The Open University

Open Research Online

Citation

Lopez, Tamara; Sharp, Helen and Wermelinger, Michel (2024). Co-Designing Resilient Socio-Technical Software Systems. In: Designing '24: Proceedings of the 1st International Workshop on Designing Software, ACM, New York, NY, USA, pp. 60–63.

URL

https://oro.open.ac.uk/95335/

License

(CC-BY 4.0) Creative Commons: Attribution 4.0

https://creativecommons.org/licenses/by/4.0/

Policy

This document has been downloaded from Open Research Online, The Open University's repository of research publications. This version is being made available in accordance with Open Research Online policies available from Open Research Online (ORO) Policies

Versions

If this document is identified as the Author Accepted Manuscript it is the version after peer review but before type setting, copy editing or publisher branding

Co-Designing Resilient Socio-Technical Software Systems

Tamara Lopez
The Open University
Milton Keynes, United Kingdom
tamara.lopez@open.ac.uk

Helen Sharp
The Open University
Milton Keynes, United Kingdom
helen.sharp@open.ac.uk

Michel Wermelinger
The Open University
Milton Keynes, United Kingdom
michel.wermelinger@open.ac.uk

ABSTRACT

Socio-technical resilience is increasingly a design goal for software that integrates new kinds of automation. Using an example drawn from air traffic control, this paper explores how a co-design approach can be used alongside resilience engineering principles to provide structure and focus to software design activities. The report aligns the steps associated with resilience engineering to the co-design activities of probing, provoking, projecting and prototyping, and suggests methods for engagement with stakeholders that can be used to collect and synthesise data.

CCS CONCEPTS

• **Software and its engineering** → **Designing software**; • **Human-centered computing** → **Empirical studies in collaborative and social computing**.

KEYWORDS

collaborative and social computing, designing software, human-centred computing

ACM Reference Format:
Tamara Lopez, Helen Sharp, and Michel Wermelinger. 2024. Co-Designing Resilient Socio-Technical Software Systems. In *2024 International Workshop on Designing Software (Designing '24), April 15–14, 2024, Lisbon, Portugal.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3643660.3643948

1 INTRODUCTION

Software is changing, and working lives are changing as a result. Deeply embedded within social worlds, software increasingly uses model-based reasoning to undertake and support decision making that was once handled exclusively by humans. The challenge within software engineering is to design machine learning (ML) and other model-based systems that are accurate, fair, and transparent [2]. These systems must also reflect the values of the social worlds in which they are embedded [16]. In addition, though software must still be engineered to perform, to be reliable, and secure, it is also expected to be able to adapt to support humans within conditions that are uncertain [6]. In the context of work, people, long recognised to contribute to systemic resilience as users or operators [9], are now required to interact with ML and other model-based software in ways that influence and direct the behaviour of systems at a greater scale than ever before.

Alongside the growth in scale and complexity of automation within software has come a growing awareness that software must remain resilient in the face of changes that have social and technical aspects [10].

This paper contributes to understanding about methods to support the design of software systems that have the goal of preserving and accounting for socio-technical resilience. Using an example drawn from air traffic control and employing principles of resilience engineering [9], this work explores how socio-technical analysis methods[1] can be applied with stakeholders to create co-designed outputs and to maintain focus within software design. Extending prior work that focused on identifying resilient performance in current software development practice [11], this study considers how software systems can be designed to sustain resilient performance in the future.

Resilience engineering (ResE) provides a way to model human performance capabilities that are evident in resilient performance and to link these capabilities to system level goals within a domain [9]. The approach is socio-technical, examining how individuals and teams work on tasks; the organisational efforts to coordinate, support and manage operations; and the industrial system which designs and produces technologies that are used in work [12]. The potential for resilience [5, 9] lies within effective behavioral adaptation and compensation by people to manage disturbances in practice, rather than in managing the risk or consequences of failure.

