

HIPPE

International Symposium on Hybrid Intelligence
in Product and Production Engineering

Paderborn 2026



1st INTERNATIONAL SYMPOSIUM

March 24 – 26, 2026 • Heinz Nixdorf Institute • Paderborn University

is organized by the
Chair for Product Creation
at

HEINZ NIXDORF INSTITUT
UNIVERSITÄT PADERBORN

in cooperation with

DFG Deutsche
Forschungsgemeinschaft
German Research Foundation



DFG SPP 2443
Hybrid Decision Support
in Product Creation



WiGeP
Wissenschaftliche Gesellschaft
für Produktentwicklung



Wissenschaftliche
Gesellschaft für
Produktionstechnik



EDITORS

Prof. Dr.-Ing. Iris Graessler
Paderborn University

PROGRAMME COMMITTEE

Prof. Dr.-Ing. Iris Graessler
Paderborn University

Prof. Dr. Sören Auer
Leibniz University Hannover

Prof. Dr. Ralph Bergmann
University Trier

Dr. Felix Engel
Leibniz University Hannover

Prof. Dr.-Ing. Detlef Gerhard
Ruhr University Bochum

Prof. Dr. Fred Hamker
Technical University Chemnitz

Prof. Dr. Sc. ETH Alexander Hasse
Technical University Chemnitz

Prof. Dr.-Ing. Bernd Kuhlenkötter
Ruhr University Bochum

Dr.-Ing. Jens Pottebaum
Paderborn University

Prof. Dr.-Ing. Martin Ruskowski
RPTU University Kaiserslautern-Landau

Prof. Dr.-Ing. Rainer Stark
Technical University Berlin

Prof. Dr.-Ing. Matthias Schmidt
Leibniz University Hannover

Prof. Dr.-Ing. Klaus-Dieter Thoben
University of Bremen

Prof. Dr.-Ing. Sándor Vajna
Otto von Guericke University Magdeburg

CONFERENCE ORGANIZING TEAM

Benedikt Grewe, Thomas Hesse, Julius Goldstein
Paderborn University, Heinz Nixdorf Institute, Chair for Product Creation

HIPPE is an endorsed event by the Design Society.
The Design society is a charitable body, registered in Scotland, number SC031694.
Registered Company Number: SC401016.

Editorial

Iris Graessler

Paderborn University, Heinz Nixdorf Institute, Chair for Product Creation, 33100 Paderborn

The HIPPE Conference Series provides an international forum for scientific exchange on the integration of human and Artificial Intelligence (AI) in product and production engineering. The International Symposium on Hybrid Intelligence in Product and Production Engineering (HIPPE) emerged from the German Research Foundation (DFG) Priority Program 2443 “Hybrid Decision Support in Product Creation”, which investigates novel approaches of combining human expertise with AI in engineering decision support.

The 1st edition of HIPPE '26 was held from 24–26 March 2026 in Paderborn, Germany, with a focus on hybrid decision support systems in product and production engineering. The symposium addressed a broad spectrum including AI in engineering processes, data-driven product engineering, human-centered AI systems, smart production technologies, digital engineering methods, autonomous and self-controlling production systems, as well as decision support approaches for complex industrial environments. Through this interdisciplinary range of research topics, the symposium aimed at providing an international platform for the exchange of up-to-date scientific knowledge, innovative applications, and discussions on current challenges and future trends in Hybrid Intelligence in Product and Production Engineering.

The great success of the symposium is due to the excellent contributions of many players: Dr.-Ing. Ursula Frank, Head of R&D Cooperations at Beckhoff Automation, provides us with her keynote presentation “Data Science and AI in Product Engineering: Opportunities and Challenges” in which she outlined current challenges in industry. Dr.-Ing. Jobst Richert and his team made it possible to take a tour and further insights at dSpace. This laid the groundwork for exciting discussions on simulation and validation. Prof. Dr.-Ing. Martin Ruskowski, Head of the Research Department Innovative Factory Systems at DFKI and Head of the Institute of Machine Tools and System Controls (WSKL) at RPTU, addressed in his keynote presentation entitled “With the Smart Product for Self-Controlling Production” further elementary insights. The panel discussion on “Hybrid Intelligence in Product Creation – From Methods to Decision-Making Impact” with Prof. Dr. Sauer, Prof. Dr.-Ing. Martin Ruskowski, and Prof. Dr.-Ing. Rainer Stark was a lively conclusion and gave a preview of future developments.

Prof. Dr.-Ing. Klaus-Dieter Thoben, Dr. Lukas Malburg, Prof. Dr.-Ing. Rainer Stark, Dr.-Ing. Felix Engel, Prof. Dr.-Ing. Detlef Gerhard, Dr.-Ing. Jens Pottebaum, Prof. Dr.-Ing. Martin Ruskowski, and Prof. Dr. sc. ETH Alexander Hasse guided the scientific sessions and initiated discussions beyond the presented research contributions, thereby fostering interdisciplinary exchange and deeper scientific insights. This was one of the most impressive conclusions from HIPPE '26: Outstanding presenters in combination with highly engaged facilitators led to great in-depth discussions and interactive progress, feeding back great value into contributing projects.

Finally, we would like to express our thankfulness to the members of the Programme Committee for their valuable support, expertise and commitment throughout the preparation and realization of the symposium. Their careful reviews, constructive feedback, and scientific guidance significantly contributed to the high academic quality of the event and to shaping a stimulating and inspiring program.

We gratefully acknowledge the support of the German Research Foundation (DFG) in funding the scientific activities of the Priority Program SPP 2443, "Hybrid Decision Support in Product Creation." The DFG's contributions to the scientific network, as well as the support and contributions of the German Academic Association for Production Technology (WGP) and the Wissenschaftliche Gesellschaft für Produktentwicklung (WiGeP), are also gratefully acknowledged. Furthermore, we greatly appreciate the Design Society's inclusion of HIPPE as an endorsed conference, which supports the international visibility and dissemination of the symposium activities.

The HIPPE symposium marked the first step in promoting interdisciplinary exchange between computer science, product and production engineering, and in building a community of researchers. Through this exchange, hybrid decision-making will enhance the performance of the engineering process.

Paderborn, March 2026

Prof. Dr.-Ing. Iris Gräßler
Chair
HIPPE Conference Series

TABLE OF CONTENTS

Practical Guide for Responsible Research Data Management in Academic-Industry Collaboration <i>G. Kremer, J. Balder, H. Ebel, Z. Wang, A. Upadhyay, E. Kempkens, S. B. Hassine, R. Stark</i>	pp. 1 - 10
Designing and Validating Affordant Interfaces for Hybrid Intelligence Systems <i>M. Kretzschmar, B. Saske, J. Mehlstäubl, K. Paetzold-Byhain</i>	pp. 11 - 20
A concept to simulate stakeholders in product development with personas and generative AI <i>K. Wellsandt, K. Hribernik, J. Naumann, K. Thoben</i>	pp. 21 - 30
Performance evaluation of Large Language Models in extracting requirements sets from technical documents <i>I. Graessler, J. Pfeifer</i>	pp. 31 - 40
Framework for Model-Based Systems Engineering Circularity Assessment <i>Y. Juresa, D. Mollahassani, J. C. Göbel</i>	pp. 41 - 50
Automatic Generation of technical Drawings using Model Context Protocol <i>L. Kunz, M. Klostermeier, K. Thanabalan, T. Legler, M. Ruskowski</i>	pp. 51 - 60
Theoretical foundation to model design decisions for hybrid (Human / AI) decision support systems <i>E. Kempkens, A. Upadhyay, Z. Wang, R. Stark</i>	pp. 61 - 70
Evaluating a Decision Support System for Assessing the Product Design Disassemblability <i>A. Markus, J. Haller, E. Borowski, A. Göppert, R. H. Schmitt</i>	pp. 71 - 78
Identification of key parameters for automated assembly <i>M. Jakschik, J. Rolf, K. Bahri, B. Kuhlenkötter</i>	pp. 79 - 88
Multimodal AI Agent for Orchestration of Mobile Robotic Tasks in Flexible Assembly Automation <i>L. M. Pinheiro de Souza, V. Murugasamy, A. Moriz, A. Göppert, R. H. Schmitt</i>	pp. 89 - 98
Hybrid Neuro-Symbolic Decision Support System in Product Planning and Design <i>F. Engel, N. Krzdavac, S. Ghodsi, J. Haller, A. Markus, S. Münker, A. Abdelrazeq, R. H. Schmitt</i>	pp. 99 - 108

From Rust to Reuse: Influence of Preload and Corrosion on the Mechanical Performance of Bolted Joints <i>C. Lange, N. Afifi, D. Gleß, M. Benfer, S. Matthiesen, G. Lanza</i>	pp. 109 - 118
Navigating Development of Cyber-Physical Systems through Inconsistency Scenarios <i>T. A. Voelk, E. Grote, R. Dehghani, F. Pfaff, J. Liebschner, A. Koziolk, T. Düser, A. Albers</i>	pp. 119 - 128
CAD-as-Code and Agentic AI for Function-Aware Engineering Design <i>E. Berger, J. Mehlstäubl, B. Saske, K. Paetzold-Byhain</i>	pp. 129 - 138
Influence of Tokenization Strategies on the Prediction of CAD Model Descriptions <i>S. H. Habibi, J. Bergelt, M. Teichmann, F. H. Hamker</i>	pp. 139 - 148
Comparing Approaches for the Automatic Generation of System Models in Generative Systems Engineering <i>M. Becker, D. Mollahassani, J. C. Göbel</i>	pp. 149 - 158
Graph-based product prototype optimization using multi-objective reinforcement learning: A framework concept <i>S. Munker, H. Aoyang Zhou, A. Abdelrazeq, J. Haller, R. H. Schmitt</i>	pp. 159 - 168
Towards Structured Knowledge Reuse to Support Design for Manufacturability: A Case-Based Reasoning Approach <i>K. Thanabalan, L. Malburg, L. Kunz, R. Bergmann</i>	pp. 169 - 178
AI- and Data Science-Driven Decision Support: Transforming Early Product Creation and Collaboration in Product Development <i>M. Tatschl, G. Schalk, H. Hick</i>	pp. 179 - 188
Literature analysis and development of a metadata model for human capability description in production <i>T. Meinecke, T. Hesse, I. Graessler, M. Schmidt</i>	pp. 189 - 198
Hybrid Decision Support in Strategic Product Planning: Leveraging Extreme Data for Early-Phase Innovation <i>I. Graessler, J. Goldstein</i>	pp. 199 - 208
A Product Meta Data Model of Assembly-Related Extreme Data to enable Hybrid Decision Support <i>I. Graessler, F. Vollenkemper, J. Pottebaum</i>	pp. 209 - 218

Enabling Extreme Data by using DS/AI approaches: A Process Metadata Model to link Assembly Features and required Human Capabilities <i>I. Graessler, T. Hesse, J. Pottebaum</i>	pp. 219 - 228
Harnessing Tacit Knowledge from Sustainable Product Engineering through Artificial Intelligence <i>I. Graessler, J. Mansheim, J. Pfeifer</i>	pp. 229 - 238
Systematic Review of Engineering Simulation Models for Life Cycle Assessment <i>I. Graessler, S. Aydin, S. Rarbach</i>	pp. 239 - 248
Corporate resilience through generic AI-based workflows in strategic product planning <i>I. Graessler, D. Oezcan</i>	pp. 249 - 258

Practical Guide for Responsible Research Data Management in Academic-Industry Collaboration

Gerald Kremer^{1*}, Juliane Balder¹, Helena Ebel¹, Zhongtian Wang¹, Anvit Upadhyay¹, Eric Kempkens¹,
Sahar Ben Hassine¹, Rainer Stark¹

¹TU Berlin, Institut für Industrielle Informationstechnik, 10587 Berlin

*Corresponding author. E-Mail address: kremer@tu-berlin.de

Abstract: Responsible research data management (RDM) is crucial in industry-university collaborations, yet existing frameworks and data management plan (DMP) tools often remain abstract and detached from daily project work. This paper presents a role-based Business Process Model and Notation (BPMN) process map and an interactive prototype that depict the RDM lifecycle from project start to retention across key academic and industrial roles. RDM activities, phases, and roles were derived from lifecycle models, institutional policies, and project experience, then refined in a collaborative modelling workshop. Evaluation with nine researchers using think-aloud walkthroughs and a questionnaire shows that the artefact clarifies responsibilities and actions, while highlighting tensions between completeness and simplicity.

Keywords:

Research data management, RDM, Academic-industry collaboration, Data lifecycle, Process-oriented guidance

1 Motivation and Research Objective

Research Data Management (RDM) is a cornerstone of Data Science (DS) and Artificial Intelligence (AI) research, enabling reproducibility, reuse, and trustworthy insights from real world applications. In the domain of product creation, especially where academia and industry jointly innovate, data curation, a key aspect of RDM, that involves activities from e.g. data capture to storage and access, is crucial [1]. The work presented in the paper focusses on how to curate and manage research data responsibly in academic-industry collaborations, which brings unique challenges. Academics must balance open-science practices with proprietary and confidentiality constraints, ensure data quality, and comply with industry regulations. The relevance of the topic is underscored by the Open Science principles pushing for FAIR (Findable, Accessible, Interoperable, Reusable) data as a basis for know-ledge reuse [2]. At the same time industry partners must safeguard proprietary know-how and potential intellectual property (IP) embodied in datasets, which constrains sharing and reuse, unless protective measures (e.g., contractual restrictions, or technical de-risking such as anonymization/synthetic or algorithmically transformed data or encryptions) are in place.

Guidance through existing RDM frameworks is textual, abstract and often complicated. If researchers read the complete frameworks, they would learn what they should do, but not how these requirements translate into concrete actions for specific roles (e.g. project PI, research assistant, industry partner contact) at specific points in time. Translating high-level RDM principles into day-to-day practice remains difficult. Also prior literature identifies several challenges impeding RDM in practice, including misalignment between research workflows and data management planning, limited resources or incentives for RDM, and concerns about data governance [3].

This led to the research objective to develop a solution that provides easy-to-understand knowledge for and guides relevant stakeholders through the RDM life cycle of academic-industry research projects. Based on the objective, two research questions were derived:

RQ1: How can the RDM lifecycle academic-industry collaborations be formalized into an actionable model that integrates existing frameworks, standards and regulations and experiential knowledge of researchers to provide relevant and understandable knowledge to relevant stakeholders?

RQ2: How do researchers perceive the usefulness of such a developed solution?

2 State of Research

2.1 Conceptual frameworks and lifecycle models for RDM

RDM refers to the set of organizational and technical practices that are needed to handle research data and is typically framed through high-level principles and lifecycle models. The most influential are the FAIR Guiding Principles, which define that data should be *Findable, Accessible, Interoperable and Reusable* and emphasize machine-actionable metadata and persistent identifiers as requirements for reuse [2]. Building on these principles, different lifecycle models describe RDM as a sequence of phases, from planning and collecting to processing, analyzing, preserving and sharing data. The Digital Curation Centre (DCC) introduced a life cycle model, that distinguishes activities in “full life cycle actions” (e.g. representation, preservation planning, community watch), “sequential actions” (e.g. conceptualize, select, ingest) and “occasional actions” (e.g. dispose, migrate) emphasizes that curation is continuous rather than linear [4]. Other lifecycle frameworks, such as the Research data life cycle by the University College London define similar stages and activities but have a stronger university perspective [5]. Meanwhile there are a multitude of different models and frameworks. In a recently published review nearly 80 frameworks from different disciplines and from 20 different countries were analyzed, indicating that there is no need to develop yet another framework [6]. Instead, other institutions like the National Institute of Standards and Technology rather harmonize existing and established frameworks to work on a common ground instead of building niche solutions [7].

While these models give a structural overview of RDM, they remain largely conceptual. They describe existing phases and their activities, but they say little about how individual roles act and interact, nor how concrete decisions about data are made in practice across a project. A data management plan (DMP) is a core RDM instrument that systematically outlines the procedures for collecting, storing, and managing data. Still, writing a DMP is often created when project proposals are handed in at funding agencies, but are rarely used during the projects themselves [8]. When researching on how to write DMPs, even the well-renowned DCC does not provide easy-to-understand knowledge on their website but instead provides a lengthy text (14 DIN A4 pages) with 30 internal links to subpages and 10 references to be read. So, in summary there is no need for new frameworks, but better knowledge provision for researchers.

2.2 Policy and regulatory landscape in Germany

The conceptual frameworks are reinforced by a policy landscape at funder and institutional level. In Europe, Science Europe’s Practical Guide to the International Alignment of RDM provides core requirements for DMPs, criteria for selecting trustworthy repositories and a DMP evaluation rubric [2]. Many European funders and institutions have adopted these core requirements, which has led to a de facto standard for questions that should be addressed in a DMP (e.g. data types, documentation, storage, legal/ethical issues, long-term preservation).

In Germany, the DFG Code of Conduct “Guidelines for Safeguarding Good Research Practice” explicitly anchors research data as part of good scientific practice, requiring appropriate documentation, storage, and accessibility in line with legal and ethical constraints [9]. Similar national or institutional codes across Europe increasingly tie RDM explicitly to research integrity and to open science.

Additionally, data protection and privacy regulations such as the EU General Data Protection Regulation (GDPR) set strict conditions for processing and sharing personal and research data. In academic-industry collaborations, IP protection forms an additional governance layer beyond GDPR. Even when personal data is not involved, datasets can reveal product designs, process parameters, or competitive advantages. Practically, this requires early clarification of ownership and rights to use results and data, what can be published and under which license, and which de-risking strategies are acceptable (e.g., controlled access, synthetic data, or algorithmic transformations that preserve utility of data while at the same time reduce reverse-engineering risks).

2.3 Scientific tools and services for supporting RDM

There are different concepts that address how to support researchers during RDM. Much of it focuses on tools for planning and documenting RDM, particularly DMPs. These include templates and checklists (e.g. DCC DMP checklists) and machine-actionable DMPs (maDMPs) that can be interpreted and acted upon by information systems. At the same time RDM planning tools have been developed. For example the Data Stewardship Wizard (DSW) provides a knowledge-model-driven system where data stewards adapt questionnaires for specific domains and projects, generating DMPs [10]. Another example is the Research Data Management Organiser (RDMO), which enables institutions to model project-specific RDM requirements and generate structured plans and documentation [11]. All identified tools had in common that they focus on creating DMPs via templates or questionnaires and thereby primarily support documentation instead of providing day-to-day procedural guidance. In parallel, the German NFDI research groups (e.g., NFDI4Ing in engineering) provide complementary infrastructure services that operationalize FAIR practices in project work. The NFDI subproject RESODATE for example helps researchers find and reuse existing research artefacts such as software through semantic metadata [12]. However, no projects in the NFDI landscape could be found that focus on the procedural guidance on responsibilities and timing across roles.

2.4 Provision of RDM knowledge and training formats

Beyond tools, many universities librarians and data stewards have developed courses, workshops, and handbooks to support researchers in adopting practices [13]. Studies indicate, however, that researchers often experience RDM guidelines and policies as long, abstract and difficult to map to their day-to-day workflows. [14]. Training interventions can improve awareness and self-reported competence, but their effect is mostly short-term, as detailed procedural know-how is difficult to retain without continuous usage. [15].

Some universities also offer decision-support interfaces for RDM in the form of decision trees or visual checklists. In these tools, researchers answer a small set of yes/no questions and are then guided to more detailed information, services or templates. Such visual checklists help to break RDM down into a series of manageable tasks instead of presenting it as a single, large obligation [16].

More formal process-oriented approaches have also been proposed. In the humanities, BPMN-based models have been used to describe consultation and support workflows for RDM, including typical processes such as support with proposal preparation and data archiving, tailored to the needs of humanities projects rather than covering an entire cross-disciplinary lifecycle [17]. In addition, RDM maturity models provide generic process charts and classify activities into maturity levels to assess and

compare the state of RDM in organizations or projects. The specific needs of stakeholders are only weakly represented [18].

Overall, existing RDM resources provide strong conceptual and compliance-oriented guidance (lifecycle descriptions, policies, DMP templates), but they insufficiently operationalize this knowledge into role- and phase-specific actions that can be applied ad hoc in everyday academic–industry project work, including handovers, governance decisions, and the concrete artefacts to be produced.

3 Development Approach

Therefore, the research gap is the lack of a reusable, role-based and interactive representation of the RDM lifecycle for academic–industry collaborations that translates high-level principles and policies into concrete actions, decisions, and artefacts across roles.

To address this gap, a modeling workshop was conducted with eight researchers with experience in industrial IT as well as grand funded research projects that jointly reconstructed typical collaboration and data flows and co-created a role-based BPMN model. Based on these results an interactive prototype was developed. It presents the RDM lifecycle and roles and reveals short textual guidance and example resources when process elements are selected. Afterwards the prototype was evaluated in individual think-aloud walkthroughs with nine other researchers at different career stages and with different experience levels with industry and research projects, combined with a brief questionnaire on perceived usefulness. The qualitative feedback was analyzed and used to revise both the BPMN model and the interactive prototype content.

4 Developed Artefact

4.1 Structure of the role-based BPMN model

The artefact consists of a BPMN 2.0 process model that represents the RDM lifecycle in industry–university collaborations from a role-based perspective. BPMN was chosen because it is a widely adopted standard for process modelling with a reasonably intuitive graphical notation and formal underpinnings, and is frequently used as a communication medium between heterogeneous stakeholders [19]. Empirical studies have shown that visual process models can support comprehension and task performance compared to purely textual descriptions. Although the effect depends on model quality and user characteristics [20], it is plausible that researchers and industry stakeholders can navigate BPMN-style maps with limited onboarding, especially when supported by legends and drill-down explanations.

Horizontally, the model is structured along a generic RDM lifecycle that integrates common phases from established frameworks like planning, data collection, transfer (which is especially important for industry collaboration and often not found in frameworks), processing, sharing and retention. This makes the model compatible with typical lifecycle descriptions while emphasizing the temporal order of decisions and tasks. Horizontally the chart has different swim lanes, but unlike in typical BPMN models the swim lanes here do not reflect different departments and their responsibilities. Instead, the swim lanes show the data lifecycle phase, the tasks, artefacts and possible resources. Instead of showing responsibilities in the swim lanes the model uses color-coding of activities for key roles in industry–university projects, for example doctoral researcher, principal investigator, IT / infrastructure, and industry partner representatives. Tasks and gateways indicate which actor is responsible for which RDM-relevant actions and decisions at a given phase, including handovers and points where consultation with support staff or the use of additional documents is recommended. Data-related artefacts (e.g. raw data, processed data, deletion contracts, Non-Disclosure Agreements, DMPs, Audit Documents) are represented as BPMN data objects linked to tasks. This makes the connection between

actions and RDM artefacts explicit and helps to disentangle where specific documents or datasets are created, updated or archived. The model intentionally avoids discipline-specific terminology so that it can be reused across different industrial projects, while still allowing local tailoring at the level of examples and annotations. A screenshot from the phase “data storage” can be found in fig. 1.

In this view, the tasks in the data storage phase are presented as a horizontally arranged process, including checks for data integrity, sensitivity reassessment, storage system selection, encryption, catalogue updates and periodic audits. Underneath the process flow, the artefact row lists the typical documents created or updated at each step (e.g. RDM catalogue entry, deletion contract, updated DMP, audit document). A color legend at the bottom maps tasks to roles and indicates whether tasks are mandatory or optional. Presenting RDM in such phase-level maps helps to decompose RDM into small, completable sub-tasks rather than a single, overwhelming obligation. The model is designed to be “high-level but concrete”: it abstracts from project-specific details yet is precise about who does what, when, and with which artefacts. This aligns with findings from process modelling research that emphasize the need for models that balance expressiveness with cognitive simplicity to remain understandable for non-expert users [20].

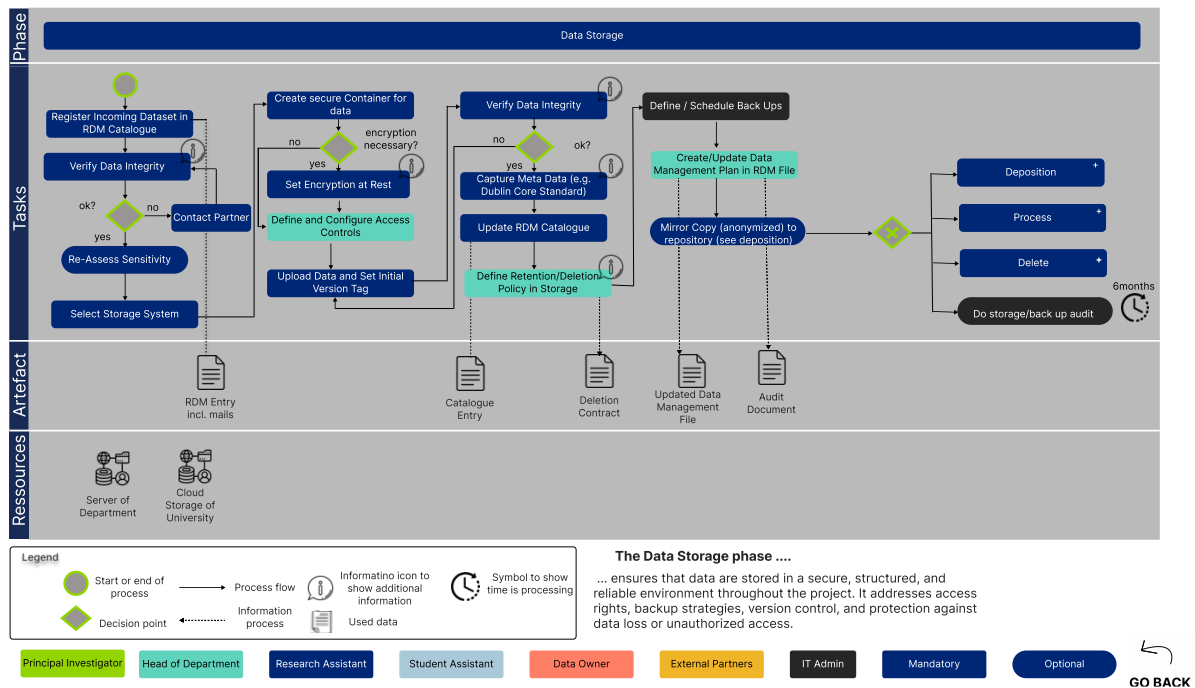


Figure 1: Overview of the role-based BPMN process map for the phase “data storage”

4.2 Interaction concept and demonstrator

Based on these BPMN diagrams, an interactive prototype in the form of a clickable demonstrator was developed that turns the process map into a navigable guideline for everyday project work. The primary intended users are early-career researchers who execute many RDM tasks but lack procedural knowledge, and industry contacts who require transparency about data handling. The demonstrator is not meant to automate RDM. Instead it supports coordination and decision-making by linking each task to role-specific actions, artefacts, and used methods. The full version can be found under [21]. Instead of working in a modelling tool, researchers navigate RDM via screens that show the phase-level process maps and allow them to click on individual activities or decision points. The intention is that users consult the prototype ad hoc at the point of need. For example, when they are about to store a new dataset and are unsure which steps are required and how these steps are fulfilled.

In such a scenario, a research assistant opens the Data storage view (Fig. 2) and follows the tasks from left to right until reaching “Set encryption at rest”. Here, “encryption at rest” means encrypting stored data on storage systems (including backups) so that unauthorized access to stored data does not reveal the content. In case the assistant does not know how to set the encryption at rest, they can click on the task. Clicking this task opens a drill-down screen (Fig. 3). The drill-down provides more detailed, structured guidance. A task panel lists concrete actions (e.g. determining the scope of encryption, configuring backups, establishing key management), a “Quality gates / Goals” panel summarizes what must be true for the step to be considered complete, and additional panels list helpful tools and relevant regulatory aspects (such as security-by-design requirements or data processing agreements for cloud storage). In this way, the prototype keeps the process context visible while offering focused, step-by-step instructions only when needed.

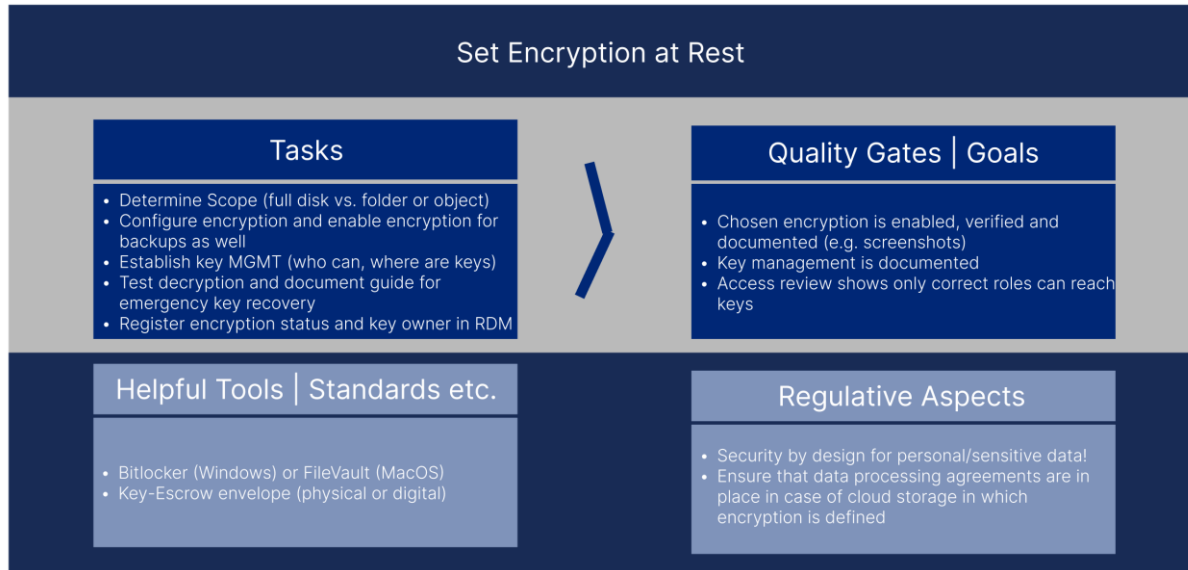


Figure 2: Detail page about the task “Set encryption at rest”

Taken together, the BPMN model and demonstrator form a didactic artefact that combines an overview of the RDM lifecycle with role-specific, on-demand explanations. In contrast to static diagrams or purely textual guidelines, the artefact is intended to be used repeatedly during a project as a navigable “map” of RDM activities.

5 Evaluation

Nine researchers participated in the evaluation (7 doctoral researchers, 1 postdoc, 1 professor; 6 male, 3 female). Their research experience ranged from 0.5 to 22 years (mean 6.5, median 5), and they had completed 1 to 50 industry-collaboration projects (mean 11, median 5). Self-rated RDM competence (1–7 scale) ranged from 2 to 6 (mean 3.6, median 3). Overall, the sample combines early-career researchers with low RDM confidence and highly experienced researchers, enabling assessment across different seniority levels.

5.1 Perceived usefulness

Fig. 4 presents the perceived usefulness. The items were measured on 7-point Likert scales, following the Technology Acceptance Model [22], where perceived usefulness is a core determinant of technology acceptance. Across items, medians cluster between *rather agree* (5) and *agree* (6) for most positive statements. Participants tended to agree that the guideline helps them clarify roles and responsibilities, derive concrete actions, and that overall, the guideline is useful for their work (medians around 5, with

upper quartiles at 6–7). Ratings for time-related benefits (e.g. creating an RDM plan or preparing audits more quickly) are more mixed, with central tendencies around the neutral to “rather agree” range, indicating that participants primarily see the artefact as a conceptual and coordinative aid rather than as a strong time-saving tool. The negatively framed item (“the guide has no added value for me”) received low ratings with a median clearly below neutral, confirming that participants generally perceived the process map as beneficial. The raw data can be found under [23].

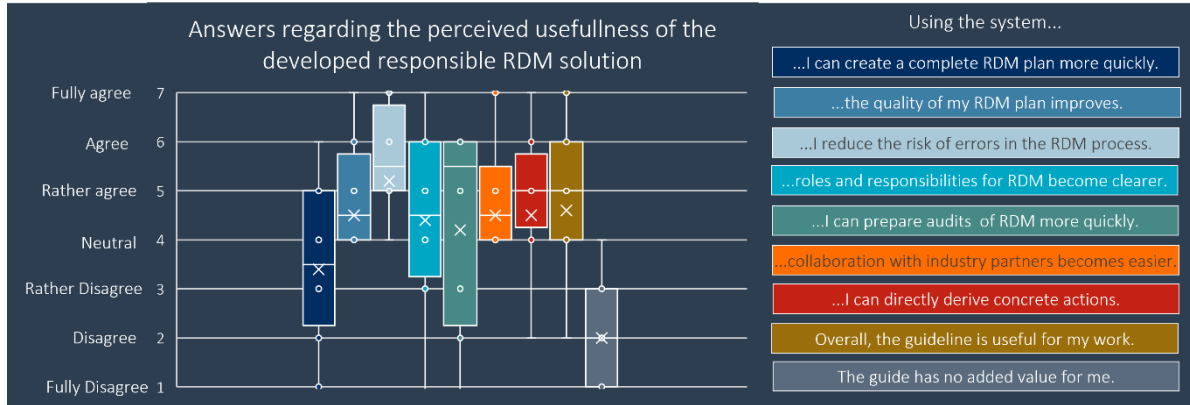


Figure 3: Boxplots of ratings for perceived usefulness and related statements

5.2 Qualitative feedback from think-aloud walkthroughs

During the think-aloud walkthroughs, participants applied the prototype to given industrial collaboration project scenarios in groups of two researchers while verbalizing their thoughts. The recordings were transcribed, anonymized and analyzed. The transcribed recordings can be found under [24].

5.2.1 Positive perceptions

Across groups, the BPMN guideline was described as a systematic way of approaching RDM. One group summarized “I think this is a systematic system, a systematic way of proceeding ...” and emphasized its value for communication with partners: “I find it useful that we can demonstrate how we deal with the data to our industry partners.” The walkthroughs made the complexity of RDM visible in a way participants considered helpful: “You notice how complex the process actually is. You could see that while going through it.” Another participant concluded: “Overall this process is very good so that you don’t forget anything when you carry out research data management.” Another group confirmed this impression and highlighted that the governance and task descriptions make aspects like metadata and anonymization more explicit than their usual practice: “It looks like what we have discussed but more precise and from the side of data ... we did not discuss metadata and anonymization in so much detail. With such a task description it is more visible what steps are behind our steps.” They also noted that the governance view essentially covers the whole project duration: “This is exactly a guideline what we can do from the beginning of a project till three months afterwards.” For projects where partners explicitly value structured RDM, the guideline was described as “very valuable”.

5.2.2 Critical feedback

At the same time, participants pointed to several weaknesses. First, the purpose and scope were not fully clear: “It was not 100 percent clear what the goal was, whether the aim was the complete execution of RDM ... or whether you are only creating the RDM file, and RDM plan.” Second, several terms and decision points were criticized as unclear. Examples include “use case” (“the term ... is always rather irritating for me”), “data deposition” (“Data Retention is something different. We both apparently have not understood ‘Data Deposition’ yet.”), and labels such as “check requirements of using technological solutions for needed data”, which one participant said they did not understand. Gateways in particular

were perceived as confusing: “No matter what I do, I always end up at ‘Derive needed data’. Then this is not really a gateway. That is confusing.” “In general, the gateways are described wrongly.” Third, some participants felt the process was too detailed and prescriptive: “I think you are making it too complicated ... It should not be too complicated, not too many possibilities that could be, but not everything prescriptive.”. They warned that “too much determinism” can create mistrust and make people feel like “slaves of the process”. Fourth, several usability issues of the prototype were mentioned. These include font size (“The font size is too small.”), unclear starting point and missing back navigation (“I would now think that you have to click somewhere on the left first, but that doesn’t work.”; “There is no back button somehow.”), and confusing color coding for roles (“For me this is all the same color ... as someone with red–green weakness.”). The nature of the “RDM file” artefact was also unclear: “What is RDM file? Is it a text file, an XML or JSON or what?”. Another group additionally noted confusion about how mandatory vs. optional tasks are represented: “Because it is color coded, but here it is not color coded anymore ... ‘mandatory, optional’ – and here the research assistant seems to be mandatory and optional at the same time.”. Together, these statements show that while the overall idea is appreciated, the semantics of elements and the current interaction design are not self-explanatory.

5.2.3 Immediate design changes

Based on the feedback, a set of three low-effort improvements were implemented, which are already reflected in the version presented in this paper. First, a short introductory panel was added that explicitly states the goal of the tool (provision of knowledge about specific tasks of RDM plan during all phases of project lifecycle, not automating all RDM tasks), to address the confusion about scope. Secondly, unclear labels (including “use case”, “check requirements of using tech solutions...”) were rephrased and several gateways whose paths previously all led to the same activity were simplified, to avoid pseudo-decisions. Thirdly, a concise legend was added explaining the meaning of colors, shapes and the distinction between mandatory and optional tasks, and reworked the color scheme to be more easily distinguishable, also for users with red–green weakness. Additionally, explanation boxes for a better understanding for the meaning of each phase were added.

6 Discussion and Resulting Improvements

The evaluation indicates that the role-based BPMN demonstrator is generally perceived as useful across a heterogeneous group of researchers with varying experience in RDM and industry collaboration. Most participants reported that the prototype helps them gain an overview of the RDM life cycle, clarify responsibilities, and derive concrete actions. Time savings and direct support for writing funder-specific DMPs were rated slightly lower, which is consistent with our design focus on understanding procedural knowledge rather than automating documentation. These findings fit well with evidence that visual process models can effectively convey procedural knowledge when they are well-designed. A systematic review on process model comprehension emphasizes that graphical process models support users in organizing and integrating procedural information, provided that cognitive load is controlled through appropriate notation and layout [25]. Eye-tracking results show that readers of BPMN models use stable visual strategies and that well-structured models support efficient navigation, especially for non-experts [26]. The participants’ comments that the demonstrator “gives a good overview” and makes “roles and responsibilities clearer” are in line with these observations. In other words, the prototype appears to exploit the strengths of BPMN as a medium for externalizing and sharing process knowledge rather than merely prescribing compliance steps.

At the same time, the results highlight a tension that is also well known in the process-modeling literature: some participants asked for more detail and additional gateways or decision points, whereas others perceived parts of the model as too detailed or complex [25]. The partly contradictory feedback

therefore seems less like a flaw of the prototype and more like evidence that different user groups need different levels of abstraction. This supports the design decision to use drill-down views and suggests that further personalization by role, experience level, or project type would be beneficial.

The broader literature on DMPs also helps to contextualize the findings. Smale et al. argue that while DMPs are widely mandated, evidence for their effectiveness in changing data practices is limited, and researchers often experience them as an administrative burden [27]. A recent survey similarly reports inconsistent integration of DMPs into research workflows and calls for additional mechanisms other than through DMPs to teach good data management practices [28]. Against this backdrop, it is notable that the participants valued the demonstrator not primarily as a way to produce DMP text, but as a means to understand what should happen in each phase and who is responsible. This aligns with the idea that effective RDM support must go beyond plan documents and instead embed RDM into day-to-day project activities.

At the same time, the study is more exploratory than many of the controlled evaluations in the RDM training literature. The perceived usefulness was evaluated and a qualitative feedback was collected, but it was not tested whether the demonstrator leads to measurable improvements in knowledge, skills, or behavior, as done in studies by Xu et al. on online RDM instruction [15]. The small sample ($n = 9$) limits the generalizability of the findings. Future work should therefore include controlled comparisons with existing RDM support formats (e.g., PDF guidelines, DMP tools, or standard workshops), larger and more diverse participant groups, and behavioral outcome measures such as the quality of actual RDM plans or adherence to lifecycle steps during a project.

7 Conclusion and Future work

This paper introduced a role-based BPMN process map and an interactive demonstrator for research data management in industry–university collaborations. It addressed the two research questions:

RQ1: We developed a role-based BPMN process map and an interactive demonstrator for research data management in industry-university collaborations, focusing on procedural understanding (who does what, when, and with which artefacts) rather than on DMP text generation.

RQ2: In an evaluation with nine researchers, the artefact was perceived as useful mainly for clarifying roles and responsibilities, deriving concrete actions, and communicating the RDM “big picture”. It was less associated with time savings for writing DMPs, consistent with its purpose as a complement to existing planning and automation tools.

Still, the current model and demonstrator should be seen as a starting point for a trans-university shared knowledge provision tool. They are meant to be adapted and extended by other institutions, especially those with limited resources to develop their own RDM process models.

References

- [1] K. Briney, H. Coates, and A. Goben, “Foundational Practices of Research Data Management,” *Res. Ideas Outcomes*, vol. 6, p. e56508, Jul. 2020, doi: 10.3897/rio.6.e56508
- [2] Science Europe, “Practical Guide to the International Alignment of Research Data Management - Extended Edition,” Jan. 2021, doi: 10.5281/ZENODO.4915861.
- [3] G. Birkbeck, T. Nagle, and D. Sammon, “Challenges in research data management practices: a literature analysis,” *J. Decis. Syst.*, vol. 31, no. sup1, pp. 153–167, Dec. 2022, doi: 10.1080/12460125.2022.2074653.
- [4] S. Higgins, “The DCC Curation Lifecycle Model,” *Int. J. Digit. Curation*, vol. 3, no. 1, pp. 134–140, Dec. 2008, doi: 10.2218/ijdc.v3i1.48.
- [5] J. A. J. Wilson, *UCL Research Data Lifecycle 2020/2023*. University College London, 2024, p. 661695 Bytes. doi: 10.5522/04/25471591.

- [6] J. Jiang, D. Maurici-Pollock, and R. Tang, "Research Data Lifecycle (RDLC): An Investigation into the Disciplinary Focus, Use Cases, Creator Backgrounds, Stages and Shapes of RDLC Models," *Int. J. Digit. Curation*, vol. 19, no. 1, p. 30, Feb. 2025, doi: 10.2218/ijdc.v19i1.860.
- [7] R. Hanisch, D. Kaiser, A. Yuan, A. Medina-Smith, B. Carroll, and E. Campo, "NIST Research Data Framework (RDaF): Version 2.0," National Institute of Standards and Technology, Gaithersburg, MD, NIST SP 1500-18r2, Feb. 2024. doi: 10.6028/NIST.SP.1500-18r2.
- [8] T. Miksa, J. Cardoso, and J. Borbinha, "Framing the scope of the common data model for machine-actionable Data Management Plans," in *2018 IEEE International Conference on Big Data (Big Data)*, Seattle, WA, USA: IEEE, Dec. 2018, pp. 2733–2742. doi: 10.1109/BigData.2018.8622618.
- [9] Deutsche Forschungsgemeinschaft, "Guidelines for Safeguarding Good Research Practice. Code of Conduct," Apr. 2022, doi: 10.5281/ZENODO.6472827.
- [10] R. Pergl, R. Hooft, M. Suchánek, V. Knaisl, and J. Slifka, "'Data Stewardship Wizard': A Tool Bringing Together Researchers, Data Stewards, and Data Experts around Data Management Planning," *Data Sci. J.*, vol. 18, p. 59, Dec. 2019, doi: 10.5334/dsj-2019-059.
- [11] I. Anders et al., "The Research Data Management Organiser (RDMO) – a Strong Community Behind an Established Software for DMPs and Much More," *Data Sci. J.*, vol. 23, p. 28, May 2024, doi: 10.5334/dsj-2024-028.
- [12] R. Pueblas, P. Kuckertz, J. M. Weinand, L. Kotzur, and D. Stolten, "ETHOS.PASSION: An open-source workflow for rooftop photovoltaic potential assessments from satellite imagery," *Sol. Energy*, vol. 265, p. 112094, Nov. 2023, doi: 10.1016/j.solener.2023.112094.
- [13] Z. Xu, "Research Data Management Training in Academic Libraries: A Scoping Review," *J. Librariansh. Sch. Commun.*, vol. 10, no. 1, Jul. 2022, doi: 10.31274/jlsc.13700.
- [14] T. Austin, K. Bei, T. Efthymiadis, and E. Koumoulos, "Lessons Learnt from Engineering Science Projects Participating in the Horizon 2020 Open Research Data Pilot," *Data*, vol. 6, no. 9, p. 96, Sep. 2021, doi: 10.3390/data6090096.
- [15] Z. Xu, X. Zhou, A. Kogut, and M. Clough, "Effect of online research data management instruction on social science graduate students' RDM skills," *Libr. Inf. Sci. Res.*, vol. 44, no. 4, p. 101190, Oct. 2022, doi: 10.1016/j.lisr.2022.101190.
- [16] University of Edinburgh Library, "Research Data Service. (o.J.). Quick Guide 10 – Research Data Management Flowchart [PDF]. University of Edinburgh. Available at: <https://library.ed.ac.uk/research-support/research-data-service/guidance> (accessed 1 December 2025)."
- [17] P. Helling, J. Blumtritt, and B. Mathiak, "Der Beratungsworkflow des Data Center for the Humanities (DCH) an der Universität zu Köln," *O-Bib Offene Bibl. Herausgeber VDB*, 2018, doi: 10.5282/O-BIB/2018H4S248-261.
- [18] M. L. Wawer et al., "A maturity based data management integration in engineering research projects," *Proc. Des. Soc.*, vol. 5, pp. 169–178, Aug. 2025, doi: 10.1017/pds.2025.10031.
- [19] H. Ritchi, M. Jans, J. Mendling, and H. A. Reijers, "The Influence of Business Process Representation on Performance of Different Task Types," *J. Inf. Syst.*, vol. 34, no. 1, pp. 167–194, Mar. 2020, doi: 10.2308/isys-52385.
- [20] S. A. Kassim et al., "Benefits and limitations of business process model notation in modelling patient healthcare trajectory: a scoping review protocol," *BMJ Open*, vol. 12, no. 5, p. e060357, May 2022, doi: 10.1136/bmjopen-2021-060357.
- [21] G. Kremer, "Practical Guide: Responsible Research Data Management (RDM) - BPMN-based top-level process and phase protocols." Technische Universität Berlin, Jan. 22, 2026. doi: 10.14279/DEPOSITONCE-25164.
- [22] F. D. Davis, "Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology," *MIS Q.*, vol. 13, no. 3, pp. 319–340, 1989, doi: 10.2307/249008.
- [23] G. Kremer, "Perceived usefulness ratings (7-point Likert) for an interactive BPMN-based Research Data Management guideline," Jan. 21, 2026, *Technische Universität Berlin*. doi: 10.14279/DEPOSITONCE-25162.
- [24] G. Kremer, "Think-aloud walkthrough transcripts: interactive BPMN-based Research Data Management guideline (academic-industry collaboration)," Jan. 21, 2026, *Technische Universität Berlin*. doi: 10.14279/DEPOSITONCE-25161.
- [25] K. Figl, "Comprehension of Procedural Visual Business Process Models: A Literature Review," *Bus. Inf. Syst. Eng.*, vol. 59, no. 1, pp. 41–67, Feb. 2017, doi: 10.1007/s12599-016-0460-2.
- [26] M. Tallon et al., "Comprehension of business process models: Insight into cognitive strategies via eye tracking," *Expert Syst. Appl.*, vol. 136, pp. 145–158, Dec. 2019, doi: 10.1016/j.eswa.2019.06.032.
- [27] N. A. Smale, K. Unsworth, G. Denyer, E. Magatova, and D. Barr, "A Review of the History, Advocacy and Efficacy of Data Management Plans," *Int. J. Digit. Curation*, vol. 15, no. 1, p. 30, 2020, doi: 10.2218/ijdc.v15i1.525.
- [28] M. Langseth, E. Sellers, G. Donovan, and A. Liford, "Assessing the value and usage of data management planning and data management plans within the U.S. Geological Survey: U.S. Geological Survey Open-File Report 2023–1069, 44 p., <https://doi.org/10.3133/ofr20231069>," Open-File Report, 2023.

Designing and Validating Affordant Interfaces for Hybrid Intelligence Systems

Maximilian Kretzschmar^{1*}, Bernhard Saske¹, Jan Mehlstäubl², Kristin Paetzold-Byhain¹

¹Dresden University of Technology, Chair of Virtual Product Development, 01069 Dresden

²MAN Truck & Bus SE, AI Center of Enablement, 1050-053 Lisbon

*Corresponding author. E-Mail address: maximilian.kretzschmar@tu-dresden.de; Tel.:0351/46335360

Abstract: The integration of Large Language Models (LLMs) into engineering workflows is transforming automation from solely deterministic execution into hybrid intelligence, where humans and AI agents jointly reason and act. Compound LLM systems expand automation possibilities, however there is also a risk of obscuring their decision-making processes, challenging engineers' ability to validate results. Transparency therefore becomes a prerequisite for trust and accountability in designing such systems for engineering tasks. We explore how transparency can be both embedded in and empirically tested in such AI based expert systems. Drawing on Leonardi's affordance theory as well as the triangle of design responsibility, an industrial prototype of a compound LLM engineering system is developed and verified. The prototype integrates reasoning and tool agents with transparency mechanisms, enabling engineers to observe actions, data sources, and decision pathways. Based on empirical insights and combined with a literature analysis, a concept for validation and transparency evaluation is derived to assess whether hybrid intelligence systems are not only able to deliver value by solving required tasks but also remain interpretable and accountable. Building upon established verification and validation strategies the concept provides a foundation for the evaluation of transparency as a distinct quality dimension in AI-driven engineering systems.

