Navigating Development of Cyber-Physical Systems through Inconsistency Scenarios

Thomas Alexander Voelk [1]*, Eva-Maria Grote [1], Razieh Dehghani [2], Felix Pfaff [1], Jonas Liebschner [3], Anne Koziolek [2], Tobias Düser [1], Albert Albers [1]

[1]Karlsruhe Institute of Technology (KIT), IPEK – Institute of Product Engineering, Kaiserstr. 10, 76131 Karlsruhe
[2]Karlsruhe Institute of Technology (KIT), KASTEL – Institute of Information Security and Dependability, Am Fasanengarten 5, 76131 Karlsruhe
[3]Karlsruhe Institute of Technology (KIT), KSRI – Karlsruhe Digital Service Research & Innovation Hub, Kaiserstr. 89, 76133 Karlsruhe
*Corresponding author. E-Mail address: thomas.voelk@kit.edu; Tel.: +49 721 608-47253

Abstract: The development of cyber-physical systems (CPS) is characterized by high complexity, heterogeneous engineering artifacts, and strong cross-domain dependencies, making the achieving of consistency an inevitable part of engineering practice. While existing approaches primarily focus on detecting and resolving inconsistencies, engineers often need to understand, assess, and navigate them as part of ongoing decision-making. In practice, however, methodological support for dealing with inconsistency is limited and frequently depends on expert availability.

This paper proposes a scenario-based approach to support engineers and project managers in managing inconsistency in CPS development. Building on previously identified key factors for inconsistency occurrence, inconsistency scenarios are derived using established scenario creation techniques and implemented in both expert-based and AI-supported assessment formats. The approach enables structured self-assessment and reflection on development environments without requiring immediate inconsistency resolution. The proposed support is evaluated in a LiveLab study using a questionnaire-based evaluation within an ASD Innovation format. The results indicate that inconsistency scenarios effectively raise awareness of inconsistency-related risks, support self-assessment, and facilitate structured discussion of (in-)consistency management in complex CPS development contexts. By combining scenario-based reasoning with AI-supported assessment, this work contributes a scalable and human-centered decision support approach for Advanced Systems Engineering.

Keywords: Inconsistency Management, Inconsistency Scenario, Cyber-Physical Systems, AI-based Development Support

1 Introduction

Product development of advanced systems such as cyber-physical systems (CPS) remains challenging. CPS are integrations of computation and physical processes [1], connecting mechatronic systems to the possibilities of the Internet of Things [2]. Such autonomous, connected, and intelligent systems require close collaboration across engineering domains, which significantly increases development complexity, cost pressure, and regulatory demands. In the context of Advanced Systems Engineering (ASE), engineers are confronted with heterogeneous models, methods, and organizational structures that must be aligned continuously throughout the development process. [3,4]

A central challenge in this setting is inconsistency [5,6]. When two or more domain-specific models or artifacts are intended to represent the same state of the system but contradict each other, inconsistencies

International Symposium on Hybrid Intelligence in Product and Production Engineering, March 24–26 2026, Paderborn, Germany

arise. Managing such inconsistencies is a recurring and well-known task in engineering practice, yet it remains difficult due to the distributed nature of development activities and the heterogeneity of engineering artefacts [6]. While existing approaches and tools provide support for syntactic and semantic consistency checking, often in the context of model-based systems engineering (MBSE), they primarily focus on detection and resolution [7]. However, in real-world CPS development, inconsistencies often cannot be resolved immediately and instead must be understood, assessed, and managed as part of ongoing decision-making [6].

This challenge is further intensified by practical constraints in industry [8]. Design teams frequently lack direct access to methodological support through coaches or domain experts, particularly when addressing emerging technologies or future-oriented topics for which internal expertise is not yet established [6–8]. Consequently, engineers and project managers require accessible, self-applicable decision support that helps them to assess unclear situations and reflect on potential courses of action without relying on external intervention.

