RE4ES: Support Environmental Sustainability by Requirements Engineering

Birgit Penzenstadler[1], Bill Tomlinson[2] and Debra Richardson[2]

[1] Technische Universität München, Germany
penzenst@in.tum.de
[2] University of California, Irvine, US
wmt@uci.edu, djr@ics.uci.edu

Abstract. [**Motivation:**] Environmental sustainability is an important concern. Information and communication technology (ICT) innovation is ambivalently positioned with regard to our rapid development and shortening innovation cycles. On one hand, information technology facilitates the (excessive) usage of resources. On the other hand, ICT can also help to significantly reduce human impact on the environment.
[**Problem:**] Environmental sustainability is currently not supported explicitly in requirements engineering (RE). This leads to the problem that (a) environmental sustainability is not yet given sufficient importance and (b) it is difficult to manifest in requirements & design and therefore hard to assess.
[**Principal idea:**] We need to combine the knowledge of RE, environmental informatics, and further disciplines, to develop an RE approach that tailors analysis, documentation, and assessment for ICT systems where environmental sustainability is a first class quality objective.
[**Contribution:**] This paper is a research preview on an approach to help requirements engineers handle sustainability as a first class quality objective. It elaborates on how we plan to refine and validate this approach in the future.

Keywords: requirements, sustainability, environment, requirements engineering, quality modeling

1 Introduction & Motivation

The most cited definition of sustainability is to "meet the needs of the present without compromising the ability of future generations to meet their own needs" [1]. Although our approach primarily aims at environmental sustainability, it must also be socially sustainable in order to have practical significance [2]. As Mahaux [3] pointed out, we need a toolbox for supporting it in requirements engineering. We extend the idea of such a toolbox in this research preview and provide some of our drafts.

Problem: The use of information and communications technology (ICT) contributes significantly to the usage of our planet's resources [4]. However, ICT

bears a lot of potential for "greening through IT" [5]. "Greening through IT" is making our life more environmentally sustainable by technological support for our daily life; this is the context of our research. In contrast, Green IT or "greening of IT" is making hardware and software of ICT systems more resource-efficient; we do not focus on this. We must improve the environmental sustainability of humankind to protect our living space for future generations. Missing is a comprehensive understanding of how software engineering, and especially requirements engineering (RE), can help in this endeavor. We are analyzing what and how RE can contribute to the improvement of the environmental sustainability of ICT in general, but especially focus on the development of ICT systems that have environmental sustainability in their explicit system vision and abbreviate these systems with ICT4ES.

Contribution: Our goal is to support the development of ICT systems for environmental sustainability with an adequate requirements engineering approach that integrates the knowledge of environmental informatics. This enables software engineers to handle sustainability as first class quality objective. Our research questions are:

RQ1: What are the implications for RE of ICT4ES, i.e., when making environmental sustainability a first-class quality objective for development?
For ICT4ES as we defined the term, environmental sustainability is an overall development goal. However, it is not clear how that impacts the requirements for a system. We seek to understand what is necessary to be taken care of when developing ICT4ES and how the business processes and business goals differ from those of traditional products.

RQ2: How can the necessities resulting from ICT4ES be implemented in an RE approach?
We aim at a toolbox to support the demands resulting from the goal of contributing to environmental sustainability. First, we analyze which artifacts are necessary to document the newly arising demands and what their concrete contents are. Then, we investigate which concepts have to be supported and which methods are required to elaborate these artifacts and how they have to be adapted.

RQ3: How can we assess the impacts of a given software system for environmental sustainability, including both direct and indirect effects, and considering different groups of stakeholders?
We elaborate metrics to measure environmental sustainability and provide an answer as to how a system can be proven to fulfill the sustainability requirements imposed upon it. Furthermore, we investigate an appropriate way to translate the requirements into acceptance criteria and how these criteria can be incorporated into an overall quality model.

2 Related Work

Sustainability is beginning to play an important role in software engineering, with the RE'08 keynote, the ICSE'09 Software Engineering for the Planet special

session, the CAiSE'10 panel, the 2009, 2010, and 2011 Workshops on Software Research and Climate Change, and the conference slogan for ICSE'12.

Amsel et al. [6] discuss ideas on how to support sustainability in SE. Cabot et al. [7] performed a case study for sustainability as goal for the ICSE organization with i* models to support decision making for future conference chairs. Dick et al. [8] investigate how web pages can be developed with little environmental impact, i.e., energy-efficiently, and work on a respective guideline for web developers. Mahaux et al. [3] performed a case study on a business information system for an event management agency that advertises environment-friendly events. They assessed how well some current RE techniques support modeling of specific sustainability requirements in that case study.

These works look at either a specific application domain or a specific development technique and adapt them to support sustainability modeling, while this project aims at an encompassing approach to be evaluated in various domains of ICT4ES systems. No other work yet proposes solutions for how to support quality modeling of environmental sustainability for software systems.

