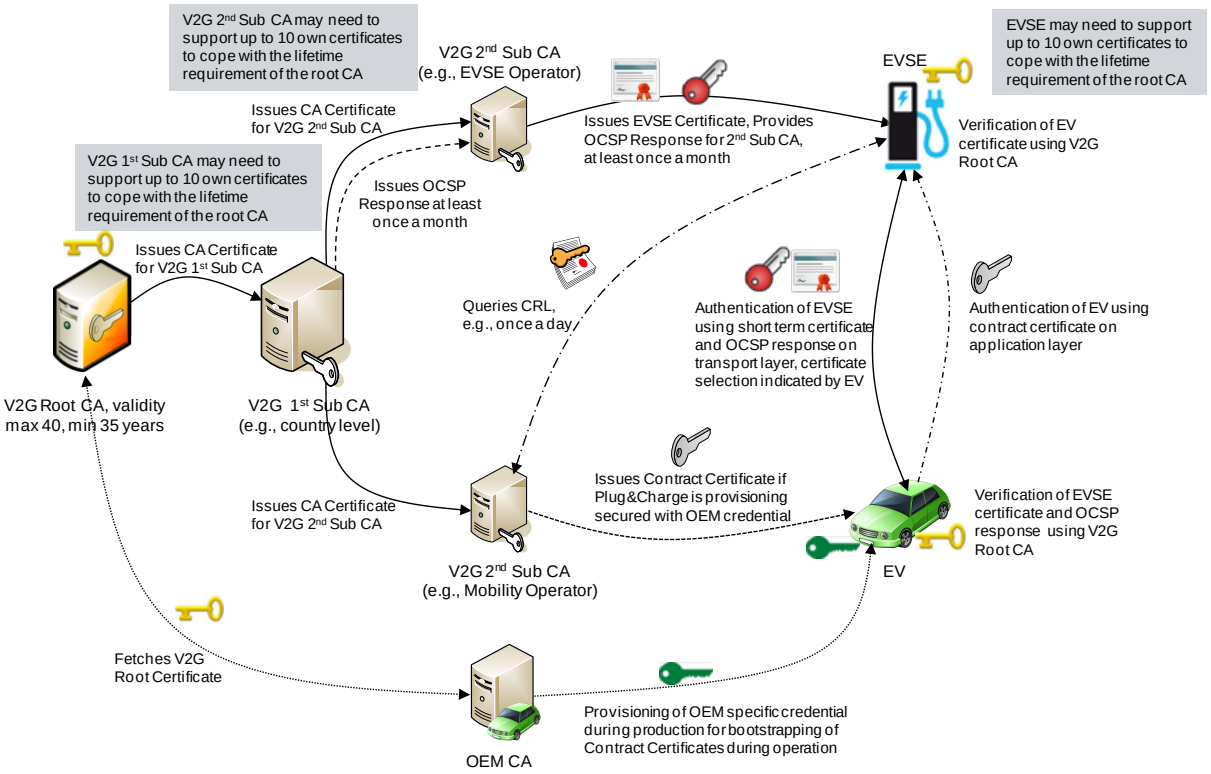


Figure 8. Credential Handling according to current state of ISO/IEC 15118 (DIS), cf. [7].

As ISO/IEC 15118 is in the process of getting finalized, it is expected that the application of certificates will be further optimized to address security on one hand and operation and maintainability on the other.

With the proposed mechanisms ISO/IEC 15118 addresses most of the threats depicted in section III B, with the focus of the interface between EV and EVSE. What is not addressed is the detection tampered of falsified components, which would support the system integrity monitoring. Also, authentication of the EV is only performed in the PnC use cases, which still leaves some possibilities for attacks from rogue EVs.

V. CONCLUSION

The focus of this paper has been the discussion of security requirements and solution approaches for the interface between an electric vehicle and a charging spot. Especially the standard ISO/IEC 15118 was in focus here addressing a variety of use cases while considering security right from the beginning. Nevertheless, to enable online control of the charging operation and also value added services, at least the charging spot needs to be connected to the Smart Grid core.

One standard, which can be directly applied for the energy automation communication is IEC 61850 [10], already applied in substation automation. This communication can be protected by security measures according to IEC62351 [11]. The security in IEC 62351 features similar protection means for TCP/IP based communication which are based on TLS as well. This eases the secure interworking between the Smart Grid communication core and the access via the charging infrastructure. All of these standards employ X.509 certificates. Thus, the key management as enabling functionality becomes a crucial point. The operational handling of an infrastructure providing and revocation information to a multitude of components can be seen as challenge here.

Another communication protocol to be named in this context is OCPP (Open Charge Point Protocol, cf. [14]), which can be used as a protocol between the charging points and the management station. This protocol uses TCP/IP for communication and XML for encoding of messages. Hence, existing security mechanism like TLS and XML security, which are also being employed to protect ISO/IEC15118 as described above, can be utilized here too.

Besides pure charging control, there may be also value-added services provided through the charging spot like multimedia services, software or firmware updates, remote diagnosis, and so on. All of these services have to be protected appropriately. The intrinsic complexity of this overall Smart Grid vehicle charging system requires a systematic approach to include required security measures right from the beginning that can be used and managed efficiently. It is expected that new use cases will enhance the existing security requirements and also influence the further development of communication standards.

VI. ACKNOWLEDGEMENT

The base version of this report (see [13]) compiled in June 2011 has been supported by the German Federal Ministry for the Environment, Nature Conservation and Nuclear Safety within the Harz.EEmobility project under contract 03KP623 (see [3] for more information). The further research and investigation leading to this update of the initial report is part of the FINSENY (Future INternet for Smart ENergY) project (see [4] for more information). The authors gratefully acknowledge the contributions of all FINSENY project partners. FINSENY is partly funded by the European Commission within the FI-PPP, which is part of the Framework Program FP7 ICT.

REFERENCES

[1] R.Falk and S.Fries, “Electric Vehicle Charging Infrastructure – Security Considerations and Approaches”, Internet 2012, June 2012, ISBN: 978-1-61208-204-2, pp.58-64

[2] The German Standardization Roadmap for Electromobility, http://www.elektromobilitaet.din.de/sixcms_upload/media/3310/Normung-Roadmap_Elektromobilit%E4t_en.pdf, last access April 2012

[3] HarzEE-mobility, https://www.harzee-mobility.de/, last access February 2013

[4] FINSENY – Future Internet for Smart Energy: http://www.fi-ppp-finseny.eu/, last access February 2013

[5] T. Dierks and E. Rescorla: “The Transport Layer Security (TLS) Protocol Version 1.2”, RFC5246, IETF, 2008.

[6] ISO/IEC 15118-1: Road vehicles — Vehicle-to-Grid Communication Interface — Part 1: General information and use-case definition, Work in Progress

[7] ISO/IEC 15118-2: Road vehicles — Vehicle-to-Grid Communication Interface — Part 2: Technical protocol description and Open Systems Interconnections (OSI) layer requirements, Work in Progress

[8] ISO/IEC 15118-3: Road vehicles — Vehicle-to-Grid Communication Interface — Part 3: Physical layer and Data Link layer requirements, Work in Progress

[9] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, and W. Polk: “Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile”, RFC 5280, May 2008

[10] ISO-IEC 61850, Part 1-9, http://www.iec.ch/cgi-bin/procgi.pl/www/iecwww.p?wwwlang=E&wwwprog=sea22.p&search=iecnumber&header=IEC&pubno=61850, last access February 2013

[11] ISO-IEC 62351, Part 1-8, http://www.iec.ch/cgi-bin/procgi.pl/www/iecwww.p?wwwlang=E&wwwprog=sea22.p&search=iecnumber&header=IEC&pubno=62351, last access February 2013

[12] IEC 61851, Part 1, www.iec.ch/cgi-bin/procgi.pl/www/iecwww.p?wwwlang=E&wwwprog=sea22.p&search=iecnumber&header=IEC&pubno=61851, last access February 2013

[13] R. Falk and S. Fries: Securing the Electric Vehicle Charging Infrastructure – Current status and potential next steps, Oct 2011, Berlin, VDI-Berichte 2131, VDI-Verlag Düsseldorf. ISBN 978-3-18-092131-0.

[14] OCPP – Open Charge Point Protocol, http://www.ocpp.nl, last access February 2013

Entity Ranking as a Search Engine Front-End

Alexandros Komninos
Department of Computer Science
University of York
York, YO10 5GH, UK
ak1153@york.ac.uk

Avi Arampatzis
Department of Electrical and Computer Engineering
Democritus University of Thrace
Xanthi 67100, Greece
avi@ee.duth.gr

Abstract — In this paper, we present a Web application for entity ranking. The application accepts as input a query in natural language and outputs a list of the most relevant entities according to the query. The system uses Web documents as data and performs extraction, formatting and ranking of entities in real time. An experiment is conducted to determine the most efficient ranking method among eleven alternatives. The experiment suggests that the total frequency of an entity in a retrieved set of documents has less to say on the entity's relevance than the number of retrieved documents it occurs in. Furthermore, for small retrieved sets such as the top-10, document rank information seems to play a little role. Four algorithms are tested for estimating the correct amount of results in the ranked list and provide a threshold. The best results are achieved by the maximum entropy algorithm applied to the distribution of scores provided by a multiplicative combination of logarithmic entity frequency and document frequency.

Keywords - Web entity ranking, entity search, information retrieval, threshold optimization.

I. INTRODUCTION

Search engines answer user queries by returning ordered lists of documents. In many occasions though, users are not searching for documents, but for some more specific information, such as "German female politicians" or "Swiss Cantons where they speak French". In this type of queries, users are looking for an answer consisting of semantically important units called *named entities*. The term named entity is used for anything that has a distinct existence and can be characterized by a name, so it can refer to people, companies, products, etc. The need for retrieving named entities as query answers has led to research for systems that can identify and return this type of information instead of whole documents.

Entity ranking is the process of finding and sorting entities according to their relevance to an information need. The difference from document retrieval is that it gives direct answers to a user's query; therefore, it is an approach for data oriented search. Another important difference is that results can come from combining information from multiple sources instead of a single one.

Different methods have been used, either separately or in combination, for the purpose of entity ranking. Most of them come from the fields of information retrieval (IR) and natural language processing, especially information extraction. The problem has been also studied from the perspective of Semantic Web technologies. Each approach offers some distinct advantages. Natural language processing techniques can capture complex relations between entities but with a computational cost that is often difficult to scale up to the volume of Web data. Semantic Web methods rely on ontologies that can also describe complex relations, but are limited to a predefined number of them. For the purpose of entity ranking on the Web, an IR approach is being presented in this paper. IR methods have been proven by Web search engines to be effective in dealing with large volumes of data and the heterogeneity of information found on the Web. They also allow free text querying and can provide fresh information that is found in Web documents.

In [1], we presented an application for entity ranking called ListCreator. Six ranking methods were formulated and their performance was evaluated. In this extended version, we perform a systematic and exhaustive analysis of possible ranking methods based on the same hypotheses and statistical measures. Therefore, we evaluate eleven ranking formulae that take into account all measure combinations, excluding ranking-wise equivalent ones. Furthermore, we address a serious limitation of this approach and ranking methods for retrieval in general. While using an effective ranking method, we expect the relevant results to be ranked higher than the non-relevant ones, but there is no further indication for how many are correct. This poses a serious problem for entity ranking when increasing the number of source documents. The ranking may get better, but the number of incorrect results on the lower part of the list also increases. A way to mitigate this problem is estimating a threshold in the results list, from which point on, the relevancy is rapidly degrading. We investigate solutions to this problem in Section V.

ListCreator can answer user queries for entities of the three major categories: persons, locations, and organizations. The application uses a search engine to obtain a small collection of Web documents that are related to the submitted query. The entities found in the documents are extracted using a named entity recogniser. The ranking is achieved by statistical information retrieval methods, taking advantage of the common information among the source

documents. The results are returned to the user as a ranked list of all the relevant entities that the application managed to extract.

The above ranking method is based on two assumptions. First, that a Web search engine will be able to retrieve documents that contain the relevant entities. The connection between the query and the entities will take place using document retrieval methods. Second, given that the Web is a collection of documents from independent authors, the desired information or some part of it will be found on several different documents. In order to obtain a ranking of entities according to relevance to the query, we find how important each entity is for the retrieved collection. The work on this paper mainly addresses the problem of measuring this importance with a statistical model.

The system relies on the technologies of Web search and named entity recognition for acquiring data in order to perform the ranking. Therefore, it can be used as a front-end to a commercial Web search engine utilizing its state-of-the-art search functionality. Our motivation is to build an entity ranking system that can use effectively the information in Web documents, and can produce results without relying on external sources. An online demo of the application can be found in [2].

The contribution of this paper is threefold. First, we build an online prototype as proof-of-concept for entity ranking as a search engine front-end, using IR methods. Such methods are simple and fast, and therefore suited for an online application, also scaling well to large amounts of data. Second, we formulate and experimentally evaluate several ranking methods that can be used in the particular system. Third, we evaluate the performance of four algorithms for threshold estimation.

The rest of this paper is organized as follows. In Section II, we review related work. In Section III, we give a detailed description of ListCreator's methods and architecture. In Section IV, we describe the different methods for ranking entities and perform experiments to compare their effectiveness. The experiments for threshold estimation are presented in Section V, followed by a discussion in Section VI. Conclusions are drawn in Section VII, together with directions for further research and improvements.

II. RELATED WORK

Entity ranking has a lot in common with automatic question answering, where the answer to a query is often a name or a list of names. Research for question answering systems took place during the TREC (Text REtrieval Conference) QA track. A method used for extracting answers from raw text is checking document snippets that are relevant to the query, and counting the frequency of each possible answer [3][4]. As the frequency of a candidate answer gets higher, so does the probability of it being the correct one. This approach is similar to ListCreator's, with the difference being that it is not a model build to produce a ranking, but only focuses on the top result, in order to provide a single answer.

Another related task to entity ranking is expert finding. For this task, a system has to automatically find an expert that meets the criteria determined in a user's query, so it is an entity retrieval problem limited to the person category. In [5], two models for expert finding were formalized. In Standard Model 1, candidate experts are described with representative documents and document retrieval methods are used to obtain the relevance of an expert according to a query. Standard Model 2 uses documents relevant to the query as latent variables for calculating the desired relevance of experts to queries. The documents are retrieved using standard document retrieval methods and each expert is assigned a total score that corresponds to the sum of the scores of all the documents that his name appears in. In [6], a combination of the two methods is proposed, creating profiles that incorporate parts of many different documents according to probability distributions.

IR methods for the purpose of entity ranking were demonstrated and evaluated during the INEX (INitiative for the Evaluation of XML retrieval) and TREC entity ranking tracks. INEX evaluated the performance of many systems from 2007 to 2009, for entity ranking in Wikipedia in two tasks [7][8]. For the entity ranking task, the requirement was retrieving entities that satisfy a topic described in natural language, while for the list completion task the objective was creating a list of entities given some examples and a description. For these tasks, an entity is anything that has a Wikipedia article dedicated to it. Participating teams used Wikipedia's semi structured format, specifically the categories and the links between articles for determining entity relations, and the infoboxes for retrieving information in a machine readable way.

TREC evaluated systems for entity ranking in the Web from 2009 to 2011 [9][10]. A name was considered to correspond to an entity, if it had its own Webpage. TREC runs a related entity search task, where the goal was to retrieve relevant entities that satisfy conditions related to another entity, and a list completion task similar to INEX. Typical approaches used the given entity to determine relation with candidate entities through co-occurrence frequency and link analysis. In [11], the structure of HTML is used to find entities in lists and tables assuming that entities found in the same format will also belong to same category, along with specific templates and filtering rules. In [12], a profile document is constructed from different parts of the corpus that mention a candidate entity, and then document retrieval is used for ranking. In [13], a document language model for estimating the probability of generating an entity from a query, a supervised and an unsupervised learning to rank approaches using SVMs are tested. In [14], Wikipedia was used as an information source for Web entity ranking, providing descriptive documents, category and link information.

A typical feature used for entity ranking from document sources is proximity measures of candidate entities and query terms. Proximity measures estimate the relevance of an entity to a query by taking into account the quantity and distance of query terms to an entity in a predefined window, aggregated over many documents. In [15], a model for ranking with proximity measures is built using non-uniform kernel functions, while in [16] and [17] proximity measures

are enhanced by patterns that consider the order of the keywords. In [18], the proximity, profile and voting methods were integrated in a single probabilistic model using a Markov Random Field.

A different approach to entity ranking is using information extraction techniques to construct structured data from text by extracting facts about entities [19][20]. This requires natural language processing, for example part of speech tagging, and is typically achieved by machine learning methods. Since applying machine learning to large volumes of text has great computational cost, the above systems constructed a database of relations between entities offline. The database is then queried for relevant entities by the user at runtime. An alternative is using data sets of existing ontologies constructed either manually or automatically using information extraction to obtain RDF data like in [21]. The database method adopts a data retrieval approach for entity ranking, where the system accepts structured queries in a query language like SPARQL or more sophisticated extensions like [22], instead of imprecise free text queries. Recent research addressing the problem of transforming keyword queries to structured ones can be found in [23] and [24].

Entity retrieval by keyword queries from datasets of ontologies was the subject of the Semantic Search Challenge (2010, 2011) [25][26] and the JIWES 2012 (Joint International Workshop on Entity-oriented and Semantic search) [27], where a related entity finding and a list search task similar to TREC took place. The objective was to rank entities belonging to the Linked Open Data according to a free text query. The participants used IR methods specifically modified for retrieving RDF data. A model for efficient usage of the structured information presented in a knowledge base is explored in [28]. Approaches that combine IR models for keyword search and the structured information presented in knowledge bases to return a ranking according to relevance can be found in [29] and [30].

The database and ontology approaches have currently certain limitations compared to the IR methods that deal directly with unstructured data (text). The relations and attributes defined in an ontology are limited in comparison to the relations found in documents and can be interesting to a user. Furthermore, a system relying in a database can only accept structured queries that impose restrictions to the user. Finally, a lot of effort is required to keep a database up to date with recent information, something that an approach dealing directly with Web documents can easily achieve.

The idea of using statistical evidence from multiple sources to certify the correctness of results has been used in the similar tasks of question answering and expert finding. However, previous work does not investigate suitable ranking methods to better utilize evidence, instead they make an arbitrary choice of ranking method without considering alternatives, e.g., term frequency [3] or document score aggregation [5]. Research on the document retrieval task has shown that different ranking methods can play an important role in the quality of retrieval. In this paper, we investigate an optimal ranking method for the entity retrieval task by using a systematic approach. In order to formulate ranking methods, we make no assumptions about the underlying distribution of relevancy for entities. Instead, we begin with two generic hypotheses about use of language from Web document authors and construct several ranking algorithms for each one, using different combinations of statistical measures. We proceed to measure effectiveness based on empirical evaluation.

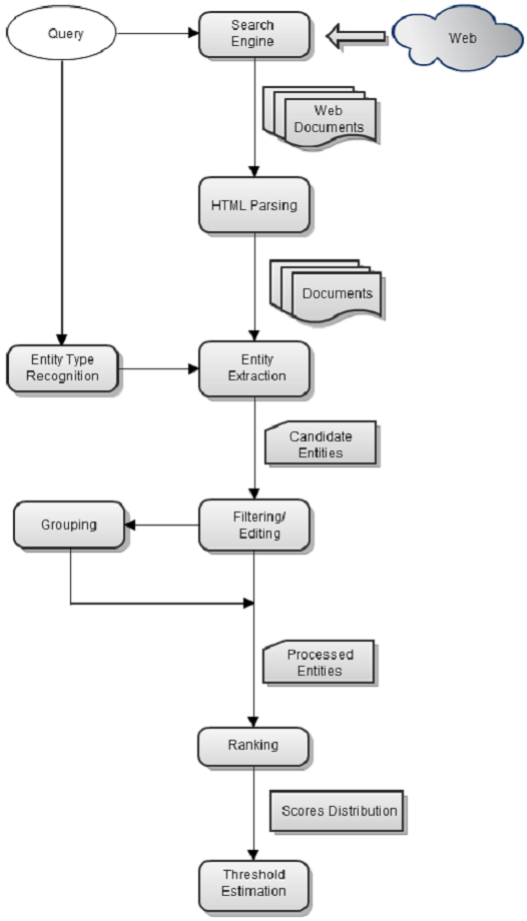


Figure 1. The system's components and dataflow.