In this work, we argue that inconsistencies can serve as a valuable entry point for such support if they are explicitly structured and represented as inconsistency scenarios. Rather than aiming for immediate resolution, inconsistency scenarios allow engineers to explore different configurations of development environments, assess inconsistency-related risks, and navigate complex project setups. Building on scenario techniques from strategic foresight, we propose a scenario-based support approach that enables self-assessment and indication of management change potential for CPS development environments. By doing so, this work contributes to decision support in product creation and addresses a gap between existing inconsistency management methods and the needs of human-centered CPS engineering practice.

To address these research questions, the paper is structured as follows. Section II provides a literature-based foundation by reviewing related work on inconsistency management, scenario-based approaches, and decision support in CPS development. Section III formulates the research goal and research questions and outlines the applied methodology. Section IV presents the derivation of inconsistency scenarios and introduces their implementation in visual scenario profiles as a means of supporting self-assessment and decision-making. Section V evaluates the proposed approach in a LiveLab study across different CPS development contexts. Section VI discusses the results, implications for engineering practice, and limitations of the approach. Finally, Section VII concludes the paper and outlines directions for future research.

2 Related Work

2.1 Inconsistency in CPS engineering

In the engineering of advanced systems, major issues arise when organizations try to homogenize domain-specific engineering activities and orchestration of activities on a cross-domain level [4]. This orchestration effort is mostly connected to the issue of consistency between the models: consistency describes the state of a set of different models which represent the same system-of-interest, but are modelled from different disciplines point of view, and do not possess any contradictions in the set of models [8,9]. While the state of consistency is the target state of product development, inconsistency (or the absence of consistency) is the regular engineering state [5]. Especially in the engineering of complex CPS, such inconsistencies propose a major challenge to engineering organizations: Engineers might not be aware of consistency relations and semantic overlaps between domain-specific models [10], introducing a need for conscious inconsistency management [6,7].

Currently, approaches supporting the conscious management of inconsistency mainly focus on two fields of actions: Tool-based support for consistency preservation, e.g. in MBSE or other applications (virtual) single underlying models [6–8,10], or methods for inconsistency management, e.g. decision support for inconsistency problem solving [6–8]. While the first is well-discussed, especially the second field of action offers potential for further investigation [6,7]: Inconsistency itself is currently not a well-understood term for practitioners from engineering [5]. As communication of specific inconsistency situations is not trivial for engineering practitioners [5], a structured canvas was developed to record inconsistency situations [11]. While workshops were able to give first insights into the occurrence of inconsistency situations in CPS engineering environments [12], limitations became obvious: While in general rated as relevant, industry experts deemed the derived insights as to general to directly implement them into engineering practices [12,13]. As inconsistency is a concept not familiar with most practitioners, the derivation of possible measures is heavily reliant on expert knowledge available in workshops rather than self-responsible assessments.

To enable the evaluation of an organizations boundary conditions that might lead to inconsistency, Albers et al. [13] introduce a set of 14 possible key factors for inconsistency occurrence in engineering organizations based on a connectivity analysis in a Design-Structure-Matrix. The resulting factors are given in Table 1.

Table 1. Key factors for inconsistency occurrence based on Albers et al. [13]

1	Homologation	8	Development time
2	Risk tolerance	9	Share of new development
3	Development process	10	Degree of maturity
4	Planning stability	11	Complexity
5	Regulatory requirements of processes	12	Number of variants of a system
6	Documentation type	13	Number of boundary artifacts
7	Documentation scope	14	Usage of modeling methodology

During validation in a systems engineering workshop, the factors and their derivation were evaluated as useful and correct. Still, a transfer to industrial practice was not obvious to the workshops participant without further methodological support.

