

# Requirements Engineering: Die Schlüsseldisziplin für nachhaltige Software- und Systementwicklung

Stefanie Betz<sup>1</sup>, Norbert Seyff<sup>2</sup>, Birgit Penzenstadler<sup>3</sup>, Christoph Becker<sup>4</sup>

<sup>1</sup>Karlsruhe Institut für Technology

<sup>2</sup>Fachhochschule Nordwestschweiz, Universität Zürich

<sup>3</sup>California State University, Long Beach

<sup>4</sup>University of Toronto, Technische Universität Wien

## Beitrags-Kategorie

Vision

## Zielgruppe des Beitrags

RE-Interessierte, -Praktiker, -Forscher

## Motivation der behandelten Fragestellung

Sozio-technische Systeme haben Einzug in alle Bereiche unseres Lebens gefunden. Dennoch werden bei der Entwicklung dieser Systeme mögliche Auswirkungen in der späteren Anwendung nicht oder zu wenig berücksichtigt. Wenn zum Beispiel der spätere Ressourcenverbrauch bei der Systementwicklung nicht berücksichtigt wird, kann dies zu Systemen führen, welche die Umwelt auf verschiedenen Ebenen belasten: Die Systeme können energiehungrig sein (Effekt 1. Ordnung, direkte Auswirkung), sie können Nutzerverhalten dahingehend ändern noch mehr Ressourcen zu verbrauchen (Effekt 2. Ordnung, indirekte Auswirkung), und das kann über einen längeren Zeitraum dazu führen, dass sich diese Auswirkungen systemweit multiplizieren (Effekt 3. Ordnung, systemisch).

## Behandelte Fragestellung oder Problem

Das Thema „Nachhaltigkeit“ gewinnt immer mehr an Bedeutung [1]. Diese Entwicklung wird auch in folgendem Zitat deutlich: *„Von einem Prinzip der Forstwirtschaft hat sich Nachhaltigkeit zu einem Leitbild für das 21. Jahrhundert entwickelt. Der Kerngedanke: Zukünftige Generationen sollen dieselben Chancen auf ein erfülltes Leben haben wie wir.“*<sup>1</sup>

Dabei beziehen sich die Auswirkungen, die ein Softwaresystem auf seine Umgebung haben kann nicht nur auf die ökologische Dimension, sondern umfassen auch die individuelle, soziale, ökonomische und technische Dimension [2]. Die Betrachtung dieser muss während der Entwicklung erfolgen und erfordert eine umfassende Analyse des zu entwickelnden Systems, um nachhaltige sozio-technische Systeme zu erstellen.

In dieser Arbeit argumentieren wir, dass Requirements Engineering (RE) als Disziplin und insbesondere RE-Verantwortlichen in Unternehmen eine Schlüsselrolle bei der Analyse und Diskussion der möglichen Auswirkungen zufällt, da Anforderungen die Grundlage für die zu entwickelnden nachhaltigen Systeme legen [3].

## Lösung und Ergebnisse

Nachhaltigkeit ist ein systemisches Konzept, das unabhängig von der Zielsetzung eines zu entwickelnden Systems berücksichtigt werden muss. Nachhaltigkeit muss demnach als fundamentaler Bestandteil der existierenden Gesellschaften und deren zugrundeliegenden Systeme angesehen werden [4]. Das zu erreichen erfordert Personen und Organisationen, die nachhaltige Software und Systeme entwickeln. In diesem Zusammenhang haben wir den Requirements Engineer als Schlüsselrolle identifiziert. Wir sehen den Requirements Engineer als „Gestalter“, der zusammen mit anderen Stakeholdern (z.B. dem Kunden) die Anforderungen an das zu entwickelnde System festlegt, überprüft und nachverfolgt und somit einen starken Einfluss auf die Entwicklung nachhaltiger Software hat. Im Folgenden erläutern wir, welche Aufgaben des Requirements Engineers dafür relevant sind. Wir stellen unsere Vision vor, wie RE-typische Verantwortungsbereiche bzw. Aktivitäten (Ermittlung, Analyse, Dokumentation, Validierung und Management von Anforderungen) im Rahmen der Entwicklung von nachhaltiger Software und Systemen angepasst und erweitert werden sollen. Diese Vision beruht auf intensiver Diskussion zwischen Wissenschaftlern und auch auf einer Interviewstudie, an der 13 RE Praktiker teilgenommen haben.

*Ermitteln von Anforderungen:* Ein essentieller Aspekt für nachhaltige Systementwicklung ist „Awareness“ (Bewusstseinsbildung). Die verschiedenen Stakeholdergruppen (Kunden, Management, Nutzer) müssen über die Bedeutung und den potentiellen Nutzen von nachhaltiger Software informiert werden. Ebenso muss der Requirements Engineer bei der Identifikation von Stakeholdern auch Organisationen und Personen berücksichtigen, die über die späteren

<sup>1</sup><http://www.bne-portal.de/was-ist-bne/grundlagen/nachhaltigkeitsbegriff/>

Auswirkungen im Einsatz Auskunft geben können (erweiterte Stakeholderidentifikation) beispielsweise ein Umweltexperte.