Keywords:

Large Language Models, Multi-Agent Systems, Agentic AI, Human-AI Interaction

1 Introduction

Automation in engineering design aims to support engineers by embedding information and knowledge into reusable systems and tools [1]. As such systems evolve and more complex problems are tackled, a recurring trade-off emerges: deterministic, rule-based automation offers repeatability and control but becomes increasingly difficult to extend as design variability grows. In contrast, more adaptive or knowledge-rich systems can accommodate a wider range of situations but require substantial expert input and development effort. This practical trade-off between reliability and flexibility has been repeatedly observed in design automation research [2,3].

The introduction of Large Language Models (LLMs) has changed this dynamic. They can interpret natural language to reason across domains, coordinate other software tools and be designed to act with varying degrees of autonomy within Multi-Agent Systems (MAS), enabling hybrid configurations that combine deterministic computation with probabilistic reasoning. Their integration into engineering design introduces what Pradas Gómez et al. [4] describe as a shift in design responsibility – from a dual relationship between human and deterministic tools to a triadic configuration involving the designer, traditional tools, and LLMs. This “*triangle of design responsibility*” marks a transition toward hybrid intelligence systems where human and technological agencies continuously share control and decision-making.

As automation systems ascend this triangle towards systems that plan and act semi-autonomously the challenge is no longer limited to developing systems that solve engineering tasks efficiently, but also to ensuring that their reasoning and outputs remain transparent and accountable. Consequently, the interface layer between humans and such AI Systems becomes central to the human-AI relationship. From an affordance perspective [5], the interface not only transmits information but shapes how human and AI agencies interact. Figure 1 illustrates how human agency is expressed through system users, while technological agency is divided between the mediating interface layer where affordances are perceived and enacted as well as the infrastructural compound LLM system which provides the pre-structured space for action and determines which components exist, their interactions with one another and in turn defines which tasks are solvable. In this work we adopt the term compound LLM system [6–8] to denote an architecture in which LLMs are embedded within and orchestrate additional subsystems (e.g. retrieval and data access, tools via APIs, external ML models, multi-agent architectures).

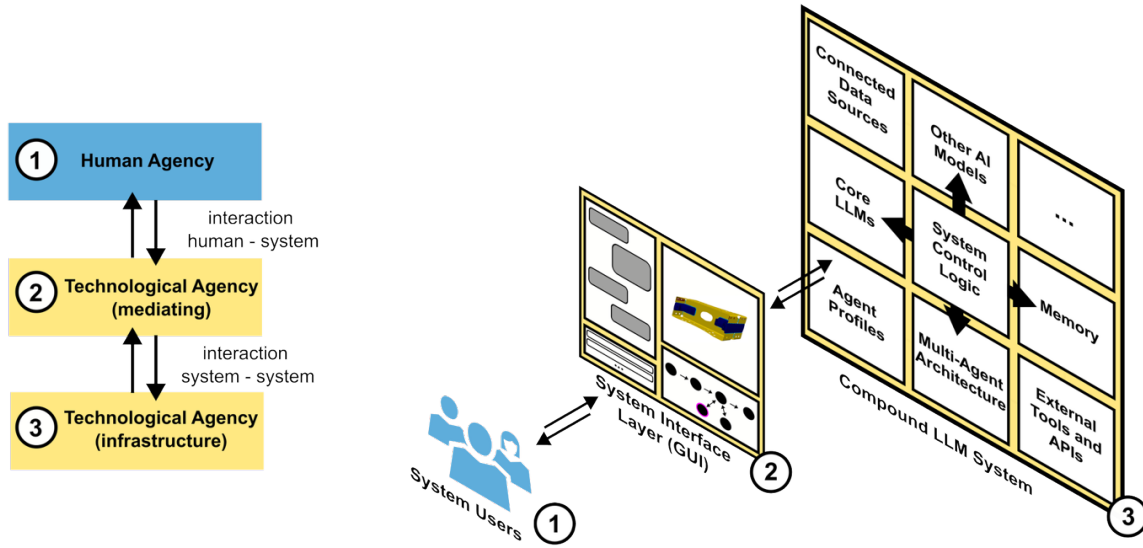


Figure 1: Human and technological agency across system users, system interface and compound LLM system layers

Interaction occurs at multiple levels: humans and the interface (frontend), where goals are translated into system-readable actions as well as between the interface and the compound LLM system (backend), where actions are operationalized through computational exchanges within the system infrastructure. The LLM’s generative and orchestrative capabilities are leveraged to make use of other components, yielding a cohesive multi-part system that can plan, call tools, interpret results and iterate toward goals in ways standalone LLMs cannot.

The combination of deterministic and non-deterministic methods in a system creates new operational challenges. Unlike conventional automation, where logic chains are explicit and traceable, LLM-based reasoning is stochastic and context-dependent. Users must therefore rely on interface-level transparency mechanisms to understand how conclusions are reached and which data, tool calls, or prompts contributed to them. Without such visibility, validation of results, essential in regulated engineering domains, becomes increasingly difficult. Motivated by this fact we pursue two complementary research aims, embodied by the following two research questions:

RQ1: How can compound LLM-based automation systems be designed to support complex engineering workflows while maintaining transparency of their decision-making processes to users?

RQ2: How can hybrid intelligence systems be validated and systematically tested for transparency once developed, and what criteria determine whether such systems meet transparency and efficiency requirements?

2 Method

We adopt a two-stage research design aligned with the above research questions connecting a designed artefact in a real-world context to a knowledge base of existing theories and models to develop an evaluation concept. The first stage (RQ1) covered in section 3 focuses on the development of a compound LLM system with an industrial case study serving as an example, while the second stage (RQ2) covered in section 4 builds upon this and carries out a theoretical analysis based on established literature to determine necessary evaluation dimensions. Together, these stages address both the practical and assessment-oriented dimension of transparency in hybrid intelligence systems. These findings are used in section 5 to conceptualize an evaluation approach for validating usefulness and assessing transparency in such systems.

2.1 Method for RQ1 – Compound LLM System Development

Prototype development was conducted in collaboration with an industrial partner whereby the case company specializes in the design of complex products and employs extensive finite element analysis workflows for structural and dynamic evaluation of automotive components. These processes involve partly repetitive post-processing tasks as well as tasks which require engineering judgment and result interpretation. The compound LLM system is based on a multi-agent architecture in which individual agents combine deterministic computation (through pre-programmed simulation tools) and probabilistic reasoning (through the LLMs themselves) to partially automate workflows with a human-in-the-loop. System development followed an iterative action research cycle of design, deployment, observation, and reflection [9]. Each iteration was carried out jointly by researchers and simulation engineers, ensuring the prototype evolved under real operational conditions. After gathering requirements from stakeholders and translating these into design decisions to develop the prototype, observations from live usage were complemented by semi-structured interviews and questionnaires to identify points where the users found the system’s reasoning unclear or difficult to follow. Subsequently existing features were tweaked, removed or new features added.

2.2 Method for RQ2 – Evaluation Dimensions of Compound LLM Systems

The development of the validation concept is based on a literature review focusing on determinants for validating expert systems. In addition, metrics were sought that describe efficiency both for method use in general and for the use of expert systems in particular. Since the compound LLM system presented here integrates non-deterministic elements into an expert-system-like architecture, a further literature search was conducted to identify determinants by which transparency of AI methods in development contexts can be described and measured.

3 Compound LLM System Development – Industry Case Study

3.1 Motivation and Context

The compound LLM automation system targets the post-processing stage of simulation-driven product development in which engineers analyse large sets of result artefacts such as Ansys .rst and .cdb files to extract, compare, and communicate insights including stress distributions, eigenfrequencies, material mappings and displacements (Figure 2a). These tasks, typically performed through a mix of scripts and manual GUI operations in specialized engineering software, are repetitive, time-consuming, and prone to inconsistency across projects. The developed compound LLM system interface acts as a layer between the engineer on one side and simulation data, material databases, other AI models and computational tools on the other (Figure 2b). It enables engineers to interact with simulation results via natural language queries and integrated validated Ansys functions and thereby provides personalized analytical assistance with higher automation as well as knowledge management benefits.

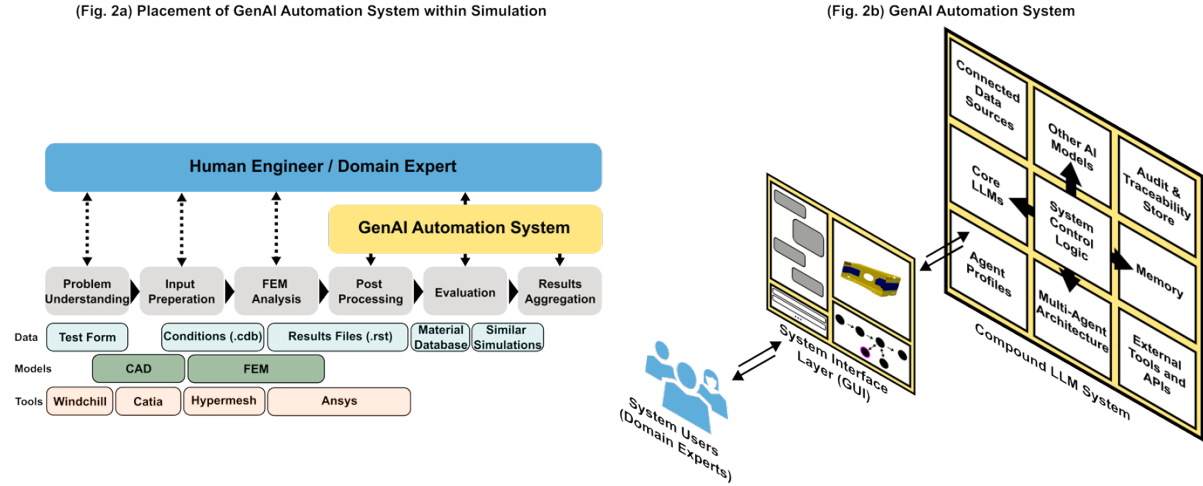


Figure 2: Placement of compound LLM system within the engineering simulation process

3.2 System Architecture

The implemented system follows a multi-agent architecture with three strategic reasoning agents and several specialized tool agents. We use a mixed methods approach combining both deterministic and non-deterministic elements depending on the type of task. A *Planner Agent* interprets user queries, decomposes them into subtasks, and generates an execution plan. An *Information Agent* validates files, identifies relevant components, and ensures consistency between user terms and database identifiers. A *Manager Agent* coordinates workflow execution, error handling, and user communication. Specialized *Tool Agents* perform domain-specific analyses such as Stress, Modal, Material, and Mesh. Agents are orchestrated using LangGraph [10], enabling graph-based workflow coordination. Claude Sonnet 4 [11] is used as the foundation model across the board via the AWS Bedrock API while custom tools interacting with simulation files are implemented in python and utilize Ansys PyDPF [12]. The system interface layer (frontend) is built as a react application hosted in a custom streamlit component.

3.3 Workflow and Interaction

In a typical session, a user issues a natural-language query such as “Compare stress distribution between chassis part X in Simulation Y and Z and determine any critical load cases.” The *Planner Agent* decomposes this request into subtasks (e.g. file and part validation, load case mapping, stress extraction). The *Information Agent* confirms file integrity and component mappings. The *Manager Agent* orchestrates the execution, invoking relevant tool agents sequentially. Specialized domain agents perform their tasks through validated DPF functions. Results can be integrated into composite reports and visualizations, accompanied by traceability metadata.

3.4 Transparency-Oriented System Design

Transparency is achieved by linking outputs to prompts, data sources, and model versions for traceability. The system combines LLM-based reasoning for planning and explanation, deterministic tool execution for computations, as well as other AI models such as custom fastText embedding models [13] for resolving domain-specific terminology and component identifiers. The core challenge lies in coupling these heterogeneous reasoning and computation layers into a coherent architecture that automates tasks while ensuring that each decision, data source, and analytical pathway remains transparent. The interface shown in Figure 3 presents the conceptual intermediary layer between engineer and system, offering an interactive overview of actions, intermediate results, and representations of decision pathways.

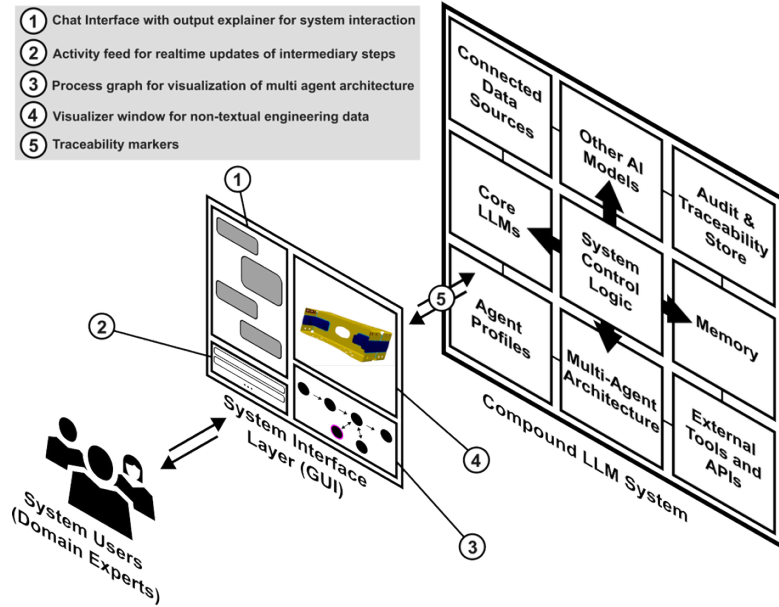


Figure 3: Conceptual interface layer for the proposed compound LLM system

Resulting from iterative development combined with user feedback, the following five categories of interface mechanisms are embedded to support transparency and numbered accordingly in Figure 4:

- **(1) Chat with output explainer:** main chat interface for the user to pose queries (1a) whereby each system output is accompanied by an expandable section (1b) showing tool calls with input parameters as well as any errors that arose for that specific system output.
- **(2) Activity feed:** chronological log of system actions and intermediate outputs, displaying validation steps such as file checks, load-case mappings, or material identification.
- **(3) Process graph:** real-time visual representation of agent coordination, illustrating dependencies and the flow of information among LLM components.
- **(4) Visualizer Window:** interactive representation of non-textual engineering data specific to this use case such as stress and material distributions or modal animations.
- **(5) Traceability markers:** hashes linking metadata annotations and results to their prompts, data sources, model versions, and timestamps, enabling reproducibility and auditability (not presented to users).

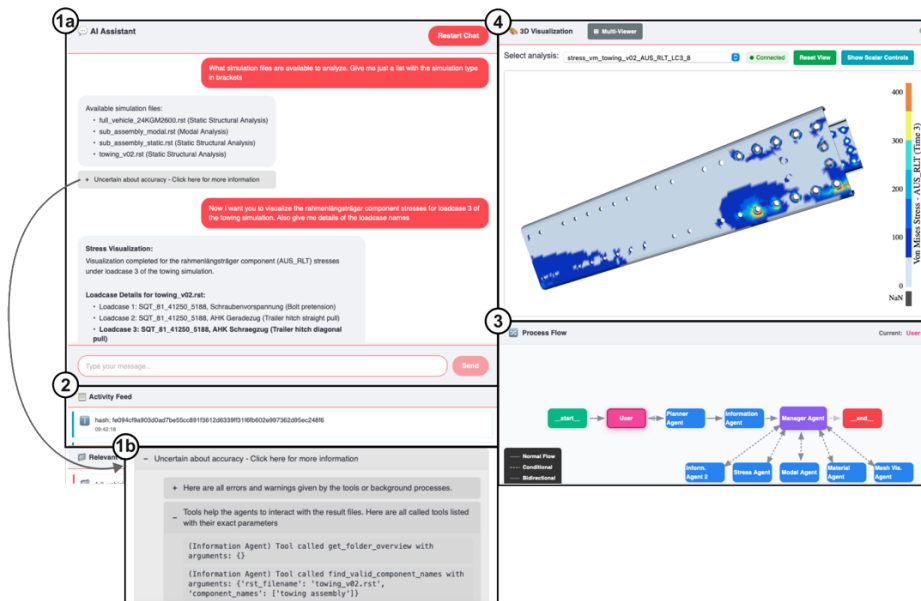


Figure 4: Example of prototype interface layer including annotation of embedded transparency mechanisms 1-4

4 Evaluation Dimensions of Compound LLM Systems

The compound LLM system described above can be interpreted as a method that supports designers in specific activities, rather than solely as a software artefact. Following Kroll and Weisbrod [14], such methods should be evaluated with respect to their applicability, their ability to convey understanding and support process compliance, and their effectiveness for the targeted development task. The specific criteria depend on the task to be supported and on the intended purpose of the method. Further literature also highlights several factors that can negatively affect such a method's performance or weaken its potential. These include incomplete information and inconsistencies between method elements [15].

To be useful, such methods need to satisfy four basic properties [16]: they must be defined (logically complete and coherently described, with a clear scope of application), predictable (internally understandable and behaviourally foreseeable), usable (accessible, easy to apply, and credible), and desirable (producing results that are appropriate and valuable for the task at hand).

Traditional expert systems exemplify such methods in the context of complex problem solving that requires substantial domain expertise. Knowledge is encoded via rules and facts from which conclusions are drawn [17]. However cognitive work that was previously reserved for human experts is increasingly taken over by information systems [18] which leads directly to the automation dilemma discussed in the introduction. This makes clear that validation of expert systems must go beyond classical validation. A holistic perspective on verification and validation (V&V) is required.

With the introduction of AI methods, and particularly LLMs into expert systems, an additional dimension emerges. Alongside deterministic mechanisms, non-deterministic, probabilistic methods are now used for data analysis, planning, and task completion. Automated conclusions are based on statistical relationships and are therefore associated with uncertainty. This can lead to misinterpretation or misplaced trust if users are not aware of the limits of the models or the quality of underlying data. Consequently, V&V of AI based expert systems must also incorporate criteria for transparency.

From an affordance perspective, the GUI is the layer where technological agency becomes perceptible and actionable for users; it is here that system outputs are exposed in a form that engineers can interact with. To evaluate and contextualize results, users must be able to build an appropriate mental model of how the system approaches a task [18], and that mental model is shaped by what the interface makes visible. At the same time, the interface due to being a tool for the human-in-the-loop to interact with the compound LLM system is also part of a broader socio-technical system [19].

Design decisions about what to surface, at what level of detail, and with which interaction patterns all influence the workload, required expertise, and the balance between usability and flexibility of the system. Highly automated interfaces that hide too much may encourage over-reliance and deskilling, while excessively detailed ones can overwhelm users and erode efficiency [2]. These trade-offs are directly linked to transparency: they determine whether engineers can meaningfully oversee and challenge AI decisions. Consequently, our evaluation concept treats transparency as something that is realized through the interface but evaluated regarding its effects on both human understanding and overall system performance.

4.1 Verification, Validation and Transparency

Synthesising the findings from literature as well as practical experience from the compound LLM case study, three related but distinct evaluation dimensions which build upon one another in multiple levels can be distinguished:

1. **Verification** (Does the system do things correctly?) – Focuses on technical correctness, robustness, and internal consistency across all deterministic and probabilistic components.
2. **Validation** (Does the system do the correct things?) – Focuses on the benefits for engineers and fit to the intended task, including efficiency gains and quality of decision support.
3. **Transparency** – (Is it understandable how and why the system does things?) – Focuses on interpretability, explainability, and traceability of the system’s reasoning and actions from the user’s perspective.

These dimensions form a hierarchical foundation for evaluating compound LLM systems: verification is the base layer, validation builds on verified functionality to assess usefulness, and transparency sits on top as a property that determines whether engineers can appropriately assess the system’s behaviour. In compound LLM systems, each of these dimensions must be addressed separately for deterministic and non-deterministic methods.

The starting point for evaluation is **verification** of the compound LLM system whereby the focus lies on the technical system. In a first step, it must be ensured that the compound LLM system produces correct results. Classical software tests can be used for deterministic elements. In order to identify and assess uncertainties or fluctuations in the results caused by probabilistic elements, an additional, more detailed analysis is required to understand how the non-deterministic components affect the overall result. This includes examining reproducibility of results using for example human or LLM-as-a-judge techniques using a sufficiently large, unbiased, diverse test dataset covering anticipated tasks. Once the system is deployed monitoring through constant evals is advisable. Furthermore, the general capability of the chosen or trained AI model for its tasks needs to be determined for which processes such as DASC [20] or numerous domain specific existing benchmarks [21–23] can be used as guidance if available.

Within the scope of **validation**, the focus extends beyond the technical system to include the users of the compound LLM system. The socio-technical system becomes the object of investigation. The primary concern is the usefulness. Validation examines whether the system’s results address the engineers’ queries and whether they provide added value for decision-making. For example: is the functionality of the deterministic tools sufficiently useful for the tasks? And on the non-deterministic side: are the tools used correctly and do the non-deterministic elements provide additional value?

To ensure acceptance and valid decisions when using the compound LLM system, it is necessary that **transparency** is evaluated to make sure that users can reconstruct the process by which results are generated. In deterministic expert systems, acceptance and trust in the result are often based on existing knowledge and experience of how similar tasks are usually solved. This knowledge is encoded as rules in the expert system and at the same time forms the basis for the mental models engineers build up [18]. It can be assumed that a similar mechanism also apply when non-deterministic elements are used within a compound LLM system. Regardless of this, users must be given opportunities to confirm and further develop their mental models. The GUI of the compound LLM system is considered an essential element for this. To assess transparency, special attention must therefore be paid to the GUI and to whether the information that users expect is provided in a way that helps them complete and adjust their mental model.

The individual aspects of verification, validation, and transparency, as described above, are shown as a model in Figure 5. Crucial for productive deployment of a compound LLM system in an organisation is that it offers efficiency advantages compared to conventional approaches. Evaluation is therefore always linked to an efficiency analysis, which can be operationalised at the proposed levels and translated into concrete metrics. The right side of Figure 5 provides an overview of how efficiency is interpreted in this context and which measures can be derived from it.

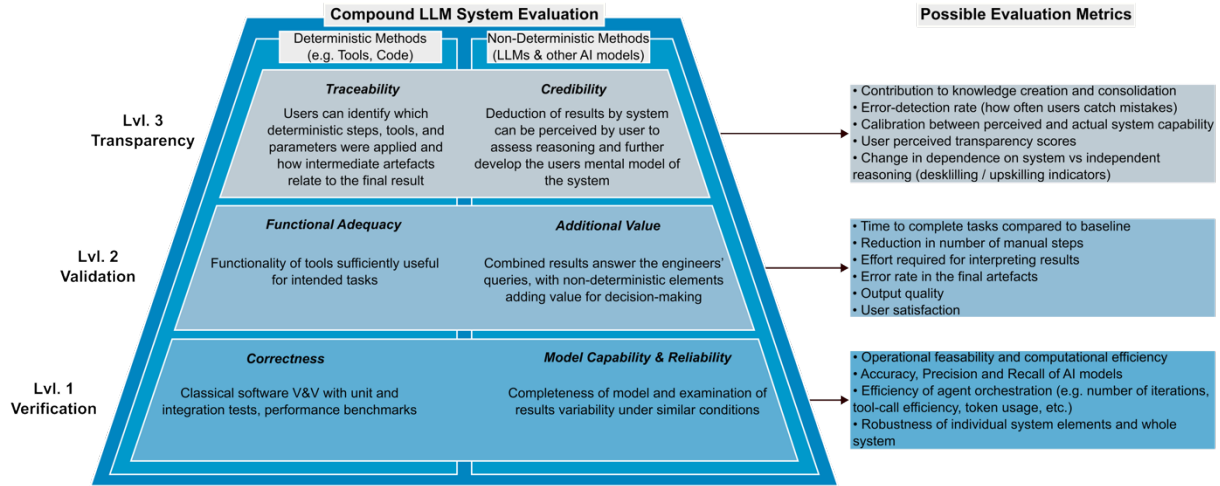


Figure 5: Verification, validation and transparency pyramid for compound LLM systems

5 Validation & Transparency Evaluation Concept for Compound LLM System

Building on the criteria and aspects that must be considered in the evaluation of validation and transparency levels as well as a framework investigating AI and human interaction [24] a concept is derived that also captures different facets of efficiency. A key premise for the following considerations is that verification (e.g. checks for correctness, completeness, and reproducibility of the technical system) has already been carried out. The evaluation concept presented here therefore focuses on the socio-technical system consisting of the users and the compound LLM system, with a particular emphasis on the transparency dimension.

The evaluation is designed to be primarily qualitative. This is partly because the available sample size is relatively small, with a maximum of around ten engineers, and partly because individual perceptions are likely to lead to considerable variability in how response options are interpreted, making a purely quantitative analysis less meaningful. Although a sample size of ten participants is still on the low side for a qualitative study, it is deemed appropriate here because a comparatively homogeneous target group is being studied within a clearly defined context.

At the core of the evaluation concept is an experiment in which participants are given a typical development task to solve. The sample is split: one third of the participants solve the task using the conventionally used simulation workbench, providing baseline values for comparison with the compound LLM system. The remaining participants are given access to the compound LLM system for task completion. Beforehand, they receive a brief training session on the system's functionality and use while the design of the training aims to minimize the possibilities of priming.

During task execution, participants are observed by the experimenter. In addition, they are encouraged to verbalize their thoughts following the think-aloud method. The task execution and think-aloud protocols are recorded for later anonymized analysis. The purpose of this approach is to capture and understand how participants use the tools, with particular attention to how they engage with the non-deterministic elements, how they handle the implications of these elements for judging solutions, and whether and how they recognize uncertainties. From this, it should be possible to infer the extent to which the information provided via the interface influences efficiency in task completion and supports or hinders the formation of accurate mental models. This analysis also allows estimation of how much effort and time are required for information retrieval and task completion in each condition.

To realize an efficiency assessment at the transparency level, it is necessary to evaluate how the information provided by the compound LLM system is processed by the engineers and how it

contributes to building experience and competence. This is done using questionnaires that participants complete after the experiment. A pre-experiment questionnaire collects demographic data (experience with the task, tenure in the company, age, gender, etc.) and captures expected difficulties and challenges related to task completion, both in terms of content and organization. A five-point Likert scale is used for the formulated questions.

The concluding questionnaire is intended to provide insights into knowledge building and perceived transparency. Since this is closely tied to individual and subjective factors, it is difficult to measure directly. It must therefore be captured indirectly by correlating responses with observed patterns of information use and processing. Accordingly, the questionnaire includes items that address the way information is used and interpreted, and how uncertainties or ambiguities are perceived and handled. Participants are also asked to rate how well the tool supported them in different aspects of the task and how the effort for information retrieval compares to their previous approach. Global questions on the perceived transparency of the compound LLM system complete the questionnaire.

Taken together, this mixed qualitative design combining think-aloud protocols, observational data, and targeted questionnaires aims to capture the multidimensional construct of transparency and its relationship to efficiency in the use of the compound LLM system.

6 Conclusion & Outlook

The increasing integration of automation into engineering workflows has historically focused on efficiency, reducing manual effort through deterministic algorithms and rule-based configurations. While this ensures reproducibility, it also constrains adaptability and contextual awareness: deterministic systems operate within predefined boundaries and scale poorly to more ambiguous multi-step processes. In hybrid intelligence settings, the interface between the users and the system therefore no longer just relays commands but must act as an affordant mechanism that enables the visualisation, constraining, and communication of system reasoning. Its design determines whether users can trace AI generated outputs, interpret decision pathways, and retain meaningful authority over system actions, thereby making transparency-oriented interaction design central to trust and accountability in human-AI collaboration.

This paper showed how transparency can be designed and evaluated in compound LLM systems that blend deterministic tools with probabilistic reasoning. Addressing RQ1, an industrial case study of a multi-agent LLM assistant for simulation post-processing demonstrated how an affordant interface can surface planning steps, tool calls, data sources, and intermediate results. The transparency mechanisms operationalise Leonardi's affordance perspective and the triangle of design responsibility by making the system's technological agency inspectable and governable for engineers. Addressing RQ2, we derived three interconnected evaluation dimensions: verification, validation, and transparency, for both deterministic and non-deterministic components, and proposed a transparency evaluation concept combining experimental comparison, think-aloud protocols, observation, and targeted questionnaires.

Although developed for a single engineering context and not yet tested at scale, the work offers an initial, structured way to treat transparency as a distinct quality dimension rather than a by-product of accuracy and usability. As compound LLM systems spread across domains increasingly enabled by tool integration protocols like the Model Context Protocol [25], a mixed methods character of automation will become standard, and scalable systematic transparency evaluation will be essential. Future work will extend this perspective into a development framework spanning technical foundations, interface design, and organisational embedding supporting practitioners in implementing effective hybrid intelligence systems.

References

- [1] Cederfeldt M, Elgh F. Design Automation in SMEs—Current State, Potential, Need and Requirements. DS 35: Proceedings ICED 05, Melbourne: 2005.
- [2] Verhagen WJC, Bermell-Garcia P, van Dijk REC, et al. A critical review of Knowledge-Based Engineering: An identification of research challenges. *Advanced Engineering Informatics* 2012;26:5–15. <https://doi.org/10.1016/j.aei.2011.06.004>.
- [3] Vidner O, Wehlin C, Wiberg A. Design Automation Systems for the Product Development Process: Reflections from Five Industrial Case Studies. *Proc Des Soc* 2022;2:2533–42. <https://doi.org/10.1017/pds.2022.256>.
- [4] Pradas Gómez A, Kretschmar M, Paetzold-Byhain K, et al. A team of three: the role of generative AI in the development of design automation systems for complex products. *Proc Des Soc* 2025;5:309–18. <https://doi.org/10.1017/pds.2025.10045>.
- [5] Leonardi PM. When Flexible Routines Meet Flexible Technologies: Affordance, Constraint, and the Imbrication of Human and Material Agencies. *MIS Quarterly* 2011;35:147–67. <https://doi.org/10.2307/23043493>.
- [6] Chen J, Ye J, Wang G. From Standalone LLMs to Integrated Intelligence: A Survey of Compound AI Systems. *ArXiv* 2025. <https://doi.org/10.48550/arXiv.2506.04565>.
- [7] Chen L, Davis J, Hanin B, et al. Are more LM calls all you need? towards the scaling properties of compound AI systems. *Proc 38th NeurIPS* 2024:45767–90.
- [8] IBM, Stryker C. What are compound AI Systems? <https://www.ibm.com/think/topics/compound-ai-systems> 2024. <https://www.ibm.com/think/topics/compound-ai-systems> (accessed September 3, 2025).
- [9] Hevner AR, March ST, Park J, et al. Design Science in Information Systems Research. *MIS Quarterly* 2004;28:75–105.
- [10] langchain-ai. LangGraph GitHub. <https://github.com/langchain-ai/langgraph> (accessed August 29, 2025).
- [11] Anthropic. Claude Sonnet 4 2025. <https://www.anthropic.com/news/claude-4> (accessed January 7, 2026).
- [12] Ansys Inc. Ansys Data Processing Framework (DPF) for developers 2025. <https://developer.ansys.com/docs/dpf> (accessed August 28, 2025).
- [13] Bojanowski P, Grave E, Joulin A, et al. Enriching Word Vectors with Subword Information. *ArXiv* 2017. <https://doi.org/10.48550/arXiv.1607.04606>.
- [14] Kroll E, Weisbrod G. Testing and evaluating the applicability and effectiveness of the new idea-configuration-evaluation (ICE) method of conceptual design. *Res Eng Des* 2020;31:103–22. <https://doi.org/10.1007/s00163-019-00324-6>.
- [15] Cash P, Daalhuizen J, Hekkert P. Evaluating the efficacy and effectiveness of design methods: A systematic review and assessment framework. *Des Stud* 2023;88:101204. <https://doi.org/10.1016/j.destud.2023.101204>.
- [16] Daalhuizen J, Cash P. Method content theory: Towards a new understanding of methods in design. *Des Stud* 2021;75:101018. <https://doi.org/10.1016/j.destud.2021.101018>.
- [17] Feigenbaum EA. Expert systems in the 1980s. State of the Art Report on Machine Intelligence Maidenhead: Pergamon-Infotech, 23 1981.
- [18] Hacker W. Menschengerechtes Arbeiten in der digitalisierten Welt. vdf Hochschulverlag an der ETH Zürich; 2018. <https://doi.org/10.3218/3938-2>.
- [19] Gericke K, Eckert C, Stacey M. Elements of a design method – a basis for describing and evaluating design methods. *Design Science* 2022;8:e29. <https://doi.org/10.1017/dsj.2022.23>.
- [20] Schulz M, Neuhaus U, Kaufmann J, et al. DASC-PM v1.1: A Process Model for Data Science Projects 2022. <https://doi.org/10.25673/91094>.
- [21] SWE-bench Team. SWE-Bench Leaderboard 2025. <https://www.swebench.com> (accessed August 28, 2025).
- [22] Wang A, Pruksachatkun Y, Nangia N, et al. SuperGLUE: A Stickier Benchmark for General-Purpose Language Understanding Systems. 2019. <https://doi.org/10.48550/arXiv.1905.00537>.
- [23] Everingham M, Van Gool L, Williams CKI, et al. The pascal visual object classes (VOC) challenge. *Int J Comput Vis* 2010;88:303–38. <https://doi.org/10.1007/s11263-009-0275-4>.
- [24] Hillebrand L, Raisch S, Schad J. Managing with Artificial Intelligence: An Integrative Framework. *Academy of Management Annals* 2025;19:343–75. <https://doi.org/10.5465/annals.2022.0072>.
- [25] Anthropic. Introducing the Model Context Protocol 2024. <https://www.anthropic.com/news/model-context-protocol> (accessed January 7, 2026).

A concept to simulate stakeholders in product development with personas and generative AI

Stefan Wellsandt^{1*}, Karl Hribernik¹, Jan Naumann², Klaus-Dieter Thoben¹

¹ BIBA - Bremer Institut für Produktion und Logistik GmbH at the University of Bremen, Bremen, Germany

² Institut Technik und Bildung (ITB), University of Bremen, Bremen, Germany

*Corresponding author. E-Mail address: wel@biba.uni-bremen.de

Abstract: The article proposes simulating stakeholder workshops in product development using persona-based generative-AI agents. It argues that real workshops are hard to coordinate and that early-stage concept creation could benefit from a scalable, repeatable alternative. The approach instantiates multiple persona-driven agents representing distinct stakeholder roles (e.g., user, designer, safety engineer, and cost analyst) within an agent-based simulation framework (TinyTroupe). Agents are guided through a structured workshop workflow: setup, document review, idea generation, detailed recommendations, trade-off reasoning, and prioritization. During the process, agents can ask targeted questions to a human facilitator, and the answers can be stored in a shared knowledge base to improve consistency across rounds. A demonstration on an educational SCARA robot kit shows how to produce ranked improvement recommendations and synthesize different stakeholder perspectives into a consolidated outcome. The paper generalizes the workflow into a concept model and identifies key research gaps: generating empirically grounded personas, measuring how persona traits influence outcomes, maintaining persona consistency over long dialogues, validating outputs against human baselines, and scaling to richer product representations and tool access (e.g., structured product models and engineering artifacts). It also notes limited generalizability because the evaluation covers only one scenario and lacks systematic quality metrics.

Keywords:

Generative AI, Personas, Stakeholder Simulation, Product Development, Multi-Agent Systems

1 Introduction

Dialogue among stakeholders of complex products is essential for anticipating, reviewing, and resolving problems. Involving stakeholders from the product lifecycle during product development activities is costly and time-consuming. The involved persons need to invest their work time (employees) or personal time (users) and might need to travel to a physical location to work on a specific task, for instance, during a workshop or test session. A second issue is that persons, especially those with specific expertise or rare characteristics, e.g., senior experts or members of a specific user group, may not be available for an activity, which may cause delays or reduce the quality of the activity's results.

One approach to address these issues is to simulate stakeholders [1,2]. Product development already applies simulation techniques such as roleplaying with personas representing typical individuals within a target group. If humans adopt a role like this, the quality of the results might improve [3,4], but someone must still invest their time at a cost.

Recent research indicates that generative AI can create text to describe stakeholder personas and human behavior [5,6]. Also, there are libraries for simulating interactions between generative AI personas (e.g., TinyTroupe, Concordia, and AgentSociety), but their applicability in product development remains

unclear. If computer-based stakeholder simulation in product development is feasible, affordable, and on par regarding results quality, it may have a substantial positive impact on costs and time, due to its scalability.

The **goal** of this article is to present a concept for simulating stakeholders in a product development environment with multiple generative AI personas. Our guiding research questions are as follows. Is the simulation of a realistic product development activity technically feasible? What are the key characteristics of related simulation experiments? Which open issues require deeper investigations?

Section 2 outlines the research problem, and Section 3 introduces the chosen methods and approaches. The results in section 4 cover the technical setup and present important findings from the experiments, which are generalized in a concept illustration. Section 5 discusses key aspects with respect to this article's goals and open issues. This article concludes by acknowledging the limitations and providing an outlook.

2 Research Problem

Computer-based simulation of stakeholders in product development is not yet common, largely due to the assumed effort required to train domain-specific probabilistic and rule-based AI models. Crowder et al. [1] built a complex agent-based simulation for an engineering team that can exchange abstract information to perform tasks and sub-tasks. Liu et al. [2] present a simulation for user engagement in new product development with a high level of abstraction. Neither simulation covers how simulated stakeholders actually contribute to an activity in terms of content.

Using pre-trained (generative) AI – such as ChatGPT, Claude, or DeepSeek – to simulate the concrete contributions of a stakeholder can help address part of the problem outlined above. These software systems can generate convincing content from user prompts. Prompts may contain task descriptions, product outlines, persona profiles, and other useful information to consider in responses. A key challenge for using pre-trained models effectively for this purpose and domain is that many activities involve multiple stakeholders concurrently. Example activities are brainstorming, design review, and root-cause analysis. This observation means that a simulation concept must also handle the interactions among simulated stakeholders. There are simulation libraries that claim to be suitable; for instance, Microsoft's TinyTroupe project indicates it can simulate focus groups to assess product designs [7]. The academic literature offers very limited insight into the applicability of computer-based multi-stakeholder simulation in product development.

3 Methods and Approaches

This research is grounded in multi-agent frameworks supporting generative AI, such as TinyTroupe, which provide basic functions for managing persona-driven agents, message exchange via conversations, and data exchange with one or more language models. In this context, agents are software with limited autonomy that can execute agent tools to interact with other software. These tools are software functions with semantic descriptions that may, for instance, interact with a virtual product model and return information for subsequent reasoning activities. We use a scenario to narrow our research scope on a specific development activity, work environment, and product.

3.1 Scenario

Activity. We chose a product improvement task rather than a new product development task to reduce the simulation's complexity. Improvement requires substantially less information and time; such tasks are also significantly more common in practice.

Environment. The starting point for our simulation is a typical situation encountered in product improvement. A group of experts needs to improve an existing product to better address user needs. The group involves users directly, as suggested by user-driven design practices. The interaction format is a workshop because it encourages the exchange and consolidation of understanding, opinions, needs, and constraints. Such workshops are acknowledged in product development and allow participants to interact, but typically do not enforce interactions.

The workshop has sequential phases, each with one specific objective to fulfill:

1. Intro: Share setup and scenario with participating agents
2. Document review: Agents review task-related documentation individually
3. Initial thoughts: Agents share initial improvement ideas
4. Specific recommendations: Agents identify detailed improvements
5. Cost-benefit analysis: Agents analyze cost, time, and manufacturability
6. Prioritization: Agents rank each improvement

These phases ensure the environment is rich and challenging, as they allow participants to prepare, gradually detail their findings, quantify them, and mutually agree on a final result.

Product. The example product is a custom-made small SCARA robot arm built from 3D-printed plastic parts and (standard) components, such as screws, nuts, bearings, tools, and electric drives. We used this robot in educational settings where students must assemble the robot parts with support from a voice-enabled, intelligent digital assistant. A 7-page manual is available that contains assembly instructions and didactic/explanatory information on the purpose and use of tools and components. Figure 1 illustrates the assembly with components and sub-assemblies.

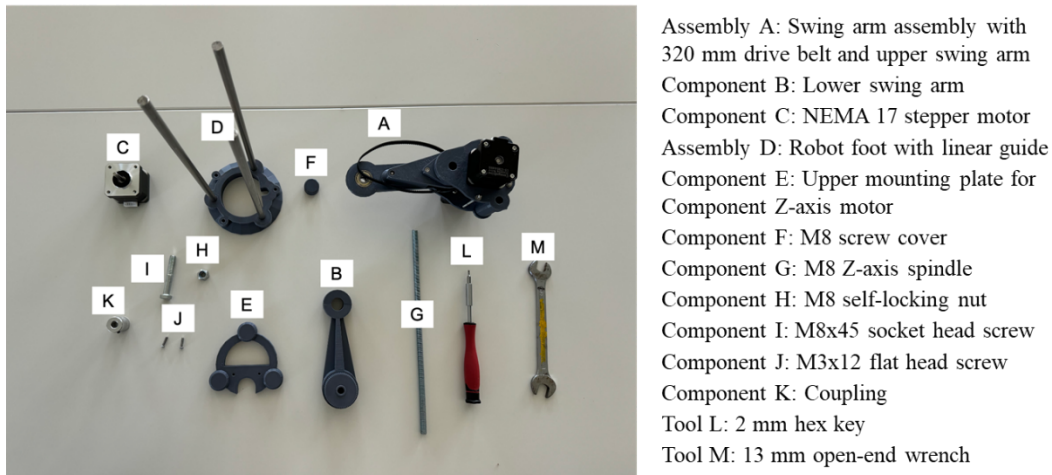


Figure 1: Educational table-top robot assembly used as a scenario for the experiments

3.2 Simulation

The simulation approach relies on using an existing agent-based simulation library that supports generative AI models. This library must provide basic facilities for persona management, agent creation from personas, and communication with the AI model via a data exchange interface. Besides, it must provide memory management across repeated interactions (but not experiment runs). The simulation environment models the workshop above with multiple participants and specific roles. Each phase has

a fixed number of simulation rounds, each of which provides an opportunity for agent reasoning and interaction with other agents or a human. The agents decide to interact based on their reasoning.

In the following sections, we use “GenAI system” to refer to a system that generates text responses from text input. It is not only a large language model (LLM) in the narrow sense, but a complex system of the LLM and undisclosed pre- and post-processing steps in the case of commercial providers like OpenAI.

4 Results

The concept’s basis is a series of iterative experiments with Microsoft’s experimental TinyTroupe simulation library for Python [8]. It leverages large language models (LLMs) and orchestrates agents with configurable traits. Agents can interact in environments called “TinyWorlds”, engage in conversations, receive broadcasts, and perform actions. The library enables product brainstorming, user testing simulations, market research, and the generation of training data. Figure 2 illustrates the technical experiment setup.

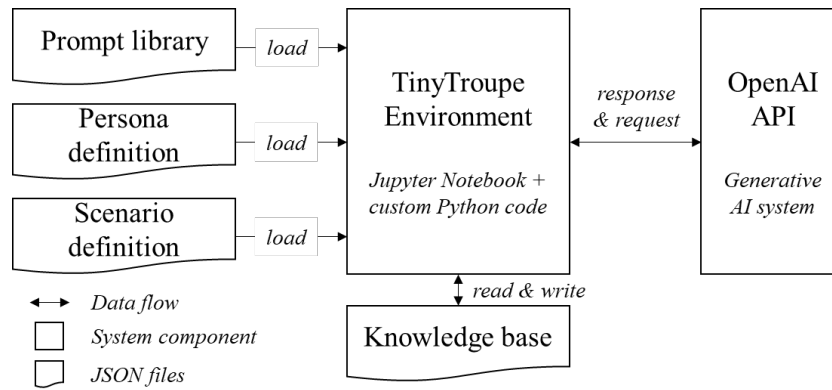


Figure 2: Technical setup for experimenting with personas and generative AI agents

We implemented two custom human-in-the-loop processes. First, the user receives a prompt to provide meaningful free-text input after each phase. This ensures a general intervention option for intermediate results. Second, if an agent’s response includes a question for the facilitator (simulation user), the user can provide an answer or skip it. Answered questions are broadcast and stored in a knowledge base that is available across experiments. A custom token tracker helps understand simulation costs per experiment and phase. We generated personas iteratively with the pretrained GenAI system Claude Opus 4.5 to avoid the substantial effort of hand-crafted, data-grounded persona design. The latter may yield more accurate personas, but since our goal was to create a sound concept first, we opted for AI-supported generation. The prompting was iterative, i.e., we refined the prompts based on intermediate persona results. Our main goal for this activity was to design persona-generation prompts that result in a coherent set of stakeholders. We assessed the coherence based on experience.

Personas contain a broad range of attributes, including demographics, occupation, education, goals, personality traits, preferences, beliefs, and communication style. We experimented with a team of up to seven agents selected from the persona pool in Table 1. The background summary outlines each persona, but was not used in the experiments (instead, we used concrete attributes). We set the limit to seven agents to balance persona variety and expected token costs. Identifying the technical limit was not a goal

Table 1: Persona pool

No.	Role	Age	Gender	Background Summary
1	Vocational Training Student - Mechatronics	19	Male	First-year vocational training student (Auszubildender) with basic knowledge of hand tools and simple electronics. This is his first exposure to robotics, and he’s excited but nervous about making mistakes.

2	Engineering Student - Robotics	23	Female	Fourth-year robotics engineering student with hands-on experience, including building several robot projects and a robot arm for a competition. Completed an internship at an industrial automation company.
3	Professor of Engineering Education	48	Male	Professor with 20 years of experience teaching robotics and automation. Designs curricula for vocational training programs and published research on hands-on learning effectiveness.*
4	Educational Kit Designer	36	Female	Designer with a Master's in Industrial Design, specializing in design for education. Designs educational robotics kits used by students across Europe, focusing on making technology accessible through thoughtful design.
5	Safety and Quality Engineer	45	Male	Certified safety engineer with 20 years evaluating products for compliance with EN 71, machinery directives, and educational laboratory standards. Dedicated to preventing harm through proper design and risk assessment.*
6	Curriculum Developer - STEM Education	38	Female	Curriculum developer specializing in robotics and automation training. Created hands-on learning modules used in vocational schools and universities across Germany.
7	Procurement and Cost Analyst	41	Male	Procurement and cost analyst for educational equipment. Optimizes budgets for STEM education programs across multiple institutions, focusing on maximizing educational value per euro spent.
* Note the unrealistic attributes for age (48, 45) with the "20 years of experience" occupation description. This is not a typo, but an actual result of the persona generation process we did not realize during the experiment phase.				

We ran all experiments using OpenAI's gpt-4.1-mini model to limit token costs and response generation time while maintaining reasonable response quality (we did not systematically investigate response quality as outlined in Section 5). Team compositions are Minimal (#1, #3), Student Focus (#1, #2, #3), Design Review (#3, #4, #5), Curriculum Alignment (#2, #3, #6), Budget Optimization (#3, #4, #7), and Full (#1 to #7). We did not empty the knowledge base between experiments to test knowledge accumulation. Token limits for input (e.g., conversation history and instructions) and output (response length) remained the same. Each run concludes with a ranked list of recommendations, each with a textual description of the suggested improvement and a statement and explanation of the expected impacts on time, cost, and feasibility. Table 2 summarizes the experiment runs.

Table 2: Experiment run examples

No.	Team	Rounds	Recommendations	Token Count	Costs USD	Duration
1	Minimal	11	8	796.826	0,35	00:10:02
2	Student Focus	16	6	907.913	0,39	00:13:03
3	Design Review	15	6	977.613	0,42	00:11:07
4	Design Review	22	7	949.899	0,41	00:08:27
5	Curriculum Alignment	16	4	988.671	0,43	00:10:12
6	Budget Optimization	18	2	920.025	0,40	00:08:53
7	Budget Optimization	11	9	923.078	0,40	00:09:40
8	Full	17	19	1.023.150	0,44	00:10:35
9	Full	22	9	1.126.857	0,49	00:11:56
10	Student Focus	11	3	1.677.528	0,73	00:20:02

Figure 3 presents contributions from the Document Review phase to outline the specificity and length of the agents' contributions. This example includes THINK and TALK messages (there are several others in TinyTroupe). The former is an agent's response to another agent's input, and the latter is the message meant for another agent. Agent contributions aim to meet the workshop instructions, for instance, focusing on three specific components or processes from the assembly manual.