3 Approach to RE for ICT4ES

Our approach to RE for ICT4ES is planned in two phases: First, we conduct an analysis of domains as well as values and goals of the respective stakeholders, then we design a tailored RE method that supports the gathered specifics for ICT4ES (see Fig. 1). All activities described in this section are in progress, which means we have started but not yet completed them.

3.1 Analysis of Domains, Values, and Goals

Environmental sustainability can be supported by software systems in different ways, e.g., (a) information systems for environmental sciences, including climate models, earthquake warning, etc., (b) information systems that support green business processes, for example environment-friendly event management, and (c) embedded systems that lower our energy consumption. Therefore, we need to analyze the different types of domains that need support in explicitly addressing environmental sustainability in their software engineering approaches.

Based on the distinction of domains, we perform structured interviews in industry and academia with representatives from different domains. The interviews are followed by a systematic analysis and an interpretation that draws conclusions for the design of the envisioned method's elements.

Starting with the results of the interview analysis, we elaborate a map of values for environmental sustainability and we detail the goals in a taxonomy, focusing on the ones that relate to requirements engineering for ICT4ES systems:

Value map for environmental sustainability in SE (RQ1) The value map shall put the value of sustainability into relation with traditional software engineering values as in the framework described by Khurum [9]. Her framework

relies on data gathered in interviews with practitioners and allows to create impact evaluation patterns from value maps.

Goal taxonomy for sustainability in SE (RQ1) The goal taxonomy decomposes and details the aspects of environmental sustainability from the point of view of software engineering. The input is the value map and for each value we can deduce supporting goals. Initially, most of these goals are independent of the system to be developed. Each of the goals is then decomposed hierarchically until the goals are sufficiently specific to be transformed into requirements.

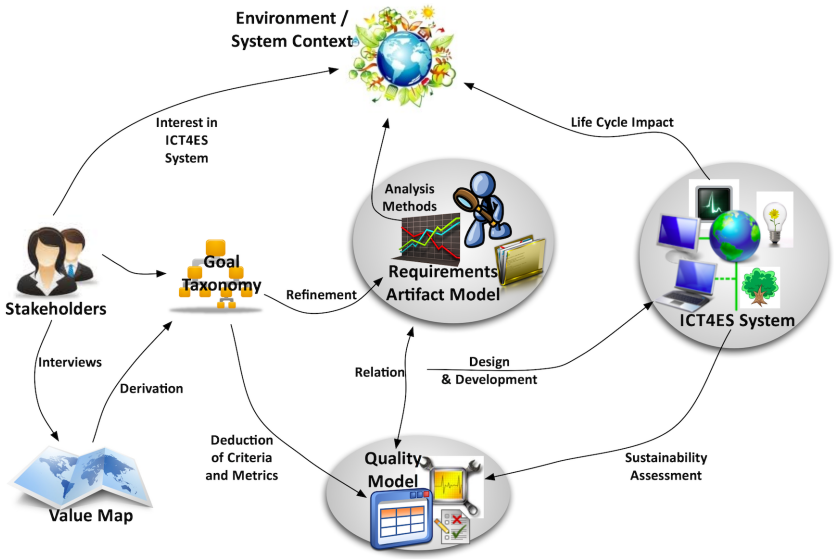

Fig. 1. Environmental Sustainability in Requirements Engineering.

3.2 Design of a Tailored RE Approach

From the goal taxonomy, we gather requirements for artifacts, methods, and models for the documentation of sustainability requirements arising by deduction from the goal taxonomy with respect to a specific ICT4ES system. Based on these requirements and the knowledge acquired in the earlier phases of the project, we conduct an analysis and evaluation of different techniques, compare existing approaches, and develop a tailored RE approach including a quality model that provides indicators and metrics to assess environmental sustainability.

Sustainability requirements artifact model (RQ2) An artifact model gives guidance on structure and content to be elaborated when documenting sustainability requirements and all related information like environmental impact, lifecycle analysis, stakeholders, rationale, regulations, etc. Based on our

experience [10], we develop an artifact model for representing sustainability requirements and related information.

Adapted analysis techniques (RQ2) To transition from goals to requirements and to adequately document these requirements according to an artifact model, we elaborate analysis techniques and documentation methods that form part of an RE approach tailored to ICT4ES. Solutions include adaptations of creativity techniques, life cycle analysis, environmental impact assessment and risk analysis techniques as well as handling of environmental information in form of data, statistics, and models.

