Report on the 7th International Workshop on Requirements Engineering for Sustainable Systems (RE4SuSy 2018)

Ahmed Alharti*, Ruzanna Chitchyan†, Abram Hindle‡, Timo Kehrer§, Birgit Penzenstadler¶, Colin C. Venters‖
*RMIT, Australia, ahmed.alharthi@rmit.edu.au
†University of Bristol, UK, r.chitchyan@bristol.ac.uk
‡University of Alberta, Canada, hindle1@ualberta.ca
§Humboldt-Universität zu Berlin, Germany, timo.kehrer@informatik.hu-berlin.de
¶California State University, Long Beach, USA, birgit.penzenstadler@csulb.edu
‖University of Huddersfield, UK, c.venters@hud.ac.uk

***Abstract*—This year's edition of the workshop was held at the Banff Centre for Arts and Creativity in Banff, Alberta, Canada at the 26th International Conference on Requirements Engineering.**

I. Introduction

The RE4SuSy workshop series has established a strong and growing research community around the different aspects of sustainability and how to support them in requirements engineering. Since requirements define how and what a software will do, we maintain that requirements engineering is the key point in software engineering through which sustainability can be fostered. Thus, the RE4SuSy workshop series is concerned with research on techniques, tools, and processes for sustainability through requirements engineering.

Last year the workshop initiated an effort to start converging the RE for sustainability community to a common set of fundamentals. This edition of the RE4SuSy workshop has built on the initial convergence effort, helping to clarify what characteristics a requirement should possess, and/or what constraints it should meet in order to qualify as a "sustainability requirement".

RE4SuSy is an interactive workshop: the contributors and participants engaged into deep discussions and provided peer feedback to all workshop submissions.

The program featured five presentations with moderated discussions and working sessions. In the following, we give a brief summary of each.

II. If you bill it, they will pay: Energy consumption in the cloud will be irrelevant until directly billed for

Don't leave the lights on! It is suggested that energy consumption of software systems and services is a significant issue as data-centers now account for a significant portion of the worldwide energy consumption (2-3%). This paper argues that until Cloud end-users have to pay for energy consumption there will be little or no demand for optimizing energy consumption. To address this, Hindle argues that Cloud services should bill customers directly in order to motivate consumers to consider the sustainability of the software services they utilise. However, the challenges here are non-trivial: e.g., How can energy consumption be measured directly in a Cloud environment? What are the economic incentives for both Cloud providers and Cloud consumers? How to create a paradigm shift towards sustainability in both Cloud providers and Cloud consumers?

Discussion: During the workshop the discussion topics around this paper focused on the economic incentives, and challenges of measuring the energy consumption, the role of software in energy consumption, and the optimization of perceptual distance. Perceptual distance denotes that information about energy consumption needs to be a direct feedback with little delay, otherwise in the users' view it will be discounted from his/her consumption. For instance, when consumption information is provided immediately at the end of a month, before the beginning of the next month, where the energy bill is seen to have increased significantly, the consumer may try to analyze and change his/her behavior accordingly for the new month. However, if he/she gets the feedback on increased consumption three months later, then he/she may not be interested in a change.

III. Towards Tool-support for Sustainability Profiling

This paper argues that the most effective decisions related to sustainability of software are made in the early stages of the software development lifecycle. To support these decisions, the authors introduce Sustainability Profiling for Software (SuSoftPro); a tool that can assist in analysing sustainability requirements. They present an analysis of the core features of SuSoftPro in comparison with two other approaches, which utilise Multi-Criteria Decision Analysis. The effectiveness of their approach is explored in a case study, which analyses the sustainability aspects of a Skin Cancer Information System.

Their results suggest that SuSoftPro provides a method for considering and prioritising sustainability requirements. This raises the question of how should we maximise the positive impact and minimise the negative impact across all sustainability dimensions?

The paper presented a study with fourteen stakeholders and 23 high-level requirements were assigned to the relevant sustainability dimensions. Fuzzy rating scales were used by participants to assign relevance to each of these requirements. Fuzzy ratings were used as the authors consider them to be more precise in expressing stakeholder preferences that numerical ones.

Discussion: The workshop discussion around energy profiling and behavior change centered around 3 main points: (i) incentivization to think about sustainability, (ii) prevalence of doomsday narrative versus a more in-depth exploration of transition stories, and (iii) observation that economic gains and losses are considered to be of prime importance (e.g. as drivers to lower one's energy bill) but it is convenience that really drives human behaviour, as humans are creatures of habit.