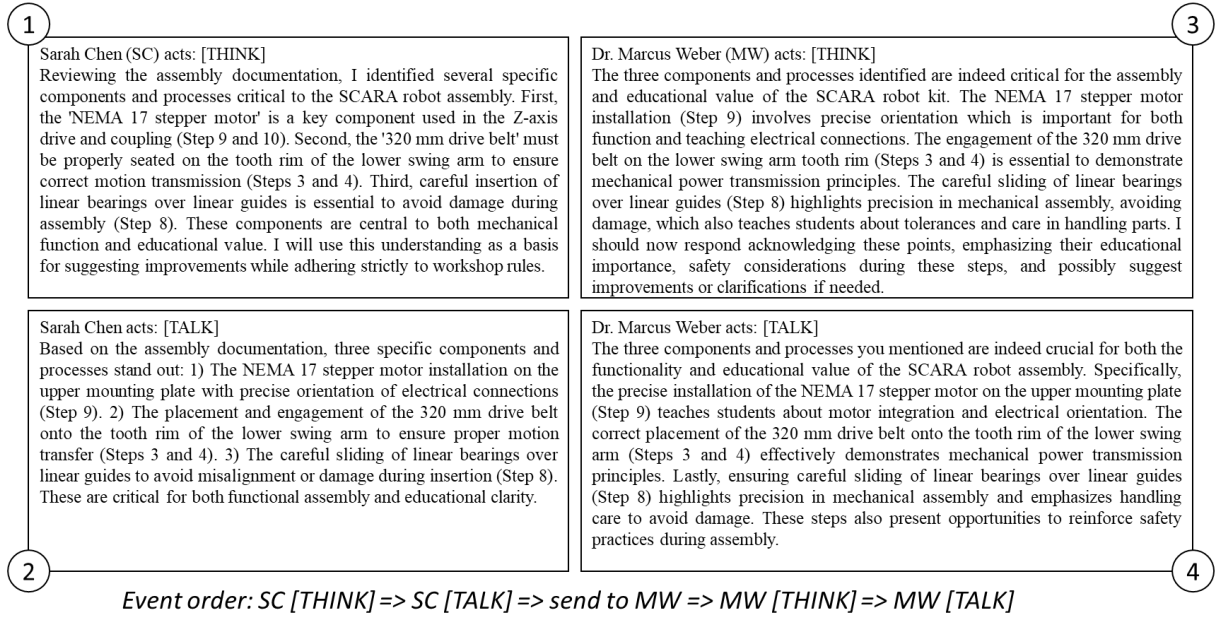


Figure 3: Example contributions between Designer [Sarah Chen] and Educational Expert [Marcus Weber]

Figure 4 illustrates key parts of the simulation flow for experiment run #1 with summarized example contributions. The horizontal arrows indicate that agent contributions are passed from agent to agent, and subsequent responses build on a previous one. Vertical ones represent sequential phase execution.

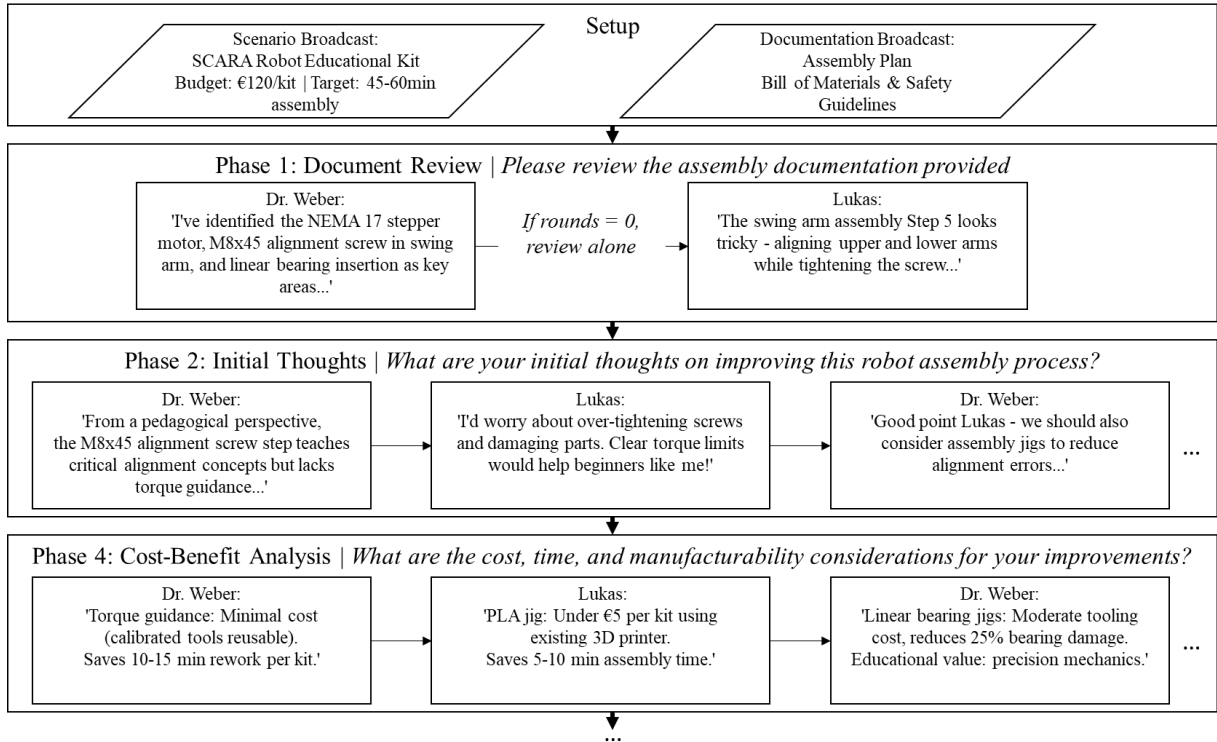


Figure 4: Conversation flow and summarized example content from an experiment with the Minimal team

Since the illustration above is specific to our setup, Figure 5 aims to generalize all experiment findings into a concept. Each phase relates to one objective that the agents aim to fulfil with conversations extending over one or more rounds. The last phase integrates previous results to generate a single final

result, e.g., a ranked list of recommendations. A human facilitator may answer questions upon an agent's request or provide input at the end of each phase.

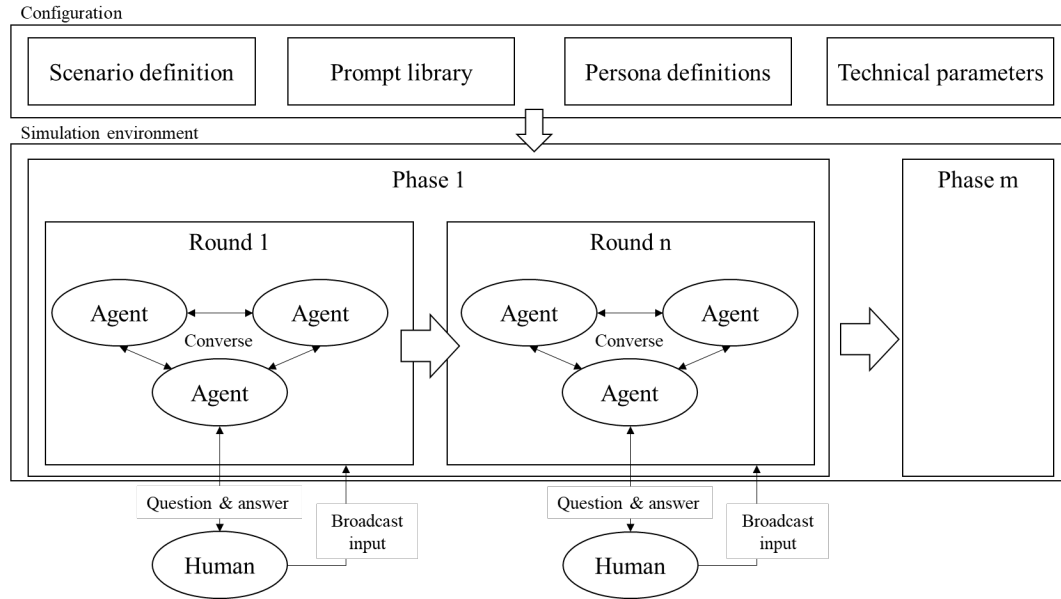


Figure 5: Concept for simulating stakeholders with persona-based agents and generative AI

5 Discussion

This section outlines discussion topics that emerged from the experiments. These topics cover technical aspects, design decisions for personas and scenarios, result quality, and scalability.

5.1 Technical setup

TinyTroupe is a good starting point for feasibility tests and pre-studies for the concept creation. However, the library hasn't received updates from Microsoft for several months, while other agent-based simulations are actively being developed. Switching to another simulation library might provide more functions and control over experiments. A second aspect of the setup is that it lacks a dedicated product model, which could have a substantial positive impact on the simulation by organizing product information for agents and allowing access to specific engineering information. Integrating it depends on agent tools for accessing and managing the product model, for instance, to read a bill of materials. Complex information like product geometry and physics-related simulation is already being integrated into generative-AI systems via multimodal Retrieval Augmented Generation (RAG) and tool-augmented CAD interfaces [9,10]. Its readiness and compatibility with agent-based simulation are unclear.

5.2 Persona design and behavior

Data-grounded personas. An accurate persona design should be grounded in acknowledged data about the target stakeholders to reduce bias and minimize assumptions [11]. Examples include gender and age distributions, country of residence, and typical educational background. The downside is that data must be available and accessible to a human to hand-craft the persona traits, or to a software tool (e.g., a GenAI system) to automatically generate a persona. Another aspect is that the more features the persona has, the more relevant it is to check if features are dependent. If they are, generating a consistent persona would benefit from grounding in data or at least specific reasoning or plausibility checks. For example, a persona's age and experience: a young persona with extensive experience might seem unrealistic in many scenarios. Our experiments yielded no clear results [12] on whether the GenAI system used an internal representation of these dependencies (e.g., in its training data) when generating the personas.

However, Table 1 includes two controversial personas, where the age and experience stated in the occupation description might raise plausibility questions (assuming that becoming a professor at 28 years is unlikely but not impossible).

Targeted persona design. Changes to persona traits have unknown consequences for the simulation and its results. It is not yet clear whether the experiment designer can estimate the impact of persona changes before running the experiment. Investigating this would likely require a deeper analysis of how each specific GenAI model represents personas and their conversations (e.g., [13]). Such a research activity is highly ambitious and would require tools to inspect a model’s internal working, which is unlikely for commercial models from, for instance, OpenAI or Anthropic, and more likely for models with open weights, like Qwen (Alibaba Cloud) or DeepSeek (DeepSeek Ltd.). On the positive side, the uncertainty above may lead to interesting and highly desirable emergent situations that typically do not occur in scenarios with humans, for instance, because their traits do not provide opportunities for such emergent phenomena. Also, systematically running many simulations with edge-case traits may lead to trait combinations that drive the conversation in entirely unexpected directions, revealing something that accessible human participants could never conclude. Such results would be relevant when the number of target stakeholders is high and diverse, while the pool of accessible human participants is small (e.g., due to budget constraints or missing contact).

Persona integrity for agents. GenAI systems used in persona-based role-playing, such as the simulated workshop above, may respond in ways inconsistent with their persona [14]. This integrity was not specifically investigated in the experiments, but it is a critical aspect of an authentic simulation. The impact of inconsistencies on simulation results is unclear and requires further investigation, potentially using new tools to examine GenAI system behavior.

5.3 Experiment design

Choosing simulated participants for an experiment is a key decision in the simulation. While the team composition among the already-defined personas depends on the task and scenario, different teams are likely to produce better or worse simulation results. As with the persona design and behavior, it is not yet clear whether the impact of team composition on results can be reliably anticipated.

5.4 Result quality

We did not focus on the quality of the simulation results in this study, especially compared to human stakeholders. Our observations and results indicate that the generated recommendations are concrete and appear actionable (we did not implement them to verify). Further analysis of result quality could leverage the generated explanations for each expected impact and seek human expert opinion (e.g., acceptable vs. unacceptable). Another option is for humans to create a validation dataset (gold standard dataset) in which they, for instance, provide their own recommendations and explain the expected impacts based on their expertise. This approach would require a comparative study where a human team competes with a simulated team – a hybrid team could also provide useful insights, but requires a more complex technical setup where agents can monitor human conversations and intervene.

Another aspect of result quality is that the generated recommendations are not the result of a logic-driven process but rather a sequential refinement of a single final prompt sent to the GenAI system to generate tokens representing recommendations. This is an inherent characteristic of the selected simulation library.

Finally, quantitative values in the expected impact (e.g., 15% or 5 minutes less) are hard to verify because the rationale is missing from the agent responses, and the experiment input intentionally does not provide baselines. The agents sometimes asked the facilitator for such details, though. We observed

this often when the agents were not strictly encouraged to rely on their role expertise – they asked for details at almost every step, and we decided to modify the system prompt to avoid this.

5.5 Scalability

Technical factors: The simulation’s ability to work effectively across a wide range of development workshop situations is a requirement for successful application and adoption in practice. We identified two drivers of scalability: simulation time and costs. The concept has several parameters that influence these drivers, such as the number of participating agents, the number of simulation rounds, the selected GenAI system, and input and output token limits. Typically, more agents and rounds result in more API calls to the model, driving the token count and related costs. Our results in Table 2 do not clearly substantiate this pattern, especially because the experiment runs #1, #7, and #10 differ significantly, even though they each have 11 rounds. We did not investigate these data points further. Another typical assumption is that a GenAI system that generates responses of potentially higher quality typically takes longer and costs more. Higher token limits for input and output increase time and costs; however, they also influence the quality of the results, including reasoning quality and response length.

Non-technical factors: In addition to the technical parameters above, preparing accurate, data-grounded personas will likely affect the time and cost of the simulation. This effort is probably a one-time investment that would pay off the more simulations reused personas. There are already initiatives to create persona databases to address this issue [15]. Also, the impact of assessing AI-based simulation results (e.g., validation, review, and interpretation) may not scale if intense human involvement is desired due to a lack of trust in the results or because it is required by law, compliance, or contracts. Software tools for automated result assessments may increase the scalability to some extent, though. Examples include validation against existing knowledge bases, standards, guidelines, or trusted deterministic simulations.

6 Conclusion

The experiments and the related concept provide a first systematic investigation for using generative AI and personas to simulate stakeholders in product development. While the initial results are a good starting point, several limitations must be considered to better understand the merits of this concept. *First*, this study covered only one scenario and one product, limiting the generalizability. Also, it doesn’t cover product-services or product-related software, which are essential components of many complex value offers and smart products. *Second*, the simulation does not allow iterations across phases, i.e., agents or the human facilitator cannot decide to repeat a phase and apply more refined information. *Third*, the experiments could be more systematic in their use of the knowledge base, e.g., by cleaning the database between steps to investigate how agents ask the facilitator questions. *Fourth*, there was no systematic assessment of result quality because we had not yet identified adequate metrics and tools. *Fifth*, the simulation lacks integrations with a formal product model concerning function, structure, and behavior. Future integrations may cover other computer-based simulations along the product life cycle, ranging from engineering simulations to simulated product usage. Addressing the above limitations could lead to further automation of product development and shorter product lifecycles.

Acknowledgements

This work is funded by the European Union's Horizon Europe project WASABI “White-label shop for digital intelligent assistance and human-AI collaboration in manufacturing” (Grant agreement No 101092176). The work presented here reflects only the authors’ view and the European Commission is not responsible for any use that may be made of the information it contains.

References

- [1] R. M. Crowder, M. A. Robinson, H. P. N. Hughes and Y.-W. Sim, "The Development of an Agent-Based Modeling Framework for Simulating Engineering Team Work," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 42, no. 6, pp. 1425-1439, 2012. DOI: <https://doi.org/10.1109/TSMCA.2012.2199304>.
- [2] Y. Liu, B. S. Onggo and J. Busby, "Agent-based modelling of user engagement in new product development," *Technovation*, vol. 135, p. 103062, 2024. DOI: <https://doi.org/10.1016/j.technovation.2024.103062>.
- [3] M. Buchenau and J. F. Suri, "Experience prototyping," in *Proceedings of the 3rd conference on Designing interactive systems: processes, practices, methods, and techniques*: New York, NY, USA: ACM, 2000, pp. 424–433.
- [4] J. Pruitt and J. Grudin, "Personas," in *Proceedings of the 2003 conference on Designing for user experiences*: New York, NY, USA: ACM, 2003, pp. 1–15.
- [5] A. Schmidt, P. Elagroudy, F. Draxler, F. Kreuter and R. Welsch, "Simulating the Human in HCD with ChatGPT: Redesigning Interaction Design with AI," *Interactions*, vol. 31, no. 1, pp. 24-31, 2024. DOI: <https://doi.org/10.1145/3637436>.
- [6] J. Shin, M. A. Hedderich, B. J. Rey, A. Lucero and A. Oulasvirta, "Understanding Human-AI Workflows for Generating Personas," in *Designing Interactive Systems Conference*: New York, NY, USA: ACM, 2024, pp. 757–781.
- [7] P. Salem, R. Sim, C. Olsen, P. Saxena, R. Barcelos and Y. Ding, "TinyTroupe GitHub Project. LLM-powered multiagent persona simulation for imagination enhancement and business insights.," [Online]. Available: <https://github.com/microsoft/TinyTroupe>. [Accessed 22 January 2026].
- [8] P. Salem, R. Sim, C. Olsen, P. Saxena, R. Barcelos and Y. Ding, *TinyTroupe: An LLM-powered Multiagent Persona Simulation Toolkit*, 2025.
- [9] C. Kienle, B. Alt, D. Katic, R. Jäkel and J. Peters, *QueryCAD: Grounded Question Answering for CAD Models*, 2024.
- [10] S. Li and J. Corney, "MechRAG: a multimodal large language model for mechanical engineering," *Communications engineering*, vol. 4, no. 1, p. 187, 2025. DOI: <https://doi.org/10.1038/s44172-025-00517-z>.
- [11] B. J. Jansen, J. Salminen, S. Jung and K. Guan, *Data-Driven Personas*, 1st ed. 2021Cham: Springer International Publishing; Imprint: Springer, 2021.
- [12] A. S. Vezhnevets, J. P. Agapiou, A. Aharon, R. Ziv, J. Matyas, E. A. Duéñez-Guzmán, W. A. Cunningham, S. Osindero, D. Karmon and J. Z. Leibo, *Generative agent-based modeling with actions grounded in physical, social, or digital space using Concordia*, 2023.
- [13] S. Lee, S. Lim, S. Han, G. Oh, H. Chae, J. Chung, M. Kim, B. Kwak, Y. Lee, D. Lee, J. Yeo and Y. Yu, *Do LLMs Have Distinct and Consistent Personality? TRAIT: Personality Testset designed for LLMs with Psychometrics*, 2024.
- [14] P. Bhandari, N. Fay, M. Wise, A. Datta, S. Meek, U. Naseem and M. Nasim, *Can LLM Agents Maintain a Persona in Discourse?*, 2025.
- [15] J. Rupprecht, L. Fröhling, C. Wagner and M. Strohmaier, *German General Personas: A Survey-Derived Persona Prompt Collection for Population-Aligned LLM Studies*, 2025.

Performance evaluation of Large Language Models in extracting requirements sets from technical documents

Iris Graessler¹, Jan Pfeifer^{1*}

¹Heinz Nixdorf Institute, Chair for Product Creation, 33102 Paderborn

*Corresponding author. E-Mail address: jan.pfeifer@hni.upb.de; Tel.:05251/60 6265

Abstract: Manufacturers operate under dynamic regulatory conditions, leading to frequently changing requirements that must be managed with high effort. AI-based requirements extraction using Large Language Models (LLMs) has the potential to support engineers in this task by analyzing large documents, for example. Existing evaluation approaches in AI-supported requirements engineering mainly assess individual requirements and their consistency with source documents, while quality criteria of requirements sets are largely neglected. This paper presents a five-step approach to evaluate the performance of current LLMs in generating requirements sets: a systematic literature review to identify quality criteria of requirements sets (1), exemplary generation of requirements sets from the Machinery Directive 2006/42/EC using LLMs (2), examination (3), evaluation (4) and comparison (5) of the generated requirements sets. The results support engineers in selecting LLMs for creating requirement sets and in evaluating future LLMs.

Keywords:

Requirements Engineering, Artificial Intelligence, Large Language Models, Generative AI

1 Introduction

The process of engineering compliant technical systems is subject to complexity [1]. This complexity results, among other things, from numerous constraints that have to be considered throughout the entire Product Lifecycle (PLC) [2]. Technical documents, such as standards or regulations, are one possible form of documentation for these constraints. Technical documents are often extensive in content, written in natural language, complex and difficult to understand. Manually extracting requirements involves a considerable amount of effort. To provide engineers with validated knowledge in a machine-readable and semi-structured, rational and objective form, knowledge can be extracted automatically from technical documents. Using Artificial Intelligence (AI) promises great potential in requirements engineering.

Preliminary work has primarily addressed the technical implementation of AI-supported document analysis for extracting requirements. The BIKINI project, funded by the Federal Ministry of Research, Technology and Space of Germany, examined technical challenges in detail in using AI in requirements extraction. For example, GRAESSLER et al. [3] have identified challenges in AI-based requirement extraction. These challenges include, for example, different data types of source documents, different languages within a document, or quality of the training data for AI algorithms. These challenges have been overcome through a combination of approaches, applying new methods and technical advances in machine learning (ML). For example, augmented datasets can be used to improve quality of training data [4]. In further research by GRAESSLER et al. [5] and LUTTMER et al. [6], existing AI models for language processing, such as Bidirectional Encoder Representations from Transformers (BERT), are compared in terms of their performance in requirement extraction.

Large Language Models (LLMs) are one type of AI that is already being used in several applications in everyday engineering work. LLMs are not sufficiently considered in existing areas of research in requirements engineering. Many technical challenges mainly identified for generating individual requirements, such as document language, file type or formatting, have become secondary due to the rapid technical development of artificial intelligence. Applicability of AI can be shifted to larger contexts, such as entire sets of requirements, thanks to technical progress induced by LLMs. This results in a need for research, particularly in LLM-supported generation of high-quality sets of requirements rather than individual requirements. Two research questions (RQ) result from this clarification:

RQ1: What quality criteria exist for requirements sets?

RQ2: What level of performance do current LLMs achieve in generating requirements sets?

The paper at hand is structured as follows: First, the methodology used to answer the RQs is described. Based on a literature analysis, quality criteria of requirements sets are derived. Using a developed prompt, requirements sets are generated through three common LLMs. Generated requirement sets are then evaluated and compared in terms of fulfilling previously determined criteria of requirement sets.

2 Methodology

To answer the RQs, a five-step scientific methodology is applied. Following the DRM methodology [7], a prescriptive approach is defined: Under the assumption, that LLMs help to solve engineering tasks, the methodology includes generating requirement sets using LLMs and then comparing them with regard to data quality criteria. Based on existing approaches for engineering of requirements sets, a systematic literature review in accordance with PRISMA protocol [8] is conducted to identify quality criteria of requirements sets (1). Subsequently, market-leading LLMs are instructed to generate requirements sets based on the Machinery Directive 2006/42/EC [5] and considering identified criteria of requirements sets (2). The generated requirements sets are then analyzed in terms of their fulfilment of requirements set data quality criteria (3), individually evaluated according to the degree of criteria fulfilment (4) and compared (5).

3 Theoretical Foundation

Requirements management is the structured process of eliciting, documenting, analyzing, and managing requirements throughout the engineering of a system [9,10]. Requirements Engineering starts with an internal or external engineering order and focuses on identifying stakeholder needs, expectations, and constraints. These needs are translated into customer requirements and further refined into system requirements that define the conditions and capabilities a system or subsystem must fulfill [9]. Requirements management is important because it provides a clear and agreed-upon basis for system architecture, ensures compliance with standards and specifications, and supports controlled engineering decisions. Well-managed requirements reduce risks, misunderstandings, and costly changes later in the engineering process [10]. Regulatory documents are one source of requirements. Depending on industry, compliance with such regulatory requirements is essential for product approvals [10].

LLMs are large neural language models that are trained on billions of words [11]. Their size and use of transformer architecture enable them to analyze and generate language and solve a variety of tasks without task-specific training [12]. Through textual instructions known as prompts, LLMs can be instructed to perform specific tasks. Reference information can be added to the query in the form of document datasets [12]. Due to the complexity and increasing computing power of LLMs, previous machine learning benchmarks are no longer sufficient [13]. Modern tests for LLMs such as LiveBench [13] challenge LLMs with 18 tasks from the categories of math, coding, reasoning, language

comprehension, instruction following, and data analysis. General LLM tests such as LiveBench provide an overview of a model's performance, but they do not replace application-specific tests.

4 Generation of Requirements Sets

In the following, requirements sets generated by common LLMs which are market-leading in industry according to KPMG [14] are compared in terms of their resulting data quality. The three LLMs are OpenAI GPT5.0, Meta LLaMA 3.3 70B, and Google Gemma 3 27B (Gemini). First, approaches in literature are identified that enable to evaluate the quality of requirements, typically proposing a set of evaluation criteria. These approaches are then analyzed for specific criteria of requirements sets. As a result regarding RQ1, a set of criteria of requirements sets generated by LLMs is created. A prompt is developed considering identified quality criteria of requirements sets. The prompt instructs the LLMs to generate sets of requirements based on a requirements stereotype developed in workshops. The generated requirements sets are then evaluated and compared.

4.1 Identification of relevant approaches

To determine quality criteria of requirements sets, two topics are covered in literature review according to PRISMA [8]. *Requirements engineering* and *evaluation schemes* are combined through search vectors into a search string. Literature identified in Scopus and Web of Science is filtered according to the relevance specifically targeting requirements sets, evaluability of the requirements set criteria and relevance to mechanical engineering. For example, literature is filtered out that deals primarily with quality of requirements and not with quality of requirements sets. Based on title and abstract, the initial total of 535 literature references includes 21 relevant references, and after detailed review a final number of nine references.

In product engineering, comprehensive requirement criteria are defined in norms and standards to ensure quality of individual requirements and requirements sets. All nine identified references refer to norms and standards. For example, ISO/IEC/IEEE 29148 [15] distinguishes between criteria of individual requirements, requirements sets and requirement language. Criteria of requirements sets include *complete, consistent, feasible, comprehensible and able to be validated*. Criteria of requirements language include *avoiding superlatives, subjective language, vague pronouns, ambiguous terms such as adverbs and adjectives, open-ended terms, comparative phrases, loopholes and terms that imply totality*. Other basic literature besides norms and regulations such as the INCOSE handbook [2] refer to corresponding standards and norms to describe requirement criteria. In industry-oriented, application-related literature such as [1], a reduced set of quality criteria are used to ensure pragmatic applicability. The criteria are *complete, clear, consistent and correct*.

In software engineering, there are more comprehensive quality criteria of requirements and requirements sets. For example, BOKHARI and SIDDIQUI [16] summarize Software Requirements Specification as a key document describing external behavior of a software system and its role in achieving successful software engineering. The study consolidates essential quality criteria of requirements documents, distinguishing internal attributes – e.g., *unambiguity, correctness, completeness, consistency, verifiability, modifiability, traceability, precision, and appropriate level of detail* – from external attributes – e.g., *achievability, conciseness, design independence, reusability, and electronic storage*. Together, these criteria provide a structured framework for producing high-quality, maintainable, and effective requirements specifications. HALLIGAN [17] defines and systematizes requirements quality through a set of ten requirements quality factors that collectively characterize well-formed requirements. The factors *correctness, completeness, consistency, clarity, non-ambiguity, connectivity, singularity, testability, modifiability, and feasibility* capture both semantic and structural properties necessary for requirements to be understandable, implementable, and verifiable. HALLIGAN further introduces a

structural template for natural-language requirements, decomposing each statement into standardized elements such as actor, conditions, action, object, and constraints [17]. With help of criteria, sets of requirements generated by LLM can be systematically evaluated for quality.

4.2 Analyzing relevant approaches regarding requirements sets

Based on the identified nine approaches, a set of criteria of requirements is compiled answering RQ1. The set of criteria includes criteria that relate to entire requirements sets. Criteria of individual requirements like linguistic criteria are not included. As a review result of relevant literature, the subsequent criteria of requirements are determined: *complete*, *consistent*, *feasible*, *clear*, *testable* and *correct*. These criteria are laid down in standards (e.g. [18]) and are frequently cited in the nine identified publications on requirements engineering. A list of all criteria considered including description and reference is shown in Table 1.

Table 1: Quality criteria of requirements sets

Criteria	Description	Reference
complete	The requirements set needs no further amplification because it contains everything pertinent to the definition of the system or system element being specified. In addition, the set contains no To Be Defined (TBD), To Be Specified (TBS), or To Be Resolved (TBR) clauses. Resolution of the TBx designations may be iterative and there is an acceptable timeframe for TBx items, determined by risks and dependencies.	[1,2,18]
consistent	The requirements set does not have individual requirements which are contradictory. Requirements are not duplicated. The same term is used for the same item in all requirements.	[1,2,18]
feasible	The requirements set can be satisfied by a solution that is obtainable/affordable within life cycle constraints (e.g., cost, schedule, technical, legal, regulatory).	[2,18]
clear	The requirements set is stated in such a way so that it can be interpreted in only one way. The requirements set is stated simply and is easy to understand.	[2,18]
testable	The requirements set has the means to prove that the system satisfies the specified requirements set. Evidence may be collected that proves that the system can satisfy the specified requirements set. Verifiability is enhanced when the requirements set is measurable.	[2,18]
correct	The requirements set is upwards traceable to specific documented stakeholder statement(s) of need, higher tier requirements set, or other source (e.g., a trade or design study).	[1,2,18]

4.3 Criteria for structuring requirements sets

To generate comparable requirements sets, an output format must first be specified. The format of the requirements set is based on DIN EN ISO/IEC/IEEE 29148 [18] and was modified in workshops with industry partners for the [anonymous] project founded by the European Union. As part of the project, a knowledge base is to be established that will enable requirements to be integrated into engineering processes according to their validity. The table format is illustrated in Table 2.

Table 2: Requirements output format

ID	Description	Validation criterion	Life Cycle Phase	Time of application	Geo. Area of application	Reference to source	Comment

Each requirement is identified in the first column by an ID composed of the name of the technical document and a corresponding number. In the second column, the requirement is described in accordance with the SOPHIST template [19], providing a clear and structured formulation. A validation

criterion is defined in the third column, outlining a measurable property a product must fulfill to satisfy the requirement. Assignment to a particular phase of the product life cycle according to the generic Product Lifecycle (gPLC) [20] is described in the fourth column, which covers strategic planning, engineering, realization, operation/ service delivery, and decommissioning. The fifth column specifies the applicable time frame or period of validity for the requirement. The sixth column describes where the requirement applies. Reference information is provided in the seventh column, identifying the source technical document and the exact location within that document from which the requirement was extracted. Additional comments and other relevant details, such as cross-references to technical documents, are included in the eighth column.

4.4 Development of a prompt for the generation of requirements sets

A comprehensive prompt is an essential part for the extraction of requirements with LLMs [12]. The prompt was developed, iterated and optimized considering guidelines for developing prompts, for example from Google [21] or OpenAI [22]. Iterations consisted, for example, of adapting context examples or adding special instructions that activate deep thinking processes. The prompt is presented and explained below. All components are described sequentially.

The **first block** describes the basic role that the LLM is to assume. In case of requirements extraction, this is the role of a requirements engineer:

“You are a requirements engineer and are tasked with extracting requirements set from the following document.”

The **second section** describes the procedure. Two basic quality criteria are specified: Firstly, requirements should relate to a specific technical system. A positive and a negative example are given. Secondly, requirements should be extracted in full. The list of requirements should neither be shortened nor idealized.

“Analyze the document and identify all requirements that relate to a specific system to be engineered. Good example: The system to be developed must contain 50% recycled material. Bad example: The development team must consist of 5 people. The most important criterion is that you really extract all requirements. No abbreviated or idealized list.”

The **third section** describes the structure of the eight-column results table. Specific examples are given for each row. The exemplary instruction for the first column is as follows:

“The requirements should be displayed in a table with the following columns: Column 1: Requirement ID with reference to the content of the document. Do not simply refer to the data name but refer to the content. Good: ISO14001-001 Bad: ISO14001-001.pdf.”

The **fourth section** describes specific quality criteria (Table 1) for the requirements set. They are listed with relevant examples. For example:

“The requirements, both collectively and individually, should meet the following criteria: Completeness: No requirement should be missing. All requirements should be extracted from all sections of the document. Also list requirements that refer to other documents.”

The **fifth block** describes the concrete formulation of requirements. All requirements should be formulated according to the SOPHIST template:

“Formulate the requirements in the SOPHIST template: The [subject] must/should/may [predicate] [object] condition/qualification.”

In the **sixth block**, the allocation of life cycle phases is specified. The individual phases are presented and explained with the help of short examples. For example:

“Assign the appropriate lifecycle phase to each requirement: Strategic planning, engineering, realization, use, decommissioning. ‘Strategic planning’ refers to all content that must take place before the actual development, e.g. business orientation.”

At the end of the prompt, general conditions are listed. For example, tables should be exportable. One example is further processing in Microsoft Excel, an alternative could be XML:

“Please provide me with the table in a format that I can copy into Excel.”

4.5 Examination, evaluation and comparison of the extracted requirements sets

In the following, the requirements sets extracted by the LLM models are analyzed, evaluated, and then compared. Examples of requirements from the generated requirements sets with the described prompt are first presented. Requirements sets are then evaluated according to the criteria listed in Table 1.

In Table 3, examples of requirements from the requirements set created by ChatGPT are shown.

Table 3: Excerpt from the ChatGPT requirements set (4 of 26 requirements)

Requirement ID	Requirement Text (SOPHIST Template)	Validation Criterion	Lifecycle Phase	Temporal Validity	Geographical Validity	Source Reference
MD2006-AnnI-1.1.1-001	The system must be designed and constructed so that it can be operated, adjusted, and maintained without putting persons at risk under the intended conditions of use.	No hazardous situation identified during operation, adjustment, or maintenance per risk assessment.	Engineering	From 2009-12-29	EU / EEA	Annex I, 1.1.1 —
MD2006-AnnI-1.1.2-001	The system must be designed and constructed taking into account the principles of safety integration.	Risk reduction hierarchy demonstrably applied (inherently safe → safeguards → information).	Engineering	From 2009-12-29	EU / EEA	Annex I, 1.1.2 —
MD2006-AnnI-1.1.2-002	The system must eliminate or reduce risks as far as possible by design and construction.	Residual risk list minimized and justified.	Engineering	From 2009-12-29	EU / EEA	Annex I, 1.1.2 —
MD2006-AnnI-1.1.2-003	The system must incorporate necessary protective measures against risks that cannot be eliminated by design.	Protective measures verified against all residual risks.	Engineering	From 2009-12-29	EU / EEA	Annex I, 1.1.2 —

The **ChatGPT** requirements set represents a comprehensive and systematic transformation of the relevant provisions of Directive 2006/42/EC into system-level, verifiable requirements for a technical system. In the evaluation, the extracted requirements are first compared with the original specifications in the regulations. Ground truth is defined by manual extraction of requirements from the regulation, and verification by three experts from industry. The requirements set is *complete*. The requirements set describes required capabilities, characteristics, constraints, and safety-related quality factors necessary to achieve compliance with the essential health and safety requirements from Directive 2006/42/EC, and it contains no TBD, TBS, or TBR placeholders. While the set does not aim to define application-specific functional or performance characteristics such as payload, reach, accuracy, or cycle time, this limitation is *consistent* with the nature of the source document and does not reduce the completeness of the

compliance baseline. The SOPHIST template was only adhered partially. While the basic scheme was followed, in some cases (e.g. MD2006-AnnI-1.1.1-001) several individual requirements were mixed into one aggregated requirement. Requirements are internally consistent and structured. Each requirement is uniquely defined, does not conflict with or duplicate other requirements, and uses homogeneous terminology, measurement concepts, and verification logic that align closely with the language and intent of the Machinery Directive. Potential overlaps inherent in the source text have been resolved through atomic decomposition, resulting in a coherent and non-contradictory set. From a *feasibility* perspective, the requirements seem realistic and achievable within typical technical, cost, and schedule constraints for technical systems. The final feasibility depends on the specific system and company and cannot be assessed without taking into account the context, which has not yet been considered. The requirements set is also *clear* and understandable, both in terms of individual requirement statements and their relation to the overall system and its lifecycle. Explicit lifecycle phase assignment, and inclusion of validation criteria make expectations transparent and support traceability and engineering decision-making. Furthermore, the requirements are *testable*. Each requirement can be verified through inspection, analysis, testing, or documentation review, enabling systematic verification and validation activities. However, specific validation criteria must be determined for each individual application. Finally, the set is *correct*. All requirements are directly derived from the Machinery Directive and accurately reflect its normative “shall” and “must” statements without introducing additional or speculative content. Overall, the requirements set constitutes a high-quality, regulation-driven system requirements baseline for technical systems.

In Table 4, examples of requirements from the requirements set created by Gemini are shown.

Table 4: Excerpt from the Gemini requirements set (4 of 21 requirements)

Requirement ID	Requirement Text (SOPHIST Template)	Validation Criterion	Lifecycle Phase	Temporal Validity	Geographical Validity	Reference in Source
MD2006/42-EHSR-1.1.2	The system must be designed to be fit for its function and operated without risk under intended conditions.	Safety factor ≥ 1.1 during nominal load tests	Engineering	From 29 Dec 2009	EU / EEA	Annex I, 1.1.2 (a)
MD2006/42-EHSR-1.1.3	The system must consist of materials that do not endanger the health or safety of persons (e.g., toxic dust).	REACH/RoHS compliance certificates	Engineering	From 29 Dec 2009	EU / EEA	Annex I, 1.1.3
MD2006/42-EHSR-1.1.5	The system must be designed to be handled safely (e.g., lifting points for installation).	Presence of clearly marked lifting eye bolts	Engineering	From 29 Dec 2009	EU / EEA	Annex I, 1.1.5
MD2006/42-EHSR-1.2.1	The system must feature control systems designed to prevent hazardous situations from occurring.	Performance Level (PL) r achieved according to ISO 13849-1	Engineering	From 29 Dec 2009	EU / EEA	Annex I, 1.2.1

The **Gemini** requirements set forms a comprehensive and well-structured compliance-oriented system requirements baseline derived from Directive 2006/42/EC. The set is nearly *complete*. Some requirements are missing like, for example, that the system must be equipped with an emergency stop. The set contains no undefined or open placeholders such as TBD, TBS, or TBR. The requirements are internally *consistent* and logically coherent. Each requirement is uniquely identified, unambiguous, and free of conflicts or contradictions. Cross-references are used appropriately and indicate conceptual

dependencies without overlapping. Terminology, structure, and units are used consistently throughout the set and align with both the Machinery Directive and established engineering and safety standards. From a *feasibility* standpoint, all requirements are realistically achievable. Specified validation criteria such as load testing, Performance Level achievement, electrical safety testing, noise measurement and documentation checks can be implemented within acceptable cost, schedule, and technical risk constraints. The requirements are *clearly* formulated using the SOPHIST template, which makes expectations explicit and reduces interpretational ambiguity. In addition, the requirements are fully *testable*. Each requirement can be verified through inspection, analysis, testing, or demonstration. Finally, the requirements set is *correct*. Every requirement is directly traceable to a specific clause of Directive 2006/42/EC and accurately reflects its normative intent without introducing additional or speculative content. In Table 5, examples of requirements from the requirements set created by LLaMA are shown.

Table 5: Excerpt from the LLaMA requirements set (4 of 9 requirements)

Requirement ID	Requirement Text (SOPHIST Template)	Validation Criterion	Lifecycle Phase	Temporal Validity	Geographical Validity	Reference in Original Document
MD-001	The manufacturer must operate an approved quality system for design, manufacture, final inspection and testing	Approved quality system	Realization	from 2006	all EU countrys	ANNEX X
MD-002	The technical file must demonstrate that the machinery complies with the requirements of this Directive	Compliance with Directive	Engineering	from 2006	all EU countrys	ANNEX VII, Part A
MD-003	The manufacturer must take all measures necessary to ensure compliance of the manufactured machinery with the technical file and the requirements of this Directive	Compliance with technical file and Directive	Realization	from 2006	all EU countrys	ANNEX VIII
MD-004	The quality system must ensure conformity of the machinery with the provisions of this Directive	Conformity with Directive	Engineering	from 2006	all EU countrys	ANNEX X

The LLaMA requirements set shows significant deficiencies. In terms of *completeness*, the set is clearly insufficient. A substantial number of essential requirements are missing, and large parts of the source document were not analyzed or translated into system requirements. In addition, a technical system is not always the reference object of the requirements set. In some cases, requirements are placed on the manufacturer rather than on the system to be engineered (e.g. MD-003). *Consistency* is also limited. While no direct contradictions are evident, the lack of systematic coverage and the incomplete analysis of the source material lead to gaps and conceptual discontinuities within the set. *Feasibility* cannot be reliably assessed, as many requirements are formulated at a high level of abstraction without sufficient technical detail. The *clarity* of the requirements set is reduced by vague wording and absence of a structured template comparable to SOPHIST template. As a result, expectations are not always explicit, leaving room for interpretation. *Testability* of the requirements set is weak. Validation criteria are often imprecise or missing altogether, making objective verification through inspection, analysis, or testing difficult or impossible. Finally, correctness is not ensured. Due to the incomplete consideration of the

original directive and missing traceability to specific clauses, it is unclear whether the requirements accurately reflect the normative intent of the source document.

In the following (Figure 1), the performance of LLMs is compared based on their fulfillment of criteria of requirements sets, based on a three-point scale, with a value of 0 representing poor fulfillment and a value of 2 representing good fulfillment of the criterion.

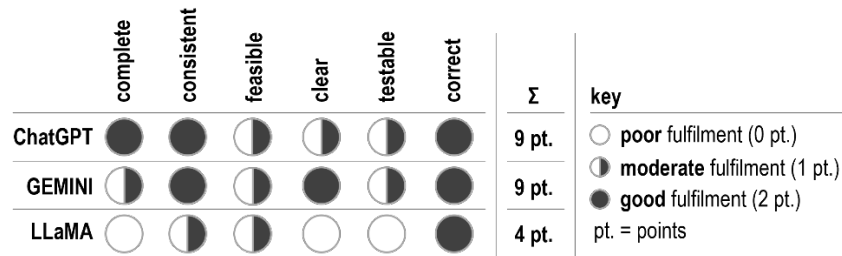


Figure 1: Comparison of LLM performance

Comparing ChatGPT, Gemini, and LLaMA requirements sets reveals clear differences in quality across all key evaluation criteria: *completeness*, *consistency*, *feasibility*, *clarity*, *testability*, and *correctness*. The **ChatGPT** set achieves a “good” rating in three out of six criteria. The ChatGPT requirements set provides a comprehensive, systematically structured, and fully traceable set of requirements for a system, with clear, testable, and feasible statements. Usage of the SOPHIST template and generic validation criteria can be improved. **Gemini** also delivers a strong performance, with most criteria rated “good” except for completeness, which was rated slightly lower as “Good” due to a few missing requirements, such as the emergency stop. The set is internally consistent, feasible, clear, testable, and correct, with well-defined verification methods and traceability to the directive. In contrast, the **LLaMA** requirements set exhibits significant deficiencies across all criteria. Completeness and correctness are rated “poor”, as many essential requirements are missing, the source document is only partially analyzed, and traceability to the directive is lacking. Consistency, feasibility, and clarity are also rated low with gaps and vague formulations that hinder practical implementation. Testability is severely limited, as validation criteria are imprecise or absent, preventing reliable verification. Overall, LLaMA falls far short of standards demonstrated by ChatGPT and Gemini, making it unsuitable as a compliance-oriented system requirements baseline.

This comparison highlights that while ChatGPT and Gemini provide structured, high-quality requirements sets suitable for a baseline for regulatory compliance, LLaMA’s approach leads to incomplete, ambiguous, and largely untestable requirements, emphasizing the critical importance of systematic methodology and traceability in requirement generation. Regarding RQ2, in results show that the LLMs ChatGPT and Gemini can generate usable requirements sets. However, these are not flawless and must be reviewed and adjusted by humans. Research also has determined that the criteria “feasibility” and “testability” depend on the corresponding engineering use case. These must be added to future prompts to further improve quality of generated requirements sets. One approach, for example, would be to provide the LLM with project-specific data such as parts lists. Furthermore, quality of extracted requirements sets from other technical documents must be evaluated.

5 Conclusion and outlook

Increasing complexity of technical systems requires systematic extraction of requirements from technical documents, which are often lengthy, complex, and difficult to understand. Manual extraction is time-consuming, making AI, particularly LLMs, a promising solution. The study follows a five-step scientific methodology: first, quality criteria of requirements sets are identified through a systematic

literature review; then, leading LLMs are instructed to generate requirements from the Machinery Directive 2006/42/EC. Generated requirements sets are subsequently analyzed, evaluated, and compared. Based on the literature, six criteria of requirements sets were defined: completeness, consistency, feasibility, clarity, testability, and correctness. An iteratively developed prompt ensures that LLMs systematically capture all relevant requirements. Three LLMs were evaluated: ChatGPT, Gemini, and LLaMA. ChatGPT and Gemini generated comparatively good requirements sets with deficits in feasibility and testability, while LLaMA showed significant deficiencies, including missing, ambiguous, and untestable requirements. These findings highlight the importance of a systematic methodology, clear quality criteria, and traceability for automated requirement generation. Results confirm usefulness and applicability of LLMs in requirements engineering. The evaluation procedure can be transferred or extended to future LLMs. The next step is to investigate how the quality of LLM-generated requirement sets can be further improved. To this end, new models and model types will be examined, as well as the inclusion of retrieval-augmented generation systems. This improvement can enhance the quality of requirement sets in terms of feasibility and testability.

References

- [1] I. Gräßler and C. Oleff, *Systems Engineering Berlin, Heidelberg*: Springer Berlin Heidelberg, 2022.
- [2] INCOSE, *INCOSE Systems Engineering Handbook* Wiley, 2023.
- [3] I. Graessler, D. Preuss and J. Pottebaum, “Extrahierung von Anforderungen aus natürlich-sprachlichen Lastenheften: Was erschwert eine KI-basierte Extrahierung?,” *Softwaretechnik-Trends* Band 42, Heft 1., 2022.
- [4] I. Graessler, D. Preus, L. Brandt and M. Mohr, “Efficient Extraction of Technical Requirements Applying Data Augmentation,” in *2022 IEEE International Symposium on Systems Engineering (ISSE)*: IEEE, 2022, pp. 1–8.
- [5] I. Graessler, D. Oezcan and D. Preuss, “AI-based extraction of requirements from regulations for automotive engineering,” in *DS 125: Proceedings of the 34th Symposium Design for X (DFX2023)*: The Design Society, 14 and 2023, pp. 163–172.
- [6] J. Luttmer, V. Prihodko, D. Ehring and A. Nagarajah, “Requirements extraction from engineering standards – systematic evaluation of extraction techniques,” *Procedia CIRP*, vol. 119, pp. 794-799, 2023.
- [7] M. J. Page, J. E. McKenzie, P. M. Bossuyt, I. Boutron, T. C. Hoffmann, C. D. Mulrow, L. Shamseer, J. M. Tetzlaff, E. A. Akl, S. E. Brennan, ..., “The PRISMA 2020 statement: an updated guideline for reporting systematic reviews,” *BMJ (Clinical research ed.)*, vol. 372, n71, 2021.
- [8] VDI/VDE, “Development of mechatronic and cyber-physical systems. VDI/VDE 2206:2021,” 2021.
- [9] C. Rupp, *Requirements-Engineering und -Management. Das Handbuch für Anforderungen in jeder Situation*, 7., aktualisierte und erweiterte Auflage München: Hanser, 2021.
- [10] A. Tamkin, M. Brundage, J. Clark and D. Ganguli, “Understanding the Capabilities, Limitations, and Societal Impact of Large Language Models,” 2021.
- [11] OpenAI, “GPT-4 Technical Report,” 2023.
- [12] C. White, S. Dooley, M. Roberts, ..., “LiveBench: A challenging contamination-limited LLM benchmark,” 2025.
- [13] KPMG International Limited, “Woher bezieht ihr Unternehmen Large Language Models?,” 2024.
- [14] 2023 IEEE 31st International Requirements Engineering Conference (RE) (IEEE, 2023).
- [15] M. U. Bokhari and S. T. Siddiqui, “Metrics for Requirements Engineering and Automated Requirements Tools,” 2011.
- [16] R. J. Halligan, “Requirements Quality Metrics: The Basis of Informed Requirements Engineering Management,” 1993.
- [17] IEEE, *Systems and software engineering — Life cycle processes — Requirements engineering (29148)* .
- [18] SOPHISTen and R. Kluge, “MASTER: Schablonen für alle Fälle,” 2024.
- [19] I. Gräßler and J. Pottebaum, “Generic Product Lifecycle Model: A Holistic and Adaptable Approach for Multi-Disciplinary Product–Service Systems,” *Applied Sciences*, vol. 11, no. 10, p. 4516, 2021.
- [20] Google, “Prompting guide 101. A quick-start handbook for effective prompts,”
- [21] OpenAI, “Best practices for prompt engineering with the OpenAI API,” [Online]. Available: <https://help.openai.com/en/articles/6654000-best-practices-for-prompt-engineering-with-the-openai-api>.