III. SYSTEM DESCRIPTION

The data flow that takes place in the system is depicted in Figure 1. The components for formatting, filtering, grouping and ranking of entities are all coded in JAVA [31]. The user Web interface is coded in HTML [32], JavaScript [33], and PHP [34].

A. The Application Website

The central Webpage consists of an input form for the user's query and gives the option to determine the type of entity (person, location, organization) that he is searching

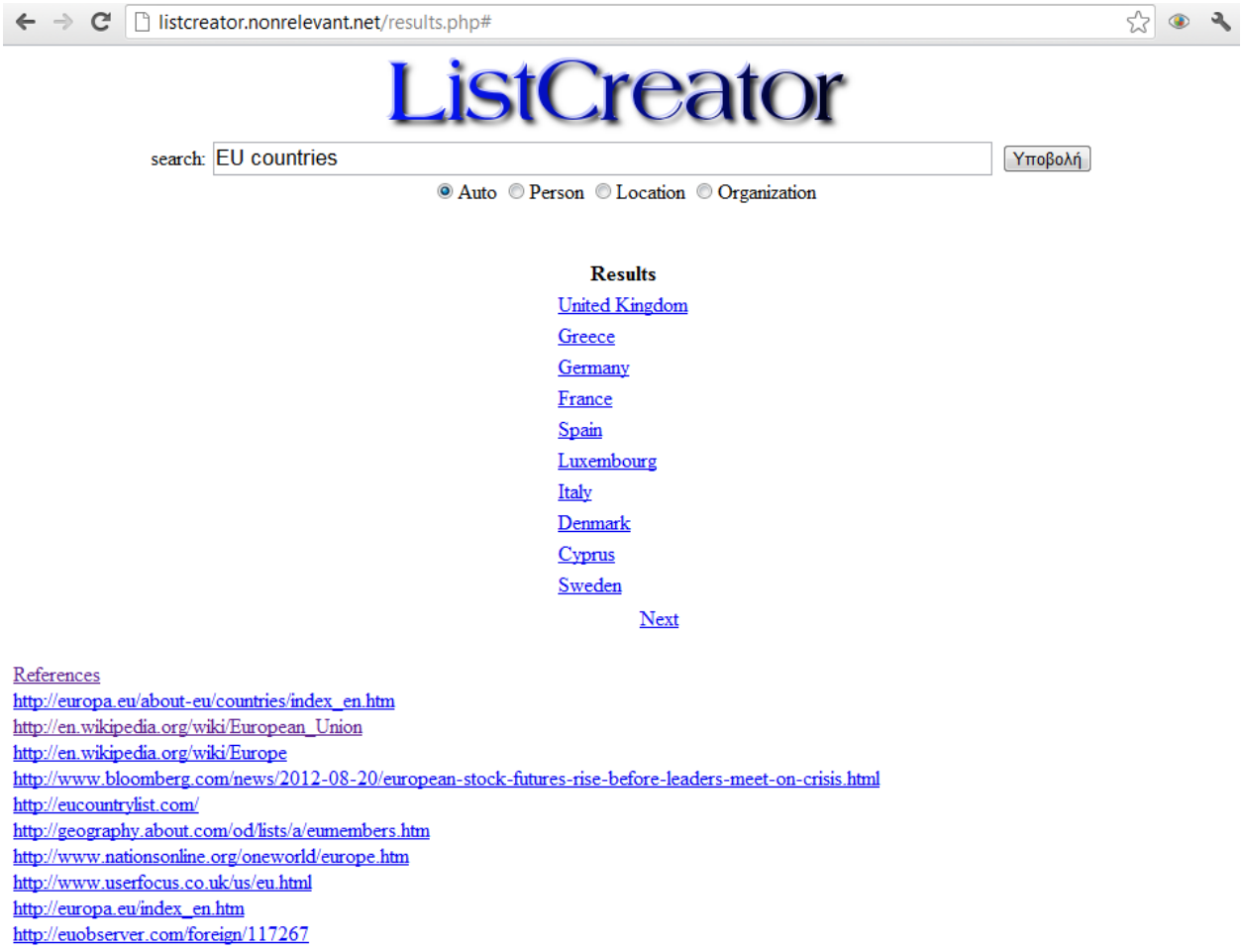


Figure 2. A results page of the application.

for. The default option is "auto", which corresponds to automatic recognition of the entity type.

The automatic recognition feature uses a list of about 100 keywords for the location type and about 50 keywords for the organization type. The collection of keywords is based on WordNet categories [35]. The system checks for the appearance of any of those keywords in the submitted query and if they exist it is assumes the user is searching for the corresponding entity type. If none of the keywords appear, the system assumes that the user is searching for persons.

The submission of a query calls the main application and the output is presented in the results Webpage with the use of PHP. Each result is linked to a corresponding Wikipedia page (if it exists), so that the user can get more information. The results Webpage also gives as references links to the Web documents where the entities were extracted from. A results page is presented in Figure 2.

B. *The Search Engine*

The search engine is a very important component of the system since it provides all the data in the form of documents for extracting and ranking the entities. The application essentially functions as a front-end in a search engine. In the current version, the search engine used is the Yahoo! BOSS API [36]. Google and Bing were also tested with similar results but Yahoo! was chosen because it combines good results with an easy to use API.

The user's query is sent to Yahoo! API without being changed and the results are returned in JSON (JavaScript Object Notation) format. The system asks for only the top-N results. Through some testing we empirically determined that N=10 retrieves enough information while, at the same time, keeps the computational cost low enough for a real time application.

C. *Entity Extraction*

In this stage, the system recognizes the entities in the documents and determines their type. For this purpose, the Stanford NER (Named Entity Recognizer) is used [37]. Stanford NER is a system for entity extraction from text coded in JAVA and distributed with GNU general public license [38] for research and education purposes. The entity recognition is done with a classifier, an algorithm that assigns words in specific categories. The categories supported by the classifier are person, location and organization.

Classification is a supervised machine learning task. The algorithm uses hand-annotated text to construct statistical

rules that can find and determine the category of names in documents. The Stanford NER classifier [39] is based on the CRF (Conditional Random Field) probabilistic model [40] and comes trained on American and British news articles. The classification process offers some very useful filtering of the entities. The usage of a NER system was considered more suitable for unknown data since it identifies entities by their context in documents, in contrast with a dictionary based approach. It is limited though in the three general entity categories.

In order to extract entities from a Web document, the HTML tags have to be removed. For the HTML parsing the JSOUP HTML Parser is used [41]. JSOUP is an open source parser also coded in JAVA that can handle html code with errors.

D. Formatting and Filtering

Each entity can appear in a document in many different ways. A person's name for example can first appear with its full name and later be referred with just the last name. In order to achieve a better ranking in the next stage, the system must recognize which names correspond to the same entity, a task called *coreference resolution*, and then assign to all of them the same canonical name. The results of this stage are also important for the final presentation since names should appear with all details and avoid listing the same names more than once. The processing of names comes in two steps. In the first step, each entry is formatted and in the second step the names referring to the same entity are grouped taking in consideration the whole set of extracted names.

The basic processing of the first step is converting the names to proper case, i.e., converting the first letter in uppercase and the rest in lowercase. For organization names with less than four letters, all of them are converted to uppercase. Furthermore, the candidate entities are filtered using an exception list. The exception list consists of about 20 entries that correspond to certain names that are often misclassified by Stanford NER. These names are popular Websites (Wikipedia, Facebook, Twitter, etc.) that are falsely classified as locations and some acronyms like FAQ, ISBN that are classified as organization. Using this exception list the results from the extraction stage are improved. Another exception list used contains all the country names. This list is checked for search of location type entities because country names appear in large numbers in documents about locations and they can have negative influence on ranking. The list is not used when the user is searching for country names. The effect of this method is splitting the location category into two, countries and other locations, providing a better filtering. The described exception lists are used solely for the purpose of improving the entity recognition task. Since Stanford NER is not trained on Web documents, its accuracy is lower when dealing with them. Some common mistakes are handled by the first list we accumulated. An alternative method would be to retrain the classifier on Web documents. The location entity category is too broad, and an approach for obtaining finer grained entities is splitting it into a geopolitical entities category and other locations. By using a list of known countries we take a step towards that direction.

The grouping of entities that happens in the second step is rule-based and is achieved by comparing each entry with all others. The system checks if an entry forms part of another in word level, and then it is substituted by its complete name. For example, the entries John Kennedy, Kennedy, John F. Kennedy and John Fitzgerald Kennedy are all grouped and substituted by the last form. In order to avoid grouping into names that may be misspelled, or into a concatenation of two names, the substitution takes place when an entry appears more than once. The grouping step is not applied for queries asking for names of countries, cities and organizations. Country and city names usually do not appear in different forms, while organization names have lots of variance to be grouped with simple rules that often lead to errors.

The above method of grouping gives good results and greatly improves performance, but in some cases the correct grouping of entries cannot be determined. Such is the case of two different candidate entities with the same last name and an entry containing this last name alone. A possible improvement could be the usage of a system that accomplishes coreference resolution utilizing machine learning, but such an approach would increase computational cost.

E. Entity Ranking

The ranking algorithm makes usage of statistical methods of IR. The input in this stage is 10 lists of candidate entities, each one corresponding to the names extracted from each document the search engine provides. The entities are then ranked according to the formula:

$$score = df \sum_{i=1}^{df} (N + 1 - r_i)$$

where *i* is the document an entity appears in, *df* is the number of the top-N documents that mention an entity, *N* is the total number of retrieved documents and in the current version is always equal to 10, *r* is the rank of the retrieved document according to the search engine and has a value from 1 to 10. The formula is based on the Borda Count preferential voting method, multiplied by the document frequency of the entity. According to the formula, an entity that appears only in the first document will get 10 points, if an entity appears on the first and second document, it will get 10 plus 9 points multiplied by 2, etc. Entities with higher score are considered more relevant to the query. This ranking formula was chosen after the experiment that will be described in the next section.

IV. EXPERIMENT

The proposed ranking method tries to solve a problem that resembles the reverse procedure of finding relevant documents to a query. Instead of searching for documents relevant to some terms, it utilizes a small collection of documents (10 in our case) with a common subject and

TABLE 1. SUMMARY OF RANKING EQUATIONS.

			Frequency Weighting			
			None	***Verbosity***	***in-between***	***scope***
			1	df	$\log(1+f_i)$	f_i
Document Frequency Weighting	***None***	1	no ranking	(1)	(3)	(2)
	Linear	df	(1)	equivalent to (1)	(4)	(6)
	Logarithmic	log(1+df)	equivalent to (1)	equivalent to (1)	(5)	(7)
	rank-based	Borda Count	(8)	(9)	(10)	(11)

searches for terms (in this case named entities) that are important for this collection. The quantities that were considered useful for the ranking according to the above line of thinking are:

- The total number of occurrences of each entity in each document (f_i). The higher the frequency of an entity, the more confidence we have in its importance for a particular document.
- Document frequency (*df*), which corresponds to the number of distinct documents where each entity occurs. This quantity shows the common information between documents. Assuming that all documents are equally relevant to the submitted query, the names that occur in most documents would also be the most relevant.
- The rank of documents that an entity appears in, according to the search engine (*r*). By taking into account this quantity the documents are no longer treated as equally relevant.

There are two opposite hypotheses regarding the frequency of a term and the importance that it has for a document [42]. According to the *verbosity hypothesis*, multiple occurrences of a term are not really important, because the document's author is more verbose: the author just used more words to express the same meaning. According to the *scope hypothesis* though, a document's author uses a specific term more times because he has more information to share on this subject.

Using the above statistical measures and hypotheses we formulate 11 ranking equations. The measures are used by itself and in multiplicative combination. We try linear and sublinear scoring, where for the latter case we use the logarithmic function as a damping factor. The logarithmic scoring for entity frequency gives an in-between approach for the two hypotheses. In all the following formulae, *i* is the document, *N* is the total number of documents and equals 10, f_i is the number of occurrences of an entity in document *i*, and *df* is document frequency.

A. Frequency-only scoring

Under the verbosity hypothesis an entity will not get extra credit for appearing more than once in a document so:

$$score = \sum_{i=1}^{df} 1 = df \quad (1)$$

The scope hypothesis suggests that each entity appearance contributes linearly to relevance:

$$score = \sum_{i=1}^{df} f_i = f \quad (2)$$

The logarithmic approach corresponds to an in-between approach and gives:

$$score = \sum_{i=1}^{df} \log(1 + f_i) \quad (3)$$

This means that we get diminishing returns on entity occurrences, so only the first few of them can contribute significantly to the score.

B. Document Frequency-only scoring

For the document frequency scoring, both the linear (*df*) and logarithmic approach (logarithm of *df*) result in ranking that is equivalent to that of (1).

C. Combination of scoring measures

Combining the two measures multiplicatively and excluding combinations that give equivalent ranking to the above scoring equations we get:
In-between frequency weighting and linear *df*:

$$score = \sum_{i=1}^{df} \log(1 + f_i) df \quad (4)$$

In-between frequency weighting and logarithmic *df*:

$$score = \sum_{i=1}^{df} \log(1 + f_i) \log(1 + df) \quad (5)$$

Scope hypothesis frequency weighting and linear *df*:

$$score = f \times df \quad (6)$$

Scope hypothesis frequency weighting and logarithmic *df*:

TABLE 2. EVALUATION RESULTS FOR THE 11 RANKING FORMULAE AVERAGED OVER THE 30 QUERIES.

Ranking Equations	P@10	R-Precision
(1)	0.4733	0.4209
(2)	0.3900	0.3675
(3)	0.4400	0.4200
(4)	0.4700	0.4314
(5)	0.4500	0.4267
(6)	0.4367	0.4178
(7)	0.4333	0.4061
(8)	0.4900	0.4216
(9)	**0.4933**	0.4200
(10)	0.4766	**0.4463**
(11)	0.4100	0.4025

$$score = f \times \log(1 + df) \quad (7)$$

Multiplicative combinations between the verbosity hypothesis weighting for entity frequency and document frequency result in the same ranking as the one provided by (1).

D. Document Rank scoring

In previous equations, the documents were seen as equivalent. To weight each document according to its rank we use the Borda Count:

$$score = \sum_{i=1}^{df} (N + 1 - r_i) \quad (8)$$

Equation (8) is equivalent to the ranking method proposed in [5], with unknown scores for the document retrieval part.

Combining the document rank with frequency weighting according to the three hypotheses we get:

$$score = df \sum_{i=1}^{df} (N + 1 - r_i) \quad (9)$$

$$score = \sum_{i=1}^{df} \log(1 + f_i)(N + 1 - r_i) \quad (10)$$

$$score = \sum_{i=1}^{df} f_i (N + 1 - r_i) \quad (11)$$

The summary of all the ranking equations according to different weightings is on Table 1.

E. Evaluation

For evaluating the performance of the various ranking formulae the measures Precision-at-10 (P@10) and R-Precision were used. P@10 shows the number of relevant answers within the top-10 results. While it does not take into account the ranking of correct answers, it offers an easy interpretation of results and does not require knowledge of the total number of correct answers to be computed. Furthermore, the P@10 measure is suitable for Web retrieval evaluation, since most users usually check only the top-10 results. A problem with P@10 is that it does not average well across queries, since the number of correct answers can have great variance. R-Precision shows the number of relevant answers within the top-R results, where as R we use the total number of relevant answers in the set. R-precision overcomes the problem of variance in the number of correct answers [43].

Each ranking formula was tested on 30 queries based on the evaluation topics for entity ranking systems from INEX

TABLE 3. THE 30 EVALUATION QUERIES WITH THE NUMBER OF CORRECT RESULTS RETRIEVED (R).

Evaluation Queries	R
Pacific navigators Australia explorers	23
List of countries in World War Two	105
Nordic authors known for children's literature	6
Makers of lawn tennis rackets	3
National capitals situated on islands	46
Poets winners of Nobel prize in literature	16
Formula 1 drivers that won the Monaco Grand Prix	32
Formula One World Constructors' Champions	11
Italian Nobel prize winners	9
Musicians who appeared in the Blues Brothers movies	29
Swiss cantons where they speak German	15
US Presidents since 1960	11
Countries which have won the FIFA world cup	8
Toy train manufacturers that are still in business	9
German female politicians	108
Actresses in Bond movies	67
Star Trek Captains characters	10
EU countries	27
Record-breaking sprinters in male 100-meter sprints	14
Professional baseball team in Japan	19
Japanese players in Major League Baseball	46
Airports in Germany	52
Universities in Catalunya	8
German cities that have been part of the hanseatic league	18
Chess world champions	20
Recording companies that now sell the Kingston Trio songs	5
Schools the Supreme Court justices received their undergraduate degrees	37
Axis powers of World War Two	6
State capitals of the United States of America	36
National Parks East Coast Canada US	10

2009 and TREC 2010. The usage of these topics was not intended to compare the results of this system to those participating on these tracks, but to evaluate on a set of queries with several degrees of difficulty, in order to determine the most effective ranking method. Chosen queries deal with entity types of the three categories that are supported in the system. The queries were slightly modified to be more specific, since they originally were followed by a narrative for more details. Most of them ask for entities that satisfy more than one condition. In order to accept an entity as relevant, it had to satisfy all the conditions of the query. The correctness of the results was manually checked. The experimental results can be seen on Table 2. The query set along with the total number of relevant entities that were retrieved by the system (R) for each one is on Table 3.

The 11 ranking methods achieved similar results, so it is not clear which one is better. The P@10 measure indicates that term frequency does not improve ranking results. As the influence of term frequency increases, P@10 decreases, suggesting that the verbosity hypothesis works better for entity ranking. However, (4) and (10) that represent the middle ground, achieve a higher R-Precision. Further increase of term frequency influence on ranking, as the scope hypothesis suggests, does not offer any improvement. The ranking of documents does not have a great impact, as expected with a small set of 10 documents, but offers some small improvement except for the case of (11).

V. THRESHOLD ESTIMATION

A large set of possible queries for entity ranking problem have answers that take a binary value of relevance, i.e., they can be described as either relevant or non-relevant. All the queries used in the experiment are of this type (factoids), in contrast with queries that ask for opinion and, therefore, their answers can rarely take binary values of relevance. We investigated ways to estimate the total number of correct results (R) for a query. The threshold can then be R. This is the breakeven-point of precision and recall, meaning that at this point the precision of the system is equal to its recall. As a result, the harmonic mean of precision and recall (the F_1 measure) is maximized. This threshold choice strikes a balance between precision and recall.

The problem of threshold estimation for document retrieval was addressed in [44], where the score distributions were used to cluster the results. The scores of relevant and non-relevant results were treated as belonging to different probability distributions and the expectation maximization algorithm was used to determine their corresponding distribution. For the problem of entity retrieval, the distributions of scores are generally unimodal, so we applied nonparametric approaches to estimate the threshold.

A great variety of algorithms are used for threshold estimation in image segmentation, where the problem is to find a threshold of the grey-level value of pixels, in order to separate the foreground and background of an image. A lot of these algorithms are nonparametric and take as input only the histogram of the values making them suitable for threshold estimation in problems unrelated to image segmentation. We applied four algorithms: Otsu's algorithm [45], Kapur et al. maximum entropy [46], Rosin's algorithm [47], and the t-point algorithm [48], and tested their accuracy to the problem of threshold estimation for each ranking method. We give a short description of each algorithm.

TABLE 4 . RELATIVE ERROR OF THE THRSHOLD ESTIMATION FOR EACH THRESHOLD ALGORITHM APPLIED TO EACH RANKING EQUATION.