In their research agenda for socio-technical systems engineering, Baxter and Sommerville argued that while many methods like resilience engineering exist for conducting socio-technical analyses, more work was needed to effectively integrate them with existing organizational systems and software engineering processes. When systematically applied, they posited that socio-technical design approaches can provide a bridge between change processes as they are undertaken within organisations and software development, promoting awareness about socio-technical concerns and lending support to software development processes [1].

Taking this position, the following sections present automation as a current design problem within air traffic control. Next, the paper draws guiding questions from this example. Resilience engineering is characterized as a method that can be applied within a co-design approach. The article concludes

This work licensed under Creative Commons Attribution International 4.0 License.
Designing '24, April 15–14, 2024, Lisbon, Portugal
© 2024 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0563-2/24/04
https://doi.org/10.1145/3643660.3643948

with a statement arguing for the use of co-design alongside resilience engineering as a step toward resilient socio-technical software design.

2 MOTIVATING EXAMPLE

Ground-based air traffic control operations must be able to manage increasing workloads during high traffic periods, and the industry is exploring how automation may support air traffic controllers (ATCOs) by taking over some of the more mundane decision-making tasks. Currently, industrial systems within air traffic are dependable and reliable, but are fixed. Though they provide support to ATCOs in the form of calculations or other information, they do not have decision-making capabilities.

The use of instruments by controllers in decision making can be seen in an incident that occurred in 2011 within the Swedish air space [14]. An aircraft was taking off from one runway at the same time that a second plane was landing on a different runway. The pilot of the landing aircraft decided to abort the landing attempt just before touchdown and commenced the procedure for a missed approach or "go around". Following published procedure, the tower controller instructed the pilot to turn right, but the aircraft turned left. The tower controller gave two additional instructions for the aircraft to turn in the correct direction; this instruction was followed on the third issue. A colleague in the tower observed on radar that the two planes were entering a near collision situation and instructed the plane that was taking off to turn in a direction that would avoid collision.

In this situation, the controllers used radio communication instruments to issue instructions and to receive information from pilots. They also used radar instruments to establish the position of the planes in relation to one another, to assess the incorrect position of the landing plane, and to determine a safe position for the plane that was taking off.

3 GUIDING QUESTIONS

The aim with this work is to understand how to support the design of new systems that will persist and preserve socio-technical resilience. Within the example above, the challenges for persisting resilient performance in system design include: 1) replacing or augmenting human decision making with technologies like machine learning, 2) meeting industry and societal requirements for being safe, and 3) preserving human capability to take and share operational control with instruments in everyday and critical situations. From these challenges, the following guiding questions can be formulated:

(1) How can socio-technical systems within domains like air traffic control be modelled in ways that allow for resilient performance to be identified?
(2) How can these models be used to support design outputs that will enable socio-technical resilience in software systems?
(3) How can appropriate levels of human involvement in the design process be maintained so that skill retention, agency, trust and accountability needs are met?

To explore these questions, the rest of the paper examines how resilience engineering, an example of a socio-technical analysis method, can be systematically applied within a co-design process [18].

4 RESILIENCE ENGINEERING USING A CO-DESIGN APPROACH

The elements of a ResE analysis [9] include understanding how work is done, identifying indicators of resilient performance within that practice, knowing the goals for the future status of the system, and determining how resilient performance can be maintained when changes are made. Analyses examine disturbances to practice that are resolved through instances of human adaptation or compensation and are guided by an understanding of what resilient performance signifies in a particular work context.

One limitation noted in prior ResE studies[5], is that they focus on finding evidence at different levels of abstraction such as within an entire industry like aviation or operations in an entire organisation. Though ResE advocates for an understanding of front-line work, at higher levels of abstraction, the nuanced details of practice within a profession are lost. Analyses cannot be used to inform the design of systems that include interaction between teammates or with technologies. A second limitation is methodological: because the discipline is conceptually oriented and lacks shared criteria or common approaches for undertaking analyses, it can be difficult to understand how to practically apply resilience engineering principles to solve design problems.