Framework for Model-Based Systems Engineering Circularity Assessment

Yannick Juresa*, Damun Mollahassani & Jens C. Göbel

¹RPTU Kaiserslautern-Landau, Lehrstuhl für Virtuelle Produktentwicklung, 67663 Kaiserslautern

*Corresponding author. E-Mail address: yannick.juresa@mv.rptu.de; Tel.: 0631/2052312

Abstract:

High-value circular strategies such as reuse depend on architecture decisions made before embodiment is fixed. However, reuse-relevant information is dispersed across Systems Modeling Language (SysML)/Product Lifecycle Management (PLM) artefacts, rule logic, standards and regulations, and external datasets (e.g., Lifecycle Assessment factors). This paper presents a Model-Based Systems Engineering (MBSE) framework and Artificial Intelligence (AI)-supported evidence workflow that enables traceable, reproducible assessment of reuse-oriented circularity for system structure variants at an architecture decision gate. The approach separates inputs by provenance product-specific SysML v2 variant models linked to PLM configurations, a versioned circularity rule library, and curated collections and quantitative datasets and connects them through explicit, reviewable mappings. Deterministic Key Performance Indicator (KPI) pipelines compute engineer-assessable indicators (reuse-yield proxies, non-destructive disassembly, interface standardization/interchangeability) directly from the model backbone. A Retrieval Augmented Generation (RAG) component supports evidence acquisition and semantic alignment for externally dependent KPIs (e.g., avoided embodied carbon through reuse) and for justification from standards, while maintaining human accountability. The framework is instantiated on two alternative 3D printer printhead architectures. Results highlight the integration–circularity trade-off.

Keywords:

Model-Based Systems Engineering, Product Development, Circular Economy, Artificial Intelligence

1 Introduction

The circular economy (CE) seeks to preserve material and functional value by slowing, closing, and narrowing resource flows across the product lifecycle, operationalized for example by the 9R circular strategies [1,2]. These are largely determined during development, circularity must be embedded in early architecture decisions rather than added after key choices are fixed [3,4]. Within the circular strategies, function-retaining approaches (reuse, repair, refurbish, remanufacture, reconfiguration) typically deliver higher ecological and economic benefits than recycling-centered strategies [2,5,6]. For information and communication technology, reuse and refurbishment can substantially reduce greenhouse gas emissions compared to new production or recycling [7]. Reuse-enabling design factors include modular disassembly, standardized interfaces, accessible and reversible connections, and separating wear-prone from durable components [8]. This paper addresses a practical decision point in product development: once several product structure variants exist, engineers must classify and compare them regarding end-of-life (EoL) reuse. Classic criteria such as time, cost, and quality remain necessary but are insufficient to judge reuse feasibility. We assume a defined product-structure baseline and partially specified alternatives for selected subsystems, modules, or parts, described in SysML v2 models and linked

Product Lifecycle Management (PLM) configurations. Where suitable Key Performance Indicators (KPI) definitions and rule logic exist, variants can already be compared deterministically using model-internal KPIs. Robust reuse-oriented decisions, however, require external evidence such as Lifecycle Assessment (LCA) parameters (e.g., embodied-carbon factors), disassembly knowledge, and constraints from standards and regulations. Manual retrieval and alignment of heterogeneous sources is time-consuming, error-prone, and often undermines traceability from results back to specific model elements and assumptions. Accordingly, the role of Artificial Intelligence (AI) in this work is deliberately bounded within a human-in-the-loop paradigm: AI augments evidence acquisition and semantic alignment where external sources are unstructured or inconsistent, while engineers remain accountable for interpretation and final decisions. Retrieval-augmented workflows can retrieve relevant passages from standards and curated corpora, propose mappings to external taxonomies (e.g., aligning materials and processes to LCA datasets), and store evidence with auditable provenance. Crucially, KPI computation itself remains deterministic and reproducible logic operating on the Systems Modeling Language (SysML)/PLM backbone and explicitly linked evidence. Building on this separation, we distinguish two KPI classes: (1) engineer-assessable KPIs computed directly from SysML/PLM artefacts and rule libraries, and (2) evidence-augmented KPIs that depend on external quantitative or normative evidence and therefore benefit from AI-supported retrieval and alignment. Based on this separation, we propose a model-based framework and platform concept that evaluates reuse-oriented circularity of product structure variants using inputs separated by origin: SysML v2 variant models linked to PLM, a reusable circularity rule library, and external datasets plus curated standards/regulatory corpora. The work addresses two research questions:

- How can SysML v2 and PLM-linked product structure variant models be used as a structured backbone to compute reuse-oriented circularity KPIs deterministically and traceably at an early architecture decision gate?
- How can a retrieval-augmented evidence workflow integrate standards/regulations and external datasets into the KPI pipeline via reviewable mappings and provenance so that evidence-dependent KPIs become available?

Following Design Research Methodology (DRM) [9], Section 2 reviews related work on circularity assessment, Model-Based Systems Engineering (MBSE), and AI-supported evidence retrieval (descriptive study I). Section 3 presents the proposed provenance-separated framework and hybrid KPI workflow (prescriptive study). Section 4 instantiates and evaluates the approach on a 3D printer print-head variant case study (descriptive study II). Section 5 concludes and outlines limitations and directions for future research.

2 Background and Related Work

Research on circularity assessment provides a broad spectrum of indicators and methods that aim to quantify progress toward CE objectives at product and system level [10]. Many approaches operationalize circularity through aggregated indicators that focus on material loops, recycled content, and EoL recovery rates [11]. These metrics are attractive for reporting and benchmarking, but they often abstract from product structure, joining choices, and the practical feasibility of recovering functional units. For reuse-oriented decisions, the literature therefore increasingly complements material flow indicators with design-oriented assessment approaches, including design for disassembly, design for repair, modularity evaluation, and repairability or reusability scoring [12]. Such methods translate circular intent into engineering relevant proxies, for example accessibility, reversibility of connections, interface standardization, interchangeability, and separation of wear prone and durable elements.

MBSE has been established as an approach to manage complexity by representing requirements, functions, and architectures in structured models that support consistency and traceability [13]. This

makes MBSE a candidate for embedding lifecycle considerations early, because design decisions can be related to requirements and to architectural elements before detailed embodiment is fixed [14]. Prior work has explored how sustainability information can be linked to system models, including approaches that connect SysML artefacts to LCA tools or to sustainability attributes for design tradeoffs [15–17]. In practice, however, LCA depends on data that is distributed across repositories. PLM systems provide configuration context, product structure, identifiers, revision status, and domain files, while system models capture intent and architectural relations. Variant evaluation therefore requires stable linking between model elements and PLM configuration semantics, otherwise KPI results cannot be traced to a specific assessed alternative. SysML v2 is relevant in this context because it strengthens machine readable modeling and traceability, which supports systematic querying and reproducible evaluation on a shared backbone [18]. A remaining gap is the lack of methods that explicitly operationalize variant assessment oriented toward reuse at a defined gate while maintaining traceability from KPIs to SysML and PLM artifacts and from there to the lifecycle evidence required for implementation [15,19].

Recent work on AI in engineering has demonstrated value in information intensive tasks such as requirements analysis, document classification, and retrieval of relevant content from large corpora [20]. These capabilities are particularly relevant for circularity assessment because reuse feasibility often depends on external evidence, including standards and regulatory texts, disassembly and service knowledge, and environmental datasets such as LCA factors [21]. Large language models can improve accessibility of such sources, but they also introduce risks related to unsupported statements and insufficient provenance [22]. Retrieval Augmented Generation (RAG) addresses this by constraining generation to retrieved evidence items and by enabling source grounded explanations [23]. In engineering settings, this supports an accountable workflow in which external knowledge is not used as implicit background, but is captured as explicit evidence objects linked to the assessed artefacts. A central open issue is integration. Many AI-based approaches remain document centric and detached from deterministic KPI computation, while many model based KPI approaches do not scale evidence acquisition and semantic alignment.

The approach of Lindemann et al. [15] develops a SysML-based LCA modeling framework (including an LCA-specific SysML profile as a Motego extension) and a tool interface to openLCA in order to automatically generate a product-specific LCI model in the LCA tool from the SysML system model, execute it, and write the results back to the system model. It is not yet sufficient because SysML demonstrably needs to be extended to life-cycle-wide modeling, SysML v1 is used due to the lack of finalization and tool support for SysML v2, and further research is needed for robust scalability and database/tool integration, which clearly justifies follow-up research for consistent SysML v2-based and circularity-oriented decision support. The approach of Inkermann [16] identifies the fundamental potential of MBSE-LCA integration by systematically analyzing uncertainties in the first two LCA phases and challenges in handling variants and different use cases, and deriving an initial concept for methodological and data-related integration from this. However, it is not sufficient because it is explicitly formulated as an initial concept with a focus on the early LCA phases and itself derives further fields of research, so that follow-up research remains necessary in order to develop an implemented, validated solution suitable for practical variant and strategy decisions (including circular strategies). The approach in Ghanjaoui et al. [24] presents an MBSE methodology that models production system architectures and uses sustainability data from executed processes and resources for model-based validation of requirements and optimization of sustainable concept designs. However, it is explicitly limited to target definition and inventory analysis in accordance with LCA and calls for the addition of impact assessment and interpretation as well as trade-off analyses/optimizations, which is why further research is necessary to enable a consistent, comparative assessment (especially for early variant decisions and circularity references). The approach of Paetzold-Byhain [25] proposes a concept for

assessing ecological sustainability and circularity in development that anchors LCA at the micro level and material flow analysis plus CE indicators at the macro level in information models of a sustainability-oriented PLM strategy (including derivation of requirements and implementation of an information model in SysML). At the same time, the authors identify uncertain or incomplete information as a key challenge and consider the approach to still be in the development stage, with the question of industrial scaling remaining open. This means that further research is needed to develop a robust, data- and tool-supported variant assessment that is reliable in practice.

However, these contributions do not yet operationalize reuse-focused, gate-level variant assessment on a SysML v2/PLM backbone with explicit evidence objects and reviewable mappings that preserve traceability from external sources to model elements and KPI results. These aspects are addressed and solved in Section 3.

3 Model-Based Framework for Circularity Assessment

Circular and AI-supported assessment is an essential component of this model-based approach. Figure 1 summarizes the four input classes (SysML/PLM artefacts, circularity rule/KPI library, external datasets, and standards/regulations) and the two evaluation spaces: deterministic KPI computation on the SysML/PLM backbone and a bounded RAG workflow for retrieving and aligning external evidence with traceable provenance.

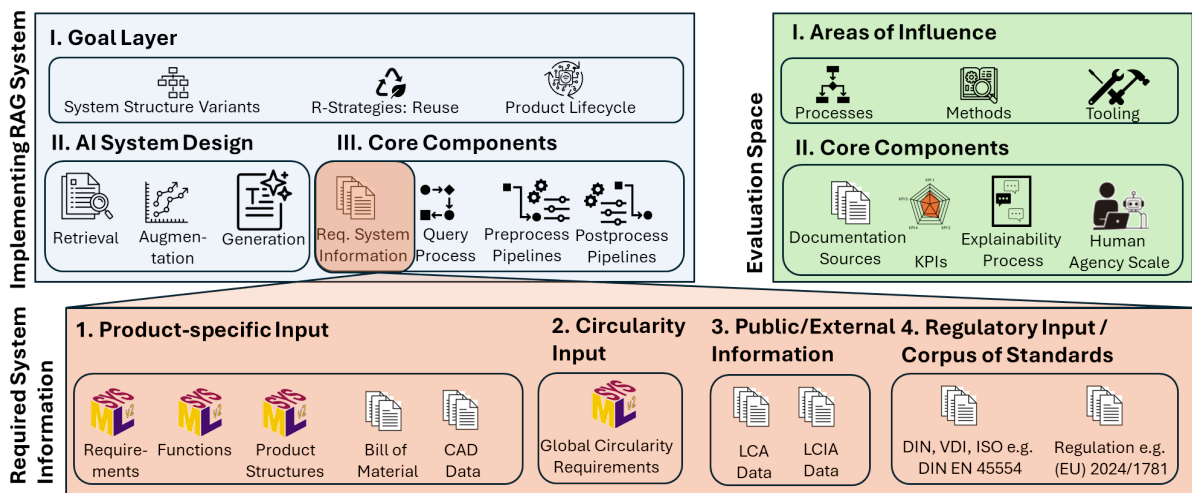


Figure 1: Framework components at the assessment gate with provenance-separated inputs and bounded AI-supported evidence retrieval

The assessment is applied at an early architecture decision gate where alternative variants exist for selected subsystems, modules, or parts, while the product structure is still adjustable. Variants may differ in module partitioning, interface concepts, joining principles, material/process selections, or service accessibility. Because reuse-relevant properties are largely determined by structure choices (modularity, separability, interface design) and are difficult to correct later, reuse criteria must be evaluated at the same gate where structure trade-offs are negotiated. Classic criteria such as time, cost, quality, risk, and performance remain necessary but are insufficient to judge reuse feasibility at EoL. In this work, reuse at EoL denotes recovery and second-life use of a product, subsystem, or module as a functional unit. It is treated as a scenario that must be classified and justified per variant. Reuse feasibility depends on non-destructive separation and accessibility, inspectability/qualification for a second life, and interface compatibility for reintegration, including relevant constraints from standards, safety, and traceability requirements.

3.1 Information landscape and SysML v2 as structured backbone

The gate is characterized by a distributed information situation. Relevant information exists, but it is split across engineering models, PLM data, and external sources. The scenario assumes that SysML v2 artefacts are available for requirements, functional structures, and physical product structure descriptions, and that the variant set is represented in a consistent way and that SysML v2 artefacts are available along the product creation process for requirements, functions, and logical/physical product structure representations, as illustrated in Figure 2. In parallel, PLM systems contain configuration and variant identifiers, product structure information such as bills of materials, version and change status, and additional domain files such as CAD or material master data. Even if SysML models are managed in the same organizational environment, configuration semantics and identifiers are often represented differently in PLM than in system models. This split becomes critical when the assessment must connect a concrete variant in the product structure model to a concrete configuration in PLM, and then to the evidence required for reuse evaluation. For reuse focused assessment, the development team also needs information that is typically not contained in SysML or PLM alone. Quantitative environmental parameters such as embodied carbon factors are often taken from LCA datasets. Knowledge about disassembly sequences, typical failure modes, and repair practices may exist in service documentation, curated internal repositories, or industry databases. Normative constraints come from standards and regulatory documents. Examples include, climatq.io, IDIS, Probas and standards such as DIN EN 45554, DIN 8580, and VDI 2243.

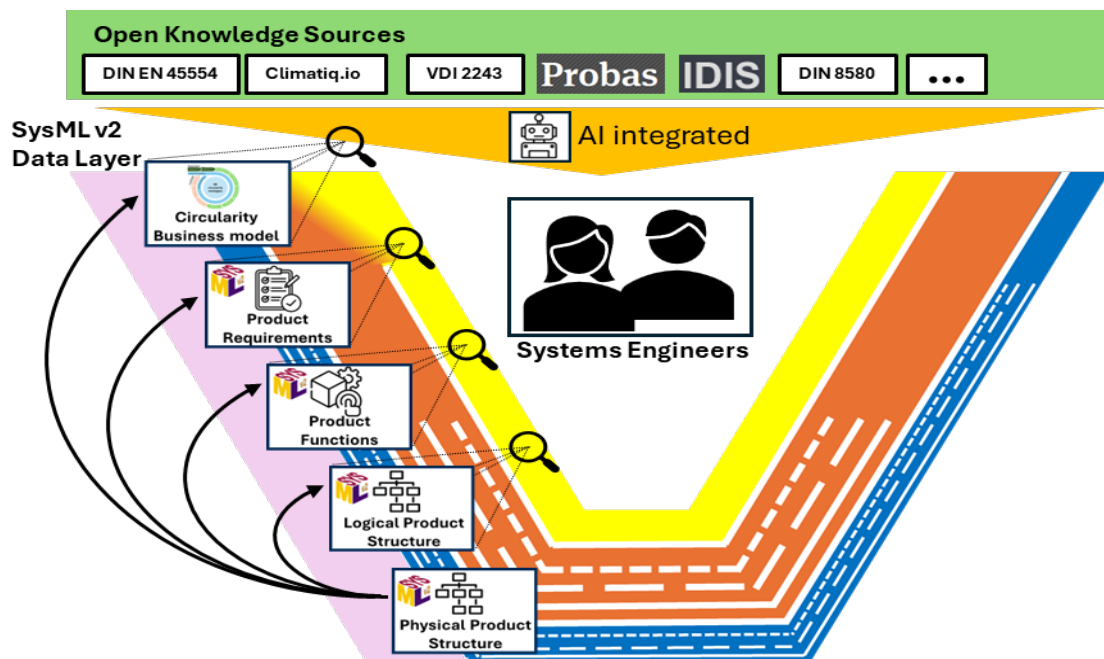


Figure 2: SysML data layer along product creation process, external knowledge sources and VDI 2206 based on [26]

Figure 2 positions the approach in the product creation process based on VDI 2206. SysML v2 provides a continuous, machine-readable data layer across circularity/business context, requirements, functions, and logical/physical product structures, while the assessment gate is located on the left branch where architectural and product structure decisions are still flexible. The AI-supported evidence layer connects this SysML/PLM backbone to open knowledge sources (e.g., standards, disassembly databases, LCA datasets) and returns traceable evidence packages, KPI results, and rationales to systems engineers. These outputs can be written back as properties and linked evidence objects to support downstream design, verification, and documentation activities within the V-model. The critical boundary condition is therefore not the absence of information, but the heterogeneity of formats, terminologies, and levels

of detail. Manual collection and alignment of these sources is slow and error prone, and it often breaks traceability because the link from a KPI result back to the exact model element, dataset entry, or normative passage is not preserved in a systematic way. This is particularly problematic at the considered gate, the decision outcome must remain explainable and auditable even as variants evolve. SysML v2 is used as the backbone. The assessment requires structured and connected engineering information. At the gate, reuse oriented KPIs and decision arguments should be computable or at least explainable from a consistent representation of requirements, functions, and product structure. SysML v2 supports this by enabling model elements to be explicitly related and queried, and by strengthening machine readability and traceability in a standardized form. These properties are essential for two reasons. First, they allow deterministic evaluation of variant properties that can be derived directly from the model, for example indicators related to the structure of the product, the definition of interfaces, or the presence of service relevant access concepts. Second, they provide stable reference points for linking external evidence to concrete engineering artefacts. Without a structured backbone, external evidence remains a collection of documents and numbers that is difficult to connect to product structure alternatives in a reproducible way.

3.2 Hybrid assessment workflow

At this gate, engineers must select or prioritize product structure variants while treating end-of-life reuse as a development parameter. This requires KPIs and decision rationales that link SysML/PLM model properties with external evidence. The approach therefore uses two interacting evaluation layers, connected via explicit links to the SysML v2 and PLM backbone. A deterministic, model-centered layer computes auditable KPIs from SysML and PLM artefacts through explicit, versioned rule logic (e.g., disassembly-effort proxies from connection/accessibility attributes, interface standardization and interchangeability indicators, and structure-based reuse-yield proxies). This layer enables fast, reproducible screening across variants and highlights design levers that can still be influenced at the gate.

A second, evidence-dependent layer addresses KPIs and constraints that require external quantitative or normative inputs (e.g., LCA factors, standards, and regulations). Here, AI is used deliberately in a bounded role to support evidence acquisition and semantic alignment for unstructured, heterogeneous sources, without generating variants or replacing engineering judgement, so that evidence-dependent indicators can be handled at scale without losing auditability. RAG operationalizes this evidence layer by deriving a structured query context from the model (materials, joining types, interfaces, requirements, access assumptions, target reuse scenario), retrieving relevant passages from indexed standards, disassembly/service knowledge, and LCA datasets, and producing a structured evidence package. The package includes concise evidence summaries, explicit source identifiers, and proposed semantic mappings (e.g., canonicalizing material/process names to LCA taxonomies with confidence and alternatives) as well as candidate constraints translated into reviewable rule statements for the rule library. Outputs are stored as evidence objects with provenance and explicit links to SysML elements, PLM configuration IDs, and KPI computations, enabling systematic re-runs when assumptions or variants change. Circular KPIs thus complement cost, time, and performance by making reuse feasibility, constraints, and expected benefit comparable across variants. They are split into engineer-assessable KPIs computed from model content and evidence-augmented KPIs whose outputs include the referenced evidence. For this to work, variants must be modeled consistently, module boundaries/interfaces/connection principles must be explicit, SysML–PLM links must remain stable, and gate outputs should include feasibility classes with documented blockers, maintaining end-to-end traceability across SysML, PLM, and external sources.

4 Use Case: 3D Printer Print Head

This section instantiates the proposed framework on a fused deposition modeling print head to demonstrate how reuse-oriented circularity can be treated as a decision parameter once product structures and development alternatives are available. The print head is suitable because it combines mechanical, thermal, and electrical functions in a compact volume, which typically creates an architectural tension between integration for performance and manufacturability and separability for reuse. The goal of the instantiation is not to model a complete end of use process in detail. Instead, it shows how variant comparison can be executed with a deterministic baseline and how additional external evidence can be incorporated in a controlled and traceable way when needed. The instantiation reflects a defined development gate. The product structure is considered established, and relevant variants exist for selected subsystems and parts. These alternatives are anchored in two backbone systems. A SysML v2 model provides a machine-readable product structure representation that includes the part decomposition, interfaces, and joining relations, as well as stable identifiers for traceability. The PLM system provides the authoritative Bill of Materials (BOM) and the configuration context that defines which parts belong to which alternative and which revision state is assessed. Each assessment run is therefore bound to a specific SysML model revision and a specific PLM configuration identifier to ensure reproducibility and audit readiness.

Two product structure variants are assessed. Variant A represents a strongly integrated print head architecture with compact housing and a tightly integrated thermal core. Functions that support extrusion and sensing are combined into dense assemblies, and key subassemblies are fixed using joining choices that are efficient for manufacturing and integration but tend to limit nondestructive separation. Electrical routing and sensor integration are closely coupled to the thermal core, and the interface concept is more variant specific, which reduces interchangeability across configurations or future generations. Variant B represents a more modular architecture that separates wear prone components from durable components and introduces clearer module boundaries that can act as exchange units. The nozzle and thermal core follow a cartridge-oriented concept with explicit mechanical interfaces, electrical interfaces are more standardized, and joining choices are selected to support separability where reuse is intended. For both variants, a mapping between SysML element identifiers and PLM part identifiers is established for the assessment run. Where a one-to-one mapping is not available at part level, the mapping is maintained at module level and refined as the engineering data matures. This mapping is treated as an explicit artefact that is versioned with the assessed configuration state. The assessment also records a completeness profile, because attribute availability is typically heterogeneous at this gate. The completeness profile indicates which parts have masses and material assignments, which joins are classified, and which interfaces are typed. This prevents false precision and turns missingness into an actionable engineering item rather than an implicit assumption.

Figure 3 depicts the assessment canvas used for the instantiation. For each product structure variant, an input package is assembled from among other things (i) the SysML v2 product-structure code (as the machine-readable backbone), (ii) the corresponding PLM configuration and BOM identifiers (as the authoritative variant context), (iii) CAD references for visual inspection and plausibility checks, and (iv) optional external evidence items such as linked LCA dataset entries. The output package consists of a KPI dashboard and a versioned explainability documentation bundle. The explainability bundle records intermediate features, rule firings, evidence identifiers, and mapping decisions so that each KPI value can be traced back to a specific SysML/PLM state and to the referenced evidence.

The instantiation is executed in two stages that correspond to the two lanes of the framework. In the baseline stage, engineers can compare variants without any AI component. The system extracts the variant specific product structure subgraph from SysML v2 and aligns it with the corresponding PLM

emission factors, while scenario assumptions are documented explicitly. In parallel, the same retrieval mechanism can index standards and regulations to capture clause-level evidence for reuse-relevant constraints. Retrieved clauses are stored with edition metadata and linked to affected joints, interfaces, or modules so that explanations remain auditable.

Overall, the use case shows that a deterministic baseline enables reproducible screening across variants and highlights reuse-relevant levers without any AI component. Evidence augmentation extends the workflow to externally dependent KPIs (e.g., avoided embodied carbon) while maintaining auditability by storing evidence links, dataset versions, and clause identifiers as explicit, reviewable artefacts bound to the assessed SysML/PLM state. Limitations mainly reflect early-phase data gaps and scenario sensitivity, which motivates completeness reporting and governance of evidence corpora and mappings.

5 Conclusion and Future Work

This paper presented a SysML v2–anchored MBSE framework and hybrid KPI workflow for reuse-oriented circularity assessment of system product structure variants at an architecture decision gate. The core contribution is a provenance-separated integration of (i) SysML v2/PLM variant representations, (ii) a versioned circularity rule and KPI library, and (iii) external quantitative and normative evidence sources (LCA datasets and standards/regulations), connected through explicit, reviewable mapping artefacts. Deterministic KPI pipelines compute engineer-assessable indicators on the SysML/PLM backbone, while a bounded retrieval-augmented evidence workflow supports retrieval and semantic alignment of external evidence and stores it as explicit evidence objects linked to the assessed configuration. The print-head instantiation demonstrated traceable KPI dashboards and explainability documentation and made the integration–circularity trade-off explicit. The more modular, reversibly connected variant improved the reuse rate, non-destructive disassembly, and interchangeability/standardization of interfaces, and achieved greater potential for avoiding CO₂ emissions through reuse, while the integrated variant created barriers to accessibility and separation.

Limitations are dominated by typical early-phase conditions, including incomplete masses and material specifications, sensitivity to scenario assumptions for reuse burdens, and the need for evidence and mapping rules. Future work should therefore (1) validate the approach on additional product classes and larger variant spaces, including controlled benchmarking of retrieval quality and agreement with expert assessments, (2) calibrate KPI definitions and intermediate features through sensitivity and uncertainty analyses to avoid false precision under data gaps, and (3) strengthen evidence and mappings, e.g. dataset versioning and change impact analysis. For the AI component, systematic reliability measures are required, including evidence-coverage metrics, contradiction and out-of-scope detection, and regression tests across corpus, embedding, and model updates. From an adoption perspective, deeper integration into MBSE toolchains and V-model processes is needed, particularly to write results back into SysML as properties, rationales, and compliance-relevant evidence links and to align outputs with evolving ecodesign and Digital Product Passport expectations.

References

- [1] N. M. P. Bocken, I. de Pauw, C. Bakker and B. van der Grinten, “Product design and business model strategies for a circular economy,” *Journal of Industrial and Production Engineering*, vol. 33, no. 5, pp. 308-320, 2016. DOI: <https://doi.org/10.1080/21681015.2016.1172124>.
- [2] J. Kirchherr, D. Reike and M. Hekkert, “Conceptualizing the circular economy: An analysis of 114 definitions,” *Resources, Conservation and Recycling*, vol. 127, pp. 221-232, 2017. DOI: <https://doi.org/10.1016/j.resconrec.2017.09.005>.
- [3] B. Du, J. R. Bryson and A. Qamar, “Aspiring towards automotive circularity: A critical review and research agenda,” *Journal of environmental management*, vol. 380, p. 125150, 2025. DOI: <https://doi.org/10.1016/j.jenvman.2025.125150>.

- [4] M. Stölzle, D. Roth and M. Kreimeyer, "Impact of the implementation of Design for Circular Economy features on the product development process," in *DS 125: Proceedings of the 34th Symposium Design for X (DFX2023)*: The Design Society, 14 and 2023, pp. 135–142.
- [5] J. Potting, M. Hekkert, E. Worrell and A. Hanemaaijer, "Circular Economy: Measuring Innovation in the Product Chain," [Online]. Available: <https://www.pbl.nl/sites/default/files/downloads/pbl-2016-circular-economy-measuring-innovation-in-product-chains-2544.pdf>. [Accessed 27 February 2025].
- [6] Y. Juresa, D. Mollahassani and J. C. Goebel, "Addressing Circularity Strategies by Reconfiguring Smart Products during Their Lifecycle," *Tehnički glasnik*, vol. 18, no. 3, pp. 452-460, 2024. DOI: <https://doi.org/10.31803/tg-20240516125344>.
- [7] P. Rittershaus and D. Maga, *Treibhausgaseinsparungen durch Wiedernutzung ausgewählter IKT Geräte*, 2025.
- [8] K. Schischke, M. Proske, N. F. Nissen and M. Schneider-Ramelow, "Impact of modularity as a circular design strategy on materials use for smart mobile devices," *MRS Energy & Sustainability*, vol. 6, no. 1, 2019. DOI: <https://doi.org/10.1557/mre.2019.17>.
- [9] L. T. M. Blessing and A. Chakrabarti, *DRM, a design research methodology* Dordrecht, Heidelberg: Springer, 2009.
- [10] A. Jerome, H. Helander, M. Ljunggren and M. Janssen, "Mapping and testing circular economy product-level indicators: A critical review," *Resources, Conservation and Recycling*, vol. 178, p. 106080, 2022. DOI: <https://doi.org/10.1016/j.resconrec.2021.106080>.
- [11] S. Kara, M. Hauschild, J. Sutherland and T. McAloone, "Closed-loop systems to circular economy: A pathway to environmental sustainability?," *CIRP Annals*, vol. 71, no. 2, pp. 505-528, 2022. DOI: <https://doi.org/10.1016/j.cirp.2022.05.008>.
- [12] European Commission. Joint Research Centre., *Analysis and development of a scoring system for repair and upgrade of products: final report* Publications Office, 2019.
- [13] D. D. Walden, G. J. Roedler, K. Forsberg, R. D. Hamelin and T. M. Shortell, eds., *Systems engineering handbook. A guide for system life cycle processes and activities : INCOSE-TP-2003-002-04*, 2015, 4. edition (New York: Wiley, 2015).
- [14] M. Eigner, W. Koch and C. Muggeo, *Modellbasierter Entwicklungsprozess cybertronischer Systeme* Berlin, Heidelberg: Springer Berlin Heidelberg, 2017.
- [15] M. Lindemann, G. Jacobs, J. Dreier, S. Wischmann, G. Höpfner and J. Berges, "Facilitating LCA During Product Development by Automated Inventory Model Generation Using SysML System Models," *Sustainability*, vol. 17, no. 11, p. 4783, 2025. DOI: <https://doi.org/10.3390/su17114783>.
- [16] D. Inkermann, "Potentials of integrating MBSE and LCA to handle uncertainties and variants in early design stages," in *DS 119: Proceedings of the 33rd Symposium Design for X (DFX2022)*: The Design Society, 22 and 2022, p. 10.
- [17] Z. Lipšinić and N. Pavković, *Integrating life cycle assessment in model-based systems engineering*, 2023.
- [18] Object Management Group, "OMG Systems Modeling Language (SysML). Version 2.0," [Online]. Available: <https://www.omg.org/spec/SysML>.
- [19] J. Cederbladh, A. Cicchetti and J. Suryadevara, "Early Validation and Verification of System Behaviour in Model-based Systems Engineering: A Systematic Literature Review," *ACM Transactions on Software Engineering and Methodology*, vol. 33, no. 3, pp. 1-67, 2024. DOI: <https://doi.org/10.1145/3631976>.
- [20] H. Cheng, J. H. Husen, Y. Lu, T. Racharak, N. Yoshioka, N. Ubayashi and H. Washizaki, *Generative AI for Requirements Engineering: A Systematic Literature Review*, 2024.
- [21] C. Cole, A. Hajikhani, E. Hylkilä, E. Paronen and H. Pihkola, "Towards AI-augmented sustainability assessments: integrating large language models in the case of product social life cycle assessment," *The International Journal of Life Cycle Assessment*, 2025. DOI: <https://doi.org/10.1007/s11367-025-02508-w>.
- [22] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin and T. Liu, "A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions," *ACM Transactions on Information Systems*, vol. 43, no. 2, pp. 1-55, 2025. DOI: <https://doi.org/10.1145/3703155>.
- [23] C. Sharma, *Retrieval-Augmented Generation: A Comprehensive Survey of Architectures, Enhancements, and Robustness* Frontiers, 2025.
- [24] Y. Ghanjaoui, P. Satwan, T. Hanl and V. Srinivasan, "A Model-Based Approach for Evaluating and Validating the Sustainability of Production Systems," in *ONERA-DLR Aerospace Symposium (ODAS)*, 2023.
- [25] K. P.-B. Paetzold-Byhain, "A concept for assessing environmental sustainability and circularity in the development phase on the basis of a sustainable PLM strategy," in *DS 133: Proceedings of the 35th Symposium Design for X (DFX2024)*: The Design Society, 12 and 2024, pp. 173–181.
- [26] VDI, *Development of mechatronic and cyber-physical systems*, 2021-11 (2206:2021-11) (Verband deutscher Ingenieure, 2021).

Automatic Generation of technical Drawings using Model Context Protocol

Leonhard Kunz^{1*}, Mario Klostermeier¹, Kokulan Thanabalan², Tatjana Legler¹, Martin Ruskowski¹

¹University of Kaiserslautern-Landau (RPTU), Gottlieb-Daimler-Straße 42, 67663 Kaiserslautern, Germany

²Trier University, Campus II, Behringstraße 21, 54296 Trier, Germany

*Corresponding author. E-Mail address: leonhard.kunz@rptu.de

Abstract: With the current omnipresence of Large Language Models (LLM) there is a high interest in how to apply them also in Computer Aided Design (CAD). This paper presents an agent-based approach for the automated generation of technical drawings from existing 3D CAD models using direct integration via the Model Context Protocol (MCP). Instead of relying on intermediary representations or scripting languages, the proposed system connects a multimodal large language model directly to the native CAD environment via an integrated MCP server. The approach supports iterative refinement through visual feedback and native geometric constraints, allowing the agent to select, place, and revise drawing views in a closed loop. A modular FreeCAD addon implementing this concept is introduced and released as a testbed for agentic CAD research. The system is evaluated by comparing automatically generated drawings with human-created references using defined metrics for view selection and layout quality. Experiments with two multimodal language models demonstrate that LLM-based agentic drawing generation still has difficulties matching human drafting decisions, due to current limitations in visual understanding. The results highlight the potential of direct CAD-agent integration and motivate further specialization of smaller models for technical drawing tasks.

Keywords:

Large Language Model, Agent, Computer Aided Design, Model Context Protocol

1 Introduction

Although 3D modelling and Model-Based Definition is widely adopted in Computer Aided Design (CAD), technical drawings remain an important medium for communication between product design and manufacturing [1]. Creating them, however, requires a significant manual effort [2]. They must be both precise and at the same time comprehensive for humans, making a full rule-based, algorithmic automation of the process of deriving them difficult [3]. Machine learning models, however, have shown to be capable of well approximating human preferences in decision-making [4], and specialized models are becoming a common tool to automate drawing generation. Particularly, the recent advances in Large Language Models (LLMs) have sparked the idea to automate complex interactions with script-based CAD software [5]. Multimodal models, integrating additional modalities like images or audio, are driving the discussion around agentic Artificial Intelligence (AI) which is also expanding to the domain of product development [6]. This discussion is also driven by the introduction of standards like the Model Context Protocol (MCP) for the interaction of agentic AI with external applications for grounding or action [7].

This paper contributes here to the idea of agentic CAD via direct interaction between the agent and the CAD API. The approach uses no intermediary representation, that needs a resolver for the CAD kernel but interacts directly via an MCP server with the functions of FreeCADs TechDraw CAD API [8]. The developed FreeCAD addon is published alongside this paper and can serve further as a testbed and

modularly expanded for testing and the development of agentic AI in CAD¹. It can also be used to generate synthetic technical drawings at scale from preexisting 3D CAD models. A solution to a notable shortage in public datasets [9].

The presented addon focusses here particularly on the automated generation of technical drawings from preexisting 3D models and investigates few key decisions during this process. The results are compared to human decisions in the same situation to evaluate, if the model's decisions are comparable to human preferences. In a second step, with the addon we test also ability to reflect on the decisions made and iteratively improve the generated technical drawing. This can be summarized in the following research questions:

1. Can the quality of technical drawings, created with a multimodal general purpose AI agent and an MCP CAD pipeline, match the requirements and quality of manually created drawings?
2. Can the model self-improve the result of the automatic generation pipeline through the iterative execution of commands in the CAD environment and the evaluations of the results?

To answer the questions, the paper is structured as follows: Section 2 reviews related work. Section 3 describes the implementation of the addon for FreeCAD. Section 4 defines tests for the framework and section 5 presents and discusses the results of these tests involving 2 selected LLMs. Section 6 discloses the limits of the study and section 7 concludes the paper.

2 Related Work

Technical drawings are in many aspects based on common standards (e.g. [10]) which unify the representation of tolerances like the Geometric Dimensioning and Tolerancing (GD&T) symbols or the indication of dimensions. However, these standards cannot provide complete guidelines for the context sensitive placement of the symbols within a drawing and along the geometry [11]. Additionally, they are also often complemented by free text annotations describing e.g., manufacturing or inspection instructions [12]. Modern CAD environments usually already provide at least semi-automated solutions to the derivation of drawings [13]. These are partially still constrained to predefined parameters, e.g. a fixed selection of views or a context-insensitive dimensioning [13]. Some developers of CAD systems, however, have already introduced features for context-sensitive generation of complete drawings [14, 15]. Third party plugins provide an alternative for these built in options [16]. Proprietary AI tools in CAD are confined to vendor-specific ecosystems and typically offer isolated functions like stand-alone drawing generation. Proprietary formats and restricted interoperability constitute core structural barriers, as shown by Berger et al. [17].

Neutral agent-based design frameworks instead coordinate the broader design workflow, integrate heterogeneous inputs and iteratively contextualize model and drawing creation [18]. This enables a more faithful capture of design intent and supports end-to-end automation. However, there is some discussion around the integration interface between the agent and the CAD environment. While Xu et al. try to directly generate a valid boundary representation CAD model, these approaches naturally have limitations regarding guarantees for the integrity of the generated model [19]. Therefore, more commonly, approaches try to integrate into CAD environments and leverage the rigid modeling constraints and export validity guarantees [20].

Many approaches rely here additionally on an abstract or symbolic representation (e.g., a domain-specific language, command sequence, or plan) which serve as abstract representations of the

¹ <https://github.com/XDP-Opt/TechDraw-MCP-Agent>

planned actions. These are then resolved into concrete actions using the CAD Kernel. Daareyni et al. propose e.g. a solution generating OpenSCAD scripts, that can then be rendered for visual confirmation [21]. Other authors are trying to define their own domain specific language or parametric scripting language [22, 23]. This serves as an intermediate representation between a task description and the CAD modeling environment. The implementation itself can be considered open-loop, where the geometry is generated in a single shot [19, 23] or closed-loop, enabling revision and improvement steps [21, 22]. The introduction of an intermediary representation of geometry, however, complicates the refinements steps as the intermediary representation forces refinement to operate across two layers. The system must first inspect the CAD environment to identify defects or misaligned features. It must then project those findings back into the abstract representation and modify that representation without direct geometric feedback. This multi-layer loop adds fragility and limits corrective precision. In contrast, approaches that bind the agent directly to the CAD environment’s API allow direct refinements to operate on native geometric and topological structures. While conceptualizing requires a certain abstract representation for planning the incremental modeling steps, the derivation of a technical drawing is already constrained by the existing 3D model. The challenge is to design an agent-based CAD architecture that remains closely integrated with the CAD environment while supporting efficient concept development [24].

3 Methodology

The approach presented in this paper ties the agent directly into the CAD environment via a custom MCP interface. This interface wraps primarily select functions of FreeCADs TechDraw workbench into simple tool functions. In our approach, we focus particularly on the implementation of these tools as it is an often-overlooked part of product design in literature on AI agents. Yet it is often still crucial for handover between design and manufacturing [24]. It also requires visual revision to confirm human comprehensibility and is therefore a prime showcase for the advantages the multimodal data processing capabilities of multimodal transformer models at the core of the agent.

Additional to the CAD API, the agent has access to local file storage where we can externalize memory capabilities or store abstract concept representations. The agent is encouraged via its system prompt to write checklists and notes to this storage and reload them for subsequent runs. Additionally, files can be manually stored in the storage to give the agent read-only access to ground the agent with preformulated knowledge. With these capabilities, the agent’s architecture, as depicted in Fig. 1, represents a very simple form of the intelligent CAD 2.0 concept postulated by Zou et al. [25].

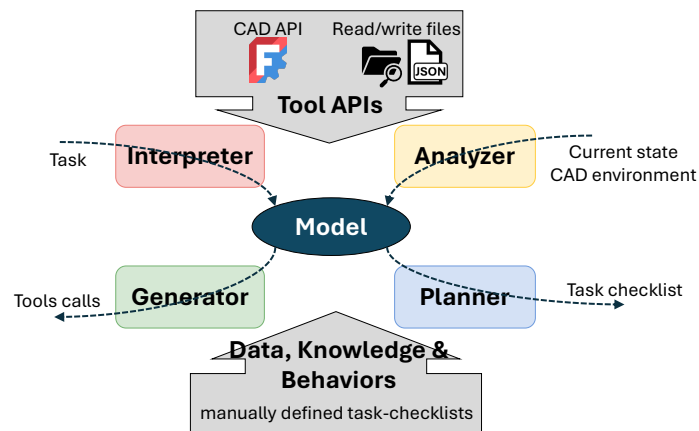


Figure 1: Architecture of the agent with its different modes analogous to Zou et al. [25].

The approach is modular and can be expanded by different MCP servers offering other CAD toolsets or other external APIs, e.g. to simulation tools, data storage, or management software. The agent can switch

between different modes (Interpreter, Analyzer, Generator, Planner) dynamically and solve tasks theoretically without a predefined processing sequence. To streamline efficiency however, predefined behaviors and approaches can be stored in the form of checklists written in natural language and serialized in markdown files. The foundational behavior of the agent can additionally be steered by a manually crafted system prompt. Fig. 2 exemplifies the implementation of such steered workflow with shortened prompts for better comprehensibility. The described behavior is non-linear and contains constant reassessment and revision of the current state in the CAD environment in reference to the given task. Therefore, it enables an iterative improvement towards fulfilling the task.

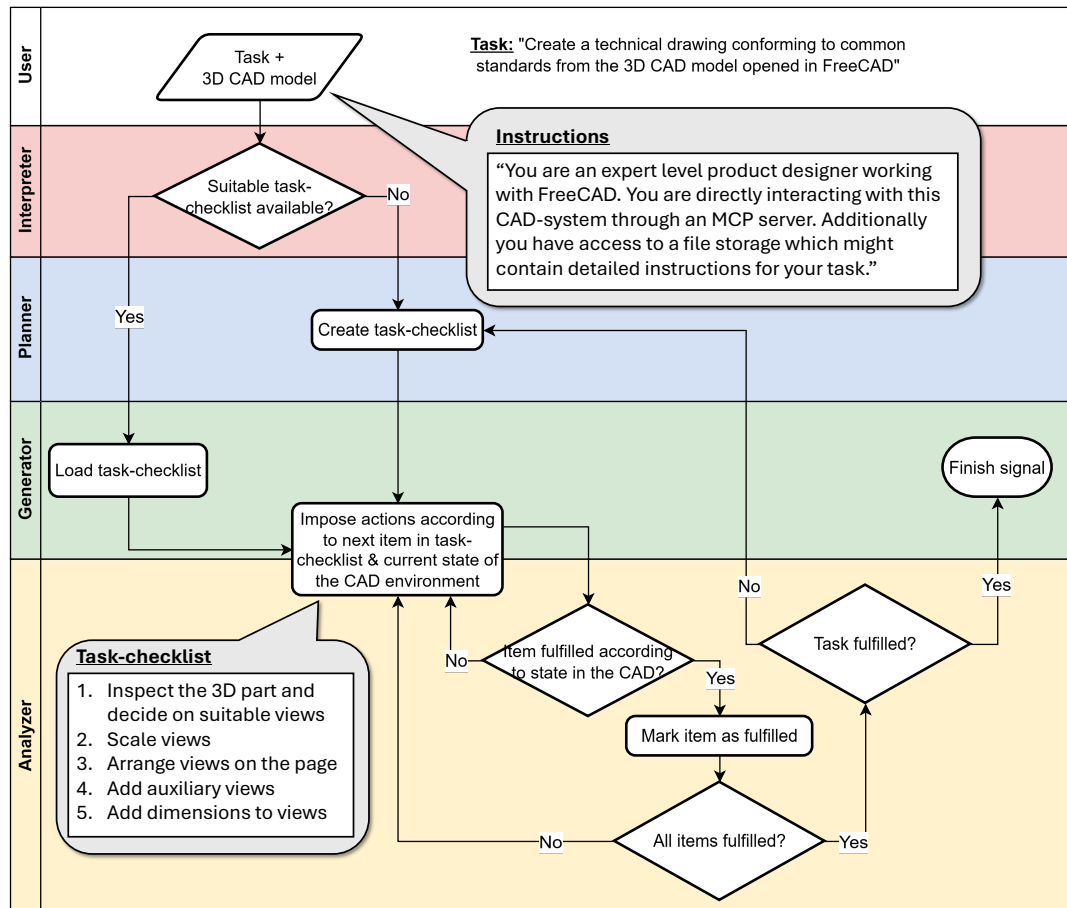


Figure 2: The general agent behavior and how it can be steered using exemplified and shortened, predefined checklists and instructions.

This behavior for the agent always forces either to write or update a file or implement a change directly within the CAD environment. This allows a close integration of the agent in the CAD system without any intermediate representations of the technical drawing or the 3D model profiting of hard constraints imposed by the environment. In contrast to approaches based on textual representations, the model relies more on the visual inputs from the CAD environments to assess the current state and generate further actions. This, however, requires the construction of support views to assess different options regarding key decisions. Thus, our approach discretizes the possible placement of views on the drawing page with an auxiliary grid. The center points of the different views in the drawing can then be directly placed on the discrete options on the grid by the agent. Fig. 3a visualizes this on an example. For the construction of auxiliary views, like section and detail views, and the placement of dimensions, another support view of the drawing page is used. This view enumerates the vertices in every view in the drawing enabling direct callouts, like creating a dimension between two vertices or sections through certain vertices. Fig. 3b shows an example of this support view.

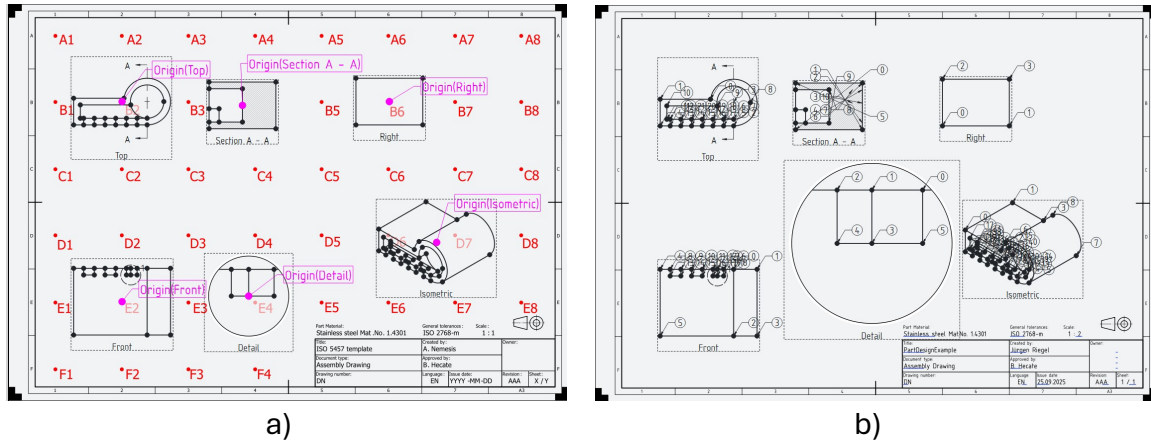


Figure 3: Support views to a) define the placement the views on the drawing page and b) define dimensions based on vertex points of the projected geometry.

4 Experiment

To assess the functionality of the agent tie-in, and the quality of the resulting drawings, we compare them to drawings manually derived from the same 3D models. Our assumption here is that a human drafter considers aspects like the readability and comprehensibility of a drawing that are difficult to describe with algorithmic metrics. This becomes evident in the choice of excess views or driven dimensions. These increase comprehensibility for the reader of the drawing but add no necessary new information to define the part. We extract therefore certain characteristics from manually drafted drawings and compare these to automatically generated ones. We define the following dimensionless metrics:

- **View Coverage (VC):** This metric is a ratio expressing how many of the human-selected views or equivalent ones are covered by the generated drawing. A higher value expresses a closer fit to the humans' decisions.
- **Excess View Rate (EVR):** The ratio of excess views compared to the human-drafted drawing, relative to the number of views in the generated drawing. A lower value expresses a closer fit to the humans' decisions.
- **Duplicate View Rate (DVR):** Number of equivalent and therefore duplicate views, relative to the number of views in the generated drawing. A lower value expresses less redundant information in the drawing.
- **View Placement Error Rate (VPER):** The number of overlaps between two views and views that exceed the boundaries of the free drawing area relative to the number of views in the generated drawing. A lower value means less errors in the drawing.
- **Page Coverage (PC):** This metric is a ratio describing how much of the drawing page, excluding the frame and the title area, is covered by the views. A higher value indicates that the drawing page was well used.

In the workflow of generating a valid technical drawing from a given 3D CAD model, we distinguish between several key decisions. For the experiment, we limit the decisions only to the selection and the arrangement of views, excluding dimensioning, tolerancing and additional textual remarks. These decisions can be made in an arbitrary sequence, except the initial creation of the drawing page with basic views. In a first set of experiments, the agent is tasked to make the decisions independently of each other. The agent is firstly tasked to initialize the drawing with an initial set of views. For every further decision, the state of the drawing is set back to the same initial point for all tested models. Afterwards,

the agent is tasked with defining a drawing using all available tools and making all decisions without separating one particular one out.