Equation	Otsu	Entropy	Rosin	T-point
(1)	0.7876	0.7256	0.8062	0.6451
(2)	0.9461	0.7625	0.7588	0.7742
(3)	1.2953	0.7783	1.0039	0.6860
(4)	0.7301	0.7840	0.6092	0.7045
(5)	0.8791	0.7091	0.9455	0.6666
(6)	0.8334	0.7674	0.719	0.7791
(7)	0.8341	**0.5984**	0.7594	0.7255
(8)	4.7738	0.9976	2.3284	1.7352
(9)	1.2242	0.7203	1.3532	1.088
(10)	2.8295	0.6114	1.2869	0.8739
(11)	1.0267	0.7341	0.6630	0.7248

A. *Otsu's algorithm*

For each value of the threshold in the histogram, the algorithm computes the variance of the two resulting classes and their sum. The optimum value for the threshold is the value that gives the minimum sum of variance for the two classes. This optimization criterion is equivalent to maximizing the between class variance.

$$T_{opt} = \arg\max\left\{\frac{P(T)[1-P(T)][\mu_r T - \mu_{nr} T]^2}{P(T)\sigma_r^2(T) + [1-P(T)]\sigma_{nr}^2(T)}\right\} \quad (12)$$

where $P(T)$ is the probability of a relevant result for a threshold value T, μ is the mean and σ is the variance of each class, relevant (r) and non-relevant (nr).

B. *Kapur's entropy based algorithm*

Every possible value for the threshold is tested and the algorithm calculates the sum of the entropy for the two resulting classes. The algorithm chooses as optimum threshold the value that maximizes the sum of the entropy of the two classes.

$$T_{opt} = \arg\max\{H_r(T) + H_{nr}(T)\} \quad (13)$$

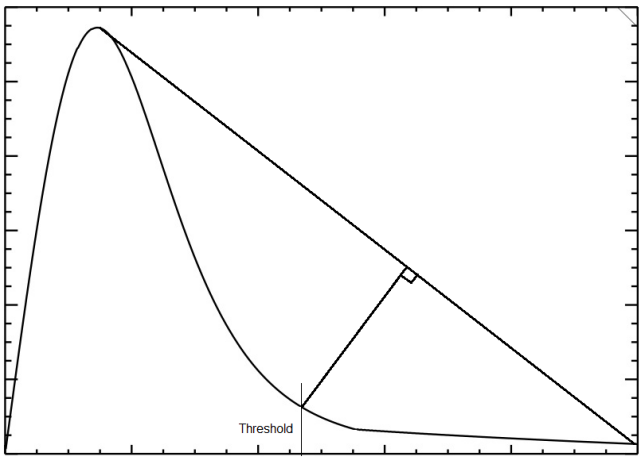


Figure 3. Rosin's algorithm. For threshold estimation in the entity ranking context, values of the x-axis represent the ranking scores and values of y-axis represent the number of entities that were assigned that score.

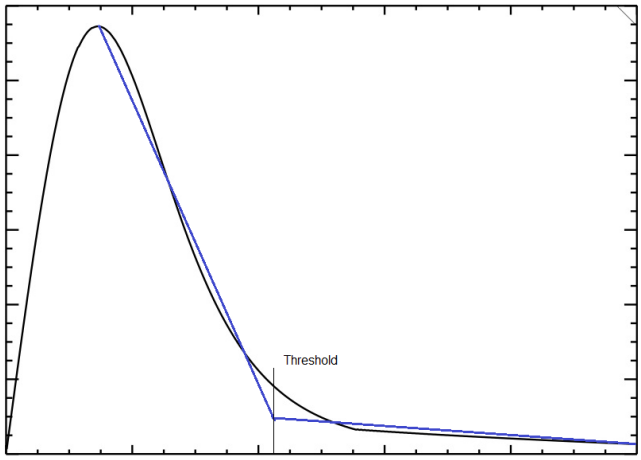


Figure 4. The T-point algorithm. Axis values are the same as in Figure 3.

$$H_r(T) = -\sum_{i=0}^{T} \frac{P(i)}{P(T)} \log \frac{P(i)}{P(T)}$$

$$H_{nr}(T) = -\sum_{i=T+1}^{S} \frac{P(i)}{1-P(T)} \log \frac{P(i)}{1-P(T)}$$

where H_r and H_{nr} are the entropies of the relevant and non-relevant class, S is the number of values of the histogram of the scores.

C. Rosin's algorithm

The algorithm considers the line that crosses the maximum value of the histogram and the last value. The threshold is determined as the point in the histogram between the aforementioned points that has the greatest Euclidian distance from the line (Figure 3).

D. T-point algorithm

For every value of the histogram between the maximum and the last value, two lines are fitted to the data using linear regression. The first is between the maximum value and the threshold, and the second between the threshold and the last value. The goodness of fit of these two lines against the points of the histogram is calculated by checking the sum of residuals. The algorithm determines the optimum threshold as the point that the two best fitted lines intersect (Figure 4).

Otsu's and Kapur's are largely cited algorithms for determining threshold values in a multimodal histogram, while Rosin's and the T-point algorithm are designed for unimodal histograms. For each algorithm and each ranking method we calculated the relative error of the estimated threshold compared to the real value and averaged over the 30 queries. The results of the experiment are in Table 4. The histogram of the scores distribution was created by grouping the values into 10 different bins. This resulted in better accuracy than using all the distinct score values, while further increase of the number of bins did not affect the results significantly.

The results suggest that each combination of ranking equation and thresholding algorithm performs differently. Although the histograms of the score distributions were mostly unimodal, the algorithms that specialize in unimodal thresholding did not perform better. The single best result is obtained with Kapur's maximum entropy algorithm when applied to the score distribution of (7).

VI. DISCUSSION

In order to decide for a better ranking method, a user model has to be taken into account. Assuming the user wants to find all relevant results, methods with higher R-Precision will work better. In case a user is interested in only a few characteristic results, then a method with higher P@10 performance will be more useful. The reason that (9) is used in the prototype is that we expect most Web users to belong in the second category.

The experiment also provided some insight in the overall system's functionality. First, we noticed the dependency of performance on the quality of retrieved documents. For queries that even one strictly relevant document was retrieved, like a Wikipedia "list of" page, the ranking was nearly optimal. In cases where partially relevant documents were retrieved, for example lists of entities according to one attribute requested by the query, the system managed to produce a combination but with reduced accuracy. The most problematic queries proved to be ones with a complex relation between attributes that cannot be well defined by simple keywords, such as "toy train companies that are still in business". A query like this would require some extra pre-processing, perhaps combined with a model for reasoning. Another problem comes with queries that have a small amount of correct answers (e.g., Axis Powers of World War Two). The threshold estimation can give valuable information in such a case so that a user can consider only a few top results as relevant. Even though only 10 documents

were used, a large number of relevant entities were retrieved for each query.

The discussed approach has the benefit of scaling well to large amounts of data. In the current implementation, we are forced to perform the named entity recognition task in real time because we do not have access to the search engine's index. Given an integrated approach, the entity extraction part can be completed offline during the preprocessing stage of indexing. The only part that always has to be computed in real time is ranking, which is accomplished by a simple formula, and is therefore not affected by the amount of data. Named entity recognition has been effectively applied in large document collections [49]. Given that all the necessary data preparation has been completed offline, we can expect that increasing the amount of data can only have a positive effect. The entity categories depend exclusively on the capabilities of the NER system. Systems for more and finer-grained categories have been described in the NER literature, e.g., in [50]. The extension of entity types can provide better filtering, but could prove problematic for the automatic type detection from the query. A more sophisticated method rather than relying on keyword detection may be needed.

VII. Conclusions and Future Work

We presented a prototype of an online application for entity ranking that uses Web documents as data and ranks the entities using IR methods. The application uses various components for recognizing the query topic, retrieving documents, extracting entities and performing coreference resolution before the ranking takes place. We formulated and evaluated several combinations of statistical quantities for ranking entities and algorithms for estimating the number of relevant results.

The experiments showed that the combination of rank position for source documents along with a measure of the common information among them yields the best results for ranking. The within-document frequency of entities did not work very well, supporting the verbosity hypothesis. Furthermore, the experiments showed that using the large data volume of the Web along with a state-of- the-art Web search engine for retrieving them, the system has little limitations in query handling. The threshold estimation experiment suggests that Kapur's maximum entropy algorithm applied to the score distribution of a multiplicative combination of entity frequency and document frequency gives the best results for estimating the number of relevant answers returned by the system.

The application currently supports search for persons, locations, and organization. The search can be easily expanded to other types of entities like products, books and movie titles by incorporating them to the extraction stage. The ranking method is very fast, but the overall speed of the application is currently confined by the entity extraction stage which uses machine learning methods. By integrating the application with a search engine the required processing for this stage could be done in advance along with the indexing stage. With this modification, the speed of the ranking method will be fully utilized.

Acknowledgment

This work was done when the first author was with the Department of Electrical and Computer Engineering, Democritus University of Thrace, Xanthi, Greece.

References

[1] A. Komninos and A. Arampatzis, "ListCreator: Entity Ranking on the Web," in *Proceedings of The Second International Conference on Advances in Information Mining and Management*, 2012, pp. 141-146.

[2] ListCreator. [Online]. Available: http://listcreator.nonrelevant.net [retrieved: May 2013].

[3] B. Sabine, "Using Grammatical Relations, Answer Frequencies and the World Wide Web for TREC Question Answering," in *Proceedings of the Tenth Text REtrieval Conference*, 2001.

[4] J. Lin, "The Web as a Resource for Question Answering: Perspectives and Challenges," in *Proceedings of the third International Conference on Language Resources and Evaluation*, 2002.

[5] B. Krisztian, L. Azzopardi, and M. De Rijke, "Formal models for expert finding in enterprise corpora," in *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieva*, 2006, pp. 43-50.

[6] P. Desislava and W. B. Croft, "Hierarchical language models for expert finding in enterprise corpora," in *International Journal on Artificial Intelligence Tools*, 2008, pp. 5-18.

[7] G. Dermatini, T. Iofciu, and A. P. de Vries, "Overview of the INEX 2008 Entity Ranking Track," in *Lecture Notes in Computer Science, Volume 5631/2009*, 2009, pp. 243-252.

[8] G. Dermatini, T. Iofciu, and A. P. de Vries, "Overview of the INEX 2008 Entity Ranking Track," in *Lecture Notes in Computer Science, Volume 6203/2010*, 2010, pp. 254-264.

[9] K. Balog, A. P. de Vries, P. Serdyukov, P. Thomas, and T. Westerveld, "Overview of the TREC 2009 Entity Track," in *Proceedings of the Eighteenth Text RErieval Conference*, 2009.

[10] K. Balog, A. P. de Vries, and P. Serdyukov, "Overview of the Trec 2010 Entity Track," in *Proceedings of the Nineteenth Text RErieval Conference*, 2010.

[11] Q. Yang, P. Jiang, C. Zhang, and Z. Niu, "Reconstruct Logical Hierarchical Sitemap for Related Entity Finding," in *Proceedings of the Nineteenth Text REtrieval Conference*, 2011.

[12] J. Guo, H. Xu, and Y. Liu, "A Novel Framework for Related Entities Finding: ICTNET at TREC 2009 Entity Track," in *Proceedings of the Eighteenth Text REtrieval Conference*, 2009.

[13] Y. Wu and H. Kashioka, "NiCT at TREC 2009: Employing Three Models for Entity Ranking Track" in *Proceedings of the Eighteenth Text REtrieval Conference*, 2009.

[14] J. Kamps, R. Kaptein, and M. Koolen, "Using Anchor Text, Spam Filtering and Wikipedia for Web Search and Entity Ranking," in *Proceedings of the Nineteenth Text REtrieval Conference*, 2011.

[15] D. Petkova and W. B. Croft, "Proximity-based document representation for named entity retrieval," in *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, 2007, pp. 731-740.

[16] Cheng, Tao, Xifeng Yan, and Kevin Chen-Chuan Chang, "EntityRank: searching entities directly and holistically," in *Proceedings of the 33rd international conference on Very large data bases*, VLDB Endowment, 2007, pp. 387-398.

[17] X. Li, C. Li, and C. Yu, "Entity-relationship queries over wikipedia," in *ACM transactions on Intelligent Systems and Technology*, volume 3, issue 4, ACM, 2012.

[18] R. Hadas, D. Carmel, and O. Kurland, "A Ranking Framework for Entity Oriented Search using Markov Random Fields," in *Proceedings of the 1st Joint International Workshop on Entit-Oriented and Semantic Search*, ACM, 2012.

[19] O. Etzioni, A. Fader, J. Christensen, S. Soderland, and Mausam, "Open Information Extraction: the Second Generation," in *Proceedings of the Twenty-second International Joint Conference in Artificial Intelligence*, 2011, pp. 3-10.

[20] M.J. Cafarella, C. Re, D. Suciu, and O. Etzioni, "Structured Querying of Web Text Data: A Technical Challenge," in *Proceedings of the Third Conference on Innovative Data Systems Research* , 2007, pp. 225-234.

[21] J. Hoffart, F.M. Suchanek, K. Berberich, E. Lewis-Kelham, G. DeMelo, and G. Weikum, "Yago2: exploring and querying world knowledge in time, space, context, and many languages," in *Proceedings of the 20th international conference companion on world wide Web, ACM*, 2011, pp. 229-232.

[22] G. Kasneci, F. M. Suchanek, G. Ifrim, M. Ramanath, and G. Weikum, "NAGA: Searching and Ranking Knowledge," in *Proceedings of the Twenty- fourth International Conference on Data Engineering*, 2008, pp. 953-962.

[23] E. Meij, M. Bron, L. Hollink, B. Huurnink, and M.D. Rijke, "Mapping queries to the Linking Open Data cloud: A case study using DBpedia," in *J. Web Semantics*, December 2011, pp. 418-433.

[24] J. Pound, I.F. Ilyas, and G. Weddell, "Expressive and flexible access to Web-extracted data: a keyword-based structured query language," in *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, 2010, pp. 423-434.

[25] M. Grobelnik, P. Mika, D. T. Tran, and H. Wang. *Proceedings of the 3rd International Semantic Search Workshop*, ACM, 2010.

[26] M. Grobelnik, P. Mika, D. T. Tran, H. Wang, "SemSearch'11: the 4th semantic search workshop," in *Proceedings of the 20th international conference companion on World Wide Web*, ACM, 2011, pp. 315-316.

[27] K. Balog, D. Carmel, A.P. de Vries, D. M. Herzig, P. Mika, H. Roitman, and T. T. Duc, "The first joint international workshop on entity-oriented and semantic search (JIWES)," *ACM SIGIR Forum*, Volume 46. No. 2. ACM, 2012, pp. 87-94.

[28] K. Balog, R. Neumayer, and K. Norvag, "On the modeling of entities for ad-hoc entity search in the Web of data," in *Advances in Information Retrieval*, 2012, pp. 133-145.

[29] R. Delbru, S. Campinas, and G. Tummarello, "Searching Web data: An entity retrieval and high-performance indexing model," in *Web Semantics: Science, Services and Agents on the World Wide Web*, Volume 10, 2012, pp. 33-58.

[30] H. Wang, Q. Liu, T. Penin, L. Fu, L. Zhang, T. Tran, Y. Yu, and Y. Pan, "Semplore: A scalable IR approach to search the Web of Data," in *Web Semantics: Science, Services and Agents on the World Wide Web*, Volume 7, Issue 3, 2009, pp. 177-188.

[31] Java. [Online]. Available: http://www.java.com/en/ [retrieved: May 2013].

[32] HTML 4.01 Specification. [Online]. Available: http://www.w3.org/TR/1999/REC-html401-19991224/ [retrieved: May 2013].

[33] JavaScript. [Online]. Available: https://developer.mozilla.org/en-US/docs/JavaScript [retrieved: May 2013].

[34] PHP. [Online]. Available: http://www.php.net/ [retrieved: May 2013].

[35] Princeton University, 2010, WordNet. [Online]. Available: http://wordnet.princeton.edu [retrieved: May 2013].

[36] Yahoo BOSS API. [Online]. Available: http://developer.yahoo.com/search/boss [retrieved: May 2013].

[37] J. R. Finkel, T. Grenager, and C. Manning, "Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling," in *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, 2005, pp. 363-370.

[38] GNU General Public License. [Online]. Available: http://www.gnu.org/licenses/gpl.html [retrieved: May 2013].

[39] Named Entity Recognition and Information Extraction [Online] http://nlp.stanford.edu/ner/index.shtml [retrieved: May 2013].

[40] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data," in *Proceedings of the Eighteenth International Conference on Machine Learning*, 2001, pp. 282-289.

[41] Jsoup: Java HTML Parser. [Online]. Available: http://jsoup.org [retrieved: May 2013].

[42] S. E. Robertson and S. Walker, "Some Simple Effective Approximations to the 2–Poisson Model for Probabilistic Weighted Retrieval," in *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1994, pp. 345–354.

[43] C. Buckley and E. M. Voorhees, "Retrieval Evaluation with Incomplete Information," in *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2004, pp. 25-32.

[44] A. Arampatzis, J. Kamps, and Stephen Robertson, "Where to Stop Reading a Ranked List? Threshold Optimization using Truncated Score Distributions," in *Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2009, pp. 524–531.

[45] N. Otsu, ''A threshold selection method from gray level histograms,'' in *Automatica*, Volume 11, no. 285-296, 1975, pp. 23-27.

[46] J. N. Kapur, P. K. Sahoo, and A. K. C. Wong, ''A new method for gray-level picture thresholding using the entropy of the histogram,'' in *Computer Vision, Grapic, and Image Processing*, Volume 29, issue 3, 1985 ,pp. 273-285.

[47] P.L. Rosin, "Unimodal thresholding," in *Pattern Recognition*, Volume 34/11, 2001, pp. 2083‑2096.

[48] N. Coudray, J.L. Buessler, J.P. Urban, "Robust threshold estimation for images with unimodal histograms," in *Pattern Recognition Letters*, Volume 31, Issue 9, 2010, pp. 1010-1019.

[49] C. Whitelaw, A. Kehlenbeck, N. Petrovic, and L. Ungar. "Web-Scale Named Entity Recognition." in *Proceedings of the 17th ACM conference on Information and knowledge management*, 2008, pp. 123-132.

[50] S. Sekine and C. Nobata, "Definition, Dictionaries and Tagger for Extended Named EntityHierarchy," in *Proceeding of Itnernational Conference on Language Resources and Evaluation*, 2004.

Computing User Importance in Web Communities by Mining Similarity Graphs

Clemens Schefels
Institute of Computer Science, Goethe-University Frankfurt am Main
Robert-Mayer-Straße 10, 60325 Frankfurt am Main, Germany
Email: schefels@dbis.cs.uni-frankfurt.de

***Abstract*—The economic success of the World Wide Web makes it a highly competitive environment for web businesses. For this reason, it is crucial for web business owners to learn what their customers want. In this paper, we provide a useful tool to the web site owner for analyzing her/his web community. In particular, the web site owner can compute the importance of the users and analyze the structure of the specific community by comparing the interests of the users. Therefore, we present the conception and implementation of a tool for building and analyzing weighted similarity graphs, e.g., for a social web community. For that, we provide measurements for user equality and user similarity. We introduce different graph types for analyzing profiles of web community users. Moreover, we propose two new algorithms for finding important users of an on-line community.**

Keywords-Computer aided analysis; World Wide Web; Data analysis; Graph theory; Application software.

I. Introduction

This paper is an extended version of [1] presented at the First International Conference on Data Analytics in Barcelona in 2012.

These days, web-based user communities enjoy great popularity. The social network Facebook[1] has more than 1 billion active users [2] and even the relatively new Google+[2] about 235 million [3]. In this highly competitive environment, it is crucial for web site owners to understand and satisfy their web community.