To address both limitations, we propose using a co-design approach to structure and organise resilience engineering analyses (see Figure 1). Co-design is a method of participatory design that works at the intersection of the dimensions of time and abstraction [18]. In working with the dimension of time, designers and stakeholders are able to generate understanding about a present situation, and to use an interpretation of the present to specify a future reality. In working with abstractions of different kinds, the goal is to progress from ideas toward more concrete representations that encapsulate instances of how the world works now and how it may work in the future. Co-design has associations with several design traditions. The use of co-design to support resilience engineering aligns with co-creative design, a tradition that learns from the collective creativity of potential users as a part of innovation and knowledge generation.

The following subsections situate the principles associated with resilience engineering analysis in relation to four co-design mechanisms, and suggest methods to collect and synthesise data.

4.1 Probing: Understanding how work is done

The techniques used within co-design should support probing or eliciting knowledge or meanings attached to actions in the world that are generally hidden or implicit. Similarly, a resilience engineering analysis begins with an understanding of how work is currently performed.

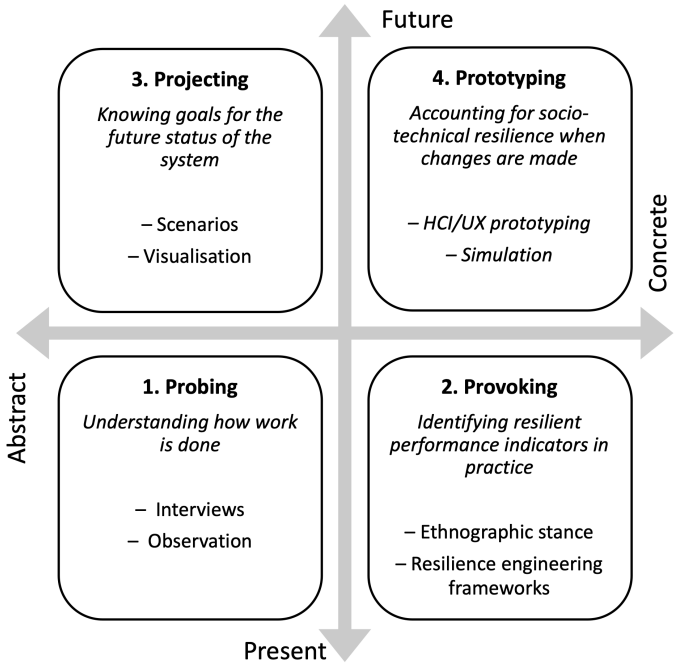

Figure 1: This matrix maps the elements of a resilience engineering analysis [9] to co-design mechanisms and activities that move from abstract (left) toward concrete representation (right), and from the present (lower) toward the future (upper). Adapted from [18].

The motivating example includes details of how work is planned to occur within air traffic control and how it unfolds in practice. The ATCOs used planned procedures to issue instructions to pilots over communication networks, and the system includes redundant instruments that provide information for calculating and triangulating calculations. The example also includes behaviour that was unplanned or spontaneous. The pilots decided to abort the landing and neglected to follow an instruction until it had been repeated three times. A second air traffic controller, who was not responsible for the landing plane, observed what was happening and intervened, passing an instruction to the second plane in order to avert a collision.

Detail about how work is done can be gathered using secondary sources such as the incident report used to generate the motivating example or through qualitative data collection methods including interviewing [4] and observation techniques [7] that reveal how people in work make decisions.

4.2 Provoking: Identifying indicators of resilient performance

Co-design methods should provoke critical thinking and engagement during design about the world as it is and how it is envisioned to be in the future [18]. ResE frameworks such as [5] provide a foundation for documenting evidence about practice that links concepts associated with resilient practice to empirical evidence of performance [11] and for building generalised analytic categories that are applicable across industries. However, a determination that practice is resilient requires analysts and other stakeholders to make a judgement about the significance of observed practices in meeting goals for resilience [5]. This assessment requires negotiation, and can raise conflicts[1].