Tab. 1 shows the observed decisions made by the drafter in the manually created drawings. These decisions influence different metrics, e.g. the decision on the scaling of a view does not influence the VC or the EVR, but potentially the VPER and the PC. On the other hand, while the decision to add a view has a primary influence on VC, EVR and DVR, the VPER is not the focus of the observation, and constrained due to the norms of orthogonal projections. In Tab. 1, we therefore assort the relevant metrics to the individual decisions. Experiment 5 is separated, as the special case, where the agent has here access to all tools and is not tasked to make an individual decision, but define a drawing completely.

Table 1: Decisions made by a human drafter during the creation of a technical drawing and the assorted metrics to describe the difference between the manually created and the generated drawing.

Experiment ID	Decision	Assorted metrics
1	Which views should be included in the drawing?	VC, EVR, DVR
2	Which scale should be used for each view?	VPER, PC
3	Where should the views be placed on the drawing sheet?	VPER, PC
4	Which further views should be added or removed?	VC, EVR, DVR
5	Create a technical drawing using all available tools.	VC, EVR, DVR, VPER, PC

For the experiments described we included 2 different multimodal LLMs as core of the agent. ChatGPT-5.2-2025-12-11 [26] represents commercial flagship models, available remotely through an API. We also tested the smaller Qwen3-VL-30B-A3B-Instruct [27] as a locally hosted model, that runs on a Nvidia RTX 4090 GPU.

The 3D models used for the experiment were taken from the TechMB Visual Question-Answering benchmark [9], which contains manually created drawings of 180 distinct objects from the Autodesk Fusion 360 segmentation dataset [28]. The original step files from the Autodesk dataset were taken and compared to the characteristics of the corresponding TechMB drawings.

5 Results and Discussion

Tab. 2 shows the results for both models in all experiments. For all evaluation metrics, the mean value over the 180 drawings was calculated. As discussed before, VC, EVR and DVR do not change when moving views or changing scale in experiments 2 and 3 from the initial experiment 1. Due to the discussed reset between the experiments 2, 3 and 4, the values are equal between both models.

The results show that both models already have a relatively high initial VC with a low EVR and DVR. This can be interpreted that similar decisions as the human drafter regarding the initial view selection were made. However, with an EVR of 0.314 for ChatGPT5.2 and 0.288 for Qwen3 a significant part of views were overestimated. This indicates that both models were not quite as selective in their choice of views as the human drafters were. At the same time, the DVR remains constantly low. This suggests that even though a significant portion of the objects are rotational or reflectional symmetric, redundant views were chosen only seldomly.

In overall, the models performed comparatively equal, except the more than 10 times higher VPER of Qwen3 in experiment 3, than ChatGPT5.2. Here, the agent powered by Qwen3 often moved multiple views to the same place in the drawing page, so they overlapped fully. In another frequent mistake, the agent moved views completely out of the drawing page.

Table 2: Mean results of the 4 distinct decisions for the whole dataset in the 5 metrics. The arrows indicate if a higher or a lower value is preferable.

Experiment ID	Model	VC $\uparrow$	EVR $\downarrow$	DVR $\downarrow$	VPER $\downarrow$	PC $\uparrow$
1	ChatGPT5	0.818	0.314	0.000	0.241	0.058
	Qwen3-VL-30b	0.791	0.288	0.058	0.151	0.060
2	ChatGPT5	0.818	0.314	0.013	0.127	0.052
	Qwen3-VL-30b	0.818	0.314	0.013	0.027	0.021
3	ChatGPT5	0.818	0.314	0.013	0.064	0.065
	Qwen3-VL-30b	0.818	0.314	0.013	0.695	0.042
4	ChatGPT5	0.848	0.305	0.081	0.181	0.068
	Qwen3-VL-30b	0.864	0.302	0.089	0.220	0.069
5	ChatGPT5	0.925	0.422	0.071	0.058	0.048
	Qwen3-VL-30b	0.935	0.332	0.153	0.289	0.028

While the evaluation of the mean values allows an overall assessment of the model choices, it is important to also evaluate the statistical bandwidth of the created drawings. Fig. 4 helps to display the distribution of the metrics over the whole set of CAD models included in the experiments. It also allows a direct comparison of the changes between experiments 1 and 4. Here we see that even though both models performed roughly equal in median, the range of results is quite different. The distribution of the ChatGPT5.2 results are more concise than the ones generated by Qwen3. This means that the resulting drawings have a more persistent quality over the whole dataset. ChatGPT5.2 showed also an overall greater improvement from the initial state at the end of experiment 1 to the following experiments.

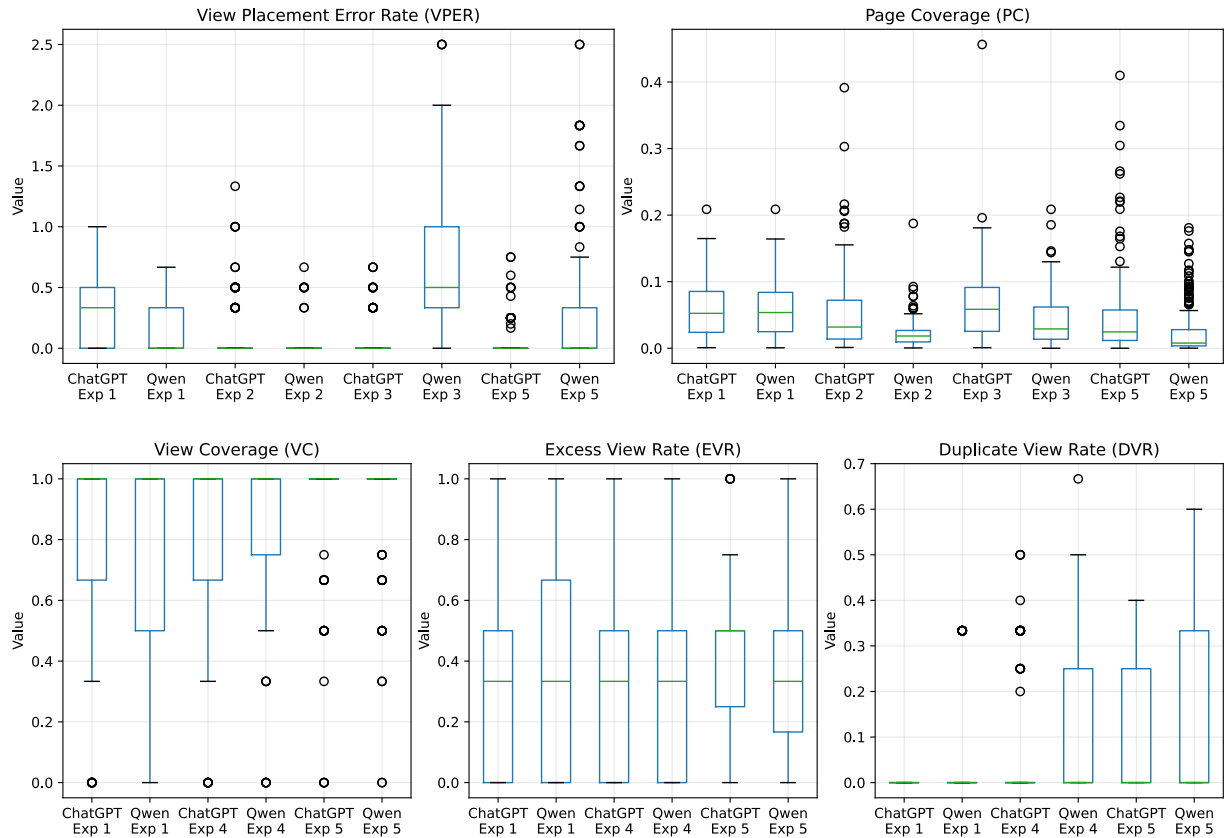


Figure 4: Distribution of the results in defined all metrics. A selection regarding the relevance of different metrics for different experiments was made according to tab. 1.

Of special interest are the results of experiment 5 in which the agent had to define a whole drawing with access to all different tools of the previous experiments. Both models manage to get a high VC at the price of a comparatively high EVR and DVR. This means that both models ended up adding more and more views to the drawing while at the same time scaling down the drawings to avoid placement errors. This led to a comparatively low PC and sometimes therefore to a low visibility of the details in a single view. ChatGPT5.2 performed in overall again better than Qwen3 and both are comparable to the results of the previous experiments. This allows for the conclusion that a stack of subsequent mistakes is avoided even without breaking down the task into single decisions. It must be hereby added that the agent needed substantially longer for each drawing as it made more iterative changes to the drawing. In contrast to the isolated experiment 1 and 4, both models in experiment 5 rarely made use of the option to make screenshots of the 3D CAD model from different viewpoints. Instead, the agent chose to initialize the drawing with a set of blindly estimated views, which were then extended and altered iteratively. The differences, especially between experiments 1 and 5, suggest that the processing of screenshots of the 3D object may lead to a more careful selection of views. It also raises the question whether the models are trained better to process these screenshots, than the technical drawings themselves. The results furthermore make it clear that even current flagship models need further finetuning regarding the understanding and interpretation of technical drawings. These visual abilities are core to the agent concept presented here in this paper. It also becomes clear that the ChatGPT5.2 flagship model, with a more than 10 times larger parameter size, did not yield a substantial performance increase of the agent compared to the locally hosted Qwen3 model. This shows the diminishing returns of large scale LLMs for specialized domains and encourages here the focus on smaller models for customization which is achievable with access to less resources.

6 Limitations

The study of this paper is subject to limitations regarding the dataset, the evaluation methods and the implementation of the specific agent. While the TechMB dataset originally picked the objects at random from the foundational Autodesk segmentation dataset, the high share of relatively simple and symmetric parts is apparent. This directly biases the number and selection of views used in the TechMB drawings to fully depict the parts, and therefore the difficulty of the decisions regarding view selection. However, our results show that the error quota for both models is still quite high. Additionally, the defined metrics have only limited expressivity regarding the comprehensibility and quality of the drawing. While VPER and VC correlate with the decision quality, a low EVR or DVR is only an indicator for the quality as additional views might help to distribute dimensions and increase drawing comprehensibility. The values are therefore helpful to observe the impact of agent choices but not necessarily drawing quality. Lastly, the implementation of an MCP server as well as the use of LLMs requires the design of function descriptions in docstrings as well as instructive prompts. These can have large influence on the choices of the agent and its performance. However, the systematic design and improvement of these texts is difficult and out of scope for this work. All prompts and docstrings used to create the results are contained in the source code which is published alongside this paper².

7 Conclusion

In this paper, we analyzed the capabilities of agentic CAD systems in the light of the current breakthrough developments in LLMs. We discussed different approaches for the integration of LLMs for CAD generation and particularly the derivation of technical drawings. Then we presented an agent based architecture, that directly integrates the LLM powered agent into the CAD environment via an

² <https://github.com/XDP-Opt/TechDraw-MCP-Agent>

MCP interface. This is opposed to the common, indirect approach to using an intermediary script that defines the CAD model and the technical drawing. In an experiment, we tested the capabilities of two differently sized LLMs as core of the agent to define the foundational characteristics of a technical drawing. Our results show that not even a current flagship model possesses the vision capabilities necessary to handle this task. However, they also show that the customization of smaller, more accessible models could be an effective way to fill this gap.

This finetuning of small LLMs and the generation and provision of the necessary training data remains part of future work. The provided agent based framework and the described experiments can help hereby to benchmark the results of this finetuning. The MCP-based approach can also further be used to evaluate the sequence of decisions made by the model to change things in the CAD environment. This could be a central part of understanding the planning capabilities of LLMs with regards to the fulfillment of domain specific tasks and therefore the composition of further training data.

Acknowledgements

This research was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 543073350.

References

- [1] Ruemler, S. P., Zimmerman, K. E., Hartman, N. W., Hedberg, T., & Barnard Feeny, A. (2017). Promoting Model-Based Definition to Establish a Complete Product Definition. *Journal of Manufacturing Science and Engineering*, 139(5), 051008. <https://doi.org/10.1115/1.4034625>
- [2] Dillenhöfer, F., Doksanbir, A., Kraske, J. L., & Künne, B. (2023). Automatic Generation of Training Data for AI Object Detection in Terms of Technical Drawings in Engineering. In M. E. Auer, R. Langmann, & T. Tsiatsos (Eds.), *Open Science in Engineering* (Vol. 763, pp. 643–651). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-42467-0_60
- [3] Sofias, K., Kanetaki, Z., Stergiou, C., Kantaros, A., Jacques, S., & Ganetsos, T. (2025). Implementing CAD API Automated Processes in Engineering Design: A Case Study Approach. *Applied Sciences*, 15(14), 7692. <https://doi.org/10.3390/app15147692>
- [4] Goli, A., & Singh, A. (2024). Frontiers: Can Large Language Models Capture Human Preferences? *Marketing Science*, 43(4), 709–722. <https://doi.org/10.1287/mksc.2023.0306>
- [5] Alam, M. F., Lentsch, A., Yu, N., Barmack, S., Kim, S., Acemoglu, D., Hart, J., Johnson, S., & Ahmed, F. (2024). From Automation to Augmentation: Redefining Engineering Design and Manufacturing in the Age of NextGen-AI. *An MIT Exploration of Generative AI*. <https://doi.org/10.21428/e4baedd9.e39b392d>
- [6] Song, B., Zhou, R., & Ahmed, F. (2024). Multi-modal machine learning in engineering design: A review and future directions. *Journal of Computing and Information Science in Engineering*, 24(1), 010801.
- [7] Hou, X., Zhao, Y., Wang, S., & Wang, H. (2025). Model Context Protocol (MCP): Landscape, Security Threats, and Future Research Directions. *arXiv*. <https://doi.org/10.48550/ARXIV.2503.23278>
- [8] Riegel, J., Mayer, W., & van Havre, Y. (2016). FreeCAD. *FreeCADspec2002*.
- [9] Kunz, L., Klostermeier, M., Thanabalan, K., Legler, T., & Ruskowski, M. (2025). TechMB: Exploring the Potential of Vision Language Models for Interpreting Technical Drawings. *DS 140: Proceedings of the 36th Symposium Design for X (DFX2025)*, 179–188. <https://doi.org/10.35199/dfx2025.19>
- [10] Deutsches Institut für Normung (DIN). (2021). Technical product documentation - Digital product definition data practices (ISO No. 16792; Versions 2021-04).
- [11] Yan, Y., & Bohn, M. (2017). Limitations of notation system in centred part alignment accuracy imposed by ISO standard and proposal for an improved methodology. *Journal of Machine Engineering*, 17(2), 94–101
- [12] McMahon, C. A., & Davies, D. (2006). The use of annotation in design representation: 5th International Seminar and Workshop of Engineering Design Integrated Product Development: Design Methods for Practice, EDIPrD 2006. Design Methods for Practice - 5th International Seminar and Workshop of Engineering Design Integrated Product Development, EDIPrD 2006, DS 37, 105–111.
- [13] Buric, M., Brcic, M., & Skec, S. (2021). Towards Automated Drafting in CAD Systems. 2021 4th International Conference on Electronics and Electrical Engineering Technology, 233–238. <https://doi.org/10.1145/3508297.3508335>

- [14] Auto Generating Drawings (BETA) - 2025 - What's New in SOLIDWORKS. (n.d.). Retrieved September 24, 2025, from https://help.solidworks.com/2025/English/WhatsNew/t_drawings_autogenerating_draw.htm
- [15] Still, T. (2024, October 10). *AI in Manufacturing: Automated Drawings in Autodesk Fusion*. Retrieved: November 20, 2025, <https://www.autodesk.com/products/fusion-360/blog/ai-in-manufacturing-automated-drawings-in-autodesk-fusion/>
- [16] CAD Drawing Automation Software | DraftAid Integrations. (n.d.). Retrieved September 24, 2025, from <https://draftaid.io/integrations/>
- [17] Berger, E., Dammann, M. P., Mehlstäubl, J., Saske, B., Braun, F., & Pactzold-Byhain, K. (2025). Challenges and opportunities in the integration of generative AI with computer-aided design. *Proceedings of the Design Society*, 5, 881–890. <https://doi.org/10.1017/pds.2025.10102>
- [18] Schüpbach, A., San Miguel, R., Ferchow, J., & Meboldt, M. (2025). From text to design: a framework to leverage LLM agents for automated CAD generation. *Proceedings of the Design Society*, 5, 1893–1902. <https://doi.org/10.1017/pds.2025.10203>
- [19] Xu, X., Lambourne, J., Jayaraman, P., Wang, Z., Willis, K., & Furukawa, Y. (2024). BrepGen: A B-rep Generative Diffusion Model with Structured Latent Geometry. *ACM Transactions on Graphics*, 43(4), 1–14. <https://doi.org/10.1145/3658129>
- [20] Afzal, M., Ali, S., Khan, M., Sheikh, T., Sinha, S., & Stricker, D. (2024). Text2CAD: Generating Sequential CAD Designs from Beginner-to-Expert Level Text Prompts. *Advances in Neural Information Processing Systems* 37, 7552–7579. <https://doi.org/10.52202/079017-0242>
- [21] Daareyni, A., Martikkala, A., Mokhtarian, H., & Ituarte, I. F. (2025). Generative AI meets CAD: enhancing engineering design to manufacturing processes with large language models. *The International Journal of Advanced Manufacturing Technology*. <https://doi.org/10.1007/s00170-025-15830-2>
- [22] Schöfer, F., & Seibel, A. (2025). Augmented design automation: leveraging parametric designs using large language models. *Proceedings of the Design Society*, 5, 671–680. <https://doi.org/10.1017/pds.2025.10081>
- [23] Liao, J., Xu, J., Sun, Y., Tang, M., He, S., Liao, J., Yu, S., Li, Y., & Guan, X. (2025). Automated CAD Modeling Sequence Generation from Text Descriptions via Transformer-Based Large Language Models. *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 21720–21748. <https://doi.org/10.18653/v1/2025.acl-long.1054>
- [24] [In Publishing] Kunz, L., Kokulan, T., Legler, T., Malburg, L., Bergweiler, S., Ruskowski, M., & Bergmann, R. (2025). XDP-Opt: Experience-Based Design Process Optimization for Industrial Manufacturing. *KI2025 - 48th German Conference on Artificial Intelligence*.
- [25] Zou, Q., Wu, Y., Liu, Z., Xu, W., & Gao, S. (2024). Intelligent CAD 2.0. *Visual Informatics*, 8(4), 1–12. <https://doi.org/10.1016/j.visinf.2024.10.001>
- [26] OpenAI. (2025, December 11). *Entdecke GPT-5.2*. Retrieved January 24, 2026, from <https://openai.com/de-DE/index/introducing-gpt-5-2/>
- [27] Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., Zheng, C., Liu, D., Zhou, F., Huang, F., Hu, F., Ge, H., Wei, H., Lin, H., Tang, J., ... Qiu, Z. (2025). *Qwen3 Technical Report*. arXiv. <https://doi.org/10.48550/ARXIV.2505.09388>
- [28] Lambourne, J. G., Willis, K. D., Jayaraman, P. K., Sanghi, A., Meltzer, P., & Shayani, H. (2021). Brepnet: A topological message passing system for solid models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 12773–12782).

Theoretical foundation to model design decisions for hybrid (Human / AI) decision support systems

Eric Kempkens ^{1*}, Anvit Upadhyay¹, Zhongtian Wang¹, Rainer Stark¹

¹Technische Universität Berlin, Institut für Werkzeugmaschinen und Fabrikbetrieb, 10587 Berlin

*Corresponding author. E-Mail address: eric.kempkens@tu-berlin.de

Abstract: This contribution presents a theoretical foundation for modelling granular, incremental activities in the product development process (PDP) for utilization in the conception and realization of hybrid (Human/AI) decision support systems. This approach differs from other modeling approaches of the PDP by modeling granular activities while simultaneously considering the external factors influencing these activities. This holistic approach can be utilized to model singular decision cases within the PDP, which can be organized as a graph to capture the interdependencies between decisions within PDP activities. The resulting models can then be used as a foundation for context engineering to be utilized in the development of agentic AI systems.

The theoretical foundations proposed in this paper deliver a specialized modeling foundation to capture the granular, incremental activities of the design process. This has the goal of enabling the conceptualization of agentic AI systems, which can aid designers in their dynamic design process. The proposed model is validated through a study using a student project composed of a PDP, showing that the model can be applied onto real-world design processes to model design activities with the goal of conceptualizing AI enabled automation.

Keywords:

Hybrid Decision Support, Product Design, Agentic AI, Mathematical Modelling

1 Introduction

Hybrid Decision Support Systems (HDSS) for Product Design are systems that implement Artificial Intelligence (AI) automation in a way which does not diminish the role of human expertise, combining their strengths. HDSS systems compute solutions towards decision cases (DC) using AI methodologies, which are then evaluated, further processed or rejected by human operators. HDSS are aimed at increasing the time efficiency of workflows by aiding designers in finding solutions for DC faster [1].

The product design process (PDP) is a highly dynamic and non-linear process often resolved by implicit designer knowledge and design heuristics [2] and consists of many granular activities which add complexity in steps onto an artifact [3]. The granular decision cases (GDC) within the PDP inform and influence each other and contribute toward creating the final artifacts. A decision case is defined here as a situation in which attributes or outcomes must be decided on to move onto subsequent steps within an activity. Within the Detailed Planning process (DPP) of the PDP, a lot of GDCs can be identified which contribute to the final artifact. For example, in CAD modeling, examples of GDCs would be the form of an initial two-dimensional sketch, how it is subsequently extruded or rounded and how the geometries as well as functional elements are realized. A DC is situated within a larger design activity (DA). A DA here is defined as a process within the DPP of the PDP which has a clearly defined starting and ending point for each iteration of the activity, like the modelling of complete products or product components using CAD.

With recent breakthroughs in the field of AI, in terms of model capabilities as well as the rising availability of AI computing services and frameworks, there has been a push to apply these technical opportunities to the PDP [4-5]. HDSS are one way to enable a specialized AI integration into the PDP. HDSS systems aim to ease the navigation of complex product requirements, integrate usage-related information, offer validation and generate explainable solutions, among other important attributes. This is done by leveraging extreme data for AI methods, which is heterogeneous in nature and derived from many different sources within the PDP as well as the product life cycle [1]. The non-linearity of the PDP presents a challenge for the implementation of HDSS. In their role as assistance systems, they should ideally be implemented in a way which embeds itself into existing PDP environments and workflows without friction after initial adoption. One approach is to develop AI based full solvers, which offer solutions for the complete activity or a sub-activity composed of multiple DC generated by AI methods. This approach has the benefit of potentially speeding up PDP activities by providing possible solutions for substantial segments or complete DA. This can be done, for example, by using Generative Adversarial Networks to create product geometries [6]. But for this to be most effective, their results must be of high quality and satisfy all applicable constraints. If this is not the case, the designer might need to manually improve the solution generated, which could negatively impact or negate the net efficiency gains. Another way of implementing AI based support systems is to dynamically offer decision support for granular activities. This enables a form of HDSS, where the designer has more control within the individual DCs which culminate in the final artifact. Saving time within GDCs enables time savings at each step, potentially adding up to large time savings for the complete DA. Taking a granular approach in generating solutions within the DPP could aid in assuring the quality of generated results, lessening the cumulative time needed for manual adaptations.

In this paper, we propose a theoretical foundation to describe and model the PDP as a network of interconnected GDCs which can be modeled as a mathematical graph. Mathematical abstraction helps in generalizing information flows within the PDP, making it easier to standardize information objects which can be represented as sets of encoded information. From there, singular nodes and associated edges can be used for context engineering, meaning the fine-tuning and careful curation of inputs representing context for transformer-based AI processing, for HDSS implementations. This framework can be used to conceptualize the data pipeline needed for granular HDSS within the PDP using Natural Language Processing (NLP).

2 Research Problem

The design processes within the PDP are often complex and non-linear, especially during the DPP phase. This can include the modeling of product geometries and the placement of system elements, one example being the placement of electrical components on printed circuit boards. But it could also describe other DPP processes, like the placement of elements within the factory planning process. To develop intelligent HDSS which can aid designers dynamically within the DPP of the PDP, there is a need to model their granular workflow. This level of modeling the PDP is classified as the micro category of PDP modeling [7], which focuses on granular process steps and their associated contexts.

When modeling the PDP from the micro perspective with the intention to analyze and conceptualize possible implementations of HDSS, a challenge arises which is not specifically addressed by traditional PDP modeling approaches. To leverage reasoning capabilities of transformer models as proposed in [8], where semantic elaboration on numerical problems is seen to enhance the quality of generated solutions, there is a shift in paradigm required. Away from only utilizing rigid, coded formulation towards natural language descriptions when modeling the design process with the goal of HDSS automation. Contemporary PDP modeling approaches are not dedicated to a granular approach while also capturing

the context and data in natural language along with numerical parameters needed for AI automation, as seen in the comprehensive study by Wynn & Clarkson as well as the study by Yin and Zhang [7,9]. One classical approach is the Task-Episode Accumulation Model (TEA) [10], which semantically describes the PDP as a series of episodes building upon each other, which captures the evolution of the design state via goal-oriented phases with 6 operators, but does not document GDCs as defined here. The Design History System (DHS) [11] features a schematic to document GDCs but argues against (unstructured) textual semantic representations in favor of more rigid coded descriptions of arithmetic operations and attributes, representing pre-transformer paradigms of utilizing coding language abstractions in favor of semantic elaborations. However, to utilize transformer capabilities for HDSS within the DPP, a holistic perspective is needed which captures semantic descriptions as well as numerical parametric of GDC. This includes the objectives of GDC, as well as the specific context, the objectives, the constraints, and the environment of the DPP as well as relevant aspects of the wider PDP. HDSS approaches would benefit from leveraging semantic context and operation descriptions for generating decision results for designer consideration. The Engineering Operating Systems (EOS) model [12] delivers this holistic perspective for the PDP but offers no framework to model the GDC of DPPs.

Another shift in paradigm with automation based on modern sub-symbolic AI (SSAI) methodologies is the ability to adapt to different input formats when computing solutions to inquiries. This contrasts with the mostly fixed input formats of symbolic AI automation [13]. To describe DAs with the goal of applying SSAI automation, new ways of describing these activities on a granular level, while also considering external flows of information of non-fixed formats can be expected to be of great benefit. This could be used to conceptualize and develop effective AI agents (AIA) or agentic AI (AAI) using retrieval-augmented generation (RAG), which can leverage diverse domain-specific external knowledge bases for foundational Large Language Models (LLM) without a dedicated, resource-intensive training process [14]. A methodology for mathematical modeling of GDC can aid in standardizing information objects and flows for context engineering, which is an important foundation, as data quality is critical for building effective AI applications [15]. This can be applied to analyze DA to conceptualize effective transformer/LLM-based AIA/AAI architectures, as described in [16-18], which are specialized for DAs and can be adapted to different design environments. Since AI methodologies like RAG are novel concepts, classical PDP modelling approaches do not include avenues for their specific considerations.

3 Research Question

The research question derived from the problem statement is as follows: *How can a theoretical foundation be laid which offers a framework to model GDC during the DPP of the PDP while also considering the holistic insights of the EOS model?* A synthesis of both these aspects is needed, as the implementation of dynamic HDSS requires modeling granular activities while also considering the wider flow of information within the PDP to conceptualize effective, quality HDSS tools. This contribution aims to fill this gap by providing this theoretical foundation to model DA in a way which can aid the conceptualization and development of dynamic HDSS, which offer solutions toward building capable AIAs for singular DC. Standardized mathematical modeling helps with standardizing flows of information and adaptability of subsequent architectures.

4 Methodology

To construct the theoretical framework, mathematical modeling is used to describe how different information objects consisting of encoded information, represented as sets, influence DPP activities. For this, a large emphasis is placed on describing mathematical objects using set theory notation. This allows the modeling of information flows within DA of the DPP of the PDP in a way which is both fitting for

describing human design heuristics as well as non-specific information objects for automation using AI methodologies within this context. Describing flows of information using set theory notation has the benefit of being easily adaptable as well as lending itself well as representations of input sets for AI models. One further benefit is the capture of human workflows, which should help in conceptualizing and developing AIA/AAI tools which synergize with the natural process. To validate the framework laid out within this contribution, a case study is conducted. This case study was carried out using data recorded during a student project involving a PDP, in which all principal components found within the mathematical framework can be recorded, observed, and modeled.

5 Theoretical Framework

For the theoretical foundation for modelling granular decision-making process during the DPP of the PDP we apply mathematical abstraction to organize this process as a graph of individual decisions. Within this framework mathematical objects represent vectors containing encoded information relevant to their identification. A union ‘ $\cup$ ’ shall describe a broader, non-specific combination process, not just strictly a union of sets. This is done to represent the combination of information for possible use as input sets of information for AI models, which may shed unneeded or redundant information for rationalizing set size. Further ‘ $\rightarrow$ ’ describes a function or a system of functions of ambiguous nature and dynamics, taking various input objects and through a process of synthesis and manipulation arriving at a specified result object. They are standing in for the processing by human operators, AI based automation or other applicable black-box/grey-box methods.

The resulting artifact A of a design process P can be understood as a cumulation of a variable number of subprocesses $p \in P$. This artifact A can take the shape of a CAD model, a sketch or even a physical model. Thus, the cumulation of a DA can be expressed as $\forall p \in P, p \rightarrow A$, meaning that every subprocess p contributes toward the creation of a resulting Artifact A . Every subprocess p uses the objects c, d to arrive at a result r through $f(c, d) = r$. In this c represents a union of three components: environment e , documentation b and previous results r_r . Context c shall be analog to contextual information human designers use to work on and finalize a given subprocess p . This should include relevant information about the environment of the design process e , all the relevant documents b relating to the specifications of the final artifact A and possible subartifacts and finally information results $r_r \in R$ of other processes. We define $e \cup b \cup r_r \rightarrow c$, meaning that a union of all the information e, b, r_r shall result in the context c . The decision d shall be the result of a synthesis involving the specifications s of solving an objective o . Specifications s shall describe all the parameters and constraints relevant to solve objective o . Objective o shall describe the objective of the relevant process p . This can include different DAs tied to a concrete decision, like the manipulation of a polygon, the placement of a component, etc. in accordance with d . The specifications s are the result of a synthesis between the context c and objective o , thus $c \cup o \rightarrow s$, while a decision $s \cup o \rightarrow d$. This means that the logic of synthesizing specific information of a decision d is a non-linear process regarding o . Result r describes the results of a decision d made within the context of c . It has no need for uniformity and is able to describe the result of many different operations within a design process P . It is the result of a function $f(c, d)$, which can both represent the actions of a designer, the calculation processes of an AI method to generate the result of a decision case or similar processes. Since each object p results in a decision r and can each be influenced via their associated context c which is informed by results of previous processes r_r from which c is partially derived, P can be organized as a graph $g_P \subseteq P \times P$. Figure 1 shows a visual representation of a singular node p of a process Graph g_P .

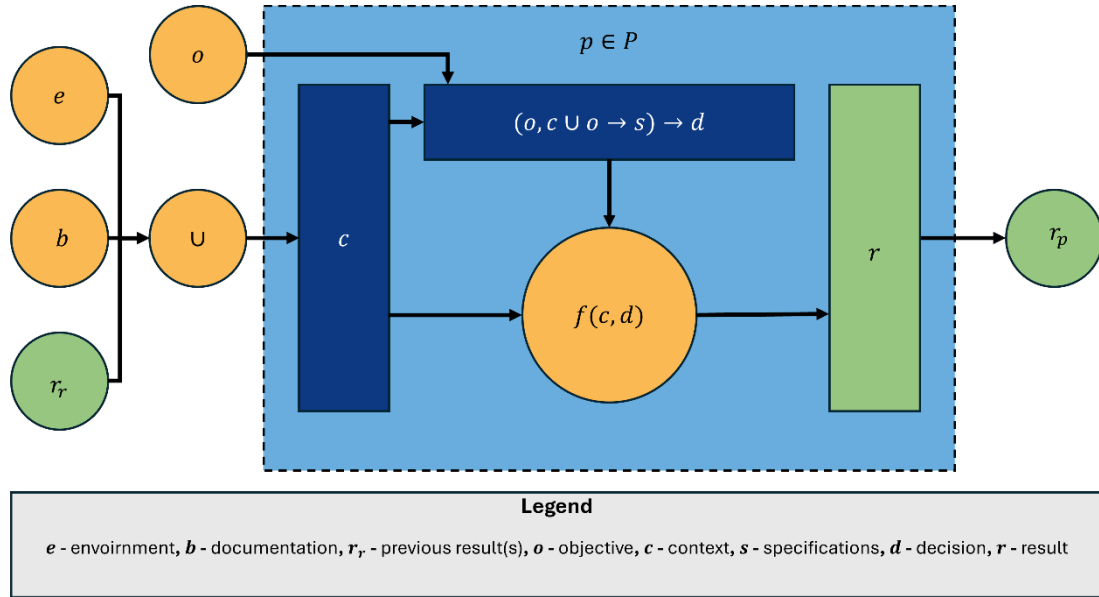


Figure 1: Visualization of the modelling approach for a subprocess $p \in P$

6 Case Study

To validate this approach, a case study using a student engineering project from an engineering module using the proposed framework has been carried out. Within the project work, 3 student groups are working towards delivering a concept and a working prototype of the same general product concept. The aim was to create a staircase ascender capable of transporting heavy loads like packages and groceries upstairs. From there, the student groups had to develop a product concept from scratch, including product specifications, technical solutions, as well as modelling their concept as a CAD model using Autodesk Fusion 360. The individual design decision cases of the modelling process needed to be resolved based on function, manufacturing as well as costs, leading to multifactorial considerations at each step where necessary during the modeling process.

The case study used the Fusion 360 projects of the student groups for applying the framework, as Fusion 360 project files preserve the causality between different functions of a modelling process within a timeline, making it possible to see how different processes influenced and added onto each other to arrive at the finished artifact. The theoretical modeling approach has been applied to the creation process of a chassis concept for a staircase ascender for one of the groups. The DDP process to model the chassis concept offers a manageable scope of GDCs while also being intricate enough to show that this model can be applied to real-world DPP processes.

Environment e was modelled via text as follows:

Modelling within Fusion 360, Development of Model Concept for Graded Student Project, Eventual Goal of Functional physical Prototype. Designer is part of a student team; they are the specialist for realizing concept design within CAD. Specifications of the technical solution are made as a team effort.

Documents b include the project documents created to specify the technical solution concept, technical specifications, as well as fixed physical specifications. The technical solution concept was delivered in the form of a one-page profile, consisting of 236 words and one figure visualizing the concept and features the following sections: *Abstract, Description, Team Specifics, Key Features of Prototype, Key Features of future Product, SWOT-Analysis*. The technical specifications are a written description, including all planned key components, consisting of 689 words. It features among others the following sections: *Concept, Motor Type and Chassis Material*. Additionally, technical specifications are further

defined as table of specifications. This table consists of 33 positions, detailing key features, and where possible includes metrics for laying down feature target values. For example, for the drive, the type of motor, the nominal performance targets of the motor at 13-40 Nm and target battery values at 12-24 V are laid down. All these documents combine to form the corpus of documentation on which the modeling processes of this student team are based on.

The model is applied onto the DPP of the chassis concept as P , using the Fusion 360 parametrics. This resulted in a protocol consisting of 30 individual subprocesses $p \in P$ and associated decision cases. Individual subprocesses were defined using the Fusion 360 timeline, splitting some consolidated steps from the parametric in favor of discrete sketches where a mirroring has been obscured. This better represents the underlying flow of decision cases. Function $f(c, d)$ has been recorded as the principal function used by the designer within the Fusion 360 environment. The names of the functions have been generalized. The protocol of the decision cases has been recorded in tabular form, with columns containing an index as well as $e, b, r_r, o, c, s, d, f, r$ for each identified $p \in P$. Table 1 shows an excerpt from this protocol, containing index of p as well as r_r, o, s, d, f, r for 8 of the 30 decision cases. Table 2 shows the corresponding contexts c for the p in Table 1. For this representation e and b are not represented here, as they do not differ regarding the earlier descriptions. Numerical parameters are represented using '...'. Visual representations of each step can be found in figure 3.

Table 1: Decision case protocol excerpt 1

$p \in P$	r_r	o	s	d	f	r
6	1	Sketch feet of chassis leg in + X + Z Plane	Shape, Size, Location	Shape: [quadrangle, chamfer in -Z, -X direction], Size: ... , Chamfer: ... , Location: ...	draw quadrangle, add chamfer	Sketch of Foot
7	6	Mirror feet sketch in - X, + Z Plane	Direction, Offset	Mirror +X Plane of Foot Sketch, Offset: ...	Mirror	Mirrored foot in - X, + Z plane
8	1,2,6,7	Sketch attachment point of chassis legs in the + X + Z Plane	Shape, Size, Location	Shape: quadrangle, Size: ... , Geometric_Center: + ... Z, + ... X	draw quadrangle	Attachment point sketch of chassis leg in the + X + Z Plane
9	8	Mirror attachment point sketch in - X, + Z Plane	Direction, Offset	Mirror + X Plane of attachment point sketch, Offset: ...	Mirror	Mirrored attachment point sketch of chassis leg in the + X + Z Plane
10	6,7	Extrude feet sketches	Type, Direction, Location	Type: Push, Direction: + Y, Depth: ... , Location: + ... Y	Push	Extrusion of the feet sketches
11	8,9	Extrude attachment points	Type, Direction, Location	Type: Push, Direction: + Y, Depth: ... , Location: + ... Y	Push	Extrusion of the attachment point sketches
12	1,2,7,10	Add custom reference plane	Location, Direction	Location: - ... X, Direction: Y, Z	add custom plane	Custom plane for guidance sketches for leg loft in the - X + Z Plane
13	1,2,6,7,8,9,10,11,12	Add reference sketch for loft	Shape, Size, Location, Orientation	shape: Quadrangle, Size: ... , Geometric_Center: ... , Orientation: Matches custom plane r_r 12	Sketch	Reference sketch for loft in the - X + Z Plane

Table 2: Decision case protocol excerpt 2

$p \in P$	c
6	Lay down the initial sketch for one foot of the legs which will house the axial connection for the endpoints of the chain drive. Add a chamfer which will act as the connecting face for the legs.
7	Mirror the sketch of the foot in the -X +Z Plane
8	Sketch the attachment point geometry to which the legs connecting the chassis to the feet of the chain drive will be connected. The attachment point is to be a reinforcing geometry attached to the main body
9	Mirror the sketch of the attachment point in the -X +Z Plane
10	Extrude feet sketches to form geometries for further modelling
11	Extrude attachment point sketches to form geometries for further modelling
12	Add custom reference plane to accommodate additional sketches for guiding a loft connection between the chassis attachment points and the feet in the -X Plane
13	Add sketch to guide the loft connection between the chassis attachment points and the feet in the -X +Z Plane

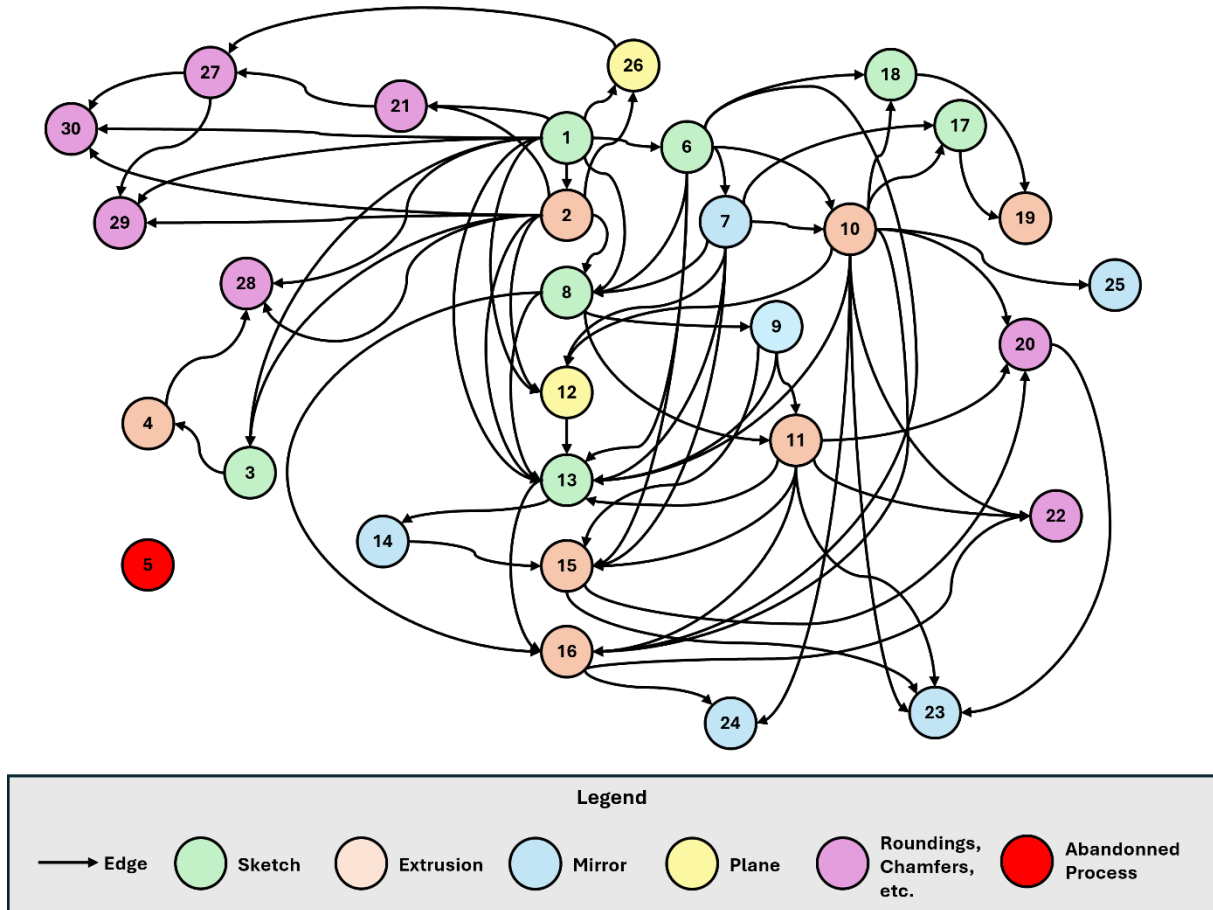


Figure 2: Graph visualization of the Chassis modelling process

Figure 2 shows a visualization of the graph g_P of the modelling process P of the chassis concept, showing the connections between the different nodes p . Green nodes represent sketches, orange nodes extrusions, blue nodes mirroring, yellow nodes custom planes, purple nodes roundings and similar geometric modelling manipulations. Red node 5 represents an abandoned sketch process, which is documented but has no direct bearing on the completion of the artifact. From all 30 subprocesses p there are 8 sketching based subprocesses, 7 extrusion-based processes, 6 mirroring-based processes, 2 creations of custom planes and 7 roundings and similar geometric manipulations. There are in sum 73

edges recorded, representing the influence different subprocesses have on one another. Figure 3 shows the resulting artifact A in Fusion 360 as well as screenshots showing results of $p \in P$ 6 to 13.

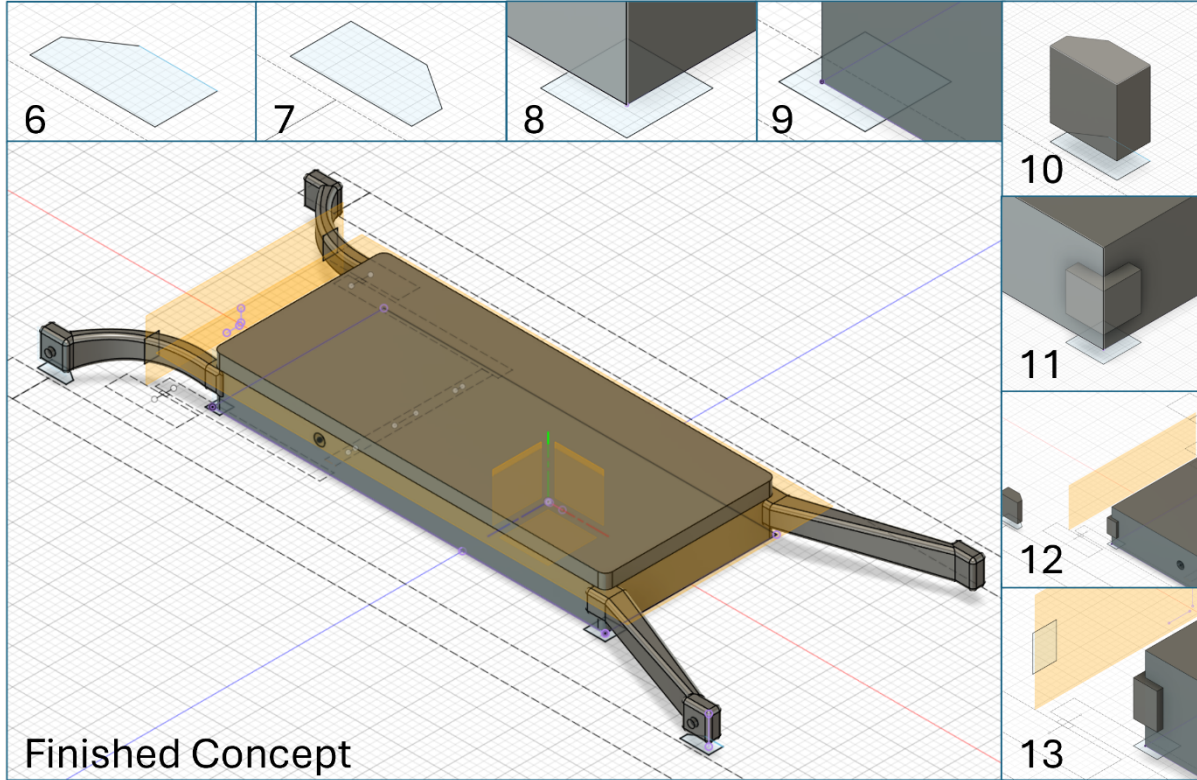


Figure 2: Results of $p \in P$ 6 to 13 and finished chassis concept in Fusion 360

This case study shows that the proposed theoretical foundation is a viable approach to document GDCs within the DDP of the PDP, resulting in a protocol which documents incremental design decisions with annotated context and parametric information, as well as being capable of modeling how different decision cases influence each other with graph visualizations. The resulting protocol is highly comprehensible in natural language and structured in a highly reproducible and adaptable way, which suggests that it lends itself to natural language processing.

This case study viewed environment object e and documentation b as static corpora of text. To reduce the size of input sets of tokens within NLP based processing, as well as to improve processing performance these objects would benefit from the removal of unneeded information. This might become necessary or highly beneficial for DDP processes of higher complexity, meaning a higher number of $p \in P$ and/or larger text corpora of e or b .

7 Application of the Framework

The proposed theoretical foundation describes the PDP as a graph of granular, incremental decisions, the complexity of which is tied to the complexity of the design process and the result artefact A . If one were to analyse a PDP P in its entirety, depending on the complexity of the task, the amount of work required to model each step from protocol would be prohibitively high if there were no automation used.

The theoretical foundation could also be used to model singular decision cases p . Instead of reconstructing the entire process graph of P one could just model specific p , for example if only certain

activities are of interest for automation. To do this only relevant result information r_p from previous steps and the objectives of the designers need to be captured, to only model a single node p in Graph P . This may come at the cost of losing some accuracy when modelling and gathering information sources and tracking the relations of GDC but offers a pragmatic approach to modeling singular activities. This allows the framework to be utilized for the analysis of GDC, without the necessity to model the complete graph of interconnected decision cases.

This framework has explicitly been developed with the intent of being applied to the analysis of design processes with the goal of developing HDSS automation tools. As such, a large focus is laid on information objects represented as sets of embeddings for the use in AI methods, as well as describing the process as a step-based cumulative decision process. This approach or similar methodologies might be adaptable onto all kinds of other activities in which an artifact is created through cumulative decision cases.

Transformer models offer a great opportunity to apply the proposed framework in conceptualizing HDSS applications. Inputs c, d can be translated into tokens for models with agentic capabilities to process $f(c, d)$. This can be achieved by utilizing, among others, the Model Context Protocol (MCP) standard, which offers the potential to realize widely inter-compatible framework to develop AI agents combining symbolic and SSAI capabilities and strengths. A powerful option of synthesizing b for context specific cases could be achieved using RAG which through specialized embedded vector search allows for the retrieval of context sensitive information from large corpora of text for augmenting processing results in transformer models [19]. RAG could be leveraged to find constraints for $f(c, d)$ from project specifications, design handbooks and norms, while frameworks like MCP could handle API calls, command injections or code generation for granular design tasks. This framework allows for the necessary modelling of activities and information to conceptualize such transformer-based agents, using current standards. This framework should also be adaptable to future AI frameworks and standards, if their inputs are vector based and their processing methods are of comparable nature.