To reach this goal, we present the conception and implementation of a tool for building and mining similarity graphs. These similarity graphs are built from the interest profiles of the users of a web community. We use the Gugubarra framework [4] [5], developed by the DBIS research group at the Goethe-University Frankfurt am Main, to compute interest profiles of web users. Our approach addresses the following research questions:

- Which users are important for the web community?
- Which users have similar interests?
- How similar are the interests of the users of the web community?
- How is this specific web community structured?

[1]https://www.facebook.com/
[2]https://plus.google.com/

To measure the similarity of the users, we are using different techniques from graph theory. First, we will introduce the similarity threshold that helps the web site owner in building the similarity graph of her/his community. This threshold sets how similar the users must be to be connected together in the similarity graph. In addition to that, it reduces the complexity of the graph. Second, we will provide several algorithms to find important users in the similarity graph. There exists not only one valid definition for importance of users because it depends–as always–on the point of view. For this reason, we provide nine algorithms to discover the importance of users. Two of these algorithms are new designed in respect to the needs of similarity graphs.

In contrast to other researches that derive the importance of users from the social structures of web communities (e.g., Trusov et al. [6]), we calculate the user importance from their interests.

The rest of the paper is structured as follows: Section II outlines related work and Section III introduces basic concepts and definitions that will be used in the rest of the paper. Section IV presents the main contribution of this paper, our analysis tool for building and mining similarity graphs and an implementation of a prototype. After we evaluate our analysis tool with a real usage dataset in Section V, we integrate it into the Gugubarra Framework in Section VI. Section VII presents the conclusion and future work.

In what follows, we assume that users are aware and have granted permission that implicit and explicit data is collected and kept in their profile for them.

II. Related Work

Previous research discovered community structures in social networks, but focused on the pure friendship or relationship structure of these communities. E.g., Rongjing Xiang et al. developed in [7] an unsupervised model to estimate relationship strength from interaction activity (e.g., communication, tagging) and user similarity. In this work, we calculate the relationship structure from the interests of the web community users. Moreover, we use their interest profiles to determine the relationship strength between the users.

To evaluate the web user's level of expertise (i.e., importance) on a given topic, Jidong Wang et al. [8] propose a link analysis. They use a unified directed graph, where the nodes of the graph are users and web pages and the directed edges represent the hyper links between web pages and user's visit of the web pages. The link analysis algorithm is derived from the algorithm presented by Kleinberg in [9] that we also use to compute the importance of the users. Moreover, we use nine different algorithms to determine the importance of web users because we think there is not only one valid definition for importance.

The detection of important users, i.e., leaders in behavior networks, is the focus of the publication of Esslimani et al. [10]. The behavior network is a graph where the nodes represent the users and the edges represent the links between users. The navigational similarities are the weights of the edges. The detection of leaders relies on their high connectivity in these behavioral networks and their potentiality of propagating accurate appreciations. We also understand high connectivity of users in a network as an indicator for their importance. Both of our new algorithms to detect important users take the connectivity of the users into account.

In [11], Paliouras uses a similarity threshold to transform a weighted user graph into an unweighted graph. As a side effect, the connectivity of the graph is reduced. In our work, we use a similarity threshold to reduce the complexity of the web community graph too. In contrast to Paliouras, we keep the resulting graph weighted and use the edge weights as additional information to calculate the important users of the web community.

III. Basic Concepts and Definitions

In this section, we introduce the analytic framework Gugubarra, which is used to calculate the interest profiles of the web community users. Furthermore, we present the definitions of the user equality and the user similarity, concepts of the graph theory, and seven algorithms to determine the importance of users.

A. The Web Analytics Software Gugubarra

The web based analytic framework *Gugubarra*, also described in [4] [5] [12], is a prototype system developed by the Databases and Information Systems (DBIS) research group at the Goethe-University Frankfurt am Main. The purpose of the system is to help the owner or manager of a web site to more fully understand the interests of the registered users on her/his web site. We use the Gugubarra interest profiles of the users to build the similarity graphs. Therefore, we introduce the basic concepts of this framework.

In this project, a *web site* is a collection of web pages, where *visitors* or *users* can register and log on. The combined group of *registered* users of this web site are called the *web community*. This web site is maintained by a web site *owner* who controls the content and decides on the business strategies or goals. During a user *session*, which is defined as the time between the log-in and the log-out of a web user, all web page requests are stored in the log files of the web server and enriched with additional information, such as zones, topics, and actions, which are explained in [13]. All of these data are used to calculate profiles describing the interests of every web site user. In Gugubarra, each user profile is stored as a vector that presents the supposed interests of a user u_m related to a topic T_i at time t_n. Each vector row contains the calculated interest value of the user for a given topic. The values of the interest are between 0 and 1, while 1 indicates high interest and 0 indicates no interest for a topic (see Figure 1). Gugubarra generates for each registered user several profiles, as follows.

The *explicit* user data are stored in *two* different profiles, in the *Obvious Profile* (OP) and in the *Feedback Profile* (FP) [4]. Explicit user data means, that the web site user is directly asked by the web site owner about the data, e.g., by an e-mail or a web form. The OP [13] contains identification and personal data, e.g., name, address of the user. The FP holds the explicit feedback of the user. The advantages of these types of data is that they come directly from the user and that the user is aware of being asked about her/his interests. Thus, the results can reflect the interests of a user very accurately. However, the disadvantages are that a user can misinterpret the topics and/or give inaccurate answers. The explicit user feedback is a valuable source for the calculation of user interest profiles.

In addition to the explicit user data, the Gugubarra Framework calculates user interests from the *implicit* user data, too. The sources of the implicit user data are the interactions of the visitors with the web site, particularly, the behavioral data. With these data, Gugubarra compensates for the constraints of the explicit user data mentioned above. The implicit user data are stored in the *Non-Obvious Profile* (NOP), which consists of the *Action Profile* (ActP) and the *Duration Profile* (DurP) [13]. In [14], the implicit user profiles of the Gugubarra Framework are extended with data form the mouse activities of the web site user.

The *Relevance Profile* (RP), introduced in [15] and [16], unites the explicit and the implicit feedback profiles of a user into a single interest profile. Figure 1 shows an example of an RP, where we calculated the data of a user u_m at time t_n, based on her/his implicit and explicit feedback, showing a supposed high interest in topic T_2 (1.0), lower interest in topic T_1 (0.3), and no interest in topic T_3 (0.0).

$$RP_{u_m,t_n} = \begin{pmatrix} 0.3 \\ 1.0 \\ 0.0 \end{pmatrix} \begin{matrix} \leftarrow T_1 \\ \leftarrow T_2 \\ \leftarrow T_3 \end{matrix}$$

Figure 1: Relevance Profile of user u_m for topic T_1, T_2, T_3.

We use the RP to calculate the graphs of the web

community. Therefore, we provide the necessary definitions in the next sections.

B. Similarity measurement

Due to the fact that the RP contains all information about the interests of the users, we want to use it to compute the similarity between the interests of *all* users. First, we have to definite the *equality of users*:

Two users u_i and u_j are equal in respect to a topic T_r of a web site at time t_n if the interest values of T_r of their RPs are equal:

$$RP_{u_i,t_n}(T_r) = RP_{u_j,t_n}(T_r) \; where \; i \neq j. \tag{1}$$

To compare users we need a measurement for *similarity*. Similarity measurements are very common in the research field of data mining. For example, documents are often represented as feature vectors [17], which contain the most significant characteristics like the frequency of important keywords or topics. To compute the similarity of documents, the feature vectors are compared with the help of distance measurements: the smaller the distance the more similar the documents are.

Gugubarra interest profiles, i.e., the RP, can be considered as feature vectors of the users, too. They contain the most significant characteristics of our users, e.g., the interests in different topics of a web site. Therefore, we can use the similarity measurements of data mining theory to compute similarity between the members of our community.

An important requirement on the similarity measurement algorithm is its performance because a web community can cover lots of users. Consequently, we have to choose a similarity measurement with a high performance so that the analysis program will scale with the high number of users. Aggarwal et al. proved in [18] that the *Manhattan Distance*, also known as *City Block Distance* or *Taxicab Geometry*, is very well suited for high dimensional data. We shared in [19] that web sites may have up to 100 topics. Thus, we have to deal with high dimensional feature vectors, i.e., one dimension per topic.

The Manhattan Distance (L_1-norm) [20] is defined as follows:

$$d_{\text{Manhattan}}(a, b) = \sum_i |a_i - b_i| \tag{2}$$

with $a = RP_{u_m,t_n}$, $b = RP_{u_r,t_n}$ and $m \neq r$.

To calculate the *user similarity* we take the RP interest value of every topic of each user and calculate the Manhattan Distance between all users of the web community as illustrated in the following example:

Let us assume we have a web site with three topics T_1, T_2, and T_3. This web site has two registered users u_1 and u_2. The RPs of the two users were calculated at time t_1:

$$RP_{u_1,t_1} = \begin{pmatrix} 1.0 \\ 0.5 \\ 0.0 \end{pmatrix}, \; RP_{u_2,t_1} = \begin{pmatrix} 0.6 \\ 0.8 \\ 0.2 \end{pmatrix} \tag{3}$$

The Manhattan Distance is calculated as follows:

$$d_{\text{Manhattan}}(RP_{u_1,t_1}, RP_{u_2,t_1}) = \tag{4}$$

$$= |1.0 - 0.6| + |0.5 - 0.8| + |0.0 - 0.2| = 0.9$$

where 0.9 is the distance of the interests of the both users, i.e., the similarity. In general, the *smaller* the calculated distance is the *more similar* are the compared users to each other.

C. Graph Theory

In this section, we present the basic definitions of graph theory, which was founded by Leonhard Euler [21], that are necessary for our tasks.

A *graph* G [22] is a tuple $(V(G),\ E(G))$. $V(G)$ is a set of *vertices* of the graph and $E(G)$ is the set of *edges*, which connects the vertices. Sometimes it is postulated [22] that $V(G)$ and $E(G)$ has to be finite but there exists also definitions about infinite graphs [23]. However, the number of web site users should be finite.

A graph G can be represented [24] by an *adjacency matrix* $A = A(G) = (a_{ij})$. This $n \times n$ matrix, n is the sum of the vertices of G, is defined as follows:

$$a_{ij} = \begin{cases} 1 & \text{if } \{v, w\} \in E(G) \\ 0 & \text{otherwise.} \end{cases} \quad with \; v,\ w \in V(G) \tag{5}$$

In a *simple graph* an edge connects always *two* vertices [25]. This means that $E(G)$ consists of unordered pairs $\{v, w\}$ with $v, w \in V(G)$ and $v \neq w$ [22]. In a social network vertices could represent the members of this network and the edges could stand for the friendship relation between these vertices–so friends are connected together.

Every pair of distinct vertices of a *complete graph* [22] are connected together.

The connections between edges can be *directed* or *undirected*. In a directed graph the edges are an ordered pair of vertices v, w and can only be traversed in the direction of its connection. This means that a *simple graph* is undirected. This feature is very useful, e.g., to model the news feed subscriptions of a user in a social network, a one-way friendship.

A *loop* is a connection from a vertex to itself [24]. A loop is not an edge.

Labeled vertices make graphs more comprehensible. Vertices can be labeled with identifiers, e.g., in the social network graph with the names of the users.

In the same way edges can be labeled to denote the kind of connection. In the social network graph example, the label could represent the kind of relation between users, e.g., friend or relative.

With *weighted graphs*, the strength of the connection between the single vertices can be modeled. Every edge has an assigned weight. In a social network the weight could be used to display the degree or importance of the relationship of the users. A weighted graph can also be represented by an adjacency matrix (see Definition 5 above) where a_{ij} is the weight of the connection of $\{v, w\}$. See Example 6 for an adjacency matrix of a similarity graph of five users:

$$A = \begin{pmatrix} 0.00 & 1.28 & 1.19 & 2.79 & 1.18 \\ 1.28 & 0.00 & 1.63 & 2.83 & 1.90 \\ 1.19 & 1.63 & 0.00 & 2.50 & 1.35 \\ 2.79 & 2.83 & 2.50 & 0.00 & 2.85 \\ 1.18 & 1.90 & 1.35 & 2.85 & 0.00 \end{pmatrix} \quad (6)$$

Every number represents the weight of the edges between two vertices, e.g., $a_{2,4} = 2.83$ represents the edge weight of the two vertices with the numbers 2 and 4. The diagonal of this matrix is 0.00 because the graph has no loops. In an undirected graph the adjacency matrix is symmetric.

A vertex w is a *neighbor* of vertex v if both are connected via the same edge. The neighborhood of v consists of *all* neighbors of v. In a social network a direct friend is a neighbor and all direct friends are the neighborhood.

A *path* [26] through a graph G is a sequence of edges $\in E(G)$ from a starting vertex $v \in V(G)$ to an end vertex $w \in V(G)$. If there exists a path form vertex v to w both vertices are connected. The number of edges on this path is called *length* of the path and the *distance* between v and w is the length of the shortest path between these two vertices. A path with the same start and end point is called *cycle*. Two vertices v and w are *reachable* from each other if there exists a path with the start point v and the end point w. If all vertices are reachable from every vertex the graph is called *connected*.

G' is a *subgraph* [24] of G if $V(G') \subset V(G)$ and $E(G') \subset E(G)$. G is than the *supergraph* of G' with $G' \subset G$.

A *community* in a graph is a *cluster* of vertices. The vertices of a community are dense connected.

D. Importance

There exist many algorithms to measure the importance of a vertex in graph. We introduce seven of the most common algorithms:

Sergin Brin and Lawrence Page [27] used their *PageRank* algorithm to rank web pages with the link graph of their search engine Google[3] by importance. This algorithm is scalable on big data sets (i.e., search engine indices). Usually the PageRank algorithm is for unweighted graphs. But there exists also implementations for weighted graphs [28]. Pujol et al. [29] developed an algorithm to calculate the reputation of users in a social network. The results of the comparison of their algorithm with the PageRank show that the PageRank is also well suited for reputation calculation, i.e., importance calculation.

The *Jaccard similarity coefficient* [30] of two vertices is the number of common neighbors divided by the number of vertices that are neighbors of at least one of the two vertices being considered [31]. Here the pairwise similarity of all vertices is calculated.

The *Dice similarity coefficient* [31] of two vertices is twice the number of common neighbors divided by the sum of the degrees of the vertices. Here the pairwise similarity of all vertices is calculated.

Nearest neighbors degree calculates the nearest neighbor degree for all vertices. In [32] Barrat et al. define a nearest neighbor degree algorithm for weighted graphs.

Closeness centrality [33] measures how many steps are required to access every other vertex from a given vertex.

Hub score [9] is defined [31] as the eigenvector of $A\ A^T$ where A is the adjacencies matrix and A^T the transposed adjacencies matrix of the graph.

Eigenvector centrality [34] [31] correspond to the values of the first eigenvector of the adjacency matrix. Vertices with high eigenvector centralities are those, which are connected to many other vertices, which are, in turn, connected to many others.

In Section V, we present an evaluation of these algorithms and compare the results with two new algorithms.

IV. Analysis of Similarity Graphs

We developed a new tool for building and analyzing similarity graphs. We integrated several algorithms from different research areas for the analysis of the graphs. The following sections should clarify research questions such as:

- Which users are important for the web community?
- Which users have similar interests?
- How similar are the interests of the users of the web community?
- How is this specific web community structured?

By answering these questions, we want to give the web site owner a useful tool to enhance her/his marketing strategies, in respect of the work of Domingos and Richardson [35], and rise as consequence the click rates of her/his portal.

Before we integrate this tool in the Gugubarra Framework, we tested our concepts with a prototype written in *R*[4]. *R* is an open source project with a huge developer community. The archetype of *R* is the statistic programing language

[3] https://www.google.com/

[4] http://www.r-project.org/

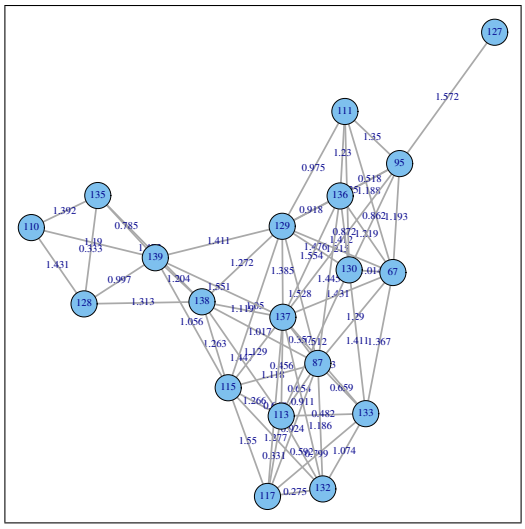


Figure 2: Smallest connection graph.

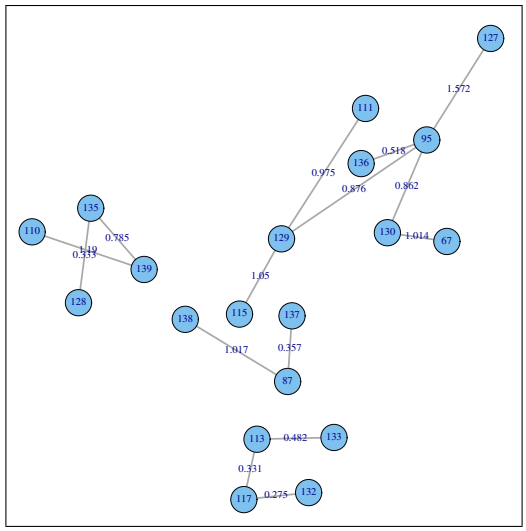


Figure 3: Closest neighbor graphs.

S[5] and the functional programming language Scheme[6]. *R* has a big variety of libraries with many different functions for statistical analytics. For graph analysis *R* provides two common libraries: the Rgraphviz[7] and the igraph[8] library. We are using the latter for our implementation because it provides more graph analytics algorithms[9] [36] and it is better applicable for large graphs. The igraph library is also available for other programming languages (e.g., *C*, Python).

Our graph analytics tool follows a two phases work flow. In the first phase, the similarity graph is built and in the second the built graph can be analyzed with different algorithms. The next paragraphs describe the work flow in more detail.

A. Building Similarity Graphs

In the first work flow phase, the similarity graph of RPs of the web community users has to be build. We use an undirected, vertices and edges labeled, weighted graph without loop to build a model for the similarity of the web community users. The weighted edges represent the similarity between the vertices, which stand for the users. The edges are labeled with the similarity value, that is the Manhattan Distance between the RPs of the users. The labels of the vertices are the user IDs. We use an undirected graph because the similarity of two users can be interpreted in both directions. Figures 2 and 3 show examples of a similarity graph. As mentioned before, in the research field of social networks graph analysis is used to detect social structures between the users, like in [37]. These graphs represent the friend relationship of the users and are different in comparison to our work. We use *weighted* graphs to embody the similarity of users where the edge weights represent the similarity between the interests of the users. So, we are not able to use the graph analytics algorithm tools from the social network analysis.

In our tool, the web site owner can chose different alternatives to build a similarity graph for the analysis. The vertices of the graph (the users) are connected via edges that represent the similarity. It is possible to connect every user to all other users so that a complete graph represents the similarity between all users. This graph is huge and not easy to understand. To reduce the complexity of this graph we introduce a *similarity threshold*. This threshold defines how similar the users must be to be connected together. Only users are connected via vertices whose Manhattan Distance of their RPs is smaller (remember: the smaller the distance the more similar users are) than the chosen threshold. Our analysis tool provides several predefined options to build different graphs with different thresholds. All these graphs are subgraphs of the complete similarity graph of the whole web community:

- **Smallest connected graph**: with this option the threshold increases until every user has at least one connection to another user. In Figure 2, user no. 127 was added last to the graph and has a Manhattan Distance of 1.572. Accordingly, all connected vertices have a similarity smaller or equal to 1.572. The result is *one* connected graph.
- **Closest neighbor graphs**: here users are only connected with their most similar neighbors. Every vertex has at least one edge to another vertex. If there exist more most similar neighbors with the same edge weight, the vertex is connected to all of them. This

[5] http://stat.bell-labs.com/S/
[6] http://www.r6rs.org/
[7] http://bioconductor.org/packages/release/bioc/html/Rgraphviz.html
[8] http://igraph.sourceforge.net/
[9] http://igraph.sourceforge.net/doc/html/index.html

can result in *many* independent graphs as displayed in Figure 3. The difference to the nearest neighbor algorithm is that the nearest neighbor algorithm calculates a path through an existing graph by choosing always the nearest neighbor of the actual vertex.