2.2 AI in engineering design support

A main hindrance in the derivation of inconsistency situation descriptions is the lack of experience and expertise around the phenomenon of inconsistency. This challenge is reportedly one of the major challenges of scaling established support measures. One way to overcome such challenges is the application of Artificial Intelligence (AI): It offers the potential to effectively support innovation activities throughout the entire process from problem exploration and ideation to concept development, selection, validation, and (digital) prototyping. [14–17]. In general, tools using artificial intelligence (AI) become more and more relevant in engineering environments: AI-tools can be identified in the whole product development process in the engineering of advanced systems [18]. Although more and more present, most small and medium-sized companies (SMEs) only use AI-tools for administrative and routine tasks and do not integrate AI-tools into their core activities [19].

Still, the introduction of AI tools comes with a serious potential: With the rise of AI-tools based on human language, expert knowledge becomes more and more accessible: Various studies show that the application of agentic AI based on large-language models (LLM) significantly supports mass-integration of expert knowledge, workshop moderation and methodological support, that previously was only possible with human intervention (e.g. [20–22]).

3 Research Goal, questions, and Methodology

As introduced in section 2, currently inconsistency management support is not sufficiently available, as expert knowledge is currently limited in this emerging field. Still, for scaled investigation of inconsistency management in practice, support in investigation of inconsistency is needed. While previous work enables the general assessment of relevant factors for managing inconsistency, it is not operationalized in a concrete design support.

As identified, the scaled application of already established support for inconsistency management currently is limited by the number of inconsistency management experts. Therefore, the goal of this work is to facilitate AI-supported inconsistency assessment to provide possible scalable support for inconsistency management.

To derive such a support, the following questions should be answered:

- How can a possible evaluation framework for inconsistency occurrence in CPS engineering environments be designed?
- How can a possible evaluation framework be implemented to support a combined expert- and AI-based evaluation of inconsistency occurrence in CPS engineering environments?
- What performance can be measured both for expert-based and AI-based evaluation of inconsistency occurrence in CPS engineering?

This work builds on the research methodology of Marxen and Albers [23] by building on previously identified key factors for inconsistency occurrence from literature and industrial practice [13] building the first phase of the research methodology. Based on those factors, a proposed support is implemented to answer the research questions.

To answer the first research question, we want to lean our framework on the established approach for scenario creation introduced in Scenario Management™ by Fink & Siebe [24]. The approach was originally developed for foresight, as interconnected and highly dependable complex networks of factors hinder the prediction of future developments. Like foresight, inconsistency occurrence is not predictable easily, but is related to a complex network of interconnected factors, as identified in previous work by Albers et al. [13].

As the definition of a concrete future is not problem-oriented (as it will probably not become reality at all), Fink & Siebe propose to focus on a space of potential futures introducing an operating space. We want to lean our approach on the same idea: if we can’t predict the inconsistency occurrence for a concrete situation and development environment, we try to set up a space of extreme scenarios, making it possible to identify the current relation of the own situation of practitioners and their corresponding engineering environment.

Fink & Siebe [24] propose the following procedure to derive such scenarios:

- Derivation of key factors based on the connectivity of influencing factors
- Derivation of (two) possible orthogonal dimensions of a key factor displaying possible development directions
- Derivation of projections as combinations of orthogonal dimensions for each key factor
- (For a top-down approach): Selection of consistent paths through a morphological matrix containing the projections for each key factor

Based on the consistent paths, plausible scenarios are build, containing a scenario name, visualizations, and descriptions.

To answer the second research question, we want to implement the generic framework developed by answering the first research question into two separate design supports: an expert-based evaluation kit consisting of a workshop presentation and scenario profiles, and an AI-based evaluation chatbot based on an AI toolbox.

To answer the third research question, the proposed support is evaluated in a LiveLab setting [25] using a structured inconsistency awareness training. The evaluation focuses on assessing the applicability, usability, and perceived benefits of the inconsistency scenarios and their representations in realistic CPS development contexts. The training combines scenario-based reflection with an AI-supported self-assessment and follows a multi-stage approach including individual assessment, collaborative discussion, and plenary reflection. Data is collected using a questionnaire-based evaluation capturing both quantitative and qualitative feedback on inconsistency awareness, decision support, and practical relevance.