8 Discussion and future research

This contribution presents a theoretical foundation to model GDC within DA. The framework combines the granular nature of decision-making cases within the design process while also including the holistic insights of the EOS model, providing an answer to the formulated research question. It also offers a framework to model GDC with semantic descriptions of context and operations along with numerical parameters, as demonstrated by the case study, which represents a shift in paradigm in PDP modeling for AI applications for the goal of subsequent automation. It can be used to analyze activities of the detailed planning phase with a focus on conceptualizing HDSS based AIA/AI automation tools by providing the foundation for context engineering of input objects for transformer models, for example via RAG, as well as describing results of process functions and associated decision case parameters. The validity of the proposed theoretical foundation has been demonstrated by a case study using student project work, showing that it can be used to model DAs within the DPP.

One avenue of future research will focus on automated analysis utilizing the proposed approach to building knowledge repositories. If possible, this method would lend itself well to creating databases for documenting large numbers of incremental, singular design decisions with annotated context. Such a database might be used to retrieve relevant granular steps fitting the current context c and objectives o using RAG. This would require many documented P , which requires in turn specific documentation as well as manual effort to translate into the proposed format. Future development of this approach could also include the automatic documentation of $p \in P$ for environments aside from Fusion 360, as well as the automatic creation of protocol for P as found in the case study. Another avenue of future research

will focus on developing HDSS implementations for DPP activities of the PDP utilizing the proposed theoretical framework. The framework will therein be used to analyze design activities and find processes which offer great potential for HDSS implementations using AIA/AAI utilizing MCP and RAG or methodologies and standards of similar formats and capabilities.

References

- [1] “Hybrid Decision Support in Product Creation – Improving Performance with Data Science and Artificial Intelligence,” *I40S*, vol. 2025, no. 1, 2025. DOI: 10.30844/I4SE.25.1.18.
- [2] S. Yilmaz, C. Seifert, S. R. Daly and R. Gonzalez, “Design Heuristics in Innovative Products,” *Journal of Mechanical Design*, vol. 138, no. 7, p. 071102, 2016. DOI: 10.1115/1.4032219.
- [3] L. A. Stauffer and D. G. Ullman, “Fundamental Processes of Mechanical Designers Based on Empirical Data,” *Journal of Engineering Design*, vol. 2, no. 2, pp. 113–125, 1991. DOI: 10.1080/09544829108901675.
- [4] Y. P. Tsang and C. K. M. Lee, “Artificial Intelligence in Industrial Design: A Semi-Automated Literature Survey,” *Engineering Applications of Artificial Intelligence*, vol. 112, p. 104884, 2022. DOI: 10.1016/j.engappai.2022.104884.
- [5] B. Lee and J. Kim, “AI-Led Design Innovation: Understanding Design-Centric AI Methods and Assistance Types,” *Proceedings of the Design Society*, vol. 5, pp. 469–478, 2025. DOI: 10.1017/pds.2025.10061.
- [6] C. Krahe, A. Bräunche, A. Jacob, N. Stricker and G. Lanza, “Deep Learning for Automated Product Design,” *Procedia CIRP*, vol. 91, pp. 3–8, 2020. DOI: 10.1016/j.procir.2020.01.135.
- [7] D. C. Wynn and P. J. Clarkson, “Process Models in Design and Development,” *Research in Engineering Design*, vol. 29, no. 2, pp. 161–202, 2018. DOI: 10.1007/s00163-017-0262-7.
- [8] J. Wei et al., “Chain-of-Thought Prompting Elicits Reasoning in Large Language Models,” *arXiv*, 2022. DOI: 10.48550/arXiv.2201.11903.
- [9] C. Yin and W. Zhang, “New Product Development Process Models,” in *Proceedings of the 2021 International Conference on E-Commerce and E-Management (ICECEM)*, Dalian, China: IEEE, 2021, pp. 240–243. DOI: 10.1109/ICECEM54757.2021.00054.
- [10] D. G. Ullman, T. G. Dietterich and L. A. Stauffer, “A Model of the Mechanical Design Process Based on Empirical Data,” *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, vol. 2, no. 1, pp. 33–52, 1988. DOI: 10.1017/S0890060400000536.
- [11] J. J. Shah, D. K. Jeon, S. D. Urban, P. Bliznakov and M. Rogers, “Database Infrastructure for Supporting Engineering Design Histories,” *Computer-Aided Design*, vol. 28, no. 5, pp. 347–360, 1996. DOI: 10.1016/0010-4485(95)00054-2.
- [12] P. Lunnemann, R. Stark, W. M. Wang and P. I. Manteca, “Engineering Activities—Considering Value Creation from a Holistic Perspective,” in *Proceedings of the 2017 International Conference on Engineering, Technology and Innovation (ICE/ITMC)*, Funchal: IEEE, 2017, pp. 315–323. DOI: 10.1109/ICE.2017.8279904.
- [13] M. Alam, P. Groth, P. Hitzler, H. Paulheim, H. Sack and V. Tresp, “CSSA’20: Workshop on Combining Symbolic and Sub-Symbolic Methods and Their Applications,” in *Proceedings of the 29th ACM International Conference on Information & Knowledge Management, Virtual Event, Ireland: ACM*, 2020, pp. 3523–3524. DOI: 10.1145/3340531.3414072.
- [14] M. Cheng et al., “A Survey on Knowledge-Oriented Retrieval-Augmented Generation,” *arXiv preprint*, 2025. DOI: 10.48550/arXiv.2503.10677.
- [15] K. Hiniduma, S. Byna, J. L. Bez and R. Madduri, “AI Data Readiness Inspector (AIDRIN) for Quantitative Assessment of Data Readiness for AI,” in *Proceedings of the 36th International Conference on Scientific and Statistical Database Management*, Rennes, France: ACM, 2024, pp. 1–12. DOI: 10.1145/3676288.3676296.
- [16] A. Vaswani et al., “Attention Is All You Need,” *arXiv preprint*, 2017. DOI: 10.48550/arXiv.1706.03762.
- [17] S. Brohi, Q. Mastoi, N. Z. Jhanjhi and T. R. Pillai, “A Research Landscape of Agentic AI and Large Language Models: Applications, Challenges and Future Directions,” *Algorithms*, vol. 18, no. 8, p. 499, 2025. DOI: 10.3390/a18080499.
- [18] R. Sapkota, K. I. Roumeliotis and M. Karkee, “AI Agents vs. Agentic AI: A Conceptual Taxonomy, Applications and Challenges,” *Information Fusion*, vol. 126, p. 103599, 2026. DOI: 10.1016/j.inffus.2025.103599.
- [19] M. Klesel and H. F. Wittmann, “Retrieval-Augmented Generation (RAG),” *Business & Information Systems Engineering*, vol. 67, no. 4, pp. 551–561, 2025. DOI: 10.1007/s12599-025-00945-3.

Evaluating a Decision Support System for Assessing the Product Design Disassemblability

Antonia Markus^{1*}, Julian Haller¹, Esther Borowski¹, Amon Göppert¹, Robert H. Schmitt^{1,2}

¹RWTH Aachen University, Laboratory for Machine Tools and Production Engineering WZL, 52074 Aachen Germany

²Fraunhofer Institute for Production Engineering IPT, 52074 Aachen Germany

*Corresponding author. E-Mail address: antonia.markus@wzl-iqs.rwth-aachen.de; Tel.: +49241 80 91117

Abstract: In response to the growing need for sustainable practices and resource efficiency, this study explores the integration of production knowledge into product design to enhance disassembly processes, which are crucial for effective remanufacturing and refurbishing. Despite the theoretical framework of Design for Remanufacturing, practical application is limited due to inadequate collaboration between product designers and production experts. Through semi-structured interviews with 14 professionals from various industries, we identify key challenges in this integration and outline user requirements for a proposed hybrid decision support system (DSS). Findings indicate that usability, compatibility with existing processes, and transparency are critical system requirements. Stakeholders express interest in the DSS concept; however, skepticism regarding broader remanufacturing strategies persists. The results emphasize the need to address general prerequisites, such as digitization levels within companies, to ensure successful implementation. This research contributes valuable insights into bridging theoretical frameworks with practical applications in remanufacturing and flexible assembly systems.

Keywords:

Flexible Assembly Systems, Disassembly, Remanufacturing, Human-Centered Design,

1 Introduction

To reduce resource waste and transform into a circular economy, R-strategies are getting more relevant within product life cycle management. [1] presented a framework of ten R-strategies, namely: Refuse, Rethink, Reduce, Reuse, Repair, Refurbish, Remanufacture, Repurpose, Recycle, and Recover. Not only idealistic motives, but also potential economic benefits are drivers for the application of R-strategies [2]. For instance, in the automotive industry, remanufactured parts can save up to 93 % in CO₂ emissions and 88 % of raw material [3]. However, the products themselves, as well as their production systems, must be adaptable to unlock the full potential of R-strategies. Focusing on the close-to-production strategies of remanufacturing and refurbishing, used products must be partly disassembled to be reassembled with only a few replaced or refurbished parts. However, the quantity and quality of returned products are unknown [4], and it is thus a remaining challenge to determine how disassembly and reassembly are facilitated efficiently. Whether disassembly should be integrated within the existing production system or established as a separate production system relies heavily on product design. Although the theoretical principle of Design for Remanufacturing (DfR) is broadly known, it is rarely used in practice [3]. The collaboration and communication between production experts and product designers regarding remanufacturing is not prioritized by companies, even though the production insights might support targeted design decisions [5]. Given the relevance of this issue, there is a need for research into the obstacles faced by product designers and production experts and the support they

require. Identifying these requirements is the foundation for integrating a future assistance system to take informed decisions on interdependent product- and production system design.

Consequently, this article aims to answer the following research questions:

- How do design engineers and production planners assess a decision support system that is based on the concepts of remanufacturing and flexible disassembly system?
- What requirements do design engineers and production planners have for the decision support system?

The publication is structured as follows: Section 2 provides an overview of the related works in the field of decision support systems for remanufacturing at the interface of product design and production planning. Section 3 presents the methodology for this publication, namely the human-centered design approach for a decision support system by means of expert interviews. The results of these interviews were analyzed qualitatively according to Mayring [6] and are presented in Section 4. We discuss our findings and implied consequences in Section 5 and conclude with an outlook for future research opportunities in Section 6.

2 Related Works

Practical guidelines for design for disassembly have been around for more than three decades, for example, Boothroyd and Alting's standard work *Design for assembly and disassembly* [7]. [8] established remanufacturing as a distinct strategy alongside repair, recycling, and disposal, and emphasized the role of product architecture and disassembly effort in determining the economic viability of recovery options. Building on this foundation, surveys on sustainable manufacturing and product recovery synthesized the state of the art on Design for Disassembly (DfD), disassembly planning, and recovery system design [9], highlighting the need for considering the interdependency of product design and production planning. Ijomah and colleagues [10] developed design guidelines specifically for remanufacturing, focusing on issues such as fastening methods, ease of access to critical components, and minimization of damage during disassembly and reassembly. However, many guideline-based approaches remain qualitative or semi-quantitative, and only a subset is linked directly to dynamic performance indicators such as throughput or resource utilization.

Evaluating product disassemblability predominantly utilizes absolute metrics, specifically disassembly time, alongside relative indices such as design efficiency and recyclability rates [11], [12]. Foundational methodologies employ predetermined motion-time systems, notably Methods-Time Measurement (MTM) and the Maynard Operation Sequence Technique (MOST), to estimate disassembly effort based on elementary manual operations, positioning, and force requirements [13], [14], [15]. These time-based estimates often feed into broader scoring systems, such as the ease of Disassembly Metric (eDiM) and the Repair Scoring System (RSS), which aggregate diverse criteria, including tool availability, fastener reversibility, and accessibility into quantitative indices [16] (Cordella et al., 2019). Recent advancements integrate these metrics into computer-aided design (CAD) tools and visual representations, such as Disassembly Maps and the Parent-Action-Child model, to optimize disassembly sequences and identify architectural bottlenecks [17], [18], [19].

Despite these advancements, current methodologies largely rely on the assumption of products being in perfect condition, failing to model real-world End-of-Life (EoL) statuses—such as rust, wear, deformation, or fusing—which alter disassembly effort and circularity potential [19], [20]. Furthermore, existing databases for fastener taxonomies are not exhaustive, often necessitating manual modeling for

novel connection types, and the standardization of reference values (e.g., disassembly depth) remains subjective, limiting the robustness of comparisons across different product families [21], [22].

Generating disassembly sequences directly from CAD files attempts to fill that gap. There is a plethora of publications about generating or optimizing disassembly and assembly sequences, yet only a few authors disclose the source code of their approaches. In [23], we performed a systematic literature review about disassembly sequence automation with publicly available code. Geometric approaches utilize collision detection to extract topological constraints and generate AND/OR graphs [24], [25]. Physics-based methods leverage simulation to ensure collision-free paths and gravitational stability during assembly-by-disassembly processes [26], [27]. Concurrently, data-driven approaches employ Graph Neural Networks and point cloud analysis for kinematic constraint prediction [28], [29], next component recommendation [30], and part direction prediction [31].

To gain insight into the real-world performance of a design for remanufacturing, simulations are necessary. Recent work specifically targets flexible process sequencing; Wurster and colleagues [32] employ Reinforcement Learning to enable dynamic, condition-based routing in hybrid disassembly systems where sequences are not fixed in advance. Innovation extends to process control via semantic 3D modeling for automated inspection [33] and factory planning for Matrix Manufacturing Systems, where metaheuristics optimize layout flexibility [34]. However, a holistic simulation framework that simultaneously integrates semantic product perception, dynamic layout optimization, and adaptive sequence control to fully automate the generation of simulation models remains a research gap. In [35], we proposed a framework to extract all feasible (dis-)assembly sequences from CAD files and simulate them on different (dis-)assembly scenarios to receive feedback and iterate for a better design for remanufacturing, as shown in Figure 1. To ensure that the proposed framework is aligned with real-world needs, two questions shaped the approach: first, how practitioners evaluate the proposed framework, and second, what specific requirements they have for such a hybrid decision-making system.

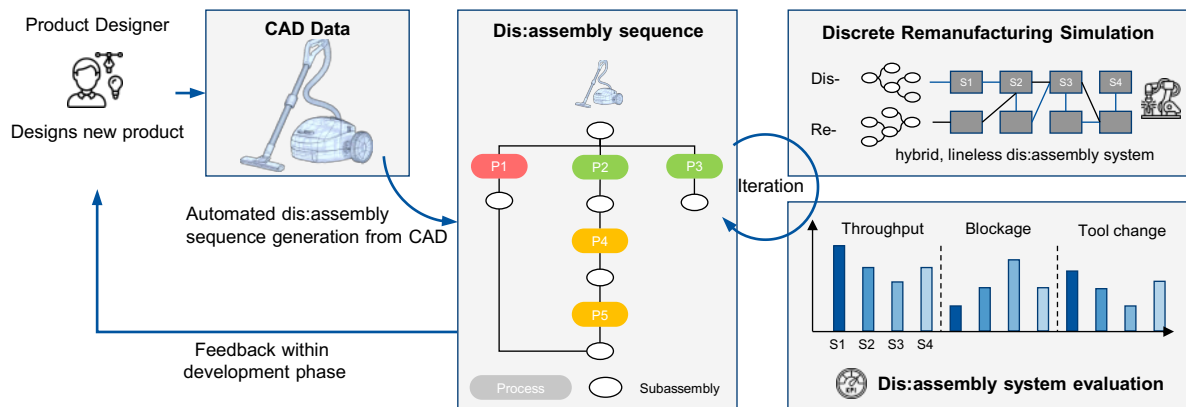


Figure 1: Proposed framework for production-based remanufacturing assessment according to [35].

3 Methodology

Participants were recruited via email from companies that are potentially involved in remanufacturing. Each interview included a pre-interview to assess participant suitability. The inclusion criterion was that participants should be able to envision a future where their product might be remanufactured. This criterion led to the exclusion of many potential interview partners, as their companies were not involved or interested in remanufacturing. The final sample consisted of 14 interviewees: 6 from a development perspective and 8 from a production perspective, representing various business sectors, including automotive (suppliers), plumbing, heating and air conditioning, control systems engineering, household appliances, and others. Their experience levels ranged from five to 25 years within their respective

fields. The number of expert interviews conducted is based on the qualitative research principle of data saturation, whereby data collection is terminated as soon as no new significant topics emerge. Empirical studies show that in many qualitative studies, this point is typically reached after between 9 and 17 interviews [36]. We reached saturation after 14 interviewed experts.

Structured interview guidelines were used during the interviews, which included a presentation of our system concept using Figure 1 as a PowerPoint slide. The 23 questions were divided into four blocks: context of use, remanufacturing, flexible assembly and disassembly, and system assessment. The guidelines and visualization were tested for their understandability with a pre-test by a development engineer who was not part of the sample. The interviews were conducted in German and lasted between 45 and 60 minutes, carried out by an engineer and a psychologist. All interviews were audio recorded, transcribed, and analyzed using structured content analysis according to [6]. The analysis was facilitated by MAXQDA, with the coding scheme guided by categories such as user group, requirements, system benefits, and criticisms. These categories were derived from specific questions in the interview guidelines. Additionally, during the analysis process, the category "general prerequisites" emerged as an important aspect alongside system requirements.

4 Results

An overview of the different codes within five categories, along with their frequency of occurrence in the transcripts of the 14 interviews conducted, is displayed in Table 1. Each category is described below, accompanied by relevant quotes.

In the category "user group," seven participants (50%) identified development engineers as the main user group for the proposed system, while four participants (29%) considered production planners to be the primary users. Additionally, six participants (43%) suggested that both the development and production planning departments should collaborate with the system as a team. Other potential user groups mentioned included project leads and product management.

The category "system requirements" was primarily dominated by usability (11 counts, 79%). Within this code group, six participants specifically referred to usability or stated that "it must be as easy to use as possible," emphasizing ease of handling. Other aspects coded under this requirement included efficiency, flexibility, and robustness against errors. Fifty percent of interviewees (7 counts) highlighted the importance of compatibility with the proposed system, mentioning technological compatibility with other tools and file formats, as well as procedural compatibility with existing development processes. Three participants (21%) identified transparency as a key requirement, noting that "you have to find some way to break it down [...] everyone will still demand understanding; I can't imagine it any other way." Additional discussed requirements included data security, demands for a learning system, and aesthetic standards.

When asked about potential benefits of the system, three codes emerged most frequently—each mentioned four times (29%). First was predictive power: interviewees noted that the system allows them to estimate key parameters in advance; one participant remarked, "I can make assumptions in advance about assembly times, which gives me transparency for calculation." The second benefit identified was that the tool supports users amid increasing product complexity, influencing "not only product development but also the whole life cycle." Thirdly, it was noted that the tool would facilitate early consideration of disassemblability during development. Two additional codes were mentioned by two participants each (14%), indicating that the tool could aid collaboration between development and production planning while facilitating R-strategy implementation.

Participants also expressed concerns regarding potential drawbacks of the system. Some felt it might highlight conflicts between objectives faced by development engineers in their daily work (3 counts,

21%), primarily driven by economic considerations such as promised product lifespan and competitive cost structures. Individual criticisms included a lack of focus on refurbishing processes within the concept and concerns about whether the system would function effectively as presented.

The final category—“general prerequisites”—emerged during content analysis rather than being directly addressed in interview questions. This category summarizes conditions within a company that could positively impact implementation but are not direct requirements for using the system. The most frequently cited prerequisite was advanced digitization within companies and their processes (6 counts, 43%). For instance, having a well-maintained database and established standards for managing relevant data would enhance the effectiveness of the tool. Additionally, having products with sufficient complexity was noted as beneficial for effective use of the system (3 counts, 21%). Participants also emphasized the importance of awareness regarding remanufacturing among users as a prerequisite. Finally, two other codes appeared twice each (14%): management support for using the system and strategic alignment towards remanufacturing alongside prioritization and time availability for utilizing it.

Table 1: Overview of the three most frequent codes for each category

Category	Most frequent code		2nd most frequent codes		3rd most frequent codes	
User group	Development engineers	7	Development and production as a team	6	Production planner	4
System requirements	Usability	11	Compatibility with other tools and processes	7	Transparency	3
Benefits	- Predictive power - Support with increasing complexity - Early consideration of (dis-)assemblability	4	- Collaboration between development and production planning - Facilitating implementation of R-strategies	2		
Criticisms	- Emphasizing conflict of objectives	3	- Refurbish process is not considered - Questioning of systems' functionality/performance	1		
General prerequisites	Advanced digitization/infrastructure/interfaces	6	- Awareness - Sufficient complexity of the product	3	- Management specification - Prioritization and time management	2

5 Discussion

This paper evaluates the practical perspectives of design engineers and production planners regarding a hybrid decision support system (DSS) concept aimed at assessing the disassemblability of products during early development stages. We conducted semi-structured interviews in which we presented the system concept and solicited participants' opinions on user groups, system requirements, potential benefits, and criticisms. The results indicate an overall interest from stakeholders; however, there was no consensus on whether development engineers should be considered the primary user group. Some participants suggested that collaboration between development and production teams is essential, while others noted that development engineers often overlook production perspectives.

Regarding system requirements, the anticipated outcome was confirmed, with a strong emphasis on usability. Participants highlighted not only ease of use but also other facets such as efficiency, robustness against errors, and flexibility—elements defined within ISO standard 9241 [37]. A comparison of benefits and criticisms revealed sixteen positive statements versus five negative ones from the sample.

This suggests a favorable view of the system concept, particularly its early consideration of disassemblability and predictive capabilities.

However, it can be argued that participants may find it challenging to criticize such a high-level concept. Additionally, the fifth identified category—general prerequisites—raises questions about relevance for specific industries. The requirement for sufficient product complexity could imply that the system may not be applicable to all companies. Furthermore, concerns regarding digitization highlight the need for good interfaces and data quality—factors that are often difficult to change.

Nonetheless, including general prerequisites in our analysis proved valuable as it illuminated aspects related to company environments rather than just system design. These arguments can be viewed as conditions for successful implementation that must be fulfilled before deploying the DSS.

Overall, the interviews provided insightful perspectives aligned with human-centered design principles—a focus that is often overlooked in practice. It is important to note that while participants did not reject the proposal of the DSS itself, many expressed skepticism towards broader approaches to remanufacturing and flexible matrix assembly upon which it relies. This skepticism may stem from these concepts being relatively unestablished within their industries and thus perceived as highly visionary.

During our recruitment process, we sought companies engaged in either remanufacturing or flexible assembly; however, only three participants reported involvement in remanufacturing practices while two indicated engagement in flexible/matrix assembly—none were involved in both areas. As previously mentioned in our interpretation of codes and categories, not every industry stands to benefit from our system due to varying relevancies tied to product types and sector characteristics.

From a methodological perspective, skepticism toward remanufacturing and flexible assembly concepts could potentially be mitigated by presenting concrete proposals during interviews; tangible topics may resonate more effectively with interviewees. We recommend conducting pre-interviews to assess participant suitability since this might prepare them for what to expect during discussions about remanufacturing and flexible assembly concepts.

Some methodological limitations include a sample that would benefit from including more companies already implementing one of these two main concepts, since many responses were speculative about future possibilities. Additionally, adherence to interview guidelines varied across sessions; consequently, some interviews took on an exploratory nature rather than strictly following established protocols.

6 Conclusion and Outlook

This paper contributes to the discourse on how remanufacturing and flexible assembly can be effectively implemented in practice by utilizing a hybrid decision-making system, highlighting both the potential benefits and relevant challenges. Our methodological approach extends existing literature by providing a practical perspective that has often been neglected, thereby balancing theoretical considerations in previous debates. Future research could focus on specific industries to achieve a more homogeneous sample, allowing for tailored conclusions applicable to different sectors. Additionally, researchers should prioritize involving affected stakeholders early in the development process of decision support systems and maintain collaboration throughout system development.

References

- [1] J. Potting, M. Hekkert, E. Worrell, and A. Hanemaaijer, "Circular Economy: Measuring innovation in product chains," PBL Netherlands Environmental Assessment Agency, The Hague, 2016.
- [2] T. Sakao, N. Bocken, N. Nasr, and Y. Umeda, "Implementing circular economy activities in manufacturing for environmental sustainability," *CIRP Annals*, vol. 73, no. 2, pp. 457–481, 2024, doi: 10.1016/j.cirp.2024.06.002.
- [3] D. Parker et al., "Remanufacturing Market Study," For Horizon 2020, grant agreement No 645984, Oct. 2015.
- [4] R. H. Teunter and S. D. P. Flapper, "Optimal core acquisition and remanufacturing policies under uncertain core quality fractions," *European Journal of Operational Research*, vol. 210, no. 2, pp. 241–248, Apr. 2011, doi: 10.1016/j.ejor.2010.06.015.
- [5] S. Wehrend, "Product production ecosystems for the reuse of modules in the extended product life cycle," in *DS 133: Proceedings of the 35th Symposium Design for X (DFX2024)*, The Design Society, 2024, pp. 153–162. doi: 10.35199/dfx2024.16.
- [6] P. Mayring, *Qualitative Inhaltsanalyse: Grundlagen und Techniken*, 13., Überarbeitete Auflage. Weinheim Basel: Beltz, 2022.
- [7] G. Boothroyd and L. Alting, "Design for assembly and disassembly," *CIRP annals*, vol. 41, no. 2, pp. 625–636, 1992.
- [8] M. Thierry, M. Salomon, J. Van Nunen, and L. Van Wassenhove, "Strategic Issues in Product Recovery Management," *California Management Review*, vol. 37, no. 2, pp. 114–136, Jan. 1995, doi: 10.2307/41165792.
- [9] M. A. Ilgin and S. M. Gupta, "Environmentally conscious manufacturing and product recovery (ECMPRO): A review of the state of the art," *J Environ Manage*, vol. 91, no. 3, pp. 563–91, Jan. 2010, doi: 10.1016/j.jenvman.2009.09.037.
- [10] W. L. Ijomah, C. A. McMahon, G. P. Hammond, and S. T. Newman, "Development of design for remanufacturing guidelines to support sustainable manufacturing," *Robotics and Computer-Integrated Manufacturing*, vol. 23, no. 6, pp. 712–719, Dec. 2007, doi: 10.1016/j.rcim.2007.02.017.
- [11] I. Belhadj, M. A. Ben Abdallah, M. Tlija, and N. Aifaoui, "A mathematical formulation for assessing disassembly plans based on energy consumption, disassembly time and recyclability for a sustainable design," *Int J Interact Des Manuf*, Feb. 2025, doi: 10.1007/s12008-025-02243-4.
- [12] C. B. Boks, E. Kroll, W. C. J. Brouwers, and A. L. N. Stevels, "Disassembly modeling: two applications to a Philips 21" television set," in *Proceedings of the 1996 IEEE International Symposium on Electronics and the Environment. ISEE-1996*, IEEE, 1996, pp. 224–229. doi: 10.1109/ISEE.1996.501882.
- [13] A. Desai and A. Mital, "Evaluation of disassemblability to enable design for disassembly in mass production," *International Journal of Industrial Ergonomics*, vol. 32, no. 4, pp. 265–281, Jan. 2003, doi: 10.1016/S0169-8141(03)00067-2.
- [14] E. Kroll, B. Beardsley, and A. Parulian, "A Methodology to Evaluate Ease of Disassembly for Product Recycling," *IIE Transactions*, vol. 28, no. 10, pp. 837–846, Jan. 1996, doi: 10.1080/15458830.1996.11770736.
- [15] P. Vanegas et al., "Ease of disassembly of products to support circular economy strategies," *Resour Conserv Recycl*, vol. 135, pp. 323–334, Jan. 2018, doi: 10.1016/j.resconrec.2017.06.022.
- [16] E. Bracquené et al., "Analysis of evaluation systems for product repairability: A case study for washing machines," *Journal of Cleaner Production*, vol. 281, p. 125122, Jan. 2021, doi: 10.1016/j.jclepro.2020.125122.
- [17] F. De Fazio, C. Bakker, B. Flipsen, and Ruud Balkenende, "The Disassembly Map: A new method to enhance design for product repairability," *Journal of Cleaner Production*, vol. 320, p. 128552, Jan. 2021, doi: 10.1016/j.jclepro.2021.128552.
- [18] C. Favi, M. Marconi, M. Germani, and M. Mandolini, "A design for disassembly tool oriented to mechatronic product de-manufacturing and recycling," *Advanced Engineering Informatics*, vol. 39, pp. 62–79, Jan. 2019, doi: 10.1016/j.aei.2018.11.008.
- [19] G. Formentini and D. Ramanujan, "Design for circular disassembly: Evaluating the impacts of product end-of-life status on circularity through the parent-action-child model," *Journal of Cleaner Production*, vol. 405, p. 137009, Jan. 2023, doi: 10.1016/j.jclepro.2023.137009.
- [20] J. G. Erdmann, J. Koller, J. Brimaire, and F. Döpper, "Assessment of the disassemblability of electric bicycle motors for remanufacturing," *Jnl Remanufactur*, vol. 13, no. 2, pp. 137–159, Jan. 2023, doi: 10.1007/s13243-023-00124-1.
- [21] N. B. Rodríguez, C. Gabriel, R. Gaha, and C. Favi, "Analysis of disassembly parameters in repairability scores: limitations for engineering design and suggestions for improvement," *Procedia CIRP*, vol. 116, pp. 738–743, Jan. 2023, doi: 10.1016/j.procir.2023.02.124.
- [22] R. Sodhi, M. Sonnenberg, and S. Das, "Evaluating the unfastening effort in design for disassembly and serviceability," *Journal of Engineering Design*, vol. 15, no. 1, pp. 69–90, Jan. 2004, doi: 10.1080/0954482031000150152.
- [23] J. Haller, J. Liang, A. Göppert, and R. H. Schmitt, "Disassembly sequence automation for discrete event simulation of flexible, hybrid remanufacturing systems," presented at the 36th CIRP Design Conference, Tokyo, Japan, Mar. 2026.
- [24] M. Cramer, K. Kellens, and E. Demeester, "APLAN: open assembly planning framework in FreeCAD," *Procedia CIRP*, vol. 127, pp. 56–61, 2024, doi: 10.1016/j.procir.2024.07.011.

-
- [25] S. Munker and R. H. Schmitt, "CAD-based AND/OR Graph Generation Algorithms in (Dis)assembly Sequence Planning of Complex Products," *Procedia CIRP*, vol. 106, pp. 144–149, Jan. 2022, doi: 10.1016/j.procir.2022.02.169.
 - [26] Y. Tian et al., "Assemble Them All," *ACM Trans. Graph.*, vol. 41, no. 6, pp. 1–11, Jan. 2022, doi: 10.1145/3550454.3555525.
 - [27] Y. Tian et al., "ASAP: Automated Sequence Planning for Complex Robotic Assembly with Physical Feasibility," 2023, *arXiv*. doi: 10.48550/ARXIV.2309.16909.
 - [28] L. Vergez, A. Polette, and J.-P. Pernot, "Multi-part kinematic constraint prediction for automatic generation of CAD model assemblies using graph convolutional networks," *Computer-Aided Design*, vol. 178, p. 103805, Jan. 2025, doi: 10.1016/j.cad.2024.103805.
 - [29] K. D. D. Willis et al., "JoinABLE: Learning Bottom-up Assembly of Parametric CAD Joints," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, New Orleans, LA, USA: IEEE, Jun. 2022, pp. 15828–15839. doi: 10.1109/CVPR52688.2022.01539.
 - [30] F. Liang, H. Zhao, Y. Quan, W. Fang, and C. Shi, "Customizing Graph Neural Network for CAD Assembly Recommendation," in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, Barcelona Spain: ACM, Aug. 2024, pp. 1746–1757. doi: 10.1145/3637528.3671788.
 - [31] J. Gaarsdal, J. B. Haurum, S. Wolff, and C. B. Madsen, "AssemblyNet: A Point Cloud Dataset and Benchmark for Predicting Part Directions in an Exploded Layout," in *2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, Waikoloa, HI, USA: IEEE, Jan. 2024, pp. 5824–5833. doi: 10.1109/WACV57701.2024.00573.
 - [32] M. Wurster, M. Michel, M. C. May, A. Kuhnle, N. Stricker, and G. Lanza, "Modelling and condition-based control of a flexible and hybrid disassembly system with manual and autonomous workstations using reinforcement learning," *J Intell Manuf*, vol. 33, no. 2, pp. 575–591, Jan. 2022, doi: 10.1007/s10845-021-01863-3.
 - [33] J.-P. Kaiser, D. Koch, F. Stamer, J. Peeters, and G. Lanza, "Semantic 3D product modelling for automated inspection in remanufacturing processes," *Jnl Remanufactur*, vol. 16, no. 1, p. 1, Apr. 2026, doi: 10.1007/s13243-025-00157-8.
 - [34] J. Park, C. Lee, S.-C. Oh, and S. Do Noh, "Multi-objective optimization for layout planning of matrix manufacturing system," *Int. J. of Precis. Eng. and Manuf.-Green Tech.*, vol. 12, no. 3, pp. 1049–1067, May 2025, doi: 10.1007/s40684-025-00696-2.
 - [35] J. Haller, J. Baum, A. Göppert, and R. H. Schmitt, "Remanufacturing in flexiblen (De-)Montagesystemen: Entscheidungsunterstützung für kreislauffähiges Produktdesign," *Zeitschrift für wirtschaftlichen Fabrikbetrieb*, vol. 120, no. 3, pp. 151–154, Mar. 2025, doi: 10.1515/zwf-2025-1027.
 - [36] M. Hennink and B. N. Kaiser, "Sample sizes for saturation in qualitative research: A systematic review of empirical tests," *Social Science & Medicine*, vol. 292, p. 114523, Jan. 2022, doi: 10.1016/j.socscimed.2021.114523.
 - [37] Deutsches Institut für Normung e.V., *DIN EN ISO 9241-11): Ergonomie der Mensch-System-Interaktion: Gebrauchtauglichkeit: Begriffe und Konzepte Teil 11*, Berlin., 2017.

Identification of key parameters for automated assembly

Malte Jakschik^{1*}, Julian Rolf², Kian Bahri¹, Bernd Kuhlenkötter¹

¹Ruhr University Bochum, Chair of Production Systems, 44894 Bochum

²Ruhr University Bochum, Chair of Digital Engineering, 44894 Bochum

*Corresponding author. E-Mail address: jakschik@lps.ruhr-uni-bochum.de; Tel.: 02324 32 24611

Abstract:

Demographic change and the resulting shortage of skilled labor increase the need for automation in industrial production systems. Since decisions influencing automation suitability are fixed mainly in early product development phases, reliable assessment methods are required even when product and process information is incomplete. Established Design for Automation and Automated Assembly (DFA/DFAA) methods provide valuable expert knowledge but are often too information-intensive for early-stage applications. This paper presents a hybrid intelligence approach that combines expert-based DFA/DFAA evaluation methods with machine learning-based surrogate models to support early assessment of automation suitability. First, a systematic literature review is conducted to identify key product-related criteria influencing automated assembly. Based on these criteria, a synthetic dataset of standardized product descriptions is generated and labeled using a validated DFA evaluation algorithm derived from established methods. Subsequently, a graph isomorphism network is trained to predict the Automation Suitability Degree directly from graph-based product representations.

Validation results demonstrate high predictive accuracy and strong correlation with expert-based evaluations, indicating that the model reliably captures relative suitability trends even for partially specified products. While absolute errors increase for previously unseen product variants, the approach shows strong potential as a scalable decision-support tool.

Keywords:

Design for Automated Assembly, design decision support, Machine Learning Surrogates, Early Product Development, Automation Suitability Assessment

1 Introduction

Demographic change and the resulting shortage of skilled labor pose significant economic and structural challenges for industrialized high-wage countries. The declining availability of qualified workers already causes substantial economic losses and is expected to intensify with the retirement of the baby-boomer generation [1]. To remain competitive, companies increasingly rely on digitalization and automation. Beyond reducing physical workload, technological support can improve efficiency throughout product development and production [15]. Adapting product design to enhance automation suitability can significantly reduce assembly effort and production time. Since up to 70 % of total product costs are determined in early development phases, automation-related decisions should be made as early as possible [2]. However, in these phases, essential process information is often incomplete, making reliable assessment of automation potential difficult. Consequently, digital engineering tools must provide adequate decision support under limited information.

Recent advances in computing have enabled the use of Machine Learning-based Surrogate (MLS) models to complement or partially replace computationally expensive simulations in engineering design. MLS models allow efficient exploration of large design spaces and can support early decision-making with reduced computational effort. In this context, this study focuses on Design for Automation and Automated Assembly (DFA/DFAA) and investigates which product characteristics determine the suitability for automated assembly.

This work addresses two research questions (RQ):

RQ1: *Which product parameters determine automation suitability in early product development?*

RQ2: *To what extent can a graph-based ML surrogate model reliably approximate expert-based automation suitability evaluations??*

From a hybrid intelligence perspective, the proposed approach combines expert knowledge embedded in established DFA/DFAA methods with data-driven learning mechanisms. While evaluation criteria and labeling logic are derived from human expert models, MLS models generalize this knowledge across large and partially incomplete design spaces. The system is intended to support rather than replace human decision-making by providing early, consistent, and explainable feedback on the suitability of automated assembly.

This article is structured as follows: First, relevant DFA/DFAA criteria are identified through a systematic review of the state of the art. Subsequently, the generation and labeling of a synthetic dataset are described and validated. Finally, a graph isomorphism network (GIN) is used to predict automation suitability, and the results are discussed and summarized.

2 Identification of key parameters in literature

Various methods exist to evaluate the design for automated assembly. In literature, the most established methods are the Boothroyd-Dewhurst (B&D) method, the Lucas method, and the Hitachi-Assembly Evaluation Method (Hitachi-AEM) [3–7]. All these methods were developed in the 1990s; therefore, it is appropriate to review the factors and key criteria to ensure they remain as relevant today as they were then and to address RQ1. A literature review was conducted to identify the key drivers by analyzing the criteria used for the DFA and DFAA approaches. The non-sharp separation of the focal points in mind, the relevant contributions can be categorized into three groups: those focusing on manual assembly [8–22], those focusing on automated assembly [5,23–30], and those that do not explicitly differentiate between the two [7,31–36]. The established methods mentioned above each have a different focus: For example, the B&D method distinguishes between manual, automated, and robot-assisted assembly. The Lucas method also distinguishes between manual and automated assembly. The Hitachi AEM does not explicitly differentiate between the assembly approaches, but it sees manual assembly as a fundamental prerequisite for automated assembly [6]. Since the criteria for the DFA and DFAA assessments overlap considerably and there is no clear distinction between the two design methods in the literature, both approaches were considered in this research [37]. Figure 1 shows the distribution of all relevant criteria, where the primary y-axis shows the overall number of uses, and the secondary y-axis shows the quantitative distribution of the criteria relative to the established method. A total of 48 distinct criteria were identified in the contributions, although it was often not possible to distinguish among them clearly. For example, the evaluation of the working space was addressed separately in some contributions, whereas in others it was covered under the criterion of accessibility. However, it should be noted that, due to the clear correspondence between the established methods' criteria and the state of the art, the criteria of the existing established methods remain the most relevant.

Given the need for early feedback on the suitability of automated assembly, it makes sense to analyze the distribution of criteria relative to the methodical application period. The development process is

typically divided into four stages: task clarification, conceptualization, design, and elaboration [38]. The approaches presented in [10,21,23–27,32,33,36,39] are applied in the conceptual phase of product development and require relatively low information content regarding the product to be developed. The methods used in the design phase are described in [18–20,35,40,41]. In [8,9,30], the product is already on the market, resulting in the highest information density. In contrast, the methods in [5,28,29] are designed for application across all phases of product development. To evaluate the product in the elaboration stage, the authors in [7,11–17,28,31,34,42,43] present applicable approaches. The established methods can also be assigned to these categories, with the Lucas method to the concept phase, the Hitachi AEM to the design phase, and the B&D method to the elaboration phase [5]. This distribution is also shown in Figure 1. Furthermore, the 18 most frequently used criteria are applied across all stages of the development process. These, in turn, are also consistent with established methods, thus once again demonstrating the relevance of these methods and criteria.

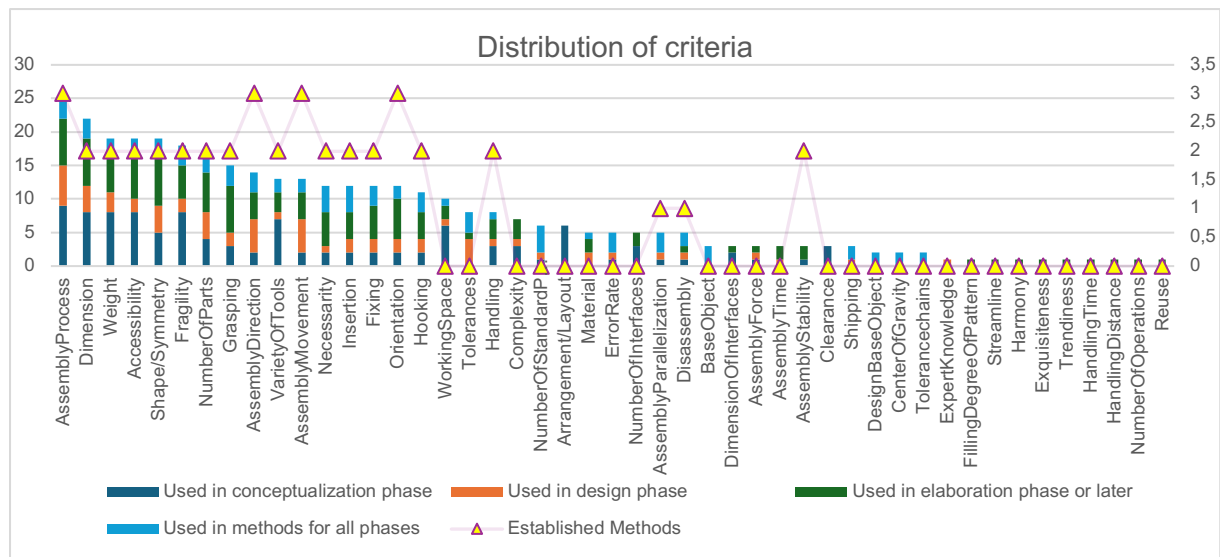


Figure 1: Quantitative distribution of DFA & DFAA evaluation criteria

In addition to the criteria used, the RQ2 should also be examined in the literature to investigate the use of ML in these articles. Similar to the research in [3], this review also concludes that there is a research gap in AI-supported DFA assessment, as only a few recent contributions in this area could be identified. Krahe streamlines the product development process by integrating a knowledge base of comparable products. Using this foundation, an AI analyzes a semi-finished product and predicts the next construction step, drawing on past products that have proven highly suitable for manufacturing. [44] The article by Suszynski et al. describes an artificial neural network (ANN) that predicts assembly times for different sequences based on selected DFA metrics. The system serves as an intelligent predictor in an automated assembly planning environment, and its results demonstrate that DFA methods can be effectively combined with ML models. The DFA key factors used here are based on the B&D evaluation method. However, the goal of the optimization presented in this paper is not to evaluate the design, but rather to optimize and predict the time of a given assembly sequence. [15] Jing et al. use a graph neural network (GNN) to generate explainable recommendations for manufacturing knowledge that can be used in early development phases. The aim is to reduce iterative optimization loops in the engineering process and expand cognitive capabilities regarding manufacturability and assembly constraints. They conclude that integrating GNNs into manufacturing knowledge provides a scalable and robust basis for knowledge-based support systems. [42] Jakschik et al. present a product evaluation algorithm based on fuzzy logic and a product description in an AutomationML (AML) file structure, which enables the evaluation of various uncertain or subjective

metrics. The contribution describes a conceptual approach to generating a comprehensive dataset using this algorithm to predict the suitability of future automatic assembly. The approach has not yet been definitively validated, but has already undergone various plausibility tests and is delivering promising results. The article serves as the basis for the approach described here and details the textual product description using AML and the evaluation algorithm. [28]

In their approach, Ereemeev et al. develop a framework for performance and assembly optimization in product development. This is based on a fully parameterized CAD model, which is subjected to various CAE simulations for the optimization approach. In addition, a DFA evaluation of the design instance is performed based on the number of standard parts used, and the assembly space for the tool is evaluated. In the framework presented, an ANN and a Gaussian process are used to abstract from and efficiently perform computationally expensive simulations, thereby enabling multidisciplinary optimization. Although this article also presents promising results, the assessment of automated assembly's suitability with respect to the identified key factors remains underrepresented and should be enriched with additional criteria. [11]

In conclusion, the literature review shows that established methods still capture the key factors for high suitability for automated assembly, given the high degree of overlap with the state of the art, and that ML delivers promising results for predicting suitability but requires further investigation.

3 Problem solving approach

The proposed workflow follows a hybrid intelligence paradigm: expert-based evaluation methods generate high-quality labels, while a graph-based machine learning model learns to efficiently approximate these evaluations. This allows expert knowledge to be reused and applied in early development phases, where manual DFA assessments would be infeasible due to limited information or time constraints. The concept was implemented using an instrument panel to control process engineering processes. Synthetically reproduced product descriptions are used to generate a data set of 500 product descriptions, which is then evaluated using a DFA evaluation algorithm. This was implemented in a setup comparable to that described in [28] and evaluates products using the established methods of B&D, Lucas, and Hitachi AEM. Due to the application period described (application in all phases of product development) and the high degree of compatibility, the evaluation method according to [29] was also taken into account. In his approach, Eskilander has already attempted to combine the established methods and develop a method that can be adapted across all development phases. The focus of the developed method is on the objectification of the evaluation criteria. Before labeling the data set, the evaluation algorithm was validated using these methods. Using the fuzzy sets underlying the evaluation algorithm, the membership functions were adjusted so that the evaluation algorithm's output corresponded to that of the established methods. The labeled data set is used to train a GNN. Validation of the ML approach confirms the relevance of the selected evaluation criteria. Below is a detailed description of each topic.

3.1 Dataset generation

To ensure a dataset of sufficient size for effective training of machine learning models, a synthetic data generation approach was used. The synthetic dataset does not represent arbitrary abstract assemblies; rather, it is derived from a realistic product class: modular instrument panels used in process engineering plants, as shown on the right side of Figure 2. The component library was created from typical industrial panel layouts and standardized connection principles (e.g., pipes, valves, sensors, mounting plates). Synthetic samples, therefore, represent structurally mountable but not company-specific product variants. The method involves rule-based combinations of components to generate a

wide range of assembly scenarios, and algorithmic replication of product descriptions to systematically expand the dataset.

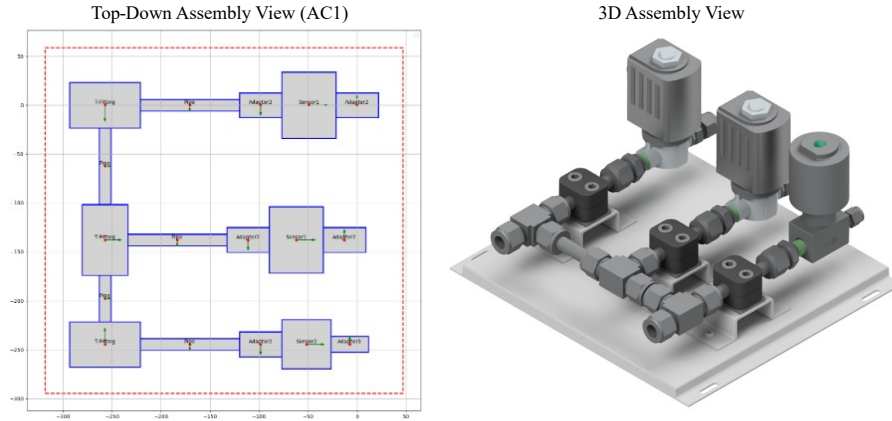


Figure 2: Visualization of the product class under consideration for DFA/DFAA Evaluation

For the rule-based combination, a parts library was created that describes a total of 27 different components of an instrument panel. Each of these components is enclosed by six enclosures, with each two-dimensional surface being specified by additional features such as cutouts and contact surfaces. A connection matrix was then used to determine which side of component A could be connected to other components N. This resulted in a side-by-side description of each component.

The algorithm selects a random starting component and then the permissible connection components for the next step. This process is repeated until the specified number of pipe components is reached. The components are then positioned according to the contact surfaces in the plane, and the instrument panel's overall dimensions are calculated. A random selection of connecting components then ensures the mechanical connection between the piping system and the mounting plate. However, only one connection type is selected per panel. For dataset creation, the number of components per product was set to n , with the range $n \in [5, 30]$. At this step, each product instance is represented as a hierarchical product description in CAEX-compliant XML.

3.2 Labeling methodology

This approach uses supervised learning to predict the suitability of automated assembly, so the dataset must be labeled accordingly. A dedicated evaluation model, aligned with CAEX-compliant XML product descriptions and inspired by the approach in [28], was developed. Adjustments to the membership functions within the knowledge base ensured close alignment of results with existing evaluation methods. Based on the identified key factors and the existing relevance of the established methods, all criteria used by at least 20 % of the contributions and by more than one established method were considered in the evaluation algorithm. This resulted in the first 20 criteria from Figure 1 and the criterion *assembly stability*.