- **Minimum spanning tree** [38]: is a subgraph where all users are connected together with the most similar users. In contrast to the "closest neighbor graph" we have *one* connected graph.
- **Threshold graph**: at last the web site owner can chose a similarity threshold on her/his own. To simplify the choice, the tool suggests two thresholds to the owner: a minimum threshold and a maximum threshold. With the minimum threshold only the most similar users are connected together and with the maximum threshold all users are connected together with every user. So the owner can chose a value between the suggested thresholds to get meaningful results.

B. Similarity Graph Mining Algorithms

In the second work flow phase, the web site owner can analyze the graph, generated in the first phase of the work flow, with different algorithms. The aim here is to detect the important users of the similarity graph.

What is an important user? There exists not only one valid definition because it depends–as always–on the point of view. In social networks, e.g., the importance of users often stands for their reputation. The reputation of a user can be measured, e.g., by its number of connectors to other users. Therefore, a connector in social networks has another meaning, i.e., the friendship, as in our similarity graphs. Thus, we can not use this definition of user importance.

In a social graph a user could be important if she/he is central in respect to the graph. Centrality means that from this very user all other users should be not far away–it should be the nearest neighbor. These highly connected users are often referred as *Hubs* or *Authorities* [9]. Hubs have many outgoing edges while Authorities have many incoming edges.

In a weighted similarity graph high importance could mean that this user is the most similar to other users–she/he should have many edges to other vertices and the edges weights should be as low as possible.

Accordingly, we provide nine algorithms to discover the importance of users. Therefore, the importance is defined by the used algorithm, which are explained below.

- **PageRank**: The vertex with the highest "PageRank" is the most important user.
- **Jaccard similarity coefficient**: We interpret the most similar vertex as the most important user.
- **Dice similarity coefficient**: Like above, we interpret the most similar vertex as the most important user.
- **Nearest neighbors degree**: If a vertex has many neighbors it can be considered as important.
- **Closeness centrality**: Vertices with a low closeness centrality value are important.
- **Hub score**: Vertices with a high score are named hubs and should be important.
- **Eigenvector centrality**: Vertices with a high eigenvector centrality score are considered as important users.

As these seven algorithms above are not *extra* designed to find the important vertices, i.e., users, in similarity graphs of user interests, we developed two new algorithms:

- **Weighted degree**: This simple algorithm choses the vertex with the most connections. Vertices with many connections are important users because they are similar to other user. Actually, they are connected with other users cause of their similarity. If there are vertices with the same number of connections it takes the vertex with the lowest edge weights. Therefore, the most unimportant or least important vertex has fewer connections to other vertices and the highest edge weights.
- **Range centrality**: The idea behind this algorithm is that a user is important who has many connections in comparison with the other users of the graph, short distance to her/his neighbors, and low edge weights. The range centrality is defined as follows:

$$C_r = \frac{range^2}{aspl + aspw} \tag{7}$$

The *range* is the fraction of the number of users that are reachable from the analyzed vertex and of all users of the graph. We take the square of the range because we consider a user as very important that is connected with many other users:

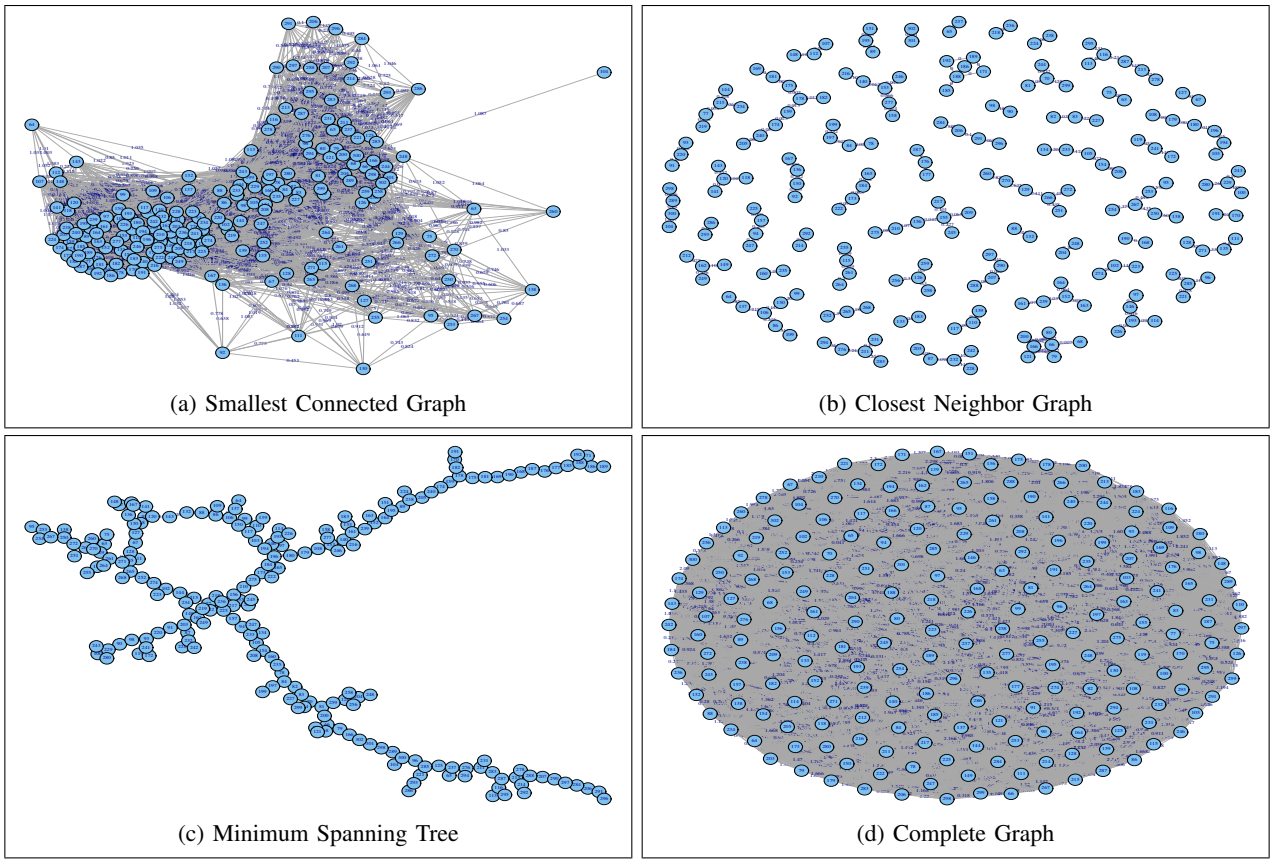
$$range = \frac{\#reachable\ user}{\#all\ user} \tag{8}$$

The average shortest path length (*aspl*) is the average length of all shortest paths divide by the number of all shortest paths. The shortest paths are calculated with the analyzed vertex as starting point:

$$aspl = \frac{average\ shortest\ paths\ length}{\#shortest\ paths} \tag{9}$$

With the average shortest path weight (*aspw*) we take into account that the weight of the connected vertices should be very low, i.e., the vertices should be very similar. It's the fraction of the sum of all shortest paths weights and of the number of all shortest paths:

$$aspw = \frac{sum\ of\ all\ shortest\ paths\ weights}{\#shortest\ paths} \tag{10}$$



(a) Smallest Connected Graph

(b) Closest Neighbor Graph

(c) Minimum Spanning Tree

(d) Complete Graph

Figure 4: Similarity graphs of all users.

In the next section we will use our analysis tool with real usage data and compare our new algorithms with the established ones.

V. Evaluation

To evaluate our algorithms, we use the real usage data from our institute web site[10], i.e., the users' session log files of the site community.

A. Material and Methods

For this evaluation the data of all registered visitors of the DBIS web site were analyzed. We observed 213 registered users over two years. For each user an RP is calculated. The data were collected during the period between June 2010 and July 2012. We used the same settings for the Gugubarra Framework as described in [12]. For each topic, zone topic weights ware associated with different *zones* [13].

Next, we use our analytics tool to build similarity graphs from the RPs of the users and calculate for every graph type the most important and the most unimportant user.

[10]http://www.dbis.cs.uni-frankfurt.de/

B. Results

First phase: In the first phase of the analysis process, we generate the similarity graphs of the users. The four graphs are displayed in Figure 4.

Second phase: In the second phase, we analyze the graph, generated in the first phase, with different algorithms. The aim here is to detect the important users in the similarity graph.

Table I displays the results of our calculations. The rows present the different graph types: *SCG* stands for *S*mallest *C*onnection *G*raph, *CNG* for *C*losest *N*eighbor *G*raph, *MST* for *M*inimum *S*panning *T*ree, and *CG* for *C*omplete *G*raph. For every graph type, the user with maximum and minimum importance is displayed. Every column presents one importance algorithm. We can observe the following fact in the dataset in respect to our algorithms, the weighted degree and the rang centrality:

In the SCG, the range centrality calculates user no. 220 as most *important* user. The weighted degree, the closeness centrality, and the PageRank select user no. 93 as most *important*. User no. 223 is *important* for the eigenvector centrality and the Dice similarity coefficient. The hub score chose user no. 91 and the nearest neighbor degree user

Table I: Evaluation Results: IDs of the users with maximum and minimum importance of every graph type (rows) for different algorithms (columns).

		Page Rank	Nearest N.D.	Dice S.C.	Jaccard S.C.	Closeness C.	Hub Score	Eigen-vector C.	Weighted Degree	Range C.
SCG	Max	93	216	223	232	93	91	223	93	220
	Min	104	138	104	104	104	104	104	104	104
CNG	Max	155	169	79,80,121,200	79,80,121,200	178	66	300	66	178
	Min	68	63,65,67,...	63,65,67,...	63,65,67,...	63,65,67,...	63,65,67,...	270	104	63,75
MST	Max	241	169	68,80,121,200	68,80,121	225	261	129	66	225
	Min	104	293	112,229,232	112,229,232	296	296	300	104	296
CG	Max	241	241	all users	all users	all users	all users	104	241	241
	Min	104	104	all users	all users	all users	all users	241	79	104

no. 216 as most *important*. The majority of algorithms calculate the same *unimportant* user (user no. 104), only the nearest neighbor degree centrality differs (user no. 138).

In the CNG, the range centrality and the closeness centrality calculates the same *important* user (user no. 178). The same *unimportant* users (user no. 63 and user no. 75) selects the range centrality, the hub score, the closeness centrality, the Jaccard and the Dice similarity coefficient, and the nearest neighbor degree. The results of the weighted degree for the most *important* user is no. 66 and for the most *unimportant* user no. 104.

In the MST, the results of the range centrality equals the closeness centrality, while the weighted degree calculates the same *unimportant* user as the PageRank. The range centrality, the hub score, and the closeness centrality select user no. 296 as most *unimportant* one.

In the CG, user no. 241 is the most *important* user for all, except for the eigenvector centrality. User no. 104 is the most *unimportant* user for the rang centrality, the PageRank, and the nearest neighbor degree. The eigenvector centrality calculates exact the opposite results. The Dice similarity coefficient, the Jaccard similarity coefficient, the closeness centrality, and the hub score are not able to find an *un-/important* user in the complete graph, because these algorithms do not include the edge weights into their calculation.

C. Discussion

Since there is no method to measure the importance objectively, we compare established algorithms with our approach. Every algorithm calculates importance in a different way, because every algorithm author has another definition of importance. Most of the algorithms are not designed for similarity or even weighted graphs. Therefore, a comparison is difficult.

The weighted degree algorithm firstly focuses on the number of connected neighbors and secondly on the weights of the connected edges. The results of the weighted degree algorithm are very different from the results of the other algorithms, only the hub score and the PageRank seem to be comparable. In contrast to the hub score the weighted degree algorithm is able to find an important user in a complete graph because it considers the edge weights of the connections (if there are users with the same number of connections, which is alway the case in a complete graph).

Similarly, the range centrality focuses on the number of connections, but also on the reachability of the user and the path length. In other words, it considers the whole graph. In comparison to the other algorithms the range centrality is very similar to the closeness centrality but the results differs at the complete graph. Here, our range centrality algorithm calculates important and unimportant users, which is similar to the PageRank algorithm, but the closeness centrality can not calculate any similarity. This is an advantage of our algorithm.

In summary, we think that our new algorithms are a good alternative for computing the importance of users in similarity graphs.

VI. Integration into the Web Analytics Software Gugubarra

After the successful prototype testing, we integrate the tool for building and mining similarity graphs in the Gugubarra Framework. The framework consists of two parts [39] [12], the Gugubarra Designer and the Gugubarra Analyzer.

The *Gugubarra Designer* helps the web site owner to include the concepts of Gugubarra into the web site and stores the feedback data of the web site users. It is realized as a plugin for the content management system Joomla!.

The other part, the *Gugubarra Analyzer*, analyzes the data of the Gugubarra Designer, to build the user profiles, and

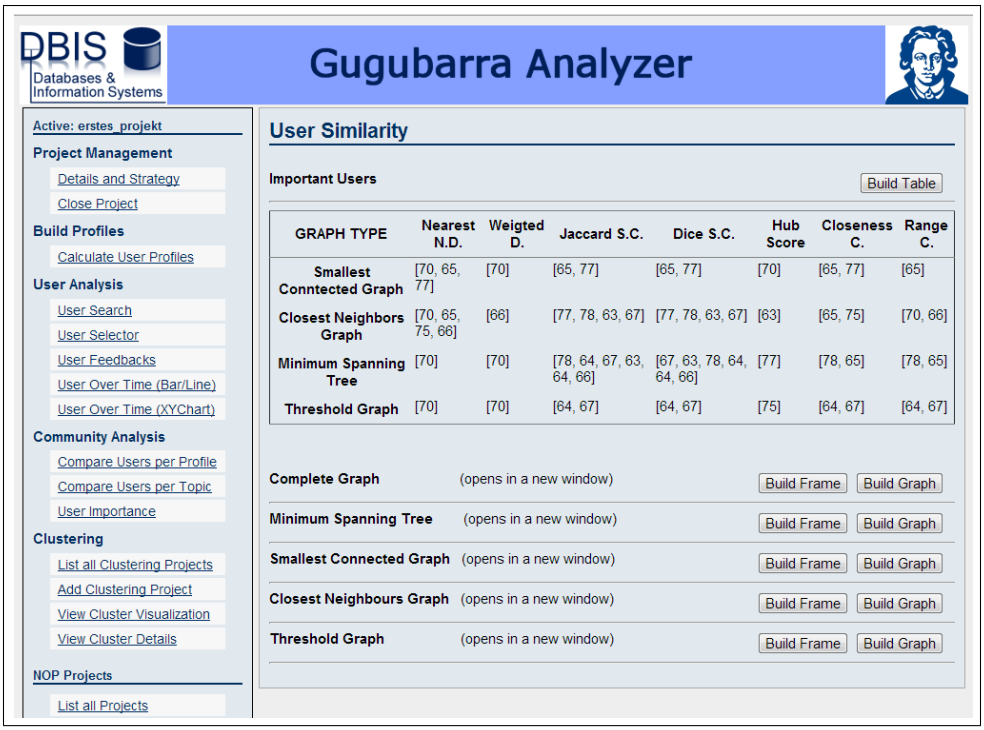


Figure 5: The GUI of the Gugubarra service for building and analyzing similarity graphs [40].

to provide the web site owner with a web application to analyze her/his web community. The Gugubarra Analyzer is a separate service and can be installed on the same or a different machine than the Gugubarra Designer.

Tapestry[11], an open source framework for creating dynamic web applications in Java[12], is used in this application to build the simple HTML[13] pages, such as the configuration dialogs. For more complex pages, the Java libraries Swing and JavaFX are used. Within the GUI configuration dialogs, the web site owner can influence the user profile calculations by changing different parameters. After configuration, the different user profiles are calculated and presented to the web site owner.

The Gugubarra Analyzer comes with a few of different analysis services, which allow the web site owner to examine some statistics about the web site community. The user profiles and the different services are provided using Spring[14], a Java framework for the development of enterprise applications. The business objects, i.e., zones, topics, and users are queried from the database and mapped to objects using Hibernate[15], a framework for the storage and retrieval of Java domain objects via object/relational mapping.

We integrated the tool for building and analyzing similarity graphs as new service into the Gugubarra Analyzer as described in detail in [40]. We used the JGraphT[16] Java library because it provides data structures, graph algorithms as well as support for the visualization of graphs. Figure 5 shows the web GUI of the Gugubarra Analyzer with the new service. The web site owner can now analyze her/his community and compute the importance of the users. The most important users are show in the table on the top ordered by the different importance algorithms. On the bottom of the page, the web site owner can choose between different graph visualizations. With the "build graph" button the selected graph representation will be calculated and presented as a static png-image. The "build frame" button will present the graph as Java-applet where the web site owner can order the single vertices manually and adapt the appearance of the graph to her/his needs.

VII. Conclusion and Future Work

With the tool for building and analyzing similarity graphs, we provide a useful service to web site owners for analyzing their web community. We showed with an evaluation the applicability of our approach. We extended the web analytics software Gugubarra with this tool. Now, with the results of graph analysis, we are able to answer the research questions of Section IV:

- Which are the important users of the web community? We provide several algorithms (see Section III-D) to

[11] https://tapestry.apache.org/
[12] http://www.java.com/
[13] HyperText Markup Language, http://www.w3.org/html/
[14] http://www.springsource.org/
[15] http://www.hibernate.org/
[16] http://jgrapht.org/

calculate the important user(s) of the community. The definition of importance is dependent on the used algorithm and on a subjective point of view. For example, vertices with many low weight connections can be considered as the important users of the community. These users are very similar to the other users, expressed by the low edge weight.

- Which users have similar interests?
  All users are connected via weighted edges. Users with similar interests have connections with low weights. The web site owner can also define, which users are connected together by selecting a similarity threshold (see work flow phase one, Section IV-A). As result only similar users are connected via edges.
- How similar are the interests of the users of the web community?
  The weights of the edges of the similarity graph represent the similarity of the users. These weights are calculated with the Manhattan Distance. Therefore, the lower the weights of the edges are the more similar are the users of the community. We give the web site owner the possibility to set thresholds to identify quickly the similarity of her/his community (see Section IV-A).
- How is the web community structured? Is it a homogeneous community where every user has similar interests or is it heterogeneous?
  The visualized graph of the community will give the web site owner an overview over the structure of the whole community of her/his web portal.

With answers to these questions, a web site owner is able to start more focused marketing campaigns. To test new contents or features for her/his web site she/he could start with the most similar users because these users can be considered as an archetype for her/his community.

Besides the extension of the tool with more algorithms for the similarity calculation, in future, the exploration for similarity (or importance) metrics would be helpful. With this type of metrics it would be possible to evaluate the similarity algorithms objectively.

ACKNOWLEDGMENT

We would like to thank Roberto V. Zicari, Natascha Hoebel, Karsten Tolle, Naveed Mushtaq, and Nikolaos Korfiatis of the Gugubarra team, for their valuable support and fruitful discussions. Furthermore, our appreciation goes to Joanna Pieper and Mitra Shamloo for their implementation work.

REFERENCES

[1] C. Schefels, "How to find important users in a web community? mining similarity graphs," in *Proceedings of the First International Conference on Data Analytics (DATA ANALYTICS 2012) / NexTech 2012*. International Academy, Research and Industry Association (IARIA), 2012, pp. 10–17.

[2] "Facebook reports fourth quarter and full year 2012 results," Menlo Park, USA, January 2013, http://investor.fb.com/releasedetail.cfm?ReleaseID=736911, accessed: June 12, 2013.