*Analyse von Anforderungen:* Grundsätzlich müssen hier Metriken entwickelt werden, die den Fortschritt im Hinblick auf Nachhaltigkeitsziele messen können, und Methoden, die es ermöglichen, die Kosten und Vorteile der Implementierung von „Nachhaltigkeits“-Anforderungen zu schätzen. Bei der Analyse der Nachhaltigkeits-Anforderungen sind dann besonders die folgenden zwei Punkte zu berücksichtigen: (1) Analyse der kurz-, mittel- und langfristigen Auswirkungen der „regulären“ (funktionalen und nicht-funktionalen) Anforderungen auf die Nachhaltigkeit von Systemen; (2) Trade-off Analysen zwischen regulären Anforderungen und Nachhaltigkeits-Anforderungen.

*Dokumentation von Anforderungen:* Existierende Standards (beispielsweise IEEE-830, ISO-25010) und Vorlagen müssen die verschiedenen Dimensionen der Nachhaltigkeit und deren Effekte berücksichtigen sowie den erweiterten Stakeholderkreis. Besonders wichtig ist es in diesem Zusammenhang, leichtgewichtige und flexible Ansätze zu entwickeln, um den Prozess nicht weiter zu verkomplizieren.

*Validierung der Anforderungen:* Anforderungen müssen bezüglich ihrer Auswirkungen auf die verschiedenen Nachhaltigkeitsdimensionen validiert werden. Dabei müssen auch die drei Auswirkungsebenen berücksichtigt werden. Mögliche Methoden dazu sind die Zielmodellierung und die Referenzmodellierung.

*Requirements Management:* Bei dieser Aktivität ist es wichtig, durchgehend sicherzustellen, dass die System-Ziele im Hinblick auf die verschiedenen Nachhaltigkeitsdimensionen erreicht werden. Dies sollte bei der Anpassung des Systems wie auch bei der Ausführung des Systems überprüft und sichergestellt werden. Daher werden fortgeschrittene Methoden und Werkzeuge zur Analyse von Auswirkungen von Systemänderungen auf die verschiedenen Nachhaltigkeitsdimensionen benötigt. Außerdem brauchen wir entsprechende Maßnahmen für das permanente System Monitoring, das die verschiedenen Auswirkungen berücksichtigt.

Wir kommen zum Schluss, dass Requirements Engineers Verantwortung für nachhaltige Software- und Systementwicklung übernehmen müssen, obwohl im Moment noch wichtige Ansätze, Methoden und Werkzeuge zur Bewältigung dieser Aufgabe fehlen. Dazu müssen Industrie und Forschung zusammenarbeiten, um gemeinsam dieses problematische Feld zu analysieren, neue Ansätze zu entwickeln, zu erproben und in realen Projekten zum Einsatz zu bringen.

Folgende Punkte sehen wir als nächste wichtige Schritte:

- Identifizierung von Gründen und Ursachen für die fehlende Berücksichtigung von Nachhaltigkeits-

dimensionen in der Software- und Systementwicklung. Mit der Hilfe von Forschern sollen aktuelle industrielle Entwicklungsprozesse und Praktiken in Bezug auf deren Nachhaltigkeit analysiert werden und Schlüsse auf mangelnde Beachtung von Nachhaltigkeitsdimensionen gezogen werden.

- Basierend darauf können exemplarische Studien durchgeführt werden, welche die Vorteile von nachhaltiger Software- und Systementwicklung analysieren. In diesen Studien können neue Methoden, Ansätze und Werkzeuge eingesetzt und erprobt werden.
- Sowohl in der Forschung als auch in der Praxis müssen Fähigkeiten für die nachhaltige Software und Systementwicklung basierend auf einem gemeinsam erarbeiteten Lehrplan entwickelt werden. Die Praxis kann dabei explizit die Nachfrage bezüglich dieser Fähigkeiten fordern.

*Let's get started.*

### **Was ist neu?**

In diesem Papier skizzieren wir, wie sich die Rolle des Requirements Engineerings in Zukunft an neue Herausforderungen im Software Engineering anpassen sollte. Weitere Forschung in diesem Feld des Requirements Engineerings ist notwendig, um aus dieser Vision Methoden und Werkzeuge für die Unterstützung des Requirements Engineerings von morgen entstehen zu lassen.

### **Danksagung**

Wir bedanken uns bei unseren Freunden und Kollegen, mit denen wir gemeinsam am Thema nachhaltige Software- und Systementwicklung forschen (siehe: <http://www.sustainabilitydesign.org/who-we-are/>).

### **Quellen**

- [1] UN GLOBAL COMPACT; ACCENTURE, "The UN global compact-accenture CEO study on sustainability: Architects of a better world." (2013). Available at: <http://www.accenture.com/Microsites/ungc-ceo-study/Pages/home.aspx>.
- [2] C. Becker, R. Chitchyan, L. Duboc, S. Easterbrook, B. Penzenstadler, N. Seyff, C. C. Venters, "Sustainability Design and Software: The Karlskrona Manifesto." Int. Conf. on Software Engineering, ICSE (2), S. 467-476, (2015).
- [3] C. Becker, S. Betz, R. Chitchyan, L. Duboc, S. Easterbrook, B. Penzenstadler, N. Seyff, C. C. Venters, "Requirements: The Key to Sustainability." IEEE Software 33(1): 56-65 (2016).
- [4] C. Becker, R. Chitchyan, L. Duboc, S. Easterbrook, M. Mahaux, B. Penzenstadler, G. Rodríguez-Navas, C. Salinesi, N. Seyff, C. C. Venters, C. Calero, S. Akinli Koçak, S. Betz, "The Karlskrona manifesto for sustainability design." CoRR abs/1410.6968 (2014).