For example, one design challenge identified in the motivating example was to meet the collective societal and industry goal that air traffic remains safe. In this example, safety is a resilience goal [9]. However, goals for resilience can be defined in relation to other key performance areas, like capacity, environmental impact or cost-efficiency [15]. Within the motivating example, the controllers exerted professional competence to avert the collision. Thus in designing new forms of automation, the primary resilience goal could be to maintain this degree of professional decision making skill and capability.

To address competing goals, provocation activities should be reflective[7], allowing designers to understand work practice from the point of view of the front-line workers, and represent the contextual factors that influence individuals, teams and organizations. Resilience engineering frameworks can be used to model resilient capabilities within individual performance [9], and the framework for small teams can be used to situate individual performance within an organizational context [5]. The resilience goals set by designers and stakeholders, and negotiated within the co-design process will influence the interpretation of practices within work episodes as being indicators of resilient or brittle practice.

4.3 Projecting: Knowing the goals for the future

As with provocation, the projecting mechanism requires negotiation between designers and stakeholders about what might constitute an improved future [18]. In the case of automation, such negotiations include imagining how the elements of practice that are indicative of resilient performance today should be included or accounted for in the future.

The motivating example illustrates limitations in the ability of the technologies used by the air traffic controllers to keep the system resilient. Though instruments supported controllers in the form of calculations and other information, they did not directly conduct decision making.

Projection activities should envision the kinds of automated futures that would be acceptable or permitted in similar situations. Though the nature of the projected automation designs may be fundamentally different, organisational values and objectives from the past may persist, and influence newer designs. For example, how would decision making or coordination in the motivating example be changed if planes were pilot-less? How would the behaviour of the observing air traffic controller have been changed if the industry used technologies that today are applied in different domains, such as gestural interfaces or chips embedded in bodies?

To imagine how automation might be used in the future and the effect this might have on practice, two methods to

use are design fictions [3] for modelling different worlds, and visualization systems inspired by superhuman powers [17].

4.4 Prototyping: Accounting for socio-technical resilience

One practical challenge in applying ResE, like other socio-technical analytic frameworks [1], is to move beyond an analysis of the system as-it-is toward a future state. Future work could plan for incremental changes in automation, but could also expand to include ambitious, expansive future scenarios that break the current working context. Projections generated within co-design about a possible future world should coalesce into concrete representations or prototypes that embody those ideas.

To preserve or account for resilience in automation design, prototypes must represent automation aims and test one or more resilient properties. The resilience goal of safety within this example was met through decisions taken by the crew of the planes to follow verbal instructions, and by an observer to the situation who issued a spontaneous instruction. These human behaviours may be regarded as resilient and worth preserving or accounting for in future systems.

Thus prototypes may include features that allow humans to direct behaviour, to observe and intervene, as in the motivating example, or may instead trial instances in which instruments alone are capable of issuing instructions, or of directing the behaviour of planes on approach and takeoff.

As a part of automation design, prototypes can serve as a lightweight form of validation [1] or assessment of the impact of proposed changes in technology on resilient performance. Prototyping is common in user experience and human-computer interaction and a number of high- and low-fidelity techniques are available [8] to test resilience properties in relation to authentic tasks. Likewise, while "In the wild" prototyping would not be appropriate in a domain like aviation, simulation is appropriate and already widely used for training [13].

5 CONCLUSION

Planning new forms of automation that take on increasing levels of decision making raises questions about what to automate and what to leave in human control. Given the fast pace of technological change, how can the future be productively envisioned by software designers to exploit the advantages of new technology while meeting the needs of the social systems in which software is used and embedded? This paper presents a step toward managing this complex process with an argument for resilient socio-technical software design. Resilience engineering gives insight to human aspects of performance that contribute to system resilience and co-design mechanisms offer a straightforward way to organise and focus design activities within domains. Used together, they offer a way to represent, imagine, and evaluate representations of human practice that should be preserved or accounted for in future software systems.

ACKNOWLEDGMENTS

We thank the contributors to project WREN[1] and the reviewers for their generous insights. Work supported by the Engineering and Physical Sciences Research Council (EP/T017465/1).