4 Deriving inconsistency scenarios leaned on scenario creation

In this section, the first research questions is answered based on the introduced approach for scenario creation by Fink & Siebe [24]. As Albers et al. [13] already provide a set of key factors, this work starts with the second step of the procedure. In an online workshop conducted in Miro, the introduced factors were discussed by the authors regarding two possible orthogonal (meaning independent) dimensions of the factors. The dimensions identified are given in Table 2.

Table 2: Identified orthogonal dimensions of key factors

Factor	**Dimension 1**	**Dimension 2**
Homologation	Number of regulations	Complexity of testing procedures
Regulatory Requirements of the Process	Number of regulations	Source of requirements
Risk Tolerance	Independence of developers	Error tolerance of managers
Documentation Scope	Scope	Level of detail
Development Process Model	Level of detail	Generalizability
Planning Stability	External volatility	Internal volatility
Documentation Type	Structure	Form
Time to Market	Industry cycle time	Resource availability
Degree of Maturity of the SiD	Development phase within the company	Novelty of the technology within the company
Share of New Development	GV share	PV share
Complexity	Number of links	Dynamics of the system
Number of Variants of a System	Degree of individualization	Degree of modularization
Number of Boundary Artifacts	Integration of development domains	Number of domains
Usage of Modeling Methods	Necessary methods	Competence in application

Based on those dimensions, different projections were constructed by the authors: According to Fink & Siebe [24] for each dimensions, two extreme values were defined. The values are per default set to “Low” and “High” where applicable. This results in a 2x2 matrix of possible value combinations, resulting in four projections per key factor. Due to special limitations of the given work, we only give an example of a possible projection in Table 3.

Table 3: Example for possible dimensions and corresponding projections for key factor 4 “Documentation Scope”

Factor 4: Documentation Scope	Dimension 1: Level of detail - Low	Dimension 1: Level of detail - High
Dimension 2: Scope - Low	"Compact basic recording: Only selected content is documented in a concise and simple form."	"Focused in-depth analysis: The documentation is limited to a few topics, but covers them in great detail."
Dimension 2: Scope - High	"Broad overview: A large part of the subject area is documented, but with a rather superficial presentation."	"Complete and comprehensive documentation: The documentation covers all relevant content comprehensively and in great depth."

Based on the resulting projections, a morphological matrix was constructed: For each factor, each projection was defined with an identifier. Now, alternative paths are constructed through a sequence of key factors. For each path, each key factor only is allowed to be represented by one projection. The resulting eight inconsistency scenarios are given in Table 4.

Table 4: Resulting Inconsistency Scenarios

Scenario	Name	Short Description
1	Little Kingdoms – Management Controlled	Isolated powerful project managers compete against each other, but keep order in their development environment
2	Lone Wolf Projects	Isolated and experienced engineers are relying on their knowledge and are mostly operating ad-hoc
3	Overwhelmed Engineers	Developers have high freedom to operate, but are not sufficiently equipped for the liabilities coming with it
4	High-Performer Teams	Self-responsible and highly adaptable expert engineering teams perform complex tasks, but tend to ignore broad documentation standards.
5	Creative and Explorative	Exploration-friendly environment with high risk affinity, where processes are mostly ad-hoc and learning cycles are fast, but uncoordinated
6	Core Team with Strong Tool Support – Quality Oriented	Precisly defined processes, standards and tool environments, focusing on the development of the best possible quality outcome oriented to customer demands
7	Core Team with Strong Tool Support – High Tech Dissemination	Precisly defined processes, standards, and tool environments, focusing on the development of systems with the highest complexity to enable technology push.
8	Classic Supplier Business	Development environment that is defined by high outer volatility and dependence on OEMs.

5 Implementing inconsistency scenarios into expert- and AI-based support

In this section, the second research questions should be answered:

2. How can a possible evaluation framework be implemented to support both expert-based and AI-based evaluation of inconsistency occurrence in CPS engineering environments?