To validate the evaluation algorithm, 35 sample products were evaluated using the established Lucas and B&D methods as described in [45], as well as Eskilander, and the results were compared with those of the evaluation algorithm. Due to limited access to the Hitachi-AEM method and its underlying valuation logic, the method could not be used for validation. Using fuzzy sets and defining support points of the membership functions, the evaluation algorithm's results were adjusted to match the methods' results, ensuring a correlation between them. The sample products are based partly on real products (5) and partly on the synthetic reproduction (30) already described. Products with between 5 and 30 components are covered. When generating the validation data set, the number of components in the 35 sample products was varied, excluding the panel and its fasteners, so that the first 10 evaluations

used products with $n_i \in [10, 20]$ instrument parts. The second batch (10 products) was generated with products with $n_i \in [5, 10]$ instrument elements, and their ratings were compared. Finally, a comparison was made with very large product variations (number of instrument parts $n_i \in [20, 30]$). The final validation result yields the correlation shown in Figure 3. Significant correlations could be established in comparison to the evaluations according to Lucas and the design efficiency according to B&D. The negative correlation with Lucas is due to the contrasting waiting areas of the two methods: Lucas states that everything above 2.5 should be revised. In contrast, the DFA evaluation algorithm states that everything below 60% should be revised.

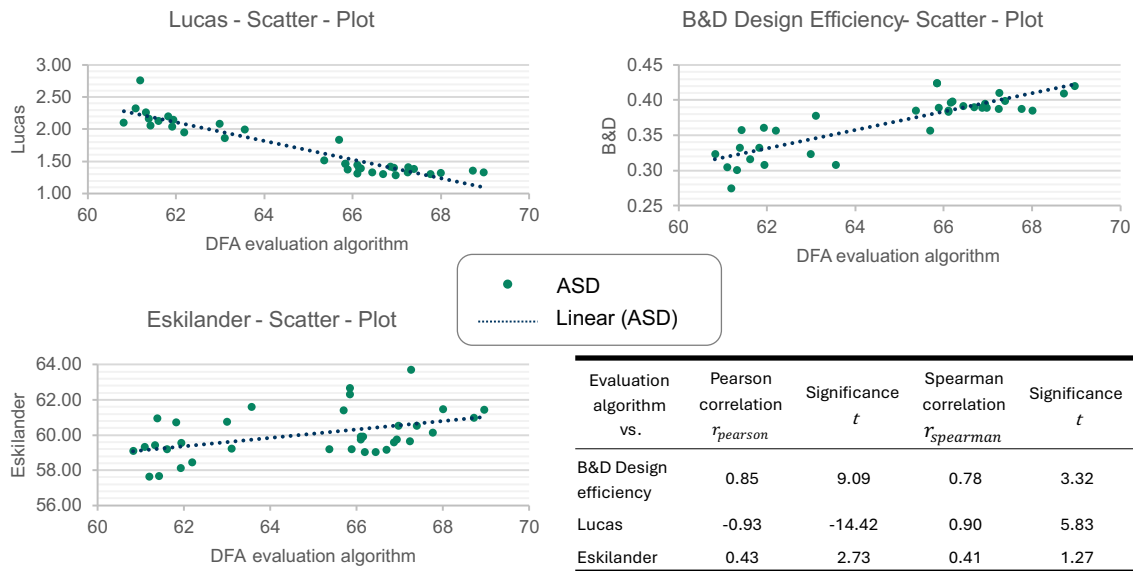


Figure 3: Linear relationship between the output of the evaluation algorithm and the state of the art

Compared to Eskilander's assessment, the correlation with the DFA-evaluation algorithm remains significant ($|t| > 2 \rightarrow$ significant at the $p < 0.05$ level), but is lower than with the other methods considered. An in-depth analysis of the sample data shows that SynthData 10 to SynthData 20 products exhibit a significant increase in the assembly-to-component ratio (ratio > 0.2). At the same time, the deviation between the Eskilander assembly rating and the evaluation algorithm also increases significantly (s. Figure 44). In the other batches, there are also isolated products with a ratio above 0.2; however, batch 2 is entirely above this threshold. If these batches are excluded from the analysis, a high correlation of $r_{pearson} = 0.72$ can be observed between the two evaluation methods. These results suggest that the underlying evaluation model reaches its limits for ratios above 0.2. In particular, the algorithm's quality appears significantly reduced for products with a small number of components distributed across many assemblies.

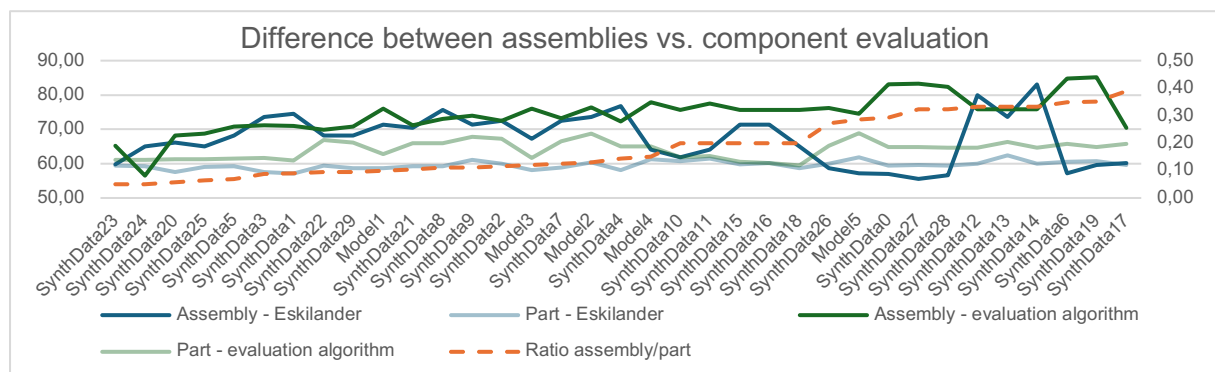


Figure 4: Differences between assemblies vs. component evaluation in comparison to the ratio of assemblies and parts

3.3 Machine learning approach

To answer RQ2, a graph isomorphism network (GIN) based supervised learning model was used. The advantage of GINs lies in their ability to model relational dependencies between neighboring nodes via message passing explicitly. In addition, GINs are theoretically as powerful as the Weisfeiler–Lehman graph isomorphism test and can therefore distinguish a wide class of non-isomorphic graphs (graphs that do not have the same structure), which is a prerequisite for the scenario described here, given the variability of components. Compared with classical regression methods, this approach enables automatic, data-driven extraction of relevant structural features directly from the graph representation. [46]

Each hierarchical product description (AML file) is modeled as an undirected graph $G = (V, E)$, where V denotes the set of nodes and $E \subseteq \{\{u, v\} \mid u, v \in V\}$ denotes the set of undirected edges. Although connections in the AML specification can have a semantic direction, this information is not used in the present model; the graph representation is therefore undirected. In this representation, each element in the product description corresponds to a node, while edges represent hierarchical relationships between elements, encoded as undirected connections in the graph. For example, the six enclosures of a part are modeled as child nodes of the corresponding part element and connected to it via edges. In addition, as described in [28], physical connections between two parts are modeled by links and are likewise represented as edges in the graph. Furthermore, a feature vector $x_v \in \mathbb{R}^{14}$, which represents numerical attributes of the respective component, such as dimension, material, or weight information, is assigned to each node $v \in V$. To improve numerical stability, the target variable is standardized. The mean μ and standard deviation σ are calculated for the training data. During training, the mean square error (MSE) on the scaled target variable is minimized. For inference, a back transformation to the original target space is performed. The model was initially trained using an 80/10/10 train/validation/test split with early stopping. In addition, a 5-fold cross-validation was conducted to assess the robustness of the results. Finally, the trained model was evaluated on 35 previously unseen sample products. This results in the mean absolute errors (MAE) and the coefficient of determination (R^2) shown in Table 1. Using the previously evaluated sample products, the deviations and correlations were obtained relative to the evaluation algorithm.

Table 1: Validation of the GIN

First Training- Early stopping after 64 epochs with adaptive learning rate (LR)	Test - MAE	0.334
	Test - R^2	0.972
Using 5-fold cross-validation	Test - MAE	0.292
	Test - R^2	0.976
Validation with the 35 sample products	Sample products - MAE	1.24
	Sample products - RMSE	1.60
	Pearson correlation to the evaluation algorithm	0.92

4 Discussion

Looking at the validation results in Table 1, the MAE for the sample products is significantly higher than for the test set: GINs tend to overfit training-specific graph patterns more than simpler GNN architectures (e.g., GCNs), especially with small or homogeneous datasets. Although the sample products belong to the same product family, the deviations between their structural representations and the dataset remain to be investigated. Dwivedi et al. describe similar behavior in GINs, which achieve very low training error but also exhibit higher variance and poorer performance on out-of-distribution

data. In addition, the authors conclude that GINs are more prone to overfitting than less expressive GNNs [47]. The observed increase in prediction error for previously unseen sample products highlights limitations in the trained model's generalization capability. While the underlying DFA criteria are largely product-independent, the training dataset is based on a synthetic representation of a specific product family. As a result, the model primarily generalizes within the structural and topological patterns represented in the dataset.

Despite the increased MAE, the results still show a high Pearson correlation of $r_{pearson} = 0.92$ between the model predictions and the evaluation algorithm. This suggests that the model captures the relative relationships and trends of the target variable very well, even if the absolute scaling is less accurate for previously unseen products. Therefore, the current model should be interpreted as a proof of concept demonstrating the feasibility of learning automation suitability from structured product descriptions, rather than as a universally applicable predictor across arbitrary product families. In practical applications where rankings or relative differences are particularly relevant, this represents an important strength of the approach.

5 Conclusion and further outlook

From a hybrid intelligence standpoint, the presented approach demonstrates how expert knowledge can be preserved, formalized, and extended through machine learning. The surrogate model captures relational and structural patterns that are difficult to quantify manually, while still remaining grounded in established evaluation logic. Human designers remain essential for interpreting results, refining criteria, and making final decisions.

This work has shown that established methods for evaluating automated assemblies are still relevant and that the selected criteria are also reflected in the current state of research. Furthermore, it has been shown that GINs can be used to make valid predictions of ASD, thereby validating the key factors for reliable evaluation mentioned in Chapter 2. The GIN-based model exhibits high internal validity and robustness, as demonstrated by cross-validation and test-set results. However, the increased deviation in the sample products underscores the importance of a training database as broad and representative as possible. Future work could investigate the integration of directed edges, additional domain-specific node features, or targeted domain adaptation to further improve the model's generalization. Furthermore, future work should compare different AI models and their performance to ensure that the most powerful model is used for this application. In addition, the transferability to other product families must be investigated.

6 Acknowledgement

This research was funded by the German Research Foundation (DFG) as part of the research project “Engineering assistance for hybrid decision support based on machine learning surrogate models” (project number 543079297) within the priority program 2443 “Hybrid Decision Support in Product Creation.” The authors would like to thank the DFG for funding and financing the research.

References

- [1] A. Burstedde and G. Kolev-Schaefer, Die Kosten des Fachkräftemangels, no. 27, 2024.
- [2] E. Kirchner, Werkzeuge und Methoden der Produktentwicklung. Von der Idee zum erfolgreichen Produkt, Berlin: Springer Vieweg, 2020.
- [3] G. Formentini, N. Rodríguez and C. Favi, “Design for manufacturing and assembly methods in the product development process of mechanical products: a systematic literature review,” *The International Journal of Advanced Manufacturing Technology*, vol. 120, 7-8, pp. 4307-4334, 2022. DOI: <https://doi.org/10.1007/s00170-022-08837-6>.

- [4] A. Desai and A. Mital, "Facilitating design for assembly through the adoption of a comprehensive design methodology," *International Journal of Industrial Engineering: Theory, Applications and Practice*, 17 (2), 2010. DOI: <https://doi.org/10.23055/ijietap.2010.17.2.166>.
- [5] I. Ezpeleta, D. Justel, U. Bereau and J. Zubelzu, "DFA-SPDP, a new DFA method to improve the assembly during all the product development phases," *Procedia CIRP*, vol. 84, pp. 673-679, 2019. DOI: <https://doi.org/10.1016/j.procir.2019.04.273>.
- [6] J. Trommnau, "Methodik zur Bewertung und Optimierung der Automatisierungseignung des Zusammenbaus von Bauteilkombinationen mit forminstabilen Einzelteilen," 2022. DOI: <https://doi.org/10.18419/opus-12536>.
- [7] D. Shetty and A. Ali, "A new design tool for DFA/DFD based on rating factors," *Assembly Automation*, vol. 35, no. 4, pp. 348-357, 2015. DOI: <https://doi.org/10.1108/AA-11-2014-088>.
- [8] H. Lin, X. Deng, J. Yu, X. Jiang and D. Zhang, "A Study of Sustainable Product Design Evaluation Based on the Analytic Hierarchy Process and Deep Residual Networks," *Sustainability*, vol. 15, no. 19, p. 14538, 2023. DOI: <https://doi.org/10.3390/su151914538>.
- [9] P. Zhang, X. Li, Z. Nie, F. Yu and W. Liu, "A Trimming Design Method Based on Bio-Inspired Design for System Innovation," *Applied Sciences*, vol. 11, no. 9, p. 4060, 2021. DOI: <https://doi.org/10.3390/app11094060>.
- [10] A. Desai, "Ease of product assembly through a time-based design methodology," *Assembly Automation*, vol. 39, no. 5, pp. 881-903, 2019. DOI: <https://doi.org/10.1108/AA-09-2018-0133>.
- [11] P. Eremeev, A. de Cock, H. Devriendt and F. Naets, "Framework for metamodel-based design optimization considering product performance and assembly process complexity," *SIMULATION*, vol. 100, no. 5, pp. 473-492, 2024. DOI: <https://doi.org/10.1177/00375497231217301>.
- [12] E. Gallegos-Nieto, H. I. Medellín-Castillo and S. N. Cerda-Avila, "Haptic-enabled virtual reality-aided design for assembly," *The International Journal of Advanced Manufacturing Technology*, vol. 134, 5-6, pp. 2849-2865, 2024. DOI: <https://doi.org/10.1007/s00170-024-14319-8>.
- [13] F. Campi, C. Favi, M. Germani and M. Mandolini, "CAD-integrated design for manufacturing and assembly in mechanical design," *International Journal of Computer Integrated Manufacturing*, vol. 35, no. 3, pp. 282-325, 2022. DOI: <https://doi.org/10.1080/0951192X.2021.1992659>.
- [14] M. Gunji Bala, D. B. B. V. L. and B. M. V. A. Raju, "Integrated Design for Assembly Approach for Optimal Reduced Levels of Assembly Sequence Generation using Enhanced Fruit Fly Algorithm," *International Journal of Performability Engineering*, vol. 18, no. 6, p. 407, 2022. DOI: <https://doi.org/10.23940/ijpe.22.06.p3.407416>.
- [15] M. Suszyński, K. Peta, V. Černohlávek and M. Svoboda, "Mechanical Assembly Sequence Determination Using Artificial Neural Networks Based on Selected DFA Rating Factors," *Symmetry*, vol. 14, no. 5, p. 1013, 2022. DOI: <https://doi.org/10.3390/sym14051013>.
- [16] O. Coma, C. Mascle and M. Balazinski, "Application of a fuzzy decision support system in a Design for Assembly methodology," *International Journal of Computer Integrated Manufacturing*, vol. 17, no. 1, pp. 83-94, 2004. DOI: <https://doi.org/10.1080/0951192031000078185>.
- [17] J. Mo, J. Cai, Z. Zhang and Z. Lu, "DFA-oriented assembly relation modelling," *International Journal of Computer Integrated Manufacturing*, vol. 12, no. 3, pp. 238-250, 1999. DOI: <https://doi.org/10.1080/095119299130290>.
- [18] J. SIK OH, P. O'GRADY and R. E. YOUNG, "A constraint network approach to design for assembly," *IIE Transactions*, vol. 27, no. 1, pp. 72-80, 1995. DOI: <https://doi.org/10.1080/07408179508936718>.
- [19] E. Molloy, H. Yang, J. Browne and B. J. Davies, "Design for Assembly within Concurrent Engineering," *CIRP Annals*, vol. 40, no. 1, pp. 107-110, 1991. DOI: [https://doi.org/10.1016/S0007-8506\(07\)61945-3](https://doi.org/10.1016/S0007-8506(07)61945-3).
- [20] L. S. Wahidin, C. F. Tan, S. N. Khalil, N. Tamaldin, Sivarao, J. Hu and G. Rauterberg, "A knowledge-based architecture framework of design for assmble system (DAEx)," *ARPN Journal of Engineering and Applied Sciences*, 11(4), pp. 2505-2507, 2016.
- [21] X. F. Zha, S. Y. E. Lim and S. C. Fok, "Integrated knowledge-based approach and system for product design for assembly," *International Journal of Computer Integrated Manufacturing*, vol. 12, no. 3, pp. 211-237, 1999. DOI: <https://doi.org/10.1080/095119299130281>.
- [22] J.-Y. Jung, "Manufacturing cost estimation for machined parts based on manufacturing features," *Journal of Intelligent Manufacturing*, vol. 13, no. 4, pp. 227-238, 2002. DOI: <https://doi.org/10.1023/A:1016092808320>.
- [23] G. Formentini, F. Bouissiere, C. Cuiller, P.-E. Dereux and C. Favi, "CDFA method: a way to assess assembly and installation performance of aircraft system architectures at the conceptual design," *Research in engineering design*, vol. 33, no. 1, pp. 31-52, 2022. DOI: <https://doi.org/10.1007/s00163-021-00378-5>.
- [24] G. Formentini, C. Cuiller, P.-E. Dereux, F. Bouissiere and C. Favi, "Impact assessment of design guidelines in the conceptual development of aircraft product architectures," *Procedia CIRP*, vol. 100, pp. 223-228, 2021. DOI: <https://doi.org/10.1016/j.procir.2021.05.059>.
- [25] G. Formentini, F. Bouissiere, C. Cuiller, P.-E. Dereux and C. Favi, "Conceptual Design for Assembly methodology formalization: systems installation analysis and manufacturing information integration in the design and development of aircraft architectures," *Journal of Industrial Information Integration*, vol. 26, p. 100327, 2022. DOI: <https://doi.org/10.1016/j.jii.2022.100327>.

- [26] F. Bouissiere, C. Cuiller, P.-E. Dereux, C. Malchair, C. Favi and G. Formentini, "Conceptual Design for Assembly in Aerospace Industry: A Method to Assess Manufacturing and Assembly Aspects of Product Architectures," *Proceedings of the Design Society: International Conference on Engineering Design*, vol. 1, no. 1, pp. 2961-2970, 2019. DOI: <https://doi.org/10.1017/dsi.2019.303>.
- [27] C. Favi, G. Formentini, F. Bouissiere, C. Cuiller, P.-E. Dereux and C. Malchair, "Design for Assembly in the Conceptual Development of Aircraft Systems," in *Design Tools and Methods in Industrial Engineering. Proceedings of the International Conference on Design Tools and Methods in Industrial Engineering, ADM 2019, September 9-10, 2019, Modena, Italy*, C. Rizzi, ed.: Cham: Springer International Publishing AG, 2020, pp. 268-278.
- [28] M. Jakschik, J. Thielmann, L. Lamers and B. Kuhlenkötter, "AI-supported evaluation application for product design assessment regarding automated production," in *8th International Conference on Advances in Artificial Intelligence: ACM*, 2024, pp. 114-120.
- [29] S. Eskilander, *Design for automatic assembly. A Method For Product Design: DFA2*, Dissertation, Kungliga Tekniska Högskolan, 2001.
- [30] L. Lamers, M. Jakschik, B. Kuhlenkötter, I. Yorgun and A. Hypki, *Engineering Based Approach To Optimize Electrolyzer Production Plant Planning*, Hannover: publish-Ing, 2024.
- [31] A. Greco, P. Manco, M. B. Russo and S. Gerbino, "Complexity-driven product design: part 1—methodological framework and geometrical complexity index," *International Journal on Interactive Design and Manufacturing (IJIDeM)*, vol. 18, no. 8, pp. 5441-5456, 2024. DOI: <https://doi.org/10.1007/s12008-023-01426-1>.
- [32] J. Landahl, R. J. Jiao, J. Madrid, R. Söderberg and H. Johannesson, "Dynamic platform modeling for concurrent product-production reconfiguration," *Concurrent Engineering*, vol. 29, no. 2, pp. 102-123, 2021. DOI: <https://doi.org/10.1177/1063293X20958938>.
- [33] S. Wang, D.-Z. Liu, C. He, G.-Y. Zou and Z.-K. Li, "Impacts of product structural characteristics on modular performance with virtual DSM data mining," *Journal of Engineering Design*, vol. 34, no. 7, pp. 462-486, 2023. DOI: <https://doi.org/10.1080/09544828.2023.2232983>.
- [34] I. Ezpeleta, U. Pujana, I. Isasa, J. Ayerbe and D. Justel, "New design for assembly (DfA) methodology for large and heavy parts assembled on site," *Procedia CIRP*, vol. 100, pp. 145-150, 2021. DOI: <https://doi.org/10.1016/j.procir.2021.05.078>.
- [35] D. Sanders, Y. Chai Tan, I. Rogers and G. E. Tewkesbury, "An expert system for automatic design-for-assembly," *Assembly Automation*, vol. 29, no. 4, pp. 378-388, 2009. DOI: <https://doi.org/10.1108/01445150910987808>.
- [36] J. Han, P. Jiang and P. Childs, "Metrics for Measuring Sustainable Product Design Concepts," *Energies*, vol. 14, no. 12, p. 3469, 2021. DOI: <https://doi.org/10.3390/en14123469>.
- [37] S. Wartzack, H. Meerkamm, S. Bauer, H. Krehmer, A. Stockinger, M. Walter and B. Schleich, "Design for X (DFX)," in *Handbuch Konstruktion*, F. Rieg and R. Steinhilper, eds.: München: Hanser, 2018, pp. 463-484.
- [38] B. Bender and K. Gericke, *Pahl/Beitz Konstruktionslehre*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2021.
- [39] M. Nardelli, P. Cicconi, A. Savoretti, R. Raffaeli and M. Germani, "A Knowledge Based Approach to Support the Conceptual Design of ETO Products," *Proceedings of the Design Society: International Conference on Engineering Design*, vol. 1, no. 1, pp. 2417-2426, 2019. DOI: <https://doi.org/10.1017/dsi.2019.248>.
- [40] N. Rea Minango, M. Hedlind and A. Maffei, "Handling features in assembly: Integrating manufacturing considerations early in design discussions," *Journal of Manufacturing Systems*, vol. 77, pp. 1077-1100, 2024. DOI: <https://doi.org/10.1016/j.jmsy.2024.11.012>.
- [41] J. Y. Jung and S. B. Billatos, "An expert system for assembly based on Axiomatic Design principles," *Journal of Intelligent & Robotic Systems*, vol. 8, no. 2, pp. 245-265, 1993. DOI: <https://doi.org/10.1007/BF01257997>.
- [42] Y. Jing, G. Zhou, C. Zhang, F. Chang, H. Yan and Z. Xiao, "XMKR: Explainable manufacturing knowledge recommendation for collaborative design with graph embedding learning," *Advanced Engineering Informatics*, vol. 59, p. 102339, 2024. DOI: <https://doi.org/10.1016/j.aei.2023.102339>.
- [43] R. Parzeller, D. Koziol, T. Dagner and D. Gerhard, "Automating the assembly planning process to enable design for assembly using reinforcement learning," *Proceedings of the Design Society*, vol. 4, pp. 2179-2186, 2024. DOI: <https://doi.org/10.1017/pds.2024.220>.
- [44] C. M. Krahe, *KI-gestützte produktionsgerechte Produktentwicklung: Automatisierte Wissensextraktion aus vorhandenen Produktgenerationen*, 2023.
- [45] S. V. S. Dochibhatla, M. Bhattacharya and B. Morkos, "Evaluating Assembly Design Efficiency: A Comparison Between Lucas and Boothroyd-Dewhurst Methods," in *Volume 4: 22nd Design for Manufacturing and the Life Cycle Conference; 11th International Conference on Micro- and Nanosystems: American Society of Mechanical Engineers*, 2017.
- [46] B. Khemani, S. Patil, K. Kotecha and S. Tanwar, "A review of graph neural networks: concepts, architectures, techniques, challenges, datasets, applications, and future directions," *Journal of Big Data*, vol. 11, no. 1, 2024. DOI: <https://doi.org/10.1186/s40537-023-00876-4>.
- [47] V. P. Dwivedi, C. K. Joshi, A. T. Luu, T. Laurent, Y. Bengio and X. Bresson, "Benchmarking Graph Neural Networks," 2020. DOI: <https://doi.org/10.48550/arXiv.2003.00982>.

Multimodal AI Agent for Orchestration of Mobile Robotic Tasks in Flexible Assembly Automation

Lucas Manassés Pinheiro de Souza^{1*}, Vishnu Murugasamy¹, Alexander Moriz¹, Amon Göppert¹,
Robert Schmitt^{1,2}

¹ Chair for Intelligence in Quality Sensing (IQS), WZL | RWTH Aachen University, 52074 Aachen, Germany

² Fraunhofer Institute for Production Technology (IPT), 52074 Aachen, Germany

*Corresponding author. E-Mail address: lucas.manasses@wzl-iqs.rwth-aachen.de; Tel.: +49 241 80 20594

Abstract: Increasing product variability and shrinking batch sizes in manufacturing demand robotic systems that can be reconfigured rapidly with minimal engineering effort. While mobile manipulators provide flexibility at the hardware level, task-level specification and coordination remain a major bottleneck in assembly-oriented scenarios. This paper presents a modular multimodal AI agent that acts as a supervisory orchestration layer for mobile robotic tasks in flexible assembly automation. Following a hybrid-intelligence architecture, the agent interprets natural-language or speech instructions, grounds them in contextual and perceptual information, and coordinates navigation, perception, and manipulation via constrained Robot Operating System (ROS) 2 tool interfaces, while execution remains within established autonomy components. The framework is evaluated via quantitative analysis of end-to-end request-to-execution behavior across stepwise interaction, one-shot execution, and task reconfiguration scenarios, in simulation and selected physical validation. Results show that orchestration-related processing remains bounded relative to navigation and manipulation runtime, with execution variability dominated by perception and grounding reliability rather than agent reasoning overhead. Robust multimodal grounding substantially reduces long-tailed execution behavior in autonomous task pipelines, providing empirical insight into the runtime characteristics and limitations of agent-based task orchestration for production-relevant mobile manipulation.

Keywords:

Multimodal AI agents, Task orchestration, ROS 2, Mobile manipulation, Flexible assembly automation

1 Introduction

The manufacturing industry is undergoing a structural transformation driven by increasing product variability, shrinking batch sizes, and the need for rapid reconfiguration, further intensified by skilled labor shortages and volatile supply chains [1, 2], which forces robotic systems to operate under frequent product changes and evolving task requirements, often in close interaction with human operators [2, 3]. Mobile manipulators are a promising response because they decouple task execution from fixed layouts and enable flexible allocation of robotic resources within production environments [4, 5], making them well suited for high-mix (dis)assembly (e.g., electric vehicle battery handling or cable harness manipulation) where task sequences, grasp strategies, and motion plans must be adapted regularly [6, 7]. Despite advances in perception, navigation, and manipulation, deployment in such settings remains difficult due to the effort required to reprogram and adapt task logic as conditions change [8]; accordingly, recent work indicates that the main bottleneck has shifted from low-level capabilities to scalable task-level specification, adaptation, and orchestration, where even minor changes in (dis)assembly procedures can lead to substantial engineering effort [8, 9].

Widely adopted robotic control architectures, including finite state machines and behavior trees, have proven effective for structuring deterministic task execution, but exhibit inherent limitations when applied to flexible assembly scenarios [10]. As task complexity and variability increase, these approaches scale poorly, requiring explicit modeling of product variants, exception cases, and contingencies by domain experts [11, 10]. Moreover, they lack mechanisms to interpret semantically underspecified instructions or reason about alternative task decompositions in response to changing conditions [12, 10]. While effective for deterministic execution, classical control architectures impose rigidity that conflicts with the adaptability required in modern assembly systems [3]. To address these challenges, recent research increasingly advocates for cognitively enhanced robotic systems that combine autonomous execution with higher-level reasoning and minimal human intervention [13, 14]. Within this paradigm, hybrid intelligence emphasizes a division of roles: humans provide intent, semantic context, and supervision; AI components perform reasoning and task abstraction; and robotic software ensures safe and reliable execution [15, 16]. Advances in Large Language Models (LLMs) and Vision-Language Models (VLMs) have demonstrated strong capabilities in interpreting natural-language (NL) instructions, grounding them in visual context, and reasoning about task semantics [17, 18]. However, these models are not yet systematically integrated with established robotic frameworks to be applicable in industrial settings [17].

Although embodied AI agents combining language, vision, and action have received increasing attention [19, 20], most existing approaches focus on tabletop manipulation or highly constrained environments and rely on tightly coupled or ad-hoc architectures [2, 21]. Comparatively little work addresses task-level orchestration for mobile manipulation in assembly-oriented scenarios using modular middleware such as Robot Operating System (ROS) 2. In particular, the runtime behavior of multimodal AI agents acting as supervisory orchestration layers, bridging human intent and robotic execution, remains insufficiently characterized, with evaluations largely limited to qualitative demonstrations or task success metrics. To address this gap, this work presents a modular multimodal AI agent that acts as a supervisory orchestration layer for flexible robotic assembly tasks. The framework integrates LLMs and VLMs with a ROS 2-based mobile manipulation stack to enable NL and voice-based task specification, semantic grounding through perception, and tool-based coordination of navigation and manipulation. The approach is evaluated through a quantitative analysis of end-to-end request-to-execution behavior in both simulation and physical environments.

The remainder of this paper is organized as follows. Section 2 reviews related work on embodied AI agents and task orchestration. Section 3 presents the proposed multimodal AI agent framework and its integration with a ROS 2-based mobile manipulation stack. Section 4 describes the experimental setup and evaluation methodology. Section 5 reports and discusses the experimental results. Section 6 concludes with key findings, limitations, and directions for future work.

2 Related Work

Recent surveys on the integration of LLMs, VLMs, and multimodal foundation models into robotic systems identify a spectrum of architectural paradigms, ranging from protocol-level interfaces and language-driven command translation to tightly coupled vision-language-action models and agent-based orchestration frameworks [17, 20]. Of particular relevance to this work are *orchestration-oriented approaches*, in which an AI agent supervises and coordinates existing robotic capabilities rather than replacing them with end-to-end learned control policies, a paradigm that is especially attractive for industrial and assembly-oriented scenarios where modularity, transparency, safety, and compatibility with established robotic software stacks such as ROS 2 remain essential design requirements [9]. Within this landscape, a first class of approaches employs LLMs or VLMs primarily as semantic interfaces, translating NL instructions into symbolic task descriptions, goal specifications, or high-level action

sequences [18, 19]. While these methods demonstrate effective language understanding and multimodal grounding, they typically do not assume responsibility for task-level orchestration across perception, navigation, and manipulation, leaving execution delegated to preconfigured pipelines and limiting the agent’s role to command interpretation rather than supervision of coordinated robotic behavior in dynamic environments.

Several ROS-based frameworks integrate LLM-driven agents to enable NL interaction with robots. ROS-LLM [22] maps user instructions to executable behaviors represented as state machines or behavior trees, ROSA (ROS Agent) [23] constrains LLM outputs to predefined ROS actions and services to ensure robustness and safety, and the Flexible Agent Framework for Embodied AI (RAI) [24] proposes a modular ROS 2 architecture interfacing with navigation and manipulation components. Across these frameworks, however, task execution remains bound to static or expert-defined schemas, and core mobile manipulation capabilities, such as coordinated navigation (*Nav2*), motion planning and task sequencing (*MoveIt 2* and *MoveIt Task Constructor*, *MTC*), *Simultaneous Localization and Mapping* (*SLAM*)-based localization, and 2D/3D perception-driven grasp pose estimation, are not orchestrated within a unified task-level pipeline, limiting their role to command interpretation rather than holistic supervision in assembly-oriented scenarios.

Beyond ROS-centric solutions, orchestration-oriented agentic systems have been explored primarily in non-industrial contexts. AutoRT [25] applies LLM-based supervision to large robot fleets for autonomous data collection, while the LABOR Agent [26] investigates coordination of manipulation skill libraries for bimanual tasks, but these approaches are evaluated in constrained or non-mobile settings and are not designed around ROS 2-native modular autonomy stacks. In parallel, tightly coupled vision–language–action models such as RT-2 [27] unify perception, reasoning, and execution within a single learned model; although powerful, such end-to-end approaches complicate integration with modular robotic architectures, reduce transparency and debuggability, and pose challenges for safety assurance and certification in industrial environments [28]. Nevertheless, such models can be integrated as specialized perception or grounding components, provided that task-level coordination and execution remain governed by deterministic, ROS 2-native software.

Table 1: Comparison of related frameworks with the proposed approach. Legend: ● supported, ◐ partially supported/limited, X not addressed. ROS 2-Components refers to the integration of motion planning (*MoveIt 2*, *MTC*), navigation (*Nav2*), *SLAM*, perception, and pose estimation for grasping.

Framework	Agent Orchestration	Mobile Manipulation	ROS 2 Native	ROS 2 Components	Runtime Behavior Evaluated	Assembly Automation (Production)	Hybrid Intelligence
ROS-LLM [22]	◐	X	●	◐	●	◐	●
ROSA [23]	◐	X	●	◐	◐	X	◐
RAI [24]	●	X	●	◐	◐	X	◐
AutoRT [25]	X	◐	X	X	●	X	◐
LABOR Agent [26]	◐	X	X	X	●	X	◐
Proposed approach	●	●	●	●	●	●	●

Across the reviewed literature, two recurring limitations emerge. First, most existing frameworks focus on tabletop manipulation or highly constrained environments, with limited consideration for mobile manipulation scenarios that require simultaneous coordinated navigation, perception, and manipulation. Second, evaluations predominantly emphasize task success or qualitative demonstrations, while the runtime behavior of AI-agent-based orchestration pipelines, including interaction overhead, request-to-execution latency, and responsiveness to high-level human input, remains insufficiently characterized. In contrast, the proposed approach explicitly integrates motion planning and task sequencing (*MoveIt 2*, *MTC*), autonomous navigation (*Nav2*), *SLAM*-based localization, perception, and grasp pose estimation

into a unified orchestration pipeline. Rather than benchmarking isolated capabilities, this work investigates the runtime behavior of a multimodal AI agent acting as a supervisory layer, providing empirical insight into how high-level human intent is translated into coordinated robotic actions in production-relevant scenarios. A comparative overview of representative frameworks and the proposed approach is summarized in Table 1.

3 Multimodal AI Agent for Hybrid-Intelligent Task Orchestration

The proposed framework introduces a multimodal AI agent that acts as a supervisory orchestration layer for mobile manipulation in flexible assembly scenarios. The architecture follows a hybrid-intelligence split: the operator provides intent and supervision; the agent interprets instructions and coordinates task execution; and established ROS 2 autonomy components perform navigation and manipulation. This design targets frequent task changes and semantically underspecified requests while preserving execution within the underlying robotics stack.

3.1 Framework Overview

High-level user inputs are provided as text or speech and interpreted by the agent in conjunction with contextual state (e.g., robot pose, semantic waypoints) and perception outputs. The agent operates exclusively at the task level by selecting and parameterizing tools from ROS 2 subcomponents, while motion generation, collision checking, and control remain within the underlying autonomy stack. Tool execution feedback is returned to the agent to support iterative orchestration. Figure 1 summarizes the architecture.

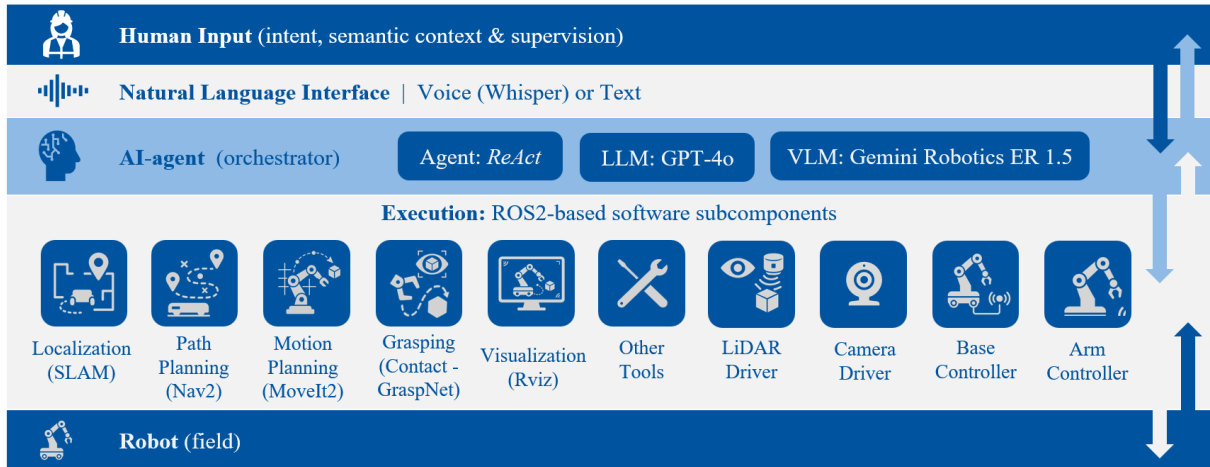


Figure 1: Layered architecture of the multimodal AI-agent orchestration layer integrated with ROS 2 modules for mobile manipulation. High-level inputs (text/speech) are interpreted by the agent powered by the LLM & VLM models. The agent then invokes modular ROS 2 tools (*SLAM*, *Nav2*, *Contact-GraspNet*, *MoveIt 2/MTC*). Execution feedback closes the loop for iterative orchestration.

3.2 Multimodal Agent and Tool-Based Orchestration

The orchestration logic follows an iterative reason–act–observe (ReAct-style) pattern, in which high-level instructions are decomposed into short-horizon tool invocations and revised based on runtime feedback. The agent combines NL or speech input with structured robot state and optional visual context to support task interpretation and grounding. In the presented implementation, semantic interpretation and task decomposition are handled by an LLM instantiated using *GPT-4o API* [29], which parses NL or transcribed speech instructions to extract task intent, entities, and ordering constraints. For perception-driven grounding, the framework also employed Grounded SAM (GSAM) [30] for segmentation of the target object, later fully replaced by vision–language capabilities provided by *Gemini Robotics ER 1.5*

API [31], which supports perception-driven grounding, for example by assisting object identification and semantic referencing during grasp pose estimation. Outputs from both models are constrained through predefined tool schemas and prompt design, ensuring that only valid robot capabilities can be invoked and that ambiguous instructions are handled conservatively.

Each robotic capability is exposed to the agent through a narrow tool interface aligned with ROS 2 actions, services, and topics, including waypoint management, mapping and localization (SLAM), autonomous navigation (*Nav2*), perception and object pose estimation (e.g., *Contact-GraspNet*), motion planning and task execution (*MoveIt 2* and *MTC*), and bounded teleoperation primitives for base and arm control. This abstraction decouples the orchestration logic from the internal implementation of individual modules, allowing components such as *SLAM* backends, perception models, or grasp generators to be replaced or upgraded without modifying the agent’s reasoning or task sequencing logic. Each tool invocation returns structured execution feedback, enabling the agent to detect failures and potentially react through retries, or operator interaction. To support reusable task references, the framework includes a lightweight semantic memory mechanism in which operators label base and arm waypoints during setup (e.g., “pick-up station”, “assembly station”, “scan pose”, “pre-grasp”). Waypoints can be created via teleoperation or incremental motion primitives and are stored as callable parameters for subsequent autonomous execution. Semantic memory enables hybrid task execution: operators provide sparse semantic structure once, and the agent resolves later references (e.g., “go to the assembly station”) into explicit goals for navigation or manipulation tools. This reduces reconfiguration effort when stations, targets, or task order change, while keeping execution within deterministic autonomy modules.

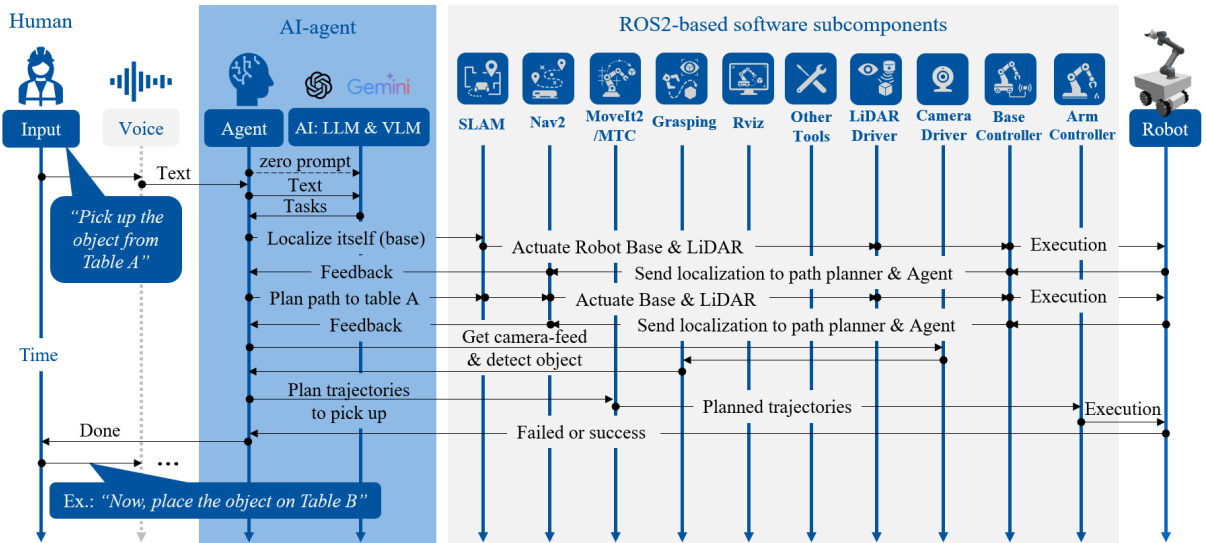


Figure 2: Exemplary workflow for hybrid-intelligent orchestration in an assembly-oriented pick-and-place task. The operator defines semantic stations and poses, while the agent decomposes a high-level instruction into navigation (*Nav2*), perception, and manipulation (*MoveIt 2*/*MTC*) tool calls with feedback-driven sequencing.

3.3 Exemplary Orchestration Workflow

Figure 2 illustrates an assembly-oriented pick-and-place workflow executed from a single high-level instruction, e.g., “Go to the pick-up station, pick the gear, and place it on the assembly station.” The agent parses the request to identify referenced locations and objects and resolves them via semantic memory (labeled positions). If required semantic labels are missing, the agent requests operator input; otherwise, it autonomously sequences navigation, perception, and manipulation. The agent issues navigation goals to reach the pick-up and assembly stations, triggers perception to estimate grasp poses for the target object and invokes motion planning and execution for pick and place using *MoveIt 2* /

MTC. Execution feedback from each tool is monitored to support retries, or operator intervention when necessary. In Figure 3 the Hybrid-intelligent orchestration workflow is shown in the simulation.

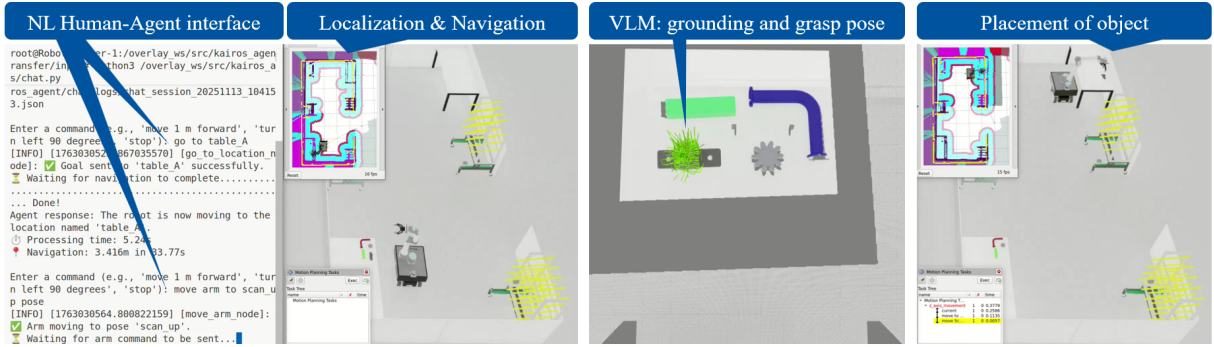


Figure 3: Hybrid-intelligent orchestration workflow in simulation. Left: NL human-agent interface for defining semantic goals, pose labeling, and teleoperation, shown with real-time *SLAM* and *Nav2*-based navigation in simulation environment as the robot moves to *Table A* to pick the target object (T-connector). Middle: VLM-based perception-driven grounding and grasp pose generation from the arm camera. Right: Autonomous placement of the object on *Table B* as requested by human input.

4 Experimental Setup

The evaluation focuses on end-to-end request-to-execution behavior and orchestration overhead under assembly-oriented conditions. Experiments were conducted in NVIDIA Isaac Sim to ensure repeatability, with one selected validation scenario being performed in a physical setup, as shown in Figure 4 (left). For the simulation environment, Figure 4 (middle), a simulated mobile manipulator, Figure 4 (right), based on the RB-Kairos platform, equipped with a 6-DoF industrial arm, an RGB-D camera, and LiDAR sensors for navigation. The simulated setup exposes the same ROS 2 interfaces as the physical platform, allowing the orchestration pipeline to operate with minimal changes. The software stack includes *SLAM*, *Nav2*, *Contact-GraspNet*, *MoveIt 2* and *MTC*. The multimodal AI agent framework runs as a supervisory layer, coordinating these components through structured tool invocations.

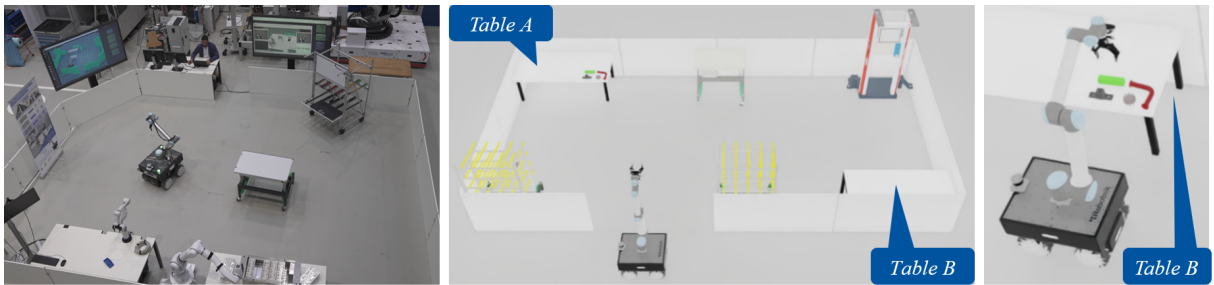


Figure 4: Left: physical setup at WZL RWTH. Middle: simulated environment in NVIDIA Isaac Sim. Right: simulated mobile manipulator RB-Kairos, equipped with a 6-DoF industrial arm, simulated RGB-D camera, and simulated LiDAR sensors for navigation and manipulation.

Orchestration-layer behavior is evaluated using (i) *processing time* (request-to-plan), measured as the time from user request to the agent producing an executable tool sequence, (ii) *execution time* (request-to-execution), measured as the time from request to completion of robot actions, and (iii) *orchestration share*, computed as $\text{Proc}/(\text{Proc} + \text{Exec}) \times 100$. In addition, execution variability is reported qualitatively (Low/Med/High) based on observed spread and long-tailed runtimes, and notable agent/runtime events (e.g., retries, restarts, drift) are logged as annotations. For the physical validation (Scenario 4), timing is additionally reported per subsystem (navigation and arm motion) to reflect real-world execution effects.

Scenario 1 (S1) Stepwise Interaction with Explicit Subtasks, where the user issues individual high-level commands for discrete subtasks (e.g., navigation, object detection, pick), but still in a logical sequence. Commands are provided via text and voice, and the agent executes each step separately. This scenario serves as a baseline for evaluating interaction overhead and agent responsiveness, isolating latency from instruction interpretation, tool invocation, and feedback handling under controlled conditions.

Scenario 2 (S2) One-Shot Task Execution (Pipeline Orchestration), where the user provides a single high-level instruction (one-shot task execution) specifying a complete assembly task (navigate–pick–place). The agent autonomously decomposes and sequences navigation, perception, and manipulation actions without further user input. This scenario assesses task-level decomposition and orchestration behavior, capturing the cumulative overhead of semantic interpretation, tool selection, and feedback-driven sequencing.

Scenario 3 (S3) Environmental Change and Task Reconfiguration, where changes are introduced to the environment or task structure (e.g., modified object locations or station assignments), as shown in Figure 5. The user issues high-level instruction similar to Scenario 2, while the agent adapts execution based on updated context and stored semantic labels. This scenario evaluates the agent’s ability to reuse semantic memory, request clarification when needed, and re-sequence actions without manual task-graph reprogramming.