[3] V. Gundotra, "Google+: Communities and photos," Mountain View, USA, December 2012, http://googleblog.blogspot.de/2012/12/google-communities-and-photos.html, accessed: June 12, 2013.

[4] N. Mushtaq, P. Werner, K. Tolle, and R. V. Zicari, "Building and evaluating non-obvious user profiles for visitors of web sites," in *IEEE Conference on E-Commerce Technology (CEC 2004)*. Washington, DC, USA: IEEE Computer Society, July 2004, pp. 9–15.

[5] N. Hoebel and R. V. Zicari, "Creating user profiles of web visitors using zones, weights and actions," in *Tenth IEEE Conference On E-Commerce Technology (CEC 2008) And The Fifth Enterprise Computing, E-Commerce And E-Services (EEE 2008)*. Washington, DC, USA: IEEE Computer Society, July 2008, pp. 190–197.

[6] M. Trusov, A. V. Bodapati, and R. E. Bucklin, "Determining influential users in internet social networks," *Journal of Marketing Research*, vol. 47, no. 4, pp. 643–658, 2010.

[7] R. Xiang, J. Neville, and M. Rogati, "Modeling relationship strength in online social networks," in *Proceedings of the 19th international Conference on World Wide Web*, ser. WWW'10. New York, USA: ACM, 2010, pp. 981–990.

[8] J. Wang, Z. Chen, L. Tao, W.-Y. Ma, and L. Wenyin, "Ranking user's relevance to a topic through link analysis on web logs," in *Proceedings of the 4th International Workshop on Web Information and Data Management*, ser. WIDM '02. New York, USA: ACM, 2002, pp. 49–54.

[9] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *Journal of the ACM (JACM)*, vol. 46, no. 5, pp. 604–632, September 1999.

[10] I. Esslimani, A. Brun, and A. Boyer, "Detecting leaders in behavioral networks," in *Proceedings of the 2010 International Conference on Advances in Social Networks Analysis and Mining (ASONAM '10)*. Washington, DC, USA: IEEE Computer Society, 2010, pp. 281–285.

[11] G. Paliouras, "Discovery of web user communities and their role in personalization," *User Modeling and User-Adapted Interaction*, vol. 22, no. 1-2, pp. 151–175, 2012.

[12] C. Schefels, "Analyzing user feedback of on-line communities," Ph.D. dissertation, Goethe-University Frankfurt am Main, 2012.

[13] N. Hoebel, S. Kaufmann, K. Tolle, and R. V. Zicari, "The gugubarra project: Building and evaluating user profiles for visitors of web sites," in *HotWeb 2006 - First IEEE Workshop on Hot Topics in Web Systems and Technologies*. Washington, DC, USA: IEEE Computer Society, November 2006, pp. 1–7.

[14] C. Schefels, S. Eschenberg, and C. Schöneberger, "Behavioral analysis of registered web site visitors with help of mouse tracking," in *Proceedings of the 14th IEEE International Conference on Commerce and Enterprise Computing (CEC2012)*. Los Alamitos, USA: IEEE Computer Society Press, 2012, pp. 33–40.

[15] C. Schefels and R. V. Zicari, "A framework analysis for managing explicit feedback of visitors of a web site," in *Proceedings of the 12th International Conference on Information Integration and Web-based Applications & Services (iiWAS2010)*. New York, USA: ACM, November 2010, pp. 481–488.

[16] C. Schefels and R. V. Zicari, "A framework analysis for managing feedback of visitors of a web site," *International Journal of Web Information Systems (IJWIS)*, vol. 8, no. 1, pp. 127–150, 2012.

[17] L. Yi and B. Liu, "Web page cleaning for web mining through feature weighting," in *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI'03)*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2003, pp. 43–48.

[18] C. C. Aggarwal, A. Hinneburg, and D. A. Keim, "On the surprising behavior of distance metrics in high dimensional spaces," in *Proceedings of the 8th International Conference on Database Theory (ICDT '01)*. Berlin, Germany: Springer, 2001, pp. 420–434.

[19] N. Hoebel, N. Mushtaq, C. Schefels, K. Tolle, and R. V. Zicari, "Introducing zones to a web site: A test based evaluation on semantics, content, and business goals," in *Proceedings of the 2009 IEEE Conference on Commerce and Enterprise Computing (CEC2009)*. Washington, DC, USA: IEEE Computer Society, July 2009, pp. 265–272.

[20] S.-H. Cha, "Comprehensive survey on distance / similarity measures between probability density functions," *International Journal of Mathematical Models and Methods in Applied Sciences*, vol. 1, no. 4, pp. 300–307, 2007.

[21] L. Euler, "Solutio problematis ad geometriam situs pertinentis," *Commentarii Academiae Scientiarum Imperialis Petropolitanae*, vol. 8, pp. 128–140, 1736.

[22] R. J. Wilson, *Introduction to Graph Theory*. London, UK: Longman, 1979.

[23] D. Jungnickel, *Graphen, Netzwerke und Algorithmen*, 3rd ed. Mannheim, Germany: BI-Wissenschaftsverlag, 1994.

[24] B. Bollobás, *Modern Graph Theory*, ser. Graduate Texts in Mathematics. Berlin, Germany: Springer, 1998.

[25] M. A. Rodriguez and P. Neubauer, "Constructions from dots and lines," *Bulletin of the American Society for Information Science and Technology*, vol. 36, no. 6, pp. 35–41, August 2010.

[26] S. E. Schaeffer, "Graph clustering," *Computer Science Review*, vol. 1, no. 1, pp. 27–64, 2007.

[27] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," in *Proceedings of the Seventh International World Wide Web Conference (WWW7)*, vol. 30, no. 1–7. Amsterdam, The Netherlands: Elsevier, 1998, pp. 107–117.

[28] D. Nemirovsky and K. Avrachenkov, "Weighted pagerank: Cluster-related weights," in *Proceedings of The Seventeenth Text REtrieval Conference (TREC 2008)*. Gaithersburg, USA: National Institute of Standards and Technology (NIST), November 2008.

[29] J. M. Pujol, R. Sangüesa, and J. Delgado, "Extracting reputation in multi agent systems by means of social network topology," in *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS '02): Part 1*. New York, USA: ACM, 2002, pp. 467–474.

[30] P. Jaccard, "The distribution of the flora in the alpine zone," *New Phytologist*, vol. 11, no. 2, pp. 37–50, February 1912.

[31] G. Csardi, *Network Analysis and Visualization*, 0th ed., http://igraph.sourceforge.net/, August 2010, package 'igraph'.

[32] A. Barrat, M. Barthèlemy, R. Pastor-Satorras, and A. Vespignani, "The architecture of complex weighted networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101, no. 11, pp. 3747–3752, 2004.

[33] L. C. Freeman, "Centrality in social networks conceptual clarification," *Social Networks*, vol. 1, no. 3, pp. 215–239, 1978.

[34] P. Bonacich, "Power and centrality: A family of measures," *American Journal of Sociology*, vol. 92, no. 5, pp. 1170–1182, March 1987.

[35] P. Domingos and M. Richardson, "Mining the network value of customers," in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, ser. KDD '01. New York, USA: ACM, 2001, pp. 57–66.

[36] J. Marcus, "Rgraphviz," Presentation, 2011, http://files.meetup.com/1781511/RgraphViz.ppt, accessed: June 12, 2013.

[37] L. A. Adamic and E. Adar, "Friends and neighbors on the web," *Social Networks*, vol. 25, no. 3, pp. 211–230, July 2003.

[38] R. C. Prim, "Shortest connection networks and some generalizations," *Bell Systems Technical Journal*, pp. 1389–1401, November 1957.

[39] N. Hoebel, "User interests and behavior on the web: Measurements and framing strategies," Ph.D. dissertation, Goethe-University Frankfurt am Main, 2011.

[40] M. Shamloo and J. Pieper, "Ähnlichkeitsgraphen und wichtige nutzer einer web-community," bachelor thesis, Goethe-University Frankfurt am Main, December 2012.

End-user Facilitated Interoperability in Internet of Things

Visually-enriched User-assisted Ontology Alignment

Oleksiy Khriyenko, Vagan Terziyan, and Olena Kaikova
IOG group, MIT Department and Agora Center
University of Jyväskylä, P.O. Box 35 (Agora)
FIN-40014 Jyväskylä, Finland
e-mail: oleksiy.khriyenko@jyu.fi, vagan.terziyan@jyu.fi, olena.o.kaikova@jyu.fi

Abstract — **Nowadays, we make a separation between the real/physical world and the Internet. It is time for these two be blended and provide ubiquitous access and interoperability online. We are approaching Internet of Things - a forthcoming technological revolution that will radically change our environment and enable innovative applications and services. To make this happen, we have to eliminate the fragmentation in used technologies and have to make the devices be used across various applications and services. We need to find a way to actually carry out the necessary and massive deployment of ubiquitous devices. So we need to put more effort into the design of tools to automate deployment and configuration of devices. This paper tackled a problem of an effective way to support interoperability in Internet of Things. We consider human as a very powerful asset in the world of ubiquitous systems and services that may provide his/her knowledge, experience and expertise. At the same time, we see a lack of human-oriented systems and infrastructures to support such a new role of a human. With a respect to the above statements, authors propose visually-enriched approach for user-powered ontology alignment to facilitate semantic interoperability in the Web of Things.**

Keywords- Mashup supported semantic visual mapping; visual ontology alignment, visual semantic human interface, semantic interoperability.

I. INTRODUCTION

This paper is an extension of the original paper [1] that has been presented at the international conference UBICOMM-2012. Here, authors extend the paper with more details regarding required semantic language extension for visually-enriched ontology and resource description, and present browser for visually-enriched linked data.

Our current Digital World is changing rapidly. We are about to enter a new era of ubiquitous computing and communication that will radically transform our corporate, community, and personal spheres. Tomorrow's world of Ubiquitous Pervasive Computing and Internet of Things is a technological revolution that represents the future of computing and communications and its development depends on technical innovations in a number of important fields. In the interconnected world of computers, interactions occur, not only between humans and applications, but also between applications of various kinds, applications and equipment, low-level software units, or any other logical or physical entities. Unlimited interoperability and collaboration are important values for a multitude of areas in our daily life.

With a purpose to better understand a need of proposed contribution, and highlight possible requirements and characteristics of interoperable systems, let us start with some short samples of use case scenarios:

Scenario 1: Person is traveling by a car. Suddenly, something is happened with the car and it needs to be repaired. Instead of searching the nearest car service station, booking a time and filling a request form describing current state of the car; the car itself searches for correspondent services in the web, collects necessary data from the correspondent modules of the car and books a time for maintenance service. During the maintenance, the car gets new spare-parts and integrates them to the central diagnostic system of the car (regardless of the fact that parts are produced by different vendors). In the same manner, car might negotiate and book appropriate time for annual technical check-up taking into account timetable of the owner, been connected to his/her personal organizer. During the trip, car might suggest optimized schedule of refueling taking into account fuel consumption, location of gasoline stations and their prices, discounts and bonuses available for the driver and other relevant contextual information.

Scenario 2: Person has bought "smart-home" system from some vendor. Vendor installs smart-home network with a set of smart-entities (sensors and actuators) and one control unit. So far, all the elements of the network belong to the same vendor and interoperate via the same ontology and communication protocol. A couple of month later, house owner buys a new smart-entity for good price from another vendor and connects it to the existing network. Later, friend of the house owner suggests some generic software application, which could be used as an upgrade of the smart-home network control unit and provides new useful features in comparison to the functionality of initial software of the control unit. This software application is produced by totally different vendor, and still can be installed to the control unit and communicate with all the connected to the smart-home network entities.

Scenario 3: Person has several measurement units (produced by different vendors) that can measure his/her heartbeat rate, arterial pressure, distance person walked or run, and some other parameters related to his/her health condition and physical activities. Person easily connects all these devices to a smart-phone to be able to log and observe them. Later, from an app. store, person buys application that suggests correspondent diet, taking into account all the measured personal data. Entering a supermarket and to be connected to the local infrastructure of it, application starts to navigate person to the correspondent location of the suitable products for his/her diet or alert the person when he/she puts to the basket a product which consists inappropriate ingredients.

All mentioned above stories are not fantasies. It is our tomorrow and, in some cases, even our today. Unfortunately, in case of our today, we have integration of systems produced by the same vendor. Supporting one interoperability model in several products, vendor creates integrated environment for various applications and interaction scenarios to be run on it. All these applications should support correspondent predefined API and data model. But, it is not what we expect to be in our tomorrow. We need an open environment with possibility to integrate various systems and components (hardware, apps, communication channels, etc.) produced by different vendors (see Figure 1). With a goal to achieve such requirements, we are approaching Internet of Things - a forthcoming technological revolution that will radically change our environment and enable innovative applications and services.

Above the personal level, the IoT will also have an important impact on enterprises and on society in general. IoT will enable a global connectivity between physical objects (connecting "things", not only places or people), will bring real-time machine-published information to the Web, as well as will enable a better interaction of people with the physical environment by combining ubiquitous access with ubiquitous intelligence. IoT will consist of a heterogeneous set of devices and communication strategies between them. Such a heterogeneous system should evolve into a more structured set of solutions, where "things" are uniformly discoverable, enabled to communicate with other entities, and are closely integrated with Internet infrastructure and services, regardless of the particular way (RFIDs, sensors, embedded devices) in which they are connected to the IoT. In this context, one of the challenging bottlenecks is to support interoperability between "entities" on a semantic level [2][3].

Taking into account current state of the art in the field of innovative research and development, we see lack of human-oriented systems and services. Now, human becomes very dynamic and proactive resource of a large integration environment with a huge amount of various heterogeneous data and services. As a user, human requires a technology and tools for easy and handy information access and manipulation. Moving from industrial era to the era of ubiquitous services, human is not considered only as a user/consumer any more. Human becomes a service provider, an expert that provides her/his knowledge and expertise to the digital world. Transition from an industrial welfare society to sustainable human-centric services society requires elaboration of correspondent infrastructure and tools to allow people add value to the process. Therefore, in this paper we propose an approach towards visually-enriched semantics as an infrastructure for user-powered semantic technology enhancement. The main contributions of the paper are visually enriched ontology and a system that visually assist users to execute semantic alignment.

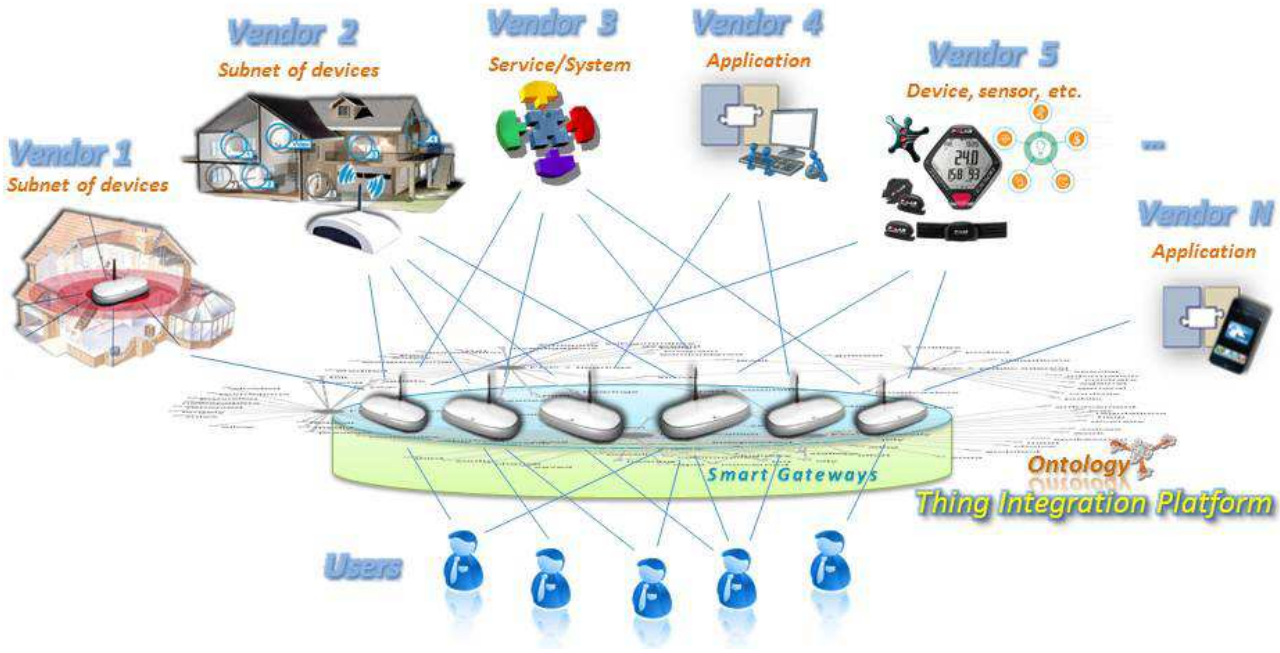


Figure 1. Thing Integration Environment.

Paper consists of two main sections: Section II and Section III. Section II addresses semantic integration platform for IoT and a vision of user-powered consumption of semantic technologies. Section III presents visually-enriched approach for user-powered ontology alignment to facilitate semantic interoperability in the Web of Things. Section IV presents author's conclusions with respect to the proposed contribution and defines future work.

II. THING INTEGRATION ENVIRONMENT

A. Smart Gateway - semantic integration platform for IoT

We are already in the middle of era of automated machine communication. There is already a lot of machine-to-machine communication going on out there; parking meters are connected, and vending machines automatically report when new supplies are needed. Every minute huge amount of data are being exchanged between machines for various purposes within various sectors. However, there is a big challenge in moving beyond application-specific devices and establishing an information model that will create re-use of the data generated by devices for new applications in different domains.

Finding the right horizontal points in the solutions is a key. There are already useful deployments within the transport, automotive, building, health and utility sectors, but everything is still very sector-specific. We need to create an infrastructure that will make information generated from a car or a building understandable not only within their own specific application/system, but across of various applications and domains. The vision of an open and interoperable IoT implicates ability:

- to have a growing environment with possibility to install and interconnect all IoT devises and software (services and applications) on the fly;
- to interconnect devises produced by different vendors;
- for third parties, to elaborate generic applications and services for IoT environments in the sense of applying them on various IoT device sets (same purpose, but different vendors).

To satisfy such requirements, IoT will require interoperability at multiple levels and rely on the benefits of the semantic technologies. On the hardware side, such problems have to be addressed as handling a capability mismatch between traditional Internet hosts and small devices, as well as handling widely differing communication and processing capabilities in different devices. In the interface between the device and network domains, IoT gateways will provide a common interface towards many heterogeneous devices and networks [4]. We assume that all "things" (devices, sensors, actuators, etc.) are connected to the web. Digital "things" such as services usually are accessible through the web. Applications might be downloaded and installed to the integration platform - Smart Gateway. Thus, we have correspondent requirements for such a platform. Smart Gateway should allow installation of applications and further configuration of communication model with it, based on accompanied annotation of the application. In case of services, Smart Gateway should be able to access semantic annotation through service access point and configure communication model with it as well.

Talking about physical world objects (device, sensors, etc.), usually they are accessible through the gateway - a control unit of a network provided by the same vendor. The only requirement - gateway should be presented in the web as a service with a set of capabilities provided by "things" connected to the gateway (providing data or doing some actions). In case we cannot have single "thing" personally connected to the web, we should deal with sub-network that consists of mentioned "thing" and correspondent gateway. Thus, we will have a set of gateways connected to the web and ready to become a part of the integration environment. Having all the gateways accessible as web-services, all connected to them real world "things" become digital entities and might be registered to the Smart Gateway.