IV. Karlskrona Manifesto: Software Requirement Engineering Good Practices

The Karlskrona Manifesto was incepted in 2014 and since then, the research group that remained steady around it conducted a number of studies with practitioners to investigate what their perceptions of sustainability were and how it could support requirements engineering. This paper takes a straightforward approach to mapping the principles to phases of the software development lifecycle and recommends a few practices for application of these principles. In addition, the paper presents a template for application of the best practices and explains these using an illustrative example study. The next step in this research is a planned confirmatory case study with application of the principles with a template to an industrial setting.

Discussion: The discussion session on the application of the Karlskrona Manifesto touched a number of topics. One promising approach suggested for use in future research was storytelling as a way to describe 'how it's done' in addition to a best practice template. As examples of this, the agile story books about XP (by Kent Beck) were proposed. For instance, the "XP Explained" book discusses how agile processes worked at Chrysler. It was also suggested to set up a web portal that would show different examples from different case study systems. One of the questions to resolve (which is outside of the scope of this particular piece of research) is how to promote best practices as opposed to engaging into 'method wars'.

Thinking of freely available materials for confirmatory case studies, it was suggested that the SQL Lite OS project has a repository of requirements which could be useful for such a study.

V. Public Policy Challenges: An RE Perspective

This paper analyzed the mass stream media around seven topics that were popular during the past 12 months. It assessed whether we can identify challenges associated with defining, formulating and realizing public policies, whether the challenges have analogs in RE for software intensive systems, and how RE techniques could help mitigate the identified public policy challenges. Furthermore, it considers whether RE techniques can be used to proactively identify possible public policy challenges during formulation and before enactment.

The investigated topics were big data algorithms that have drawn sufficient attention to warrant public policy discussions, IT projects in support of some policy goal, social (e.g. free speech, critical thinking, gender issues, fake news, radicalism), privacy (e.g. location data, social media, children's self-determination), policy (e.g. cybersecurity, copyright, taxes, housing), climate change (e.g. resilience, carbon emissions, energy, electric vehicles, pipelines), controlled substances (e.g. state versus federal law, avoiding crime, licensing, taxes), and equalization (e.g. income, taxes, resources, cost of living). The analysis identified a number of challenges that drew parallels in requirements engineering.

Discussion: Much of the discussion was about more examples of the lack of analysis used in policy decisions. One issue discussed at this session focused on the relevance of the measurement scope referred to as the "Magnitude of Unintended Consequences". A local change measured locally could or could not be felt globally, yet it will matter for the given locality; a change observed at one level of analysis could have repercussions at another level. This raises the challenge of how to measure the direct and indirect impact of decisions.

VI. An Exploration of Sustainability Thinking in Research Software Engineering

This paper proposes an avenue for research that rethinks the development of research software from a sustainability perspective. For this, the paper uses the analysis of a goal model that explores the meaning of sustainability for the domain of research software engineering. In order to get clarity on the underlying values and to extract exemplary sustainability requirements, it revisits the sustainability reference model from 2013 by Penzenstadler and Femmer and instantiates it for the case example of two dedicated research software systems from the field of Model-Driven Engineering, namely Henshin and SiLift.

The exploration of the five dimensions leads to exemplary objectives for the two tools at hand that can be implemented via activities and assessed via indicators. Next steps for this research are to implement some of identified activities for the studied tools (i.e., Henshin and SiLift).

Discussion: The discussion session topics for this paper related to licensing (which can harm the community), and the notion of 'dependency hell'. This 'hell' poses a problem for maintenance as it expects that the versions of particular libraries on which one's software depends should be locked up. For future work, there was a recommendation to look at Greg Wilson's Software Carpentry for Replicable Software and to relate to the work of Dan Katz on the WSSSPE workshop series and Jeffrey Carver's efforts for a Sustainable Scientific Software Institute in the US.

Figure 1. RE4SuSy 2018: Ahmed Alharti, Abram Hindle, Colin C. Venters, Timo Kehrer, and Birgit Penzenstadler.

VII. Next Steps

In the final session of the workshop the discussion focused on the next steps for both the workshop in general and specific actions related to the papers presented during the day. Having established a strong and growing research community around the different aspects of sustainability and how to support them in requirements engineering the discussion centered on the challenge of growing the research community and the potential steps required to achieve growth.

A potential opportunity is to demonstrate empirically the unsustainability of scientific software and link it to mechanisms for sustainability profiling underpinned by the principles set out in the Karlskrona manifesto with software and requirement engineering best practice.

A final topic for discussion was how to embed sustainability thinking into the computer science and software engineering curriculum and the need to ingrain sustainability as the core theme.