We want to answer this question by introducing two approaches to implement the derived inconsistency scenarios: Based on Fink & Siebe [24], we want to derive a first implementation based on visualization of the scenarios. We here chose a playing-card format: Each scenario is made accessible by providing two pages of information printable on DIN A5 cards (an example is given in Figure 1). Each playing card contains the following content:

- **Title** according to scenario framework in section 4
- **Picture**: Each scenario has an associated symbolic picture generated by using a generative AI toolbox. They are constructed by inserting the brief scenario description in Table 4 as well as a reference design oriented on the same graphic style
- **Summary**: The summary gives the information of the brief scenario description in Table 4 in the context of the corresponding projections of the given scenario.
- **Detailed factor description**: A list of relevant key factors and the given projection

The complete set of playing cards is accessible on GitLab [26].

Key Factor	Description
Risk Tolerance	Controlled safety structure
Development Process Model	Tailored process architecture: detailed, but case-specific
Homologation	Opaque case-by-case regulation: low regulatory density with complex, in-depth review processes
Planning Stability	Stable planning base: consistent conditions and high predictability
Regulatory Requirements	Internally defined regulation density: many internal standards
Documentation Scope	Complete and comprehensive: all topics thoroughly covered
Documentation Type	Digital full text: document-centered, text-based
Time to Market	Long and stable team journey: ample time and resources
Degree of Maturity	System optimization: improvement using familiar technologies
Share of New Development	Incremental adaptation: minor updates to existing principle and design
Complexity	Stable networked system: many links, stable behavior
Variants	Tailored single solutions: high customization, no modularity
Boundary Artifacts	Focused domain integration: few domains, strongly integrated
Modelling Methodology	Overstrained modeling: high methodological demand, insufficient competence

Figure 1: Example visualization of the inconsistency scenario in the playing-card format

Besides the well-established visualization elements described by Fink & Siebe, we want to introduce an interactive AI-based chat bot inside of an AI toolbox as an alternative solution. Based on the same used textual information available for the playing cards, a custom GPT chat bot is implemented. The chat bot provides a step-by-step routine assessing the current best fit towards one of the related projections for each key factor. After assessing the 14 key factors, a best fit towards one of the constructed scenarios is given. To reproduce the given AI-based chat bot, the JSON-file is given on GitLab [26].

6 Evaluating Inconsistency Scenarios in a LiveLab Study

The proposed inconsistency scenarios and their representations were evaluated in a LiveLab setting using a questionnaire-based evaluation. The evaluation aimed to assess whether the approach supports inconsistency awareness, self-assessment, and navigation of complex CPS development situations, as well as to evaluate the usability of the AI-supported assessment and the scenario profiles. The study was conducted within an ASD-Innovation format during the conceptualization phase of CPS development and involved hands-on application followed by structured reflection.

A total of N = 10 participants took part in the LiveLab and completed a questionnaire comprising 21 items, addressing conceptual understanding of inconsistency, self-efficacy, awareness of inconsistency-related risks, future applicability, and the usability of both the AI-based assistant and the scenario profiles. The questionnaire was administered after the LiveLab session, following individual assessment, collaborative discussion, and plenary reflection.

The results indicate that the training effectively established a shared understanding of inconsistency in CPS and strengthened participants' confidence in identifying and assessing inconsistencies. 80% of participants agreed or strongly agreed that they understand what inconsistency in cyber-physical systems means, while 20% expressed slight disagreement. Similarly, 80% of participants reported increased confidence in identifying and assessing inconsistencies themselves, whereas 20% remained neutral. Moreover, the training significantly raised awareness of inconsistency-related risks, with 80% of participants agreeing or strongly agreeing that the training clearly highlighted such risks, while 10% expressed slight disagreement and 10% remained neutral. In terms of future applicability, 80% of participants indicated that they plan to apply the learned approach in future projects, while 10% expressed slight disagreement and 10% remained neutral. This suggests that the approach is perceived as relevant for practical engineering work beyond the training context. Participants' self-assessed competence in dealing with inconsistency showed a more differentiated picture.