Depending on a business model, Smart Gateway might be a part of a gateway, provided by certain vendor. In such a way, vendor can promote own network solution as an extendable open environment that support connectivity and interoperability of various entities produced by other vendors. Having Smart Gateway as a part of local network of connected "things", services and applications is a reasonable model in case of time-limited and highly secured runtime systems. At the same time, Smart Gateway might be considered as a separate integration solution - services located in the Cloud and accessible through the web. Been easily accessible, such "thing" integration service might be very popular among ordinary people who would like to create and manage their own distributed smart spaces, integrate various services with ubiquitous "things". Relevant research has been done in "Smart Resource" and "UBIWARE" Tekes projects with respect to Global Understanding Environment (GUN) [5].

B. User-powered consumption of semantic technologies

To achieve the vision of ubiquitous 'things', the next generation of integration systems will need different methods and techniques to provide connectivity, interoperability and intelligence of distributed entities, as well as smart and intuitive mechanism of communication with a human. Among those there are technologies such as Semantic Web [6][7], Web Services [8][9], Mashups [10], Linked Data [11][12], etc. To integrate 'things' seamlessly with the existing Web infrastructure and to represent interconnected 'things' uniformly as Web resources, resulting Web of Things (WoT) is a good facilitator of interoperability. Making devises able to unambiguously exchange the meaning of data, Semantic Web technology can be used to extend WoT into Semantic Web of Things (SWoT).

Semantic based technologies are viewed today as key technologies to resolve the problems of interoperability and integration within the heterogeneous world of ubiquitously interconnected objects and systems. Semantic Web is a vision with an idea of having data on the Web defined and linked in a way that it can be used by machines not just for display purposes, but for automation, integration and reuse of data across various applications. Semantic Web is considered

as a standardized approach to achieve automated interoperability of heterogeneous systems/applications. Heterogeneity of systems and various data sources become a bottleneck for automated service integration, data processing and reuse. To make data ready to be consumed and processed by external systems, data sources and data should pass through the semantic adaptation [5][13] and be accessible in common uniform way. Due to the huge amount of application areas that Semantic Web technology tried to cover, community started to elaborate different standards and techniques to solve interoperability problems. As a result, we have a big variety of separated islands of information and management systems. These information islands internally follow the Semantic Web vision, but are heterogeneous from the general (global) interoperability point of view. This leads to the fact that society and especially its business-oriented part has started to doubt that such widely spread activity will be so much beneficial for them. Only some applications and systems in restricted domains became really useful. Most probably, the reason for this is the decentralization of uncontrolled activities, which creates new problems on the way towards ubiquitous Semantic Web. There are no doubts that Semantic Web is a very promising technology, but it definitely lacks more smart management or at least an environment that plays coordinative and supportive role and directs users towards proper technology utilization.

Services providers, as well as producers of "things", are the end users of the service-oriented technologies. They need appropriate controlled support from the infrastructure that facilitates interoperability of services/devices, integration of heterogeneous data sources, and provides platform for new services/application development. Thus, we have to provide such a coordinative and supportive environment that will facilitate development and growth of service and smart-entity market. With respect to the current state of the art, we cannot expect that community of service providers and smart-entity vendors will build one global integration infrastructure with common ontology. We cannot expect that someone else (alone or in a consortium) will do the same. Current achievements in the area of interoperability of heterogeneous systems present technologies and tools for experts to build and manage adapters between heterogeneous systems or their components. Semantic Web is a technology for machines to better perform, providing services for human in automated or semi-automated way. In a case of unavailability of a common data model, we have to deal with semi-automated performance of the system when human become involved to the process not just as a consumer, but as an expert - necessary part in the chain to supervise and correct the process performed by machines [14].

With an increase in the development of ontologies, we need tools and techniques for solving heterogeneity problems between different ontologies. Therefore, we need ontology alignment [15][16][17][18], which helps us to bring different knowledge representations into mutual agreement. With respect to the scenarios mentioned above, ontology definitions of all the smart-entities and applications/services should be (semi-)automatically aligned by control unit of the network to ensure interoperability of them in a unified way. Regarding to the mentioned ontology alignment techniques, we may expect automatic alignment for simple and similar ontologies, but in all other cases, we will definitely need a human be involved into the process. This is largely a human-mediated process. There are existing tools that can help with identifying differences among ontologies [19], but user interaction is still essential in order to control, approve, and optimize the alignment results.

Unfortunately, approaching the era of ubiquitous services and IoT, we cannot expect availability of huge amount of professional experts involved to the daily processes of "things" interoperability support. We have to find a solution to bring technology closer to the ordinary user and make him/her able to not only utilize services, but to setup, configure and supervise interoperability process. We expect a human to be not only an end-user/consumer of technology world, but also to become an integral part of it, providing own expertise and capabilities. In all mentioned scenarios, person (owner of the smart-network) should be able to help the system to perform a proper ontology alignment through correspondent human interface of the alignment system. Owner of interoperable system does not only consume a service delivered by smart machines, but also plays a valuable role as a supervisor of interoperability process. Therefore, among variety of other adapters between heterogeneous entities, bridge to the human (human-to-machine H2M and machine-to-human M2H interfaces) becomes one of the most important tools of next generation integration infrastructures.

Such an adaptation of the human to the technology world might be provided by Personal Assistant (PA) - supportive agent assigned to every user [14]. From one side, it should deal with human personality and adapts to his/her personal ontology and personal perception of environment. From another side, it should support common semantic standards and approach to be interoperable with other surrounding digital world entities (applications, services and systems). The main features of PA (among others) are:

- Enabling personal user ontology creation and ontology driven resource annotation;
- Ability to adjust to the personal user ontology, to the way user perceives the environment, information and knowledge;
- Ability to build personalized semantic mind-map based on user behavior and preferences;
- Enabling personalized natural user-driven way of querying, filtering, browsing and presentation of information.

Personalized representation of information very much concerns a human supervised ontology alignment process. Ontologies very much differ from each other. The more specific, detailed and complex ontology we make, the more semantic value it has, but, it makes harder to integrate ontology with others. Taxonomies of different ontologies are not likely to be the same. Even developed by professionals, we still have different ontologies for the same problem domain. It would seem that experts, involved to the same domain, should operate with the same terms, use the same vocabulary and knowledge representation model. But, people

are different, context and personal perception of surrounding world brings problems to interoperability process. As a part of the processes, human brings a certain level of uncertainty, and only human my help to solve the problem so far. Thus, to avoid heterogeneity in the resource annotations and simplify ontology alignment for automated interoperability between digital elements of the technology world, we may admit a necessity of personalized adaptation of every human (no matter whether it is an expert (knowledge provider) or user/customer) to the common information/data model.

In the next section we present an approach towards visually-facilitated human-assisted ontology alignment for automated interoperability among various heterogeneous entities of IoT. This approach supports the idea of end-user involvement as a powerful intelligent entity if IoT.

III. HUMAN-ASSISTED VISUAL ONTOLOGY ALIGNMENT

A. Visually-facilitated semantic matching

Let us consider a scenario of installation of a new floor-heating regulator to a "smart-home" system. Assuming that we have two different vendors (Vendor A - producer of the Control Unit for the smart-home system, and Vendor B - producer of the regulator for a floor-heating system), we have two different ontologies $Ontology^{V}_{A}$ and $Ontology^{V}_{B}$. Vendor A logically defines all the floor-heating systems with respect to the room the system is associated with. Thus, $Ontology^{V}_{A}$ might contain such concepts as: living room floor-heating system, bedroom #1 floor-heating system, bedroom #2 floor-heating system, kitchen floor-heating system, bathroom floor-heating system, etc. From the Vendor A point of view, all these concepts refer to absolutely different sub-systems in the "smart-home" network. On the other side, association of the floor-heating system with particular room/place does not matter for Vendor B. Therefore, "floor-heating system" concept in the $Ontology^{V}_{B}$ is a more general and independent entity. Moreover, most probably "floor-heating system" concept will be named very much different in those two ontologies and automated alignment will be absolutely impossible.

Since the Control Unit of the smart-home is a more general device (in comparison to specific Floor-heating system) and deals with many other devices and systems in the installed network, it utilizes more wide ontology. Therefore, to allow interoperability between the Control Unit and Floor-heating system, we have to map $Ontology^{V}_{B}$ to wider $Ontology^{V}_{A}$. At the same time, we have to pay attention to the user's ("smart-home" owner's) $Ontology^{H}_{i}$. In general case, every human has own personal ontology that will be supported by his/her PA for interaction with devices, services, applications and systems. But, for any system/application, to be a mediator between the human and some other system with its own ontology, personal human ontology itself should be mapped with ontology of mediator-system in advance. PA will collect correspondent alignments of personal human ontology with ontologies of various mediating systems that human will be interacted with.

Assuming that fully automated alignment is not possible, we do not consider the cases with very simple and self-descriptive ontologies, where automated alignment might be done based on matching of synonyms of the property manes. Correspondent example of the research at this direction is a work performed in the Tivit SHOK IoT project (funded by Tekes, Finland), where authors are trying to minimize human involvement to the process of establishing interoperability between heterogeneous systems [4]. They try to retrieve (to build) ontologies from examples of massages that systems operate in communication process (requests, response, etc.). Authors build simple plane ontologies based on names of parameters used in the messages. Later, ontologies are automatically aligned and correspondent alignments are used for automatic interoperability between heterogeneous systems in runtime. But, as was mentioned, it might work in case of self-descriptive messages, where parameters are named by words that make sense, without abbreviations and shortenings, and preferably in the same language. In all other cases (cases with complex hierarchy of sub-classes, cases of different domain description models, cases of multilingual and multicultural ontology definitions, etc.), this would not work automatically and will require human assistance. Thus, in cases of human-assisted alignment of personal human ontology or ontologies provided by different vendors, we need an innovative suitable for non-expert mechanism and correspondent user interface for ontology alignment.

With respect to the research [20][21][22][23], there are some available Ontology Alignment and visualization tools: Foam algorithm [24], multiple-view plug-in for Protégé [25] - AlViz [26], BLOOMB system [22] and Knowledge Modeler[27]. The very good overview of visually supported ontology alignment tools is presented in the paper [28]. There graphical primitives such as point, line, area, or volume are currently utilized to encode information. These objects are characterized by position in space, size, connections & enclosures, shape, orientation, and visual cues like color and texture, with temporal changes, and viewpoint transformations. Unfortunately, all these tools were elaborated for domain experts who know what ontology is and what information models might be used. Such tools present a lot of statistical data and analytics that might be very useful for the ontology engineer, but not for the ordinary user of a service. Information visualization should aim at making complex data easy accessible and understood for interactive investigation by the user. In case of smart-home, we expect that user has a basic knowledge about a domain and functionality of the system. Like in the scenario above, house owner knows already available sub-system of his/her smart-home, their purposes and functionality, as well as he/she knows a purpose of the new parts that he/she wants to be added to the system. Therefore, we have to find more suitable approach for user-assisted ontology alignment.

To be easily recognized by human, concepts and properties of different ontologies must be presented in the most understood form - in a form of image. An image (or other visual form) is the most common information representation model for human. It helps to understand the meaning and avoid verbal uncertainty presented in textual form. Therefore, user interface should be able to present semantics through interactive image mash-ups and user-

friendly browsing mechanism. Talking about data visualization, we would like to admit existence of some domain-oriented software applications, which try to visualize data in domain specific and suitable for human way (graphics software from SmartDraw®, concept-browser Conzilla and Human Semantic Web browser Conzilla2, etc.). But still, they are developed for specific standalone domain-oriented applications. And when we face a real need in an open unlimited collaboration environment, we have to develop much more visualization tools and modules that are aimed to visualize various resource properties, contexts, situations and associations to provide human flexible and handy Human-Machine interaction interface. Thus, semantic-based context-dependent multidimensional resource visualization approach and 4i (FOR EYE) technology [29][30][31] can be a basis for the development of such interface. The idea of intelligent resource visualization is to simplify the search and browsing processes via associative resource visualization. Multidimensional associative resource visualization means visualization of a resource depending on a context, via association with various aspects of resource being (relations with the other resources, domains, areas of interest, etc.). Sometimes, we cannot specify exactly what we are looking for, but we feel that it is somehow related to certain stuff, certain situation, certain context. Such visualization can give us a hint, turn to the right direction, show us related objects and provide links to them. In other words, visualization will utilize context-based filtering and enrichment of the visualized scene with the relevant links. Such approach provides an opportunity to create intelligent visual interface that presents relevant information in more suitable and personalized for user form. Context-awareness and intelligence of such interface brings a new feature that gives a possibility for user to get not just raw data, but required information based on a specified context. Now it has become evident that we cannot separate visual aspects of both data representation and graphical interface from interaction mechanisms that help user to browse and query a data set through its visual representation.

Figure 2 shows us possible visual interpretations of the Vendor A, Vendor B, and user (smart-home owner) ontologies with respect to the scenario of adding the living room floor heating system to the “smart-home” network. Since we are not consider “smart-home” owner as an expert in ontologies and complex control systems, we cannot expect that it would be possible for him/her to utilize currently available solutions for ontology alignment. Only we can expect is awareness of the user about purpose, capabilities and main functionality of the “smart-home” Control Unit and floor-heating system that he/she would like to add to the “smart-home” network. Having even such limited expertise of the problem domain, user is able to browse visual description of the Control Unit (the structure of sub-systems,

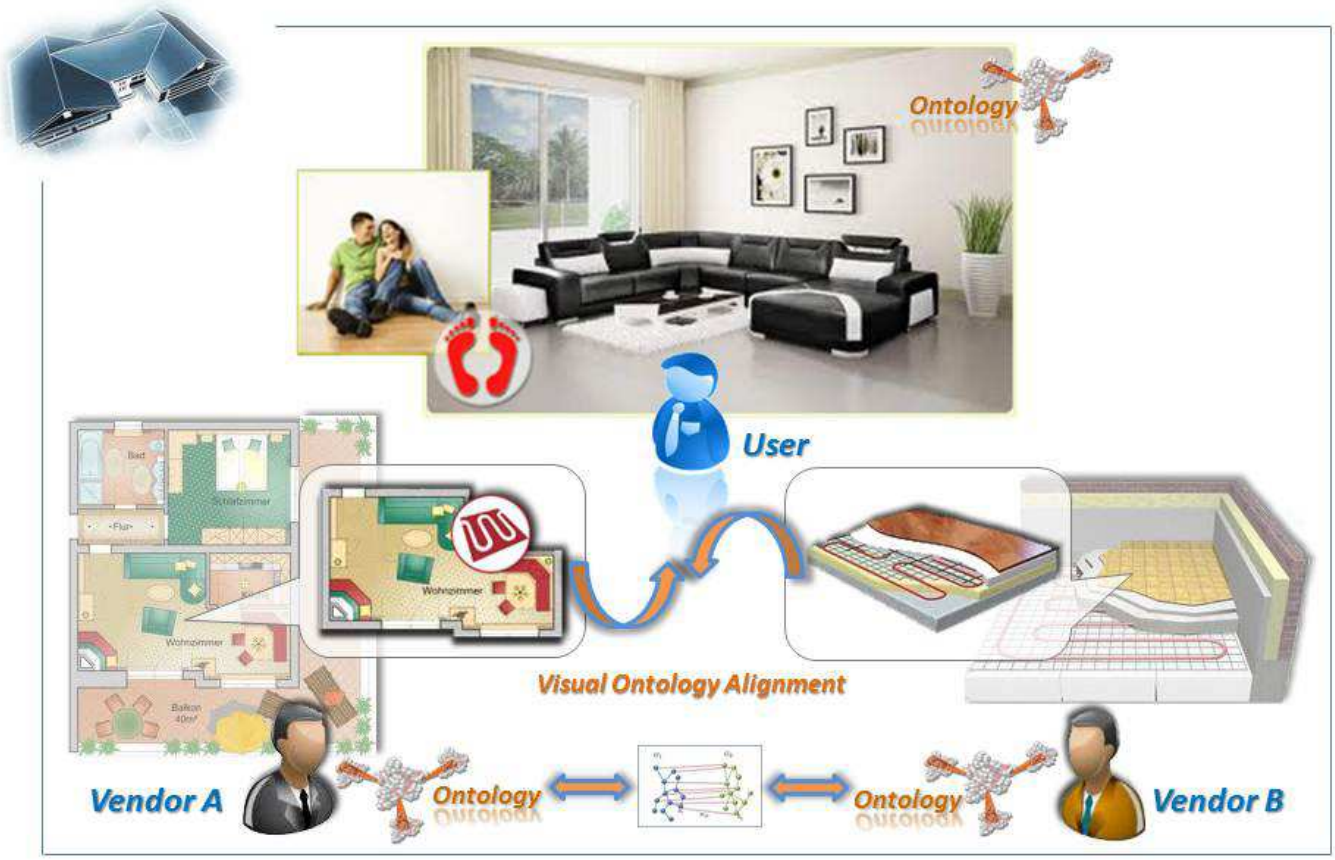


Figure 2. Visually-facilitated Ontology Alignment.

capabilities, inputs and outputs, properties, etc.) and description of the floor-heating system from another vendor to provide appropriate matching. User can intuitively map concepts and properties presented by images. Applying possible results, achieved by integrated modules of automated ontology alignment (as a background process), Visual Ontology Alignment Tool assists user with suggestions and requests next necessary alignments caused by alignments made on the previous steps. Additional textual descriptions of visual annotations support user to make correct mapping. As a result, correspondent parts of ontologies OntologyVA and OntologyVB, which are related to the communication scenario between "smart-home" Control Unit (Vendor A) and floor-heating system (Vendor B), will be mapped and correspondent alignment will be used in runtime operation of the "smart-home" network.

B. Visually-enriched ontology and resource description

To operate with visual representation model in a smart way, visualization tool should retrieve correspondent images together with ontologies. It means that ontologies should be extended with additional layer that contains visual definition of the concepts (classes and properties). Later in the text, we will call such visually-enriched ontology as Visual Ontology (VisOntology). We consider two scenarios of human-assisted visual enrichment of data (see Figure 3). In the first case, Ontology/Domain Expert creates VisOntology using Ontology Visual Enrichment Tool that adds correspondent image layer to the ontology. Later, Vendor provides annotation of the Service/System that was produced by Vendor. In the second case, Vendor itself provides visually-enriched annotation (VisAnnotation) of the produced Service/System using Visually-enriched Resource Description Tool based on regular domain ontology provided by Ontology/Domain Expert. In this case, visually-enriched ontology might be automatically created from the visually-enriched resource description during the annotation process. In case when it is difficult to associate any image with some of the concepts, tool will create an image with a correspondent text (word, character, sign, etc.) retrieved from the name of ontology element. One more scenario might have a place if we consider possibility for some third party to substitute Vendor in the Service/System annotation process and provide visual annotation in both previous cases. Both tools that were mentioned in the above scenarios have the same nature and similar functionality. Thus, let us consider them as a single tool for visual semantic enrichment.

The best way to provide visual layer to the ontology is to extend existing ontology editors (the most popular - Protégé ontology creation and editing tool) with possibility to assign appropriate visual element to every entity of ontology (class and property). Talking about instance annotation process, visually-enriched description might be performed via various RDF creation tools. It might be Protégé as well as any other more customized RDF creation tool which is more suitable for specific application domain. The main purpose of the tool is to help user to brows ontology and assign "visSemantics" property to every entity of ontology: class, property and instance. Taking into account formal aspects of a visual layer in ontology definition and resource description, we have to extend the RDF Schema (RDFS) [32] and Web Ontology Language (OWL) [33]. Figure 4 presents possible extension of RDFS or OWL with "visSemantics" property used for VisOntology and VisDescription. Talking about resource annotation, tool creates an annotation template based on assigned ontology and provides possibility to add visual description. In such a way, tool extends the concepts of the ontology with "visSemantics" property and correspondent value in a form of image. Currently we consider the range of this property as a literal URL of an image. In more advance version of VisOntology and VisDescription of the resource, the range might be extended to video, audio or any other multimedia content. Such extended range of the property might be regulated by rich datatypes that will restrict the type (file extension) of the file mentioned as a Literal value of the visSemantics property.