REFERENCES

[1] Gordon Baxter and Ian Sommerville. 2011. Socio-technical systems: From design methods to systems engineering. *Interacting with computers* 23, 1 (2011), 4–17. Publisher: OUP.
[2] Stevie Chancellor. 2023. Toward Practices for Human-Centered Machine Learning. *Commun. ACM* 66, 3 (2023), 78–85.
[3] Paul Coulton, Joseph Galen Lindley, Miriam Sturdee, and Michael Stead. 2017. Design fiction as world building. GBR, 16.
[4] Beth Crandall, Gary A. Klein, and Robert R. Hoffman. 2006. *Working minds: A practitioner's guide to cognitive task analysis*. Mit Press.
[5] Dominic Furniss, Jonathan Back, Ann Blandford, Michael Hildebrandt, and Helena Broberg. 2011. A resilience markers framework for small teams. *Reliability Engineering & System Safety* 96 (2011), 2–10. Publisher: Elsevier.
[6] Carlos Gavidia-Calderon, Amel Bennaceur, Anastasia Kordoni, Mark Levine, and Bashar Nuseibeh. 2022. What do you want from me?: adapting systems to the uncertainty of human preferences. In *Proceedings of the ACM/IEEE 44th International Conference on Software Engineering: New Ideas and Emerging Results*. ACM, Pittsburgh Pennsylvania, 126–130.
[7] Martyn Hammersley and Paul Atkinson. 2019. *Ethnography: Principles in practice*. Routledge.
[8] Sharp Helen, Preece Jenny, and Rogers Yvonne. 2023. Interaction Design: Beyond Human-Computer Interaction.
[9] Erik Hollnagel. 2017. *Safety-II in practice: developing the resilience potentials*. Taylor & Francis.
[10] Jean-Claude Laprie. 2008. From dependability to resilience. In *38th IEEE/IFIP Int. Conf. On dependable systems and networks*. G8–G9.
[11] Tamara Lopez, Helen Sharp, Michel Wermelinger, Melanie Langer, Mark Levine, Caroline Jay, Yijun Yu, and Bashar Nuseibeh. 2023. Accounting for socio-technical resilience in software engineering. In *2023 IEEE/ACM 16th International Conference on Cooperative and Human Aspects of Software Engineering (CHASE)*. 31–36. ISSN: 2574-1837.
[12] Nick McDonald. 2006. Organisational resilience and industrial risk. In *Resilience Engineering - Concepts and Precepts*, Woods D. Hollnagel, E. and N. Leveson (Eds.). CRC Press, 155–180.
[13] Maryam Safi and Joon Chung. 2023. Augmented Reality Uses and Applications in Aerospace and Aviation. In *Springer Handbook of Augmented Reality*, Andrew Yeh Ching Nee and Soh Khim Ong (Eds.). Springer International Publishing, 473–494.
[14] Swedish Accident Investigation Authority (SHK). [n. d.]. *Final report RL 2012: 03e – Serious incident on 21 January 2011*. Final Report RL 2012: 03e.
[15] S. H. Stroeve, B. A. Van Doorn, and M. H. C. Everdij. 2013. The human contribution-Analysis of the human role in resilience in ATM. In *Deliverable D1.2. EU FP7 Resilience 2050*.
[16] Jon Whittle, Maria Angela Ferrario, Will Simm, and Waqar Hussain. 2019. A case for human values in software engineering. *IEEE Software* 38, 1 (2019), 106–113.
[17] Wesley Willett, Bon Adriel Aseniero, Sheelagh Carpendale, Pierre Dragicevic, Yvonne Jansen, Lora Oehlberg, and Petra Isenberg. 2021. Perception! immersion! empowerment! superpowers as inspiration for visualization. *IEEE transactions on visualization and computer graphics* 28, 1 (2021), 22–32. Publisher: IEEE.
[18] Theodore Zamenopoulos and Katerina Alexiou. 2018. *Co-design as collaborative research*. Bristol University/AHRC Connected Communities Programme.

[1] https://stride.org.uk/2023/08/09/collaboration-with-nats/