The AI-based assistant used for the assessment was generally perceived as usable and supportive. 40% of participants strongly agreed and 40% agreed that the assistant helped them reflect on their own development situation, resulting in an overall agreement of 80%. Similarly, 60% of participants perceived the interaction with the assistant as intuitive, while 10% expressed slight disagreement and 10% remained neutral. The clarity of the assistant's questions received more mixed feedback, with 40% agreement, 20% slight disagreement, and 30% neutral responses. Furthermore, 50% of participants agreed that the assistant's classification of their project matched their own assessment, while 10% expressed slight disagreement and 30% remained neutral, suggesting that the automated assessment was generally plausible but not always fully aligned with subjective perceptions.

The scenario profiles were evaluated very positively. 90% of participants agreed or strongly agreed that the contents were clearly and understandably presented, and 80% found the structure of the profiles easy to follow. 80% reported that they could quickly grasp differences between scenarios, and 70% agreed that the profiles supported them in identifying potential inconsistencies within a scenario. Regarding practical applicability, 60% of participants agreed that the scenario profiles are usable in a real work context, while 20% remained neutral. 80% perceived the information provided as relevant for real development or analysis processes, and 90% indicated that the profiles could support them in assessing context conditions in future projects. Finally, 70% of participants agreed that the profiles are sufficiently compact to be used in practice.

7 Discussion

Overall, the evaluation results indicate that the proposed inconsistency scenarios, combined with the AI-supported assessment and scenario profiles, are effective in raising inconsistency awareness, supporting reflection, and enabling structured discussion of inconsistency-related risks. At the same time, the results highlight the need for continued use and repeated application to further strengthen individual competence in dealing with inconsistency in CPS development.

The evaluation results indicate that inconsistency scenarios are a promising means to support engineers and project managers in navigating complex CPS development situations. By shifting the focus from immediate inconsistency resolution toward awareness, reflection, and decision support, the proposed approach addresses a gap in existing inconsistency management methods, which predominantly emphasize detection and correction.

A key contribution of this work is demonstrating that scenario-based representations can effectively support a shared understanding of inconsistency in CPS development. The evaluation shows that participants were able to grasp the concept of inconsistency and reflect on its implications for their own development environments. By explicitly structuring inconsistency through deliberately accentuated scenarios, the approach makes inconsistency discussable and supports cross-disciplinary communication. The results further suggest that inconsistency scenarios facilitate self-assessment and collaborative reflection, particularly when used in interactive formats. The scenarios acted as boundary objects that supported the comparison of perspectives across roles and disciplines, highlighting their value as communicative artifacts in human-centered CPS engineering.

The integration of an AI-based assessment assistant increased the accessibility and scalability of the approach. While the assistant supported structured reflection, differences between automated classifications and participants' subjective assessments underline that AI-based support cannot replace contextual knowledge and expert judgment. Instead, it should be understood as a complementary decision-support mechanism**.** The scenario profiles were perceived as clear, compact, and relevant for practice, indicating that the chosen level of abstraction is suitable for early-stage assessment and discussion. However, the more differentiated responses regarding perceived competence suggest that increased awareness does not automatically translate into expertise, pointing to the need for repeated application and integration into everyday engineering practice.

The findings should be interpreted in light of several limitations. The evaluation was conducted in a LiveLab setting with a limited number of participants and relied primarily on self-reported perceptions rather than objective performance measures. Consequently, the results are exploratory and require further validation in longitudinal and industrial-scale studies.