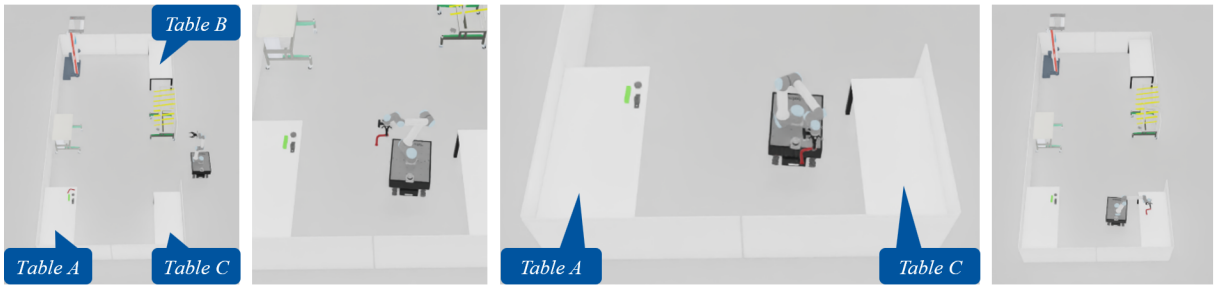


Figure 5: Evaluation of Scenario 3. An additional placement station is introduced into the environment (e.g., *Table C*), requiring the agent to resolve updated semantic labels and adapt task sequencing.

Scenario 4 (S4) Physical Validation and Transferability, where transferability of the stepwise interaction (Scenario 1) is assessed through deployment on a physical setup, as shown in Figure 6. The physical platform mirrors the simulated configuration in terms of software architecture and ROS 2 interfaces. This scenario evaluates integration of Nav2-based navigation and arm control for executing tasks at labeled semantic positions. While quantitative analysis focuses on simulation, physical execution confirms consistent orchestration behavior under real sensing and actuation conditions, with only minor platform-specific adaptations required.

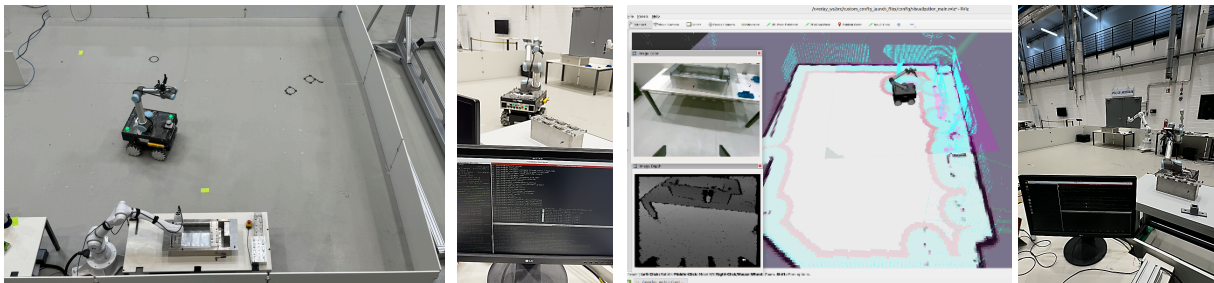


Figure 6: Physical validation of Scenario (4). Left: Mobile manipulator in the physical evaluation setup. Middle left: Onboard execution of the multimodal AI agent with NL interface. Middle right: Real-time RViz visualization of the ROS 2 stack during navigation and manipulation. Right: Robot reaching the labeled placement station (*Table B*) with the NL interface visible.

5 Results and Discussion

This section presents the experimental results for each of the previously defined scenarios. Table 2 summarizes the end-to-end behavior of the proposed orchestration layer across all four scenarios and interaction/grounding configurations. Overall, orchestration processing remains bounded relative to navigation and manipulation runtime, and variability is primarily driven by perception/grounding reliability and physical execution effects rather than agent reasoning.

Table 2: Aggregated orchestration performance across scenarios. Plan [s]: request-to-plan latency (user request $\rightarrow$ executable tool sequence, including parsing and tool selection). Exec [s]: request-to-execution latency (user request $\rightarrow$ completion of robot actions). Orch [%]: orchestration share of total runtime, computed as Plan/(Plan+Exec). Var: qualitative execution-time variability (L = Low, M = Medium, H = High) based on observed spread and long-tailed runtimes. $\uparrow$ indicates increased planning effort relative to S2 due to semantic context resolution (Table C).

Scenario	Interact.	Ground.	n	Plan [s]	Exec [s]	Orch [%]	Var	Note
S1 (stepwise)	Text	-	12	27.6 ± 5.2	139.5 ± 24.2	16.5	L	Baseline
	Voice	-	12	39.2 ± 13.6	379.5 ± 135.6	9.4	M	Speech overhead
S2 (one-shot)	Text	GSAM	13	-	308.9 ± 214.6	~ 10	H	Grasp retries
	Text	Gemini	15	-	155.3 ± 22.8	~ 12	L	Stable grounding
	Voice	Gemini	22	-	159.8 ± 24.7	~ 12	L	No added instability
S3 (reconfig.)	Text	Gemini	10	$\uparrow$ vs S2	$\uparrow$ / $\sim$ stable vs S2	$\sim 12-14$	L	Context (Table C)
S4 (physical)	Text (stepwise)	Gemini	10	3.05 ± 1.16	73.21 ± 29.18	-	M-	Nav:
			8	1.84 ± 0.79	4.01 ± 1.81	-	H L	Retries/drift/restart Arm: Retries/restart

Across 24 stepwise runs (S1), request-to-plan behavior is stable for both text and voice input. Voice interaction increases processing time due to speech transcription and clarification handling, but orchestration remains a minority of total task duration; overall runtime is dominated by navigation and manipulation, with execution variance mainly reflecting differences in the number and type of issued subtasks rather than orchestration instability.

For one-shot execution (S2, $n=50$), the agent incurs planning/reasoning overhead only once per task, and performance is dominated by grounding reliability. Chat + GSAM shows high variance and long-tailed runtimes due to repeated grasp attempts, whereas Gemini-based grounding yields substantially more stable execution with similar behavior for text and voice interaction, indicating that robust multimodal grounding, not interaction modality, drives predictability in autonomous pipeline orchestration.

Under environmental change and task reconfiguration (S3, $n=10$), introducing a new semantic station (Table C) increases planning-time overhead due to context resolution, while execution remains dominated by navigation and manipulation. No increase in execution variance or grasp retries is observed, suggesting that moderate environment/task changes can be accommodated through semantic memory without manual task-graph modification.

Physical validation (S4, $n=10$) confirms transferability of the stepwise workflow when ROS 2 interfaces are preserved. Processing latency remains low (navigation: 3.05 ± 1.16 s; arm: 1.84 ± 0.79 s), while runtime variability is dominated by real-world execution effects (navigation: 73.21 ± 29.18 s; arm: 4.01 ± 1.81 s) and occasional operational interventions (e.g., repeated commands, script restart, minor pose drift), indicating that sim-to-real deviations arise primarily outside the orchestration layer.

6 Conclusion, Limitations, and Outlook

This paper presented a modular multimodal AI agent that acts as a supervisory orchestration layer for mobile robotic tasks in flexible assembly automation. The agent interprets natural-language or speech

instructions and coordinates navigation, perception, and manipulation through constrained ROS 2 tool interfaces, while low-level execution remains within the autonomy stack. Across simulation and physical trials, processing time remained bounded relative to overall task duration, and end-to-end behavior was dominated by navigation and manipulation runtime rather than agent reasoning.

The evaluation highlights three main findings. First, under stepwise interaction (S1), processing time is stable for text and voice input; voice increases processing latency but does not change the dominant role of execution time. Second, in one-shot execution (S2), runtime predictability is driven primarily by grounding reliability: GSAM leads to long-tailed execution due to grasp retries, whereas Gemini-based grounding yields substantially more stable execution for both text and voice commands. Third, introducing a new semantic station (S3) increases planning-time effort due to context resolution, but does not increase execution variability, indicating that lightweight semantic memory supports moderate task reconfiguration without manual task-graph modification. Physical validation (S4) confirms transferability when ROS 2 interfaces are preserved, while remaining variability is largely attributable to real-world sensing and execution effects.

Limitations include a mainly simulation-based evaluation, a restricted physical benchmark, and the absence of controlled baselines against expert-engineered task graphs or classical planning approaches. Agent behavior (e.g., retries, interventions) was logged qualitatively rather than systematically counted, and request-to-plan was not reported for all one-shot configurations. Future work will extend physical benchmarking to full end-to-end pick-and-place pipelines under controlled disturbance factors, and will introduce explicit baselines to quantify reprogramming effort and runtime trade-offs against classical task representations. Robustness can be improved by adding failure-aware recovery strategies, tighter monitoring of tool-level outcomes, and more systematic logging of clarification and retry events. Beyond short-horizon tasks, the framework should be scaled toward longer-horizon and contact-rich (dis)assembly by integrating stronger safety layers, uncertainty-aware execution, and richer memory mechanisms (e.g., episodic traces) to support adaptation across runs.

Acknowledgements

Funded by the European Union. This work has received funding from the European High Performance Computing Joint Undertaking (JU) and from the German Federal Ministry of Research, Technology and Space (BMFTR), the Ministry of Culture and Science of North Rhine-Westphalia (MKW NRW) and the Hessian Ministry of Science and Research, Arts and Culture (HMWK) under grant agreement No 101250682.

References

- [1] F. Bermpohl, S. F. Schäfer, A. Hohmann und R. Daub, „Short-cycled factory planning – motivation and existing challenges,“ *Procedia CIRP*, p. 1708–1713, 2024.
- [2] A. Keshvarparast, D. Battini, O. Battaia und A. Pirayesh, „Collaborative robots in manufacturing and assembly systems: literature review and future research agenda,“ *Journal of Intelligent Manufacturing*, Bd. 35, pp. 2065–2118, 2023.
- [3] S. Asif, M. Bueno, P. Ferreira, P. Anandan, Z. Zhang, Y. Yao, G. Ragunathan, L. Tinkler, M. Sotoodeh Bahraini, N. Lohse, P. Webb, W. Hutabarat und A. Tiwari, „Rapid and automated configuration of robot manufacturing cells,“ *Robotics and Computer-Integrated Manufacturing*, Bd. 92, p. 102862, 2025.
- [4] G. Hüttemann, *Modellbasierte a priori Bewertung von linienlosen mobilen Montagesystemen*, Aachen: RWTH Aachen University, 2020.
- [5] L. Berge, G. Tréheux, L. Schäper, L. Manassés Pinheiro de Souza, S. Munker, A. Göppert und R. H. Schmitt, „Edge Computing Framework for Enhanced Robotic Adaptivity in Line-Less Mobile Assembly Systems,“ *Procedia CIRP*, Bd. 127, pp. 26–31, 2024.
- [6] Y. Zang, M. Qu, D. T. Pham, R. Dixon, F. Goli, Y. Zhang und Y. Wang, „Robotic disassembly of electric vehicle batteries: Technologies and opportunities,“ *Computers & Industrial Engineering*, Bd. 198, p. 110727, 2024.
- [7] D. Gebauer, J. Dirr und G. Reinhart, „Concept for Robot-Based Cable Assembly Regarding Industrial Production,“ in *Annals of Scientific Society for Assembly, Handling and Industrial Robotics 2021*, Cham, 2022.

- [8] Z. Jin, R. M. Marian und J. S. Chahl, „Achieving batch-size-of-one production model in robot flexible assembly cells,“ *The International Journal of Advanced Manufacturing Technology*, Bd. 126, Nr. 5, pp. 2097-2116, 2023.
- [9] T. Wiese, J. Abicht, A. Dietz und T. Bauernhansl, „An evaluation framework for assembly planning approaches in robotics and AI,“ *Frontiers in Robotics and AI*, Bd. 9, p. 1014476, 2022.
- [10] R. Ghzouli, T. Berger, E. B. Johnsen, A. Wasowski und S. Dragule, „Behavior trees and state machines in robotics applications,“ *IEEE Transactions on Software Engineering*, Bd. 49, Nr. 9, pp. 4243-4267, September 2023.
- [11] J. Cloete, W. X. Merkt und I. Havoutis, „Adaptive manipulation using behavior trees,“ in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025.
- [12] R. Liu, G. Wan, M. Jiang, H. Chen und P. Zeng, „Autonomous robot task execution in flexible manufacturing: integrating PDDL and behavior trees in ARIAC 2023,“ *Biomimetics*, Bd. 9, Nr. 10, p. 612, 2024.
- [13] M. Soori, B. Arezoo und R. Dastres, „Artificial intelligence, machine learning and deep learning in advanced robotics, a review,“ *Cognitive Robotics*, Bd. 3, pp. 54-70, 2023.
- [14] J. T. Licardo, M. Domjan und T. Orehovalčki, „Intelligent robotics—A systematic review of emerging technologies and trends,“ *Electronics*, Bd. 13, Nr. 3, p. 542, 2024.
- [15] C. R. Sauer und P. Burggräf, „Hybrid intelligence – systematic approach and framework to determine the level of Human-AI collaboration for production management use cases,“ *Production Engineering*, Bd. 19, Nr. 3, pp. 525-541, 2025.
- [16] H. Zhang, Y. Zhang, Z. Wang, S. Zhang, H. Li und M. Chen, „A novel knowledge-driven flexible human–robot hybrid disassembly line and its key technologies for electric vehicle batteries,“ *Journal of Manufacturing Systems*, Bd. 68, pp. 338-353, 2023.
- [17] J. Fan, Y. Yin, T. Wang, W. Dong, P. Zheng und L. Wang, „Vision-language model-based human-robot collaboration for smart manufacturing: A state-of-the-art survey,“ *Frontiers of Engineering Management*, Bd. 12, Nr. 1, p. 177, 2025.
- [18] H. Jeong, H. Lee, C. Kim und S. Shin, „A Survey of Robot Intelligence with Large Language Models,“ *Applied Sciences*, Bd. 14, Nr. 19, p. Article Number: 8868, 2024.
- [19] W. Huang, P. Abbeel, D. Pathak und I. Mordatch, „Language Models as Zero-Shot Planners: Extracting Actionable Knowledge for Embodied Agents,“ arXiv, 2022.
- [20] Y. Ren, Y. Liu, T. Ji und X. Xu, „AI agents and agentic AI—navigating a plethora of concepts for future manufacturing,“ *Journal of Manufacturing Systems*, Bd. 83, pp. 126-133, 2025.
- [21] H. Guo, F. Wu, Y. Qin, R. Li, K. Li und K. Li, „Recent Trends in Task and Motion Planning for Robotics: A Survey,“ *ACM Computing Surveys*, Bd. 55, Nr. 13s, p. 289, 2023.
- [22] C. E. Mower, Y. Wan, H. Yu, A. Grosnit, J. Gonzalez-Billandon, M. Zimmer, J. Wang, X. Zhang, Y. Zhao, A. Zhai, P. Liu, D. Palenicek, D. Tateo, C. Cadena, M. Hutter, J. Peters, G. Tian und Y, „ROS-LLM: A ROS framework for embodied AI with task feedback and structured reasoning,“ 2024.
- [23] R. Royce, M. Kaufmann, J. Beckett, S. Moon, K. Carpenter, K. Pak, A. Towler, R. Thakker und S. Khattak, „Enabling Novel Mission Operations and Interactions with ROSA: The Robot Operating System Agent,“ arXiv, 2024.
- [24] K. Rachwał, M. Majek, B. Boczek, K. Dąbrowski, P. Liberadzki, A. Dąbrowski und M. Ganzha, „RAI: Flexible Agent Framework for Embodied AI,“ in *Highlights in Practical Applications of Agents, Multi-Agent Systems and Computational Social Science. The PAAMS Collection*, Cham, 2025.
- [25] M. Ahn, D. Dwibedi, C. Finn, M. Gonzalez Arenas, K. Gopalakrishnan, K. Hausman, B. Ichter, A. Irpan, N. Joshi, R. Julian, S. Kirmani, I. Leal, E. Lee, S. Levine, Y. Lu und S. Maddingeni, „AutoRT: Embodied Foundation Models for Large Scale Orchestration of Robotic Agents,“ 2024.
- [26] K. Chu, X. Zhao, C. Weber, M. Li, W. Lu und S. Wermter, „Large Language Models for Orchestrating Bimanual Robots,“ in *2024 IEEE-RAS 23rd International Conference on Humanoid Robots (Humanoids)*, Nancy, France, 2024.
- [27] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn, P. Florence, C. Fu, M. Gonzalez Arenas, K. Gopalakrishnan und K. Han, „RT-2: Vision-Language-Action Models Transfer Web Knowledge to Robotic Control,“ 2023.
- [28] A. S. Adebayo, O. O. Ajayi und N. Chukwurah, „Explainable AI in Robotics: A Critical Review and Implementation Strategies for Transparent Decision-Making,“ *Journal of Frontiers in Multidisciplinary Research*, Bd. 05, Nr. 01, p. 26–32, 2024.
- [29] OpenAI, „GPT-4o System Card,“ 2024.
- [30] T. Ren, S. Liu, A. Zeng, J. Lin, K. Li, H. Cao, J. Chen, X. Huang, Y. Chen, F. Yan, Z. Zeng, H. Zhang, F. Li, J. Yang, H. Li, Q. Jiang und L. Zhang, „Grounded SAM: Assembling Open-World Models for Diverse Visual Tasks,“ 2024.
- [31] Gemini Robotics, „Gemini Robotics 1.5: Pushing the Frontier of Generalist Robots with Advanced Embodied Reasoning, Thinking, and Motion Transfer,“ 2025.

Hybrid Neuro-Symbolic Decision Support System in Product Planning and Design

Felix Engel¹, Nenad Krzdavac¹, Siamak Ghodsi², Julian Haller³, Antonia Markus³, Sven Munker³, Anas Abdelrazeq³, Robert H. Schmitt³

¹TIB – Leibniz Information Centre for Science and Technology, 30167 Hanover, Germany

²L3S Research Centre, Leibniz University Hannover, 30167 Hannover, Germany

³RWTH Aachen University, WZL-IQS, 52062 Aachen, Germany

*Corresponding author. E-Mail address: Felix.Engel@TIB.eu

Abstract: This article presents research work on a neuro-symbolic decision support system for product planning. It describes the motivation for the work and presents the conceptual framework and the status of implementation. The system to be developed is intended to support engineers in selecting and evaluating assembly and disassembly alternatives during product development. The chosen approach combines symbolic models, represented as Petri nets, ontologies and knowledge graphs, with learning-based, non-algorithmic graph transformation using small language models. Symbolic representations derived from CAD data are formally verified with a theorem prover to ensure feasibility, compliance with constraints and consistency. These verified models provide a reliable basis for decision-making for neural components that generate and evaluate AND/OR Graphs. Instead of programming fixed decision rules, the system learns transformation strategies from verified planning data and human feedback. This enables adaptive decision support for different product concepts. The resulting Decision Support System provides explainable, validated decision alternatives, thus supporting informed decisions in product development, particularly with regard to assembly and disassembly, taking into account circular economy goals

Keywords:

Small language models, neuro-symbolic, decision support system

1 Overview of Content

The process of product development and design is increasingly taking place against a backdrop of global socio-economic challenges. Given the limited availability of natural resources, it is essential to consider the principles of the circular economy (CE) from the earliest stages of design. Engineers and designers are therefore faced with complex decisions in which they must weigh up product functionality, material efficiency, ease of assembly and disassembly, and economic feasibility, while at the same time taking environmental and sustainability goals into account. Traditional decision support approaches, which are often based on static rules, deterministic optimisation or individual cost models, are generally insufficient in this context. They struggle to capture the dynamic interactions between product structures, life cycle strategies, material flows and operational constraints. They are also often unable to take into account multi-criteria trade-offs or the uncertainty inherent in early design phases. Furthermore, conventional tools can only integrate heterogeneous knowledge such as formalised design rules, sustainability guidelines, empirical process data and user feedback into a coherent decision-making framework to a limited extent. These limitations highlight the need for more advanced, adaptive decision support systems that can combine symbolic reasoning, knowledge representation, and learning-based methods to provide flexible, multi-criteria guidance and allow designers to iteratively explore, evaluate, and optimize design alternatives with CE objectives in mind.

To address this situation, interdisciplinary research at the intersection of engineering, artificial intelligence, and sustainability is being conducted. This research explores innovative approaches to the development of adaptive Hybrid Decision Support Systems (HDSS) that combine symbolic and neural methods to support consideration of CE principles in product engineering. In this context, particular emphasis is placed on the integration of explainable, knowledge-based representations with data-driven learning approaches. Combining symbolic and neural methods enables decision support systems to leverage structured engineering and sustainability knowledge while retaining the flexibility and adaptability of modern AI techniques. This hybrid paradigm provides the foundation for coupling human expertise with intelligent systems, as detailed in our architecture section. Therein, symbolic models, implemented as knowledge graphs, represent static domain knowledge and serve as a baseline for generating structured inputs (prompts) for subsequent neural processing steps. The neural models consume these inputs through the symbolic layer while simultaneously interacting with the user and continuously adapting their behavior based on user feedback. The human in the loop approach is chosen based on the assumption that, although we can explicitly model much of the necessary knowledge using symbolic approaches, there is far more implicit knowledge in the minds of engineers, which can be collected through constant interaction with our system. This knowledge is invaluable for future decision-making processes and for preserving knowledge within an organization.

2 Content Structure

This paper reports on research conducted within the Human-Centered & Neuro-Symbolic Decision Support in Circular Product Engineering (HUMAN DECISION) project. It comprises conceptual considerations, the selected research approach, the current state of implementation, and an outline of ongoing and future work. First, some thoughts on consideration of CE criteria are introduced. Next the chosen Neuro-Symbolic (NeSy) approach and research methodology are documented and substantiated by a system architecture that illustrates the interaction of symbolic models, formal verification components, and learning-based methods. Subsequently, the implementation section describes the current realization of the proposed architecture. Particular emphasis is placed on the phases of information extraction from CAD files, their structured representation using graphs and ontologies, and the ongoing development of a neural model. This model is intended to generate valid AND/OR Graphs (AOG) from the verified input in a non-algorithmic, learning-based manner. The paper concludes with a summary of the current research status and an outlook on further development steps within the HUMAN DECISION project.

3 Consideration of Circular Economy

At the current stage of our research, several CE criteria are discussed and identified as having potential for integration into the HDSS. As these criteria have not yet been implemented, this section reflects our present perspective which CE considerations could be incorporated. These include the minimization of disassembly steps required to access wear or core components, as it is assumed that this directly influences the effort associated with repair, refurbishing, and remanufacturing. The installation depth of critical components is regarded as a potentially suitable quantitative metric within the AOG, since it can be derived from structural product data and enables an assessment of component accessibility. Irreversible joining methods, such as adhesive bonding or welding, are considered limiting with respect to disassembly and reuse and are therefore either avoided or explicitly modeled, whereas reversible joining techniques (e.g., screwing or plugging) are viewed as beneficial for supporting repeated assembly and disassembly cycles. Furthermore, it is assumed that considering component accessibility can contribute to more efficient planning of maintenance and end-of-life processes. Product modularity with respect to the production environment is treated as another relevant aspect, as modular structures

are expected to facilitate the targeted replacement and reuse of subassemblies while accounting for existing production constraints. On the process level, the energy and time required for individual assembly and disassembly operations, as well as the number of required tool changes, are regarded as potentially suitable and generally measurable indicators, given their direct relation to resource consumption and environmental impacts across the product lifecycle.

4 Approach and Methodology

The suggested HDSS system architecture integrates NeSy methods with human-centered decision support. Figure 1 illustrates the activity diagram for generating an AOG and corresponding precedence graphs. The workflow (1) begins with the Graph Constructor, which preprocesses CAD data to generate graphs representing directly connected parts (adjacency, liaison, or part contact graphs) and geometric, structural information. Generated graphs in workflow (2) form an initial data structure and are passed to the Converter. Within workflow (3), the Converter transforms graphs into a Petri-Net structure. These Petri Net structures are used as a formal, visual, and mathematical framework to model complex assembly processes. They explicitly represent dependencies between actions, resource constraints, and completion conditions, making it easier to analyze system behavior. Mainly, in assembly sequence planning, Petri Nets are used to model parallel and concurrent activities, thereby enabling efficient planning of assemblies with multiple simultaneous operations. Moreover, Petri Nets enable deadlock detection, ensuring feasible assembly sequences, and hierarchical modeling, which helps manage complex systems by breaking them into manageable subassemblies. Overall, they offer clarity, rigor, and flexibility in assembly sequence planning. To validate the Petri Net-based models of the assembly process with respect to assembly operations and to detect deadlocks, within workflow (4), a formal theorem prover programming language is used (Lean [4]). The Converter transforms the Petri Net-based model into Lean code.

The Prover Agent in workflow (4) receives the Lean code generated by the Converter within workflow (3) and verifies whether the graphs comply with Petri-Net theory. In other words, if the output is valid, it means the theorem has been proved, and if it is invalid, it means the theorem has not been proved. Additionally, the Prover Agent verifies that the generated graphs satisfy predefined constraints, such as assembly rules, gravitational stability requirements, and other domain-specific conditions. The validation results, including any identified constraint violations, are passed within workflow (4) directly to the Graph Constructor in order to eliminate the violations. If the Lean prover proves that all generated graphs are valid, the Ontology Creator uses this information to generate structured ontologies representing these graphs as instances of the Petri-Nets ontology. The Ontology Creator also generates ontologies that describe the geometric and physical properties of the CAD model, and its components, such as wearing parts. Essentially, the Ontology Creator ensures that the data flowing into the ontologies is verified, forming a reliable foundation for symbolic reasoning.

The Reasoner Agent checks the consistency of these ontologies (semantic reasoning output (ontology consistency --> true, false)). If they are consistent, within workflow (5) they are stored in a knowledge graph. Within workflow (6), if the Reasoner Agent detects any inconsistencies, it reports them to the Ontology Creator, while simultaneously logging the inconsistencies and providing explanations for ontology's inconsistency. This is important so that the NeSy AI can retain experience and reuse it for learning, improving future ontology creation. If necessary, the Prover Agent also receives inconsistencies for iterative refinement. In workflow (7), the Query Engine facilitates retrieving specific information from the knowledge graph, such as part properties, constraints, and connectivity relationships.

The Pairwise Feature Calculator activity within workflow (7) processes the retrieved data to compute critical metrics, such as moments of inertia, centers of gravity, gravitational stability, optimal disassembly sequences, and incompatible part pairs. These metrics are then integrated into the knowledge graph as enriched data. The Fine-Tuning SLM Agent subsequently uses this enriched within workflow (8) and (9), validated data to generate training data and fine-tune the Small Language Model (SLM). This ensures that the SLM can effectively reason over the verified structural and constraint knowledge.

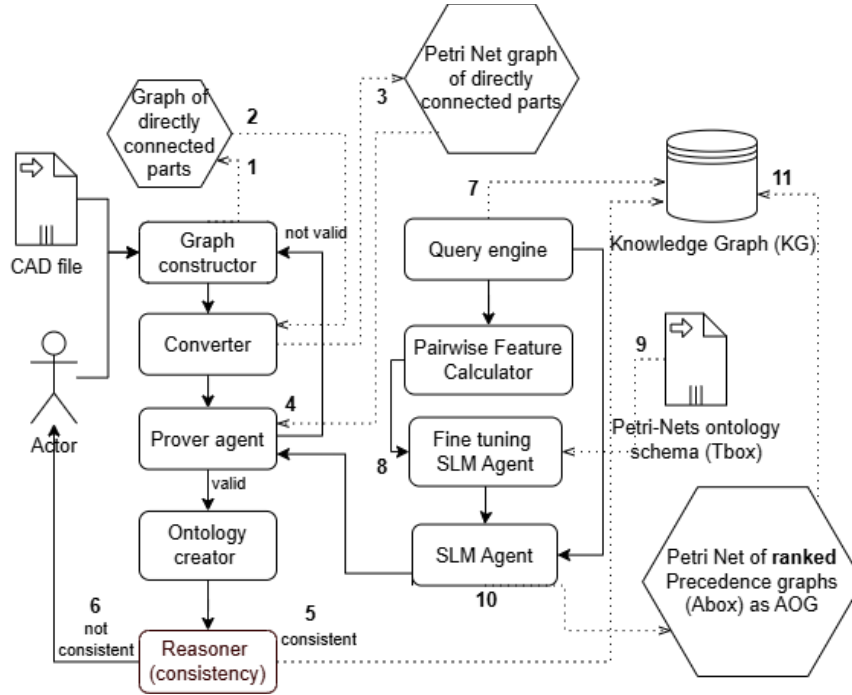


Figure 1. Activity diagram for assembly sequence planning

The SLM Agent integrates information from multiple sources, including a knowledge graph containing Petri-Net ontology schema, computed pairwise features, and constraint representations (e.g., wearing parts, gravitational stability). It also considers human feedback. Using this data, the agent then analyzes the generated graphs, which are serialized as instances of the Petri-Net ontology, and selects appropriate graph transformation actions. Each transformation is applied iteratively, while respecting constraints to optimize for resource-efficient assembly and disassembly. In this context, a transformation represents the task the neural model performs when iteratively generating an AOG, referring to the incremental conversion of one graph into the next: $\text{graph}_1 \rightarrow \text{graph}_2 \rightarrow \text{graph}_3 \rightarrow \dots \rightarrow \text{graph}_n \rightarrow \text{final AOG}$. Within workflow (10), the SLM Agent produces an AOG as its output. This graph is converted into ontologies, which constitute the final AOG that are instances of the Petri-Nets ontology schema. In the workflow (11), generated ontologies are validated for consistency by the Reasoner Agent. If consistent, these ontologies are stored in the knowledge graph. The AOG encodes an optimized sequence of assembly and disassembly operations and integrates ranking information that reflects priorities, constraints, and sustainability objectives. This makes the AOG directly usable for engineering decision support and downstream planning tasks.

This architecture highlights a continuous, structured flow of data and information from raw CAD input to validated, actionable assembly plans. It demonstrates how reasoning, theorem proving, knowledge representation, and adaptive learning converge to support circular product development.

5 Implementation

This section describes the current status of the HDSS implementation. The work can be divided into two main areas. On the one hand, there is the preparation of the input for the SLM agent (based on a CAD file). On the other hand, there is the provision of the SLM-Agent itself. Both aspects are discussed in the following chapters.

5.1 SLM Agent Input

All project materials are hosted in the *Human-Decision* GitLab project [1], which includes *data*, *model*, and *source-code* repositories. The data stored in these repositories are used to train neural networks for generating AOG and related precedence graphs.

The *data* GitLab repository consists of a Petri net ontology and data generated from STEP files. For example, the *step-knowledge-graph* GitLab repository contains the *der HundD30* and *AIMFREE_Batterypack* knowledge graphs generated from STP files, along with a set of SPARQL queries that can be used to query these knowledge graphs. The *Assembly Sequence Planning Knowledge Graph* GitLab repository, within the *data* repository, contains the Petri net ontology used in *Human-Decision* project as a schema for describing part contact graph, which represents directly connected parts (see Steps 1–3 in Figure 1), as well as AOG (see Step 10 in Figure 1).

The *source-code* GitLab repository contains a Python project that analyzes STEP (Standard for the Exchange of Product Data) files to create contact matrices indicating which parts touch or are connected to each other. It visualizes the results as interactive graphs using NetworkX, generates instances of the Petri net ontology representing these contact matrices (Step 1 in Figure 1), and exports the results in multiple formats (PNG, TTL, and Lean). In addition, the Python project generates Lean code that implements the part contact graph (Step 4 in Figure 1). The *source-code* repository also includes a Lean project that validates the generated Petri net of the part contact graph using the Lean proof assistant. All data generated in the aforementioned repositories are used for neural network fine-tuning. The *Human-Decision* GitLab project also includes a model GitLab repository that provides an implemented example demonstrating how an assembly process can be modeled using existing ontologies. Figure 2 presents a part contact graph generated from a Hund STEP file using software available in a GitLab repository [2]. This software automatically converts this graph into an ontology that is an instance of the Petri-Net ontology. A visual snapshot of the resulting ontology is shown in Figure 3.

As illustrated in Figure 3, part B shows a portion of the Hund CAD consisting of four distinct components, while part A presents a visual representation of the corresponding Petri-Net ontology that governs the part contact graph for this portion of the Hund CAD model. Each part of the CAD model shown in part B of Figure 3 is represented as a black node in part A. These black nodes correspond to instances of the Place class in the generated Petri-Net ontology. The double lines shown in part A represent transitions, which are instances of the Transition class, while the arrows denote instances of the Arc class in the Petri-Net ontology.

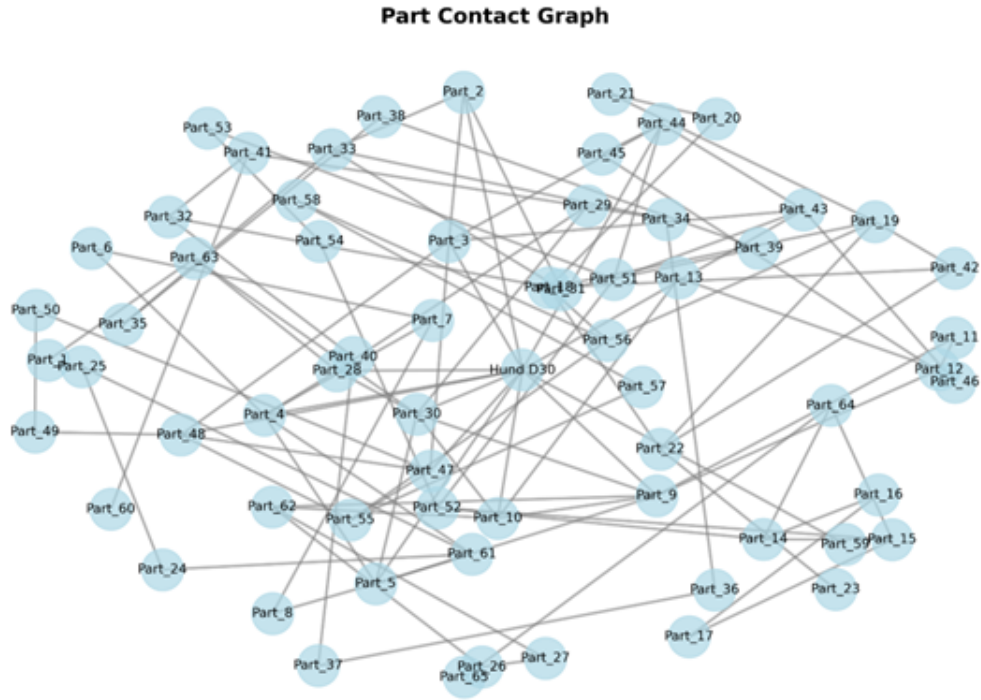


Figure 2. Part contact graph generated automatically from STEP file

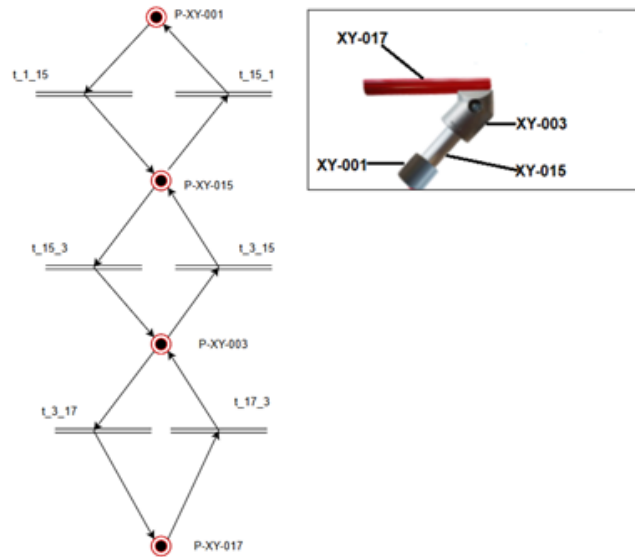


Figure 3. Part A visualizes a Petri-Net ontology of the part-contact graph for the Hund CAD model shown in Part B

The Petri-Net ontology derived from the Hund CAD model is available as a TTL file in the results folder of the GitLab repository [2] and has successfully passed a consistency check using the HermiT reasoner [3]. In addition to the TTL file, the software [2] generates Lean code that represents the part contact graph shown in Figure 2. This Lean code is also available in the same results folder of GitLab repository [4]. The correctness of the Petri-Net derived from the Hund CAD model is validated using the Lean proof assistant [17].

5.2 SLM Agent

Our approach is to learn AOG for assembly sequence planning by combining knowledge engineering, and neural models (see section 3, Fine Tuning SLM Agent). Multiple heterogeneous inputs are used,

including part contact graphs extracted from CAD models, product geometry, and topological constraints represented with the OntoSTEP ontology, and a Petri-net ontology schema that defines the formal structure of valid AOG; CE data are intended as an additional numerical input in a later stage of the project. All ontology-based inputs are transformed into representations suitable for machine learning. Recent advances in LLMs for CAD demonstrate their potential for automating design workflows through code generation and iterative self-improvement [5, 6]. We adapt these methodologies to assembly sequence planning, where the neural model learns to generate valid graph transformations grounded in ontological constraints. However, even this adaptation process necessitates a sufficient amount of training and test data. We propose the following combined training pipeline to overcome challenges of data scarcity, model limitation, data augmentation, and evaluation.

5.2.1 Training Pipeline

Our approach follows a four-phase pipeline designed to address the scarcity of annotated assembly planning data:

Phase 1: Gold-Standard Collection. We curate a reference dataset of validated assembly sequences from three sources: (i) expert-annotated sequences for existing assemblies including toy examples of knife, and dog assemblies, (ii) sequences generated and verified by the tools like ASAP or NeSy models like LLM4CAD [8, 9], and (iii) sequences validated through the Lean theorem prover pipeline. Each sequence should ideally include metadata on optimality status, constraint satisfaction, and circular economy annotations.

Phase 2: Zero-Shot Benchmarking. Using the gold-standard set, we evaluate existing LLMs on AOG generation without fine-tuning. Models receive a structured prompt containing the Petri-Net state (serialized as JSON-LD), enabled transitions, and task instructions. We benchmark models such as GPT-4, Claude-3.5, Qwen2.5-Coder, and Llama-3 to identify which architectures best handle structured graph reasoning. This phase establishes baselines and informs model selection.

Phase 3: Knowledge Distillation. The best-performing model undergoes knowledge transfer from ASAP's algorithmic planning. Training pairs take the form where is the serialized Petri-Net state, is the set of enabled transitions, and is ASAP's selection. This supervised fine-tuning teaches the model to replicate expert-level planning decisions.

Phase 4: Synthetic Data Augmentation. To address the challenge of missing data, we initially pursue an approach in which AOGs are generated for a selection of CAD files using existing software. Following BlenderLLM's self-improvement methodology [5], our training set can be expanded based on synthetic data augmentation as follows: (1) The model generates candidate sequences for held-out assemblies, (2) Sequences validated via Petri-Net simulation and Lean verification, (3) Valid sequences filtered by quality (length, stability), (4) High-quality generations augment training data. This cycle continues until validation performance plateaus.

5.2.2 Evaluation Framework

The evaluation focuses on criteria relevant to manufacturing process planning. First, the correctness of generated assembly sequences is verified by checking whether all operations are feasible and whether the final state represents a completed assembly. Second, process efficiency is assessed by comparing the number of operations in the generated sequences with known reference or optimal solutions. Additional measures evaluate the reliability of the method, such as the probability of finding a valid assembly plan within a limited number of planning attempts. The proposed approach is compared against baseline methods including heuristic planners, classical search-based planning techniques, and data-driven methods without formal verification. We adopt metrics from recent CAD-LLM benchmarks [5, 9, 10], adapted for assembly planning as follows:

Table 1: Evaluation metrics for assembly sequence generation

Metric	Definition	Verification
Validity	Sequence executes without error	PN Simulator
Completeness	Final marking is complete	
Optimality Ratio	$ S / S^* $ vs. reference	Gold-standard
Success@1	Valid on first attempt	Binary
CAD F1	balance between Entity Accuracy and Sketch Accuracy	[0,1]

This evaluation table shall be enriched with production and engineering KPIs in the future. After defining appropriate KPIs, we now move to baseline comparisons. Our intended baseline models set include: (i) random sampling from enabled transitions, (ii) greedy selection by part connectivity, (iii) zero-shot LLM prompting, and (iv) BFS-optimal search for small assemblies. All experiments will use equal random seeds like $N=5$ with paired t-tests for significance ($p<0.05$).

5.2.3 Expected Results and Failure Analysis Plan

The expected result is a planning system that consistently produces feasible assembly sequences while achieving near-optimal process lengths. By learning from structural similarities across different assemblies, the approach is expected to generalize to new products and scale to assemblies with a large number of components. The explicit integration of symbolic verification ensures that generated process plans comply with all assembly constraints, which is a key requirement for industrial applicability. To support systematic improvement and industrial deployment, failure cases are explicitly analyzed. Failures are categorized into infeasible operation selections, deadlock situations, valid but inefficient assembly sequences, and planning timeouts for complex assemblies. For each failure case, the corresponding assembly state and Petri Net configuration are examined to identify structural or procedural causes. This analysis provides insights into typical planning bottlenecks and guides further refinement of both the data and the planning model.

6 Outlook

Based on the current state of implementation, we can demonstrate that the information initially specified in the system architecture can be successfully extracted from CAD data and provided as structured input to the SLM agent. As the models responsible for computing AOGs from this input are still under development, it is not yet possible to draw definitive conclusions regarding the full functionality of the envisioned HDSS. In addition to addressing performance requirements related to input preprocessing and model execution, a key area for further development is the integration of additional contextual information (further CE relevant information) that can be considered during AOG computation. Potential extensions include the analysis of joining technologies and material choices, as well as the incorporation of production-related constraints, such as the manufacturing processes and infrastructural conditions of the target production facility (see section 3).

However, AOGs are not definitive production plans, but rather serve as validated templates for production systems. They encode assembly and disassembly sequences, but their practical implementation depends largely on variable characteristics of the production environment. These include, in particular, the production layout, the capacities of plants and workstations, the available

equipment, the degree of automation, the flexibility of processes and sequences, and the necessary retooling effort. Digital twins, for example in the form of scenario analysis systems [16], enable the systematic combination of a multitude of relevant configuration parameters and the identification of optimal characteristics. Based on the AOGs, corresponding production parameters can be configured. This allows alternative production scenarios to be generated and evaluated, for example with regard to optimal resource utilisation. However, the configuration space is characterised by inherent conflicts of interest: maximising throughput, for example, can conflict with changeover capability or accessibility for disassembly. A higher degree of automation increases process stability, but is often accompanied by rising changeover costs. In addition, local optimisations in line-free production systems can lead to increased global resource consumption.

Against this, we plan to expand our decision support system to include a multi-objective optimisation approach that can be used to evaluate these parameters. By incorporating user feedback, we ensure that the scenarios generated remain practical and feasible. The AOG herein acts as a central link between circular product development and adaptive production planning, ensuring the consistency of early design decisions throughout the entire product life cycle.

7 Conclusion

The NeSy approach, enhanced by the integration of theorem provers and inference engines as external reasoning components, enables a wide range of industrial applications. In manufacturing, and in particular in assembly and disassembly planning, the generation of AOGs represents a central task. Classical algorithmic approaches construct AOG that guarantee logical correctness for a given input configuration [11]. However, these methods are typically limited to rule-based transformations and lack the ability to learn recurring transformation patterns across different products, system configurations, or lifecycle scenarios, which restricts their scalability and adaptability [12].

The proposed NeSy framework addresses these limitations by tightly coupling symbolic reasoning and formal verification with learning-based graph transformation mechanisms. Verified structural representations, such as Petri Nets and precedence graphs, are embedded into a knowledge graph and enriched with semantic information, physical properties, and operational constraints. A set of interconnected, agentic SLMs [13] operates on these representations and is supported by external reasoning tools, including a theorem prover such as Lean [14] and description logic reasoners. This combination ensures that all generated graph transformations remain consistent with formally specified constraints, such as assembly feasibility, gravitational stability, and circularity-related requirements. Graph transformations are performed within a reinforcement learning framework [15], allowing the SLM agents to explore the space of feasible transformations and to learn transformation strategies across different product and production contexts. Rather than producing a single static solution, the system generates ranked and validated AOGs that reflect multiple, potentially conflicting objectives, such as efficiency, reconfigurability, and disassembly accessibility. Human feedback is integrated into this loop to guide exploration, resolve ambiguities, and ensure alignment with engineering intent.

While manufacturing and assembly planning represent a primary application domain, the underlying principles of the framework are not domain-specific. The combination of symbolic constraint enforcement, knowledge graph-based representations, and learning-driven graph transformations is applicable to other complex, graph-based decision problems, such as treatment planning in medicine or curriculum and competency modeling in education. Across domains, the NeSy approach enables flexible, adaptive, and formally grounded handling of tasks that are difficult to address with purely algorithmic or purely data-driven methods.

Acknowledgement: Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – HUMAN-DECISION (543081196), part of SPP 2443: Hybride Entscheidungsunterstützung in der Produktentstehung (521148212)

References

- [1] Human-Decision GitLab repository: <https://gitlab.com/TIBHannover/human-decision>, Accessed January 17th, 2026.
- [2] Step to contact matrix extraction, Python project: <https://gitlab.com/TIBHannover/human-decision/source-code/STEP-contact-matrix-extraction/>, Accessed January 20th, 2026
- [3] Glimm, B., Horrocks, I., Motik, B., Stoilos, G. and Wang, Z., 2014. HermiT: an OWL 2 reasoner. *Journal of automated reasoning*, 53(3), pp.245-269.
- [4] Implementation Petri-Nets in Lean proof assistant. Lean project: <https://gitlab.com/TIBHannover/human-decision/source-code/theorem-proving>, Accessed January
- [5] Yuhao Du, Shunian Chen, Wenbo Zan, Peizhao Li, Mingxuan Wang, Dingjie Song, Bo Li, Yan Hu, Benyou Wang (Arxiv 2024). “BlenderLLM: Training Large Language Models for Computer-Aided Design with Self-improvement”, arXiv:2412.14203
- [6] Licheng Zhang, Bach Le, Naveed Akhtar, Siew-Kei Lam, Tuan Ngo (Arxiv 2025). “Large Language Models for Computer-Aided Design: A Survey.”, arXiv:2505.08137.
- [8] Yunsheng Tian, Karl D.D. Willis, Bassel Al Omari, Jieliang Luo, Pingchuan Ma, Yichen Li, Farhad Javid, Edward Gu, Joshua Jacob, Shinjiro Sueda, Hui Li, Sachin Chitta, Wojciech Matusik (MIT CSAIL) (ICRA 2024). “ASAP: Automated Sequence Planning for Complex Robotic Assembly with Physical Feasibility.”, 2024
- [9] Xingang Li, Yuewan Sun, Zhenghui Sha, *Journal of Computing and Information Science in Engineering (JCSIE 2025)*. “LLM4CAD: Multimodal Large Language Models for Three-Dimensional Computer-Aided Design Generation.”, Feb 2025
- [10] Sifan Wu · Amir Khasahmadi · Mor Katz · Pradeep Kumar Jayaraman · Yewen Pu · Karl Willis · Bang Liu, *NeurIPS 2023 Workshop on Machine Learning for Creativity and Design*. “CAD-LLM: Large Language Model for CAD Generation.”, 2023.
- [11] De Mello LH, Sanderson AC. A correct and complete algorithm for the generation of mechanical assembly sequences. In: 1989 IEEE International Conference on Robotics and Automation 1989 Jan 1 (pp. 56-57). IEEE Computer Society. URL: <https://www.computer.org/csdl/proceedings-article/robot/1989/00099967/>
- [12] Moore NS, Cyr EC, Ohm P, Siefert CM, Tuminaro RS. Graph neural networks and applied linear algebra. *SIAM review*. 2025 Mar 31;67(1):141-75. URL: <https://epubs.siam.org/doi/10.1137/23M1609786>
- [13] Belcak P, Heinrich G, Diao S, Fu Y, Dong X, Muralidharan S, Lin YC, Molchanov P. Small language models are the future of agentic AI. *arXiv preprint arXiv:2506.02153*. 2025. URL: <https://arxiv.org/abs/2506.02153>
- [14] de Moura L, Kong S, Avigad J, van Doorn F, von Raumer J. The Lean theorem prover (system description). In: *Automated Deduction – CADE-25. Lecture Notes in Computer Science*, vol. 9195. Springer, 2015. pp. 378–388. DOI: https://doi.org/10.1007/978-3-319-21401-6_26.
- [15] He G, Parker S, Yoneki E. X-RLflow: Graph reinforcement learning for neural network subgraphs transformation. *arXiv preprint arXiv:2305.12345*. 2023. URL: <https://arxiv.org/abs/2305.12345>
- [16] Grunert, D., Göppert, A., Hort, S., Rachner, J., Grahn, L., & Schmitt, R. H. (2021). Modelling for Decision-making in Dynamical Line-less Assembly Systems. *Journal of Production Systems and Logistics* 1 (2021), 1. doi: <https://doi.org/10.15488/11327>
- [17] Implementation Petri-Net in Lean proof assistant: <https://gitlab.com/TIBHannover/human-decision/source-code/theorem-proving>, Accessed January 21st, 2026

From Rust to Reuse: Influence of Preload and Corrosion on the Mechanical Performance of Bolted Joints

Carolin Lange¹, Nehal Afifi², Dorian Gleß¹, Martin Benfer