Visual enrichment is individual, as long as a set of images, used by VisOntology and VisAnnotation providers, is individual. Tools should allow user to make key-word based image annotation/tagging for individual content and create a personal pool of annotated visual content for further reuse. Later, annotated content (used for visual enrichment)

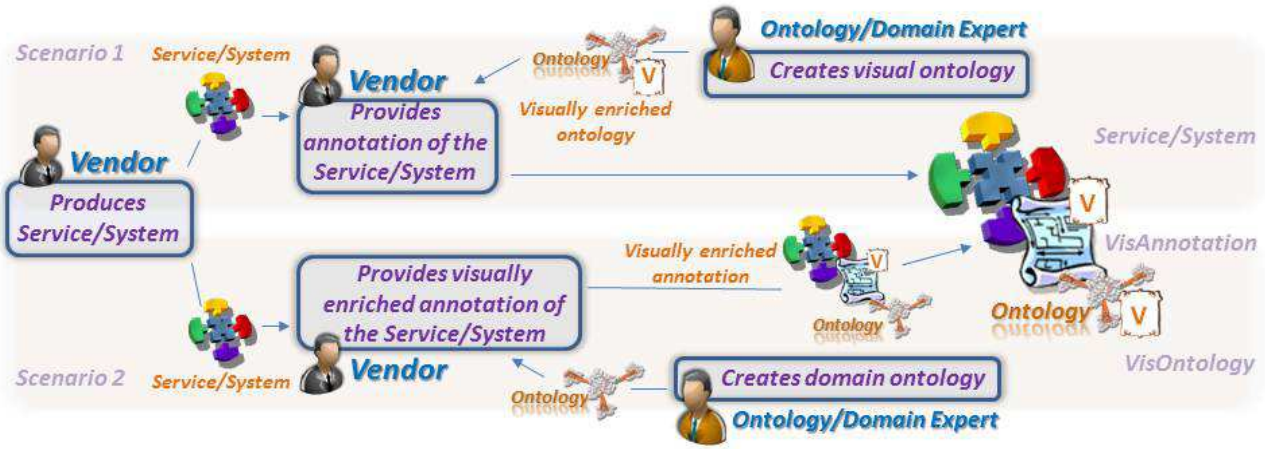


Figure 3. Human-assisted visual enrichment scenarious.

will be easily retrieved based on user search request or automatically suggested to a user based on attributes of self-descriptive elements of ontology. In case of old-fashioned service/system description which is based on ordinary ontology, enrichment of the description might be still automated on some extend. Based on the names of ontology concepts, Visual Enrichment Tool may search among visual content (annotated already), and build visual layer automatically. Quality of automated enrichment might be relatively low in comparison to human assisted enrichment. But, even in worst cases, when we do not have any human involvement at the stage of resource annotation, it might help to retrieve at least some visual content for further visual ontology alignment process. Taking into account growing trend towards sharing and reuse of content, annotated visual content might be shared through various clouds and common spaces. Thus, tool can use not only personal visual content of the user, but also will allow to manage and extend his/her virtual visual content space with external sources. In this case, Social Web might be considered as a good platform to share visual annotation content and VisOntologies.

Assuming that it might be not so popular for vendors to provide visual description manually, visually-enriched ontologies might become popular. Already having visual layer imbedded into ontology, resource annotation tool will suggest correspondent visual entity with respect to the class of annotated instance. Responsible for resource annotation expert might just simply accept such proposed visualization or provide customized visualization more suitable in particular context. Every time when expert select customized visualization, correspondent visual entity might be shared and will extend ontology with extra visual definition of correspondent class (or property). Multiple visual annotations of the classes and properties will enhance semi-automated resource definition process. Several appropriate visualization options will be proposed to annotation expert. To avoid redundancy of alternatives, they might be filtered based on automatically detected or specified (by annotation expert) context. This will require context definition for each visualization entity in the ontology (see Figure 4). In this example, we may see the definition of the contexts for two different images assigned to the HeatingSystem class (class of various heating systems). Context definition might be perform via reification mechanism and be defined either by context definition keyword(s) (via "visContextKeyWords" property) or by instance of the context definition class (via "visContext" property). Thus, the domain for both mentioned properties in rdf:Statement. In the example, we may find two different methods for applying reification mechanism. One of them uses abbreviation "{}" supported by Notation-3 [34] serialization to define a statement. Another method uses standard approach of RDFS via definition of an instance of rdf:Statement class. The range of the "visContextKeyWords" and "visContext" properties is different. Range for the first property is restricted by rdfs:Literal class and considered to be referred to keyword (key-sentence). At the same time, range of another property refers to the class of context definitions. In this paper we do not concentrate our attention on the context definition class.

```
@prefix :    <http://www.example.org/sample.owl#>.
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix rdfs:<http://www.w3.org/2000/01/rdf-schema#>.
@prefix owl: <http://www.w3.org/2002/07/owl#>.

rdfs:visSemantics
     rdf:type rdf:Property;
     rdfs:domain rdfs:Resource;
     rdfs:range rdfs:Literal.

owl:visSemantics
     rdf:type owl:DatatypeProperty;
     rdfs:domain rdfs:Resource;
     rdfs:range rdfs:Literal.

owl:visContextKeyWords
     a owl:DatatypeProperty;
     rdfs:domain rdf:Statement;
     rdfs:range rdfs:Literal.

owl:visContext
     a owl:ObjectProperty;
     rdfs:domain rdf:Statement;
     rdfs:range owl:ContextDefinition.

:HeatingSystem a rdf:Class .

owl:ContextDefinition a rdf:Class .

#---some definition of the context ---
:carHContext a owl:ContextDefinition .

#-------------------------------------

{:HeatingSystem rdfs:visSemantics
"www.example.org/FloorHSystem.jpeg"}
     owl:visContextKeyWords
     "floor heating" ,
     "home heating" ,
     "room heating" .

{:HeatingSystem rdfs:visSemantics
"www.example.org/CarHSystem.jpeg"}
     owl:visContext :carHContext .

:statement_1 a rdf:Statement ;
     rdf:subject :HeatingSystem ;
     rdf:predicate rdfs:visSemantics ;
     rdf:object
   "www.example.org/CarHSystem.jpeg" .

:statement_1 owl:visContextKeyWords
     "car heating" ,
     "vehicle heating" .
```

Figure 4. "visSemantics" ontology extension and visualization context definition.

There are several approaches towards context definition, but in this paper we are just assuming an existence of some context definition class (in our example - "ContextDefinition" class).

Sub-class and sub-property hierarchy of the ontology also can help to collect visualization alternatives and automate resource annotation process. All the visualization entities that describe sub-classes and sub-properties also describe supper-classes and supper-properties. Even if certain class does not have any visual description/representation, the list of alternatives to describe an instance of that class will be retrieved from the sub-classes and the same context-based filtering technique might be applied to find more appropriate visualization. Thus, more detailed ontology with deep sub-class hierarchy might minimize amount of multiple context-dependent visualization settings. If we reconsider example in Figure 4 and define floor heating system and car heating system as sub-classes of more general class that defines all heating systems, then "HeatingSystem" class might be described by some general visualization and additional alternatives might be collected form visual descriptions of its' sub-classes.

C. *Browsing of visually-enriched linked data*

Browsing of visually-enriched ontologies and visualization of visually-enriched resource descriptions might be performed by appropriate visualization tools. As was mentioned before, it is reasonable to have visualization of ontologies imbedded into ontology editors. At the same time, visually-enriched resource descriptions must be visualized in more intuitive way via smart integration of data mashups [29][30]. In this project, we have been used 4i (FOR EYE) Browser [31] - smart visual context-sensitive resource browser (elaborated in UBIWARE Tekes project) and Linked Data Browser (elaborated in this project for visualization of visually-enriched resource descriptions) (see Figure 5). 4i (FOR EYE) is an ensemble of Intelligent GUI Shell (smart middleware for context dependent combination of different MetaProviders) and MetaProviders, visualization modules that provide integration and context-dependent filtered representation of resource data. Context-awareness and intelligence of such interface brings a new feature that gives a possibility for user to get not just raw data, but required integrated information based on a specified context.

Figure 5. Browsing of visually-enriched linked data.

The Figure 5 shows us example of Linked Data browsing with respect to the heating systems imbedded into smart-home described in one of the scenarios in the beginning of the paper. Based on the resource description file that contains visually-enriched resource descriptions, one of the visualization modules of the 4i Browser has been used to present us all the heating systems available in the house. Another visualization tool (Linked Data Browser) has been used to browse the RDF graph from the same resource description source and show us certain smart-home with correspondent smart-home heating system that consist of four sub-heating systems for living room, bedroom, balcony, and bathroom.

IV. Conclusion and Future Work

Approaching the era of people-oriented systems, human becomes very dynamic and proactive resource of a large integration environment with a huge amount of different heterogeneous data. People are great asset to be utilized in servicing and services creation. Involving people to the process, we allow them be not only a user, but also add value to technology evolution.

With the aim to elaborate an environment that enables integration of heterogeneous "things" and intelligent distributed systems within the Internet of Things framework, authors address the mechanism of human-assisted simplification of semantic matching to allow interoperability of entities in the IoT. Assuming unavailability of a sufficient amount of professional experts to be involved to the daily "things" integration support process, authors proposed the way to make user be not just a consumer of thing-based services, but also an expert capable to compose and establish interoperability among the things. Taking into account specifics of the potential user and unsuitability of current ontology alignment tools for it, this paper presents a human-driven approach towards visually-facilitated ontology alignment through visually-enriched ontologies and resource (thing) descriptions. Authors have presented extension of RDFS and OWL ontologies to enable creation of visually-enriched ontologies and resource descriptions. Current implementation of correspondent toolset is concentrated on and consists of an interface for the final stage - Visual Ontology Alignment Tool that assumes existence of VisOntologies and VisDescriptions of Things. Implementation of the tool for visual enrichment of ontologies and resource descriptions is considered as a future continuation of presented work.

Acknowledgment

This research has been performed as part of IoT Tivit SHOK Program in the department of Mathematics and Information Technology (MIT, University of Jyvaskyla, Finland) funded by TEKES and consortium of industrial partners. Authors are grateful to the project team members from Industrial Ontologies Group (IOG) and colleagues from VTT research team (as well as other industrial partners of the project) who have been involved into the correspondent task and did provide fruitful cooperation.

References

[1] O. Khriyenko, V. Terziyan, and O. Kaikova, "User-assisted Semantic Interoperability in Internet of Things: Visually-facilitated Ontology Alignment through Visually-enriched Ontology and Thing Descriptions", In: Proceedings of the Sixth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2012), Barcelona, Spain, 23-38 September, 2012, pp. 104-110.

[2] D.J. Cook and S.K. Das, "How smart are our environments? An updated look at the state of the art". Pervasive and Mobile Computing. 3(2) , 2007, pp. 53-73.

[3] J. Honkola, H. Laine, R. Brown, and O. Tyrkko, "Smart-M3 information sharing platform". Proc. IEEE Symp. Computers and Communications (ISCC'10), 2010, pp. 1041-1046.

[4] K. Kotis and A. Katasonov, "Semantic Interoperability on the Web of Things: The Smart Gateway Framework", CISIS 2012, Palermo, Italy, 2012.

[5] O. Kaykova, O. Khriyenko, D. Kovtun, A. Naumenko, V. Terziyan, and A. Zharko, "Challenges of General Adaptation Framework for Industrial Semantic Web", In: Amit Sheth and Miltiadis Lytras (eds.), Semantic Web-Based Information Systems: State-of-the-Art Applications, CyberTech Publishing, 2007, pp. 61-97.

[6] Semantic Web, 2001. URL: http://www.w3.org/2001/sw/

[7] T. Berners-Lee, J. Hendler, and O. Lassila, "The Semantic Web", Scientific American 284(5), 2001, pp. 34-43.

[8] A. Ankolekar, M. Burstein, J.R. Hobbs, O. Lassila, D.L. Martin, D. McDermott, S.A. McIlraith, S. Narayanan, M. Paolucci, T.R. Payne, and K. Sycara, "DAML-S: Web Service Description for the Semantic Web", 2002. URL: http://www-2.cs.cmu.edu/~terryp/Pubs/ISWC2002-DAMLS.pdf

[9] M. Paolucci, T. Kawamura, T.R. Payne, and K. Sycara, "Importing the Semantic Web in UDDI", 2002. URL:http://www-2.cs.cmu.edu/~softagents/papers/Essw.pdf

[10] EM. Maximilien, A. Ranabahu, and K. Gomadam, "An Online Platform for Web APIs and Service Mashups". In IEEE INTERNET COMPUTING, IEEE Computer Society, 2008, pp. 32-43.

[11] T. Berners-Lee, "Linked Data - Design Issues". 2006. URL: http://www.w3.org/DesignIssues/LinkedData.html

[12] T. Heath and C. Bizer, "Linked Data: Evolving the Web into a Global Data Space" (1st edition). Synthesis Lectures on the Semantic Web: Theory and Technology, 1:1, 1-136. Morgan & Claypool. 2011.

[13] O. Khriyenko and M. Nagy, " Semantic Web-driven Agent-based Ecosystem for Linked Data and Services", In: Proceedings of the Third International Conferences on Advanced Service Computing, Rome, Italy, 25-30 September, 2011, 8 p.

[14] O. Khriyenko, "Collaborative Service Ecosystem - Step Towards the World of Ubiquitous Services". In: Proceedings of the IADIS International Conference Collaborative Technologies 2012, Lisbon, Portugal, 19-21 July, 2012.

[15] V. Spiliopoulos and G. A. Vouros, "Synthesizing Ontology Alignment Methods Using the Max-Sum Algorithm", Knowledge and Data Engineering, IEEE Transactions on, vol.PP, no.99, pp.1-11.

[16] K. Kotis, A. Katasonov, and J. Leino, "Aligning Smart and Control Entities in the IoT", In: Proceedings of the 5th Conference on Internet of Things and Smart Spaces, St.-Petersburg, Russia, 27-28 August, 2012.

[17] K. Kotis, A. Valarakos, and G. Vouros, "AUTOMS: Automating Ontology Mapping through Synthesis of Methods.", In: Proceedings of the International Semantic Web

Conference (ISWC'06), Ontology Matching International Workshop, Atlanta USA, 00/2006.

[18] A. Valarakos, V. Spiliopoulos, K. Kotis, and G. Vouros, "AUTOMS-F: A Java Framework for Synthesizing Ontology Mapping Methods", i-Know,07, Graz, Austria, 00/2007.

[19] P. Shvaiko and J. Euzenat, "Ontology matching: state of the art and future challenges". IEEE Transactions on Knowledge and Data Engineering, 2012.

[20] J. Pina, E. Cerezo, and F. Seron, "Semantic visualization of 3D urban environments". Multimedia Tools and Applications, Volume 59, Number 2 (2012), pp. 505-521, DOI: 10.1007/s11042-011-0776-3.

[21] M. Lanzenberger and J. Sampson, "Human-Mediated Visual Ontology Alignment". HCI (9) 2007, pp. 394-403.

[22] P. Jain, P. Hitzler, A.P. Sheth, K. Verma, and P.Z. Yeh, "Ontology Alignment for Linked Open Data". In: Proceedings of the 9th International SemanticWeb Conference, ISWC 2010, Springer-Verlag (2010), Shanghai, China, November 7-11, 2010, pp. 402–417.

[23] F. Kboubi, A.H. Chaibi, and M.B. Ahmed, 'Semantic Visualization and Navigation in Textual Corpus'. In: CoRR abs/1202.1841, 2012 .

[24] M. Ehrig and Y. Sure, "Ontology mapping - an integrated approach. In: Bussler, C., Davis, J., Fensel, D., Studer, R. (eds.) Proceedings of the First European Semantic Web Symposium, Heraklion, Greece, 10-12 May, 2004.

[25] Protégé-owl (Stanford Medical Informatics) - http://protege.stanford.edu/overview/protege-owl.html

[26] M. Lanzenberger and J. Sampson, "Alviz - a tool for visual ontology alignment". In: Society, I.C.S. (ed.) Proceedings of the IV06, 10th International Conference on Information Visualization, London, UK, July, 2006.

[27] A. Sheth and D. Avant, "Semantic Visualization: Interfaces for exploring and exploiting ontology, knowledgebase, heterogeneous content and complex relationships," NASA Virtual Iron Bird Workshop, CA, March 31 and April 2, 2004.

[28] M. Granitzer, V. Sabol, K. W. Onn, D. Lukose, and K. Tochtermann, "Ontology Alignment - A Survey with Focus on Visually Supported Semi-Automatic Techniques," Future Internet, 2, 2010, pp. 238-258.

[29] O. Khriyenko, "Context-sensitive Multidimensional Resource Visualization", In: Proceedings of the 7th IASTED International Conference on Visualization, Imaging, and Image Processing (VIIP 2007), Palma de Mallorca, Spain, 29-31 August 2007.

[30] O. Khriyenko, "4I (FOR EYE) Technology: Intelligent Interface for Integrated Information", In: Proceedings of the 9th International Conference on Enterprise Information Systems (ICEIS-2007), Funchal, Madeira – Portugal, 12-16 June 2007.

[31] O. Khriyenko, "Context-sensitive Visual Resource Browser", In: Proceedings of the IADIS International Conference on Computer Graphics and Visualization (CGV-2008), Amsterdam, The Netherlands, 24-26 July 2008.

[32] RDF Vocabulary Description Language 1.0: RDF Schema. W3C Recommendation 10 February 2004. URL: http://www.w3.org/TR/rdf-schema/

[33] OWL Web Ontology Language. W3C Recommendation 10 February 2004. URL: http://www.w3.org/standards/techs/owl#w3c_all

[34] Notation3 (N3): A readable RDF syntax. W3C Team Submission 28 March 2011. URL: http://www.w3.org/TeamSubmission/n3/

www.iariajournals.org

International Journal On Advances in Intelligent Systems

ICAS, ACHI, ICCGI, UBICOMM, ADVCOMP, CENTRIC, GEOProcessing, SEMAPRO, BIOSYSCOM, BIOINFO, BIOTECHNO, FUTURE COMPUTING, SERVICE COMPUTATION, COGNITIVE, ADAPTIVE, CONTENT, PATTERNS, CLOUD COMPUTING, COMPUTATION TOOLS, ENERGY, COLLA, IMMM, INTELLI, SMART, DATA ANALYTICS

issn: 1942-2679

International Journal On Advances in Internet Technology

ICDS, ICIW, CTRQ, UBICOMM, ICSNC, AFIN, INTERNET, AP2PS, EMERGING, MOBILITY, WEB

issn: 1942-2652

International Journal On Advances in Life Sciences

eTELEMED, eKNOW, eL&mL, BIODIV, BIOENVIRONMENT, BIOGREEN, BIOSYSCOM, BIOINFO, BIOTECHNO, SOTICS, GLOBAL HEALTH

issn: 1942-2660

International Journal On Advances in Networks and Services

ICN, ICNS, ICIW, ICWMC, SENSORCOMM, MESH, CENTRIC, MMEDIA, SERVICE COMPUTATION, VEHICULAR, INNOV

issn: 1942-2644

International Journal On Advances in Security

ICQNM, SECURWARE, MESH, DEPEND, INTERNET, CYBERLAWS

issn: 1942-2636

International Journal On Advances in Software

ICSEA, ICCGI, ADVCOMP, GEOProcessing, DBKDA, INTENSIVE, VALID, SIMUL, FUTURE COMPUTING, SERVICE COMPUTATION, COGNITIVE, ADAPTIVE, CONTENT, PATTERNS, CLOUD COMPUTING, COMPUTATION TOOLS, IMMM, MOBILITY, VEHICULAR, DATA ANALYTICS

issn: 1942-2628

International Journal On Advances in Systems and Measurements

ICQNM, ICONS, ICIMP, SENSORCOMM, CENICS, VALID, SIMUL, INFOCOMP

issn: 1942-261x

International Journal On Advances in Telecommunications

AICT, ICDT, ICWMC, ICSNC, CTRQ, SPACOMM, MMEDIA, COCORA, PESARO, INNOV

issn: 1942-2601