8 Conclusion

This paper introduced a scenario-based approach for navigating inconsistency in cyber-physical systems development. By structuring inconsistency through deliberately accentuated scenarios and implementing them in expert-based and AI-supported assessment formats, the approach supports engineers and project managers in raising awareness, reflecting on development contexts, and navigating inconsistency-related risks. The results of a LiveLab evaluation indicate that inconsistency scenarios effectively support understanding, self-assessment, and decision-making in complex CPS environments. Overall, this work demonstrates that treating inconsistency as a manageable and discussable aspect of development offers a promising contribution to human-centered and scalable decision support in Advanced Systems Engineering.

References

[1] E. A. Lee, "CPS Foundations," in *Proceedings of the 47th Design Automation Conference*: New York, NY, USA: ACM, 2010, pp. 737–742.

[2] I. Graessler and J. Hentze, "The new V-Model of VDI 2206 and its validation," at - Automatisierungstechnik, vol. 68, no. 5, pp. 312-324, 2020. DOI: https://doi.org/10.1515/auto-2020-0015.

[3] R. Dumitrescu, A. Albers, O. Riedel, R. Stark and J. Gausemeier, Engineering in Germany - The status quo in business and science, a contribution to Advanced Systems Engineering, 2021.

[4] Rethinking and Reframing Engineering. Challenges, Application Scenarios and the New Advanced Systems Engineering Model, acatech IMPULSE, 2023.

[5] A. Albers, A. Koziolek, T. A. Völk, M. Klippert, F. Pfaff, R. Stolpmann and S. E. Schwarz, "Identification of Inconsistencies in Agile CPS Engineering with Formula Student," in *Proceedings of The XXXV ISPIM Innovation Conference*, 2024, pp. 1–15.

[6] A. Cicchetti, "Consistency and Uncertainty in the Development of Cyber-Physical Systems," in *MPM4CPS: Multi-paradigm modelling for cyber-physical systems*, H. Vangheluwe, V. Amaral, H. Giese, J. Broenink, B. Schätz, A. Norta, P. Carreira, I. Lukovic, T. Mayerhofer, M. Wimmer and A. Vallecillo, eds., 2016, pp. 56–61.

[7] S. Feldmann and B. Vogel-Heuser, “Diagnose von Inkonsistenzen in heterogenen Engineeringdaten,” in *Handbuch Industrie 4.0*, B. Vogel-Heuser, M. ten Hompel and T. Bauernhansl, eds.: Berlin: Springer Vieweg, 2024, pp. 909–928.

[8] G. Spanoudakis and A. Zisman, “Inconsistency Management in Software Engineering: Survey and Open Research Issues,” in *Handbook of Software Engineering and Knowledge Engineering*, S. K. Chang, ed.: World Scientific, 2001, pp. 329–380.

[9] A. Albers and Q. Lohmeyer, “Advanced Systems Engineering-Towards a Model-Based and Human-Centered Methodology,” in *9th International Symposium on Tools and Methods of Competitive Engineering TMCE*, 2012, pp. 407–416.

[10] R. Reussner, I. Schaefer, B. Beckert, A. Koziolek and E. Burger, “Consistency in the View-Based Development of Cyber-Physical Systems (Convide),” in *2023 ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)*: IEEE, 2023, pp. 83–84.

[11] T. A. Voelk, R. Dehghani, M. Klippert, F. Pfaff, R. Stolpmann, K. Nowak, A. Koziolek and A. Albers, “Structuring Inconsistency Situations in Engineering of Cyber-Physical Systems - a Description Template Proposal,” Procedia CIRP, vol. 136, pp. 49-54, 2025. DOI: https://doi.org/10.1016/j.procir.2025.08.011.

[12] T. A. Voelk, K. Nowak, F. Pfaff, R. Dehghani, T. Düser, A. Koziolek and A. Albers, “Consistently Inconsistent: AI-Analyzed Patterns in Inconsistency-Situations of Cyber-Physical Systems Development,” in *2025 IEEE International Symposium on Systems Engineering (ISSE)*: IEEE, 2025, in press.