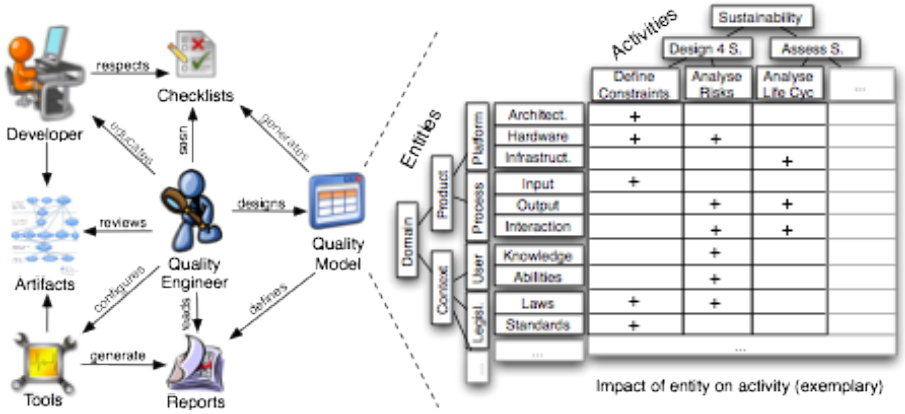

Fig. 2. Model-based Quality Assurance (adapted from [11]) and Quality Model Excerpt.

Deduced quality model (RQ3) The quality model is built upon the input from the value map and the goal taxonomy. A quality model is a model with the objective to describe, assess and/or predict quality [11]. The activity-based quality model is elaborated on the basis of concepts proposed in [12]. It includes criteria for sustainability assessment as well as indicators and metrics to evaluate and measure a software system's compliance to the sustainability requirements. Fig. 2 shows the model-based principle and an excerpt of the quality model draft.

Case studies (RQ1-3) The approach will be evaluated in industrial case studies, including the value map, the goal taxonomy, the artifact model, the analysis techniques, and the quality model. The qualitative evaluation will be implemented as a comparative study. The case study already under way is on car sharing; another one will be on an irrigation system.

4 Conclusion

In this research preview, we have introduced our ongoing research on a tailored RE method for ICT systems for environmental sustainability. The analysis phase

investigates the domains and elaborates values and goals with the respective stakeholders. The design phase provides a tailored artifact model with analysis methods and a deduced quality model. Both will be evaluated in industrial case studies from different domains.

Currently, we are completing a systematic literature review on sustainability in software engineering to extend the preliminary check in [3] and preparing a guideline for the industry interviews. Furthermore, we evaluate approaches from related disciplines in student seminars as described in [13] for preliminary studies.

Our contribution will provide software engineers with a toolbox to handle sustainability as first class quality objective. This enables "greening through IT" — to produce ICT systems that have positive impact on their surrounding eco-systems and therefore not only meet the needs of the present (by satisfying traditional quality objectives) but at the same time preserve the ability of future generations to meet their own needs (by meeting sustainability quality objectives). As software systems have a profound influence on many different facets of global civilization, including sustainability in the design of these systems has the potential to have transformative impacts on the world in which we live.

References

1. UN World Comission on Environment and Development: Our Common Future. In: UN Conference on Environment and Development. (1987)
2. Sverdrup, H., Svensson, M.G.E.: Defining the concept of sustainability. In: Systems Approaches and Their Application. Springer (2005) 143–164
3. Mahaux, M., Heymans, P., Saval, G.: Discovering Sustainability Requirements: an Experience Report. In: 17th REFSQ. (2011)
4. Group, T.C.: Smart 2020: Enabling the low carbon economy in the information age. Technical report, Global eSustainability Initiative (2008)
5. Tomlinson, B.: Greening through IT. MIT Press Association (2010)
6. Amsel, N., Ibrahim, Z., Malik, A., Tomlinson, B.: Toward sustainable software engineering. In: Proc. of the 33rd Intl. Conf. on Software Engineering. (2011)
7. Cabot et al.: Integrating sustainability in decision-making processes: A modelling strategy. In: 31st Intl. Conf. on Software Engineering. (2009) 207 –210
8. Dick, M., Naumann, S., Kuhn, N.: A model and selected instances of green and sustainable software. In: What Kind of Information Society? (2010)
9. Khurum, M., Gorschek, T.: Software value map - an exhaustive collection of value aspects for the development of software intensive products (2011)
10. Fernandez, D.M., Lochmann, K., Penzenstadler, B., Wagner, S.: A case study on the application of an artefact-based requirements engineering approach. In: 15th Intl. Conf. on Evaluation and Assessment in Software Engineering. (2011)
11. Wagner, S., Deissenboeck, F., Winter, S.: Managing quality requirements using activity-based quality models. In: Intl. Workshop on Software Quality. (2008)
12. Winter, S., Wagner, S., Deissenboeck, F.: A comprehensive model of usability. In: Proc. of Engineering Interactive Systems. (2007)
13. Penzenstadler, B., Fleischmann, A.: Teach sustainability in software engineering? In: 24th Intl. Conference on Software Engineering Education & Training. (2011)