[13] A. Albers, A. Koziolek, T. A. Voelk, E.-M. Grote, R. Dehghani and T. Düser, “How to "Measure" a Mirage: Assessing Inconsistency in Mechatronic and Cyber-Physical Development Projects,” in *2025 IEEE International Symposium on Systems Engineering (ISSE)*: IEEE, 2025, in press, pp. 1–8.

[14] S. G. Bouschery, V. Blazevic and F. T. Piller, “Augmenting human innovation teams with artificial intelligence: Exploring transformer-based language models,” Journal of Product Innovation Management, vol. 40, no. 2, pp. 139-153, 2023. DOI: https://doi.org/10.1111/jpim.12656.

[15] C. Kakatkar, V. Bilgram and J. Füller, “Innovation analytics: Leveraging artificial intelligence in the innovation process,” Business Horizons, vol. 63, no. 2, pp. 171-181, 2020. DOI: https://doi.org/10.1016/j.bushor.2019.10.006.

[16] V. Bilgram and F. Laarmann, “Accelerating Innovation With Generative AI: AI-Augmented Digital Prototyping and Innovation Methods,” IEEE Engineering Management Review, vol. 51, no. 2, pp. 18-25, 2023. DOI: https://doi.org/10.1109/EMR.2023.3272799.

[17] J. Holmström and N. Carroll, “How organizations can innovate with generative AI,” Business Horizons, vol. 68, no. 5, pp. 559-573, 2025. DOI: https://doi.org/10.1016/j.bushor.2024.02.010.

[18] N. Afifi, C. Wittig, L. Paehler, A. Lindenmann, K. Wolter, F. Leitenberger, M. Dogru, P. Grauberger, T. Düser, A. Albers and S. Matthiesen, Data-Driven Methods and AI in Engineering Design: A Systematic Literature Review Focusing on Challenges and Opportunities, 2025.

[19] Generative AI and the SME Workforce.

[20] M. Houtti, M. Zhou, L. Terveen and S. Chancellor, “Observe, Ask, Intervene: Designing AI Agents for More Inclusive Meetings,” in *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*: New York, NY, USA: ACM, 2025, pp. 1–18.

[21] S. Tegos, A. Mavridis and S. Demetriadis, “Agent-Supported Peer Collaboration in MOOCs,” Frontiers in artificial intelligence, vol. 4, p. 710856, 2021. DOI: https://doi.org/10.3389/frai.2021.710856.

[22] A. Mastroianni, L. Rampino and F. A. Figoli, “LARGE LANGUAGE MODELS IN THE DESIGN PROCESS: A WORKSHOP ON THE POSSIBILITIES OF HUMAN-AI COLLABORATION THROUGH KNOWLEDGE FORMATION,” in *ICERI2024 Proceedings*, ICERI Proceedings: IATED, 2024, pp. 5784–5794.

[23] L. Marxen and A. Albers, “Supporting Validation in the Development of Design Methods,” in *DS 70: Proceedings of DESIGN 2012, the 12th International Design Conference, Dubrovnik, Croatia*, 2012, pp. 1009–1018.

[24] A. Fink and A. Siebe, Szenario-Management. Von strategischem Vorausdenken zu zukunftsrobusten EntscheidungenFrankfurt, New York: Campus Verlag, 2016.

[25] A. Albers, T. Düser, S. Ott, M. Klippert, F. Pfaff, S. E. Schwarz, M. Kuebler, C. Thümmel, M. Schlegel and K. Bause, “Co-Creation zwischen Universitäten und Unternehmen: ASD - Agile Systems Design Innovationsformate,” in *Stuttgarter Symposium für Produktentwicklung SSP 2025 : Tagungsband zur Konferenz, Stuttgart, 22. Mai 2025*: Stuttgart : Fraunhofer IAO, 2025.

[26] T. A. Voelk and Di Gao, “Inconsistency Scenario Implementation,” [Online]. Available: https://gitlab.kit.edu/thomas.voelk/inconsistency-scenario-implementation/-/tree/29912ba9121b3b2639421bde77c0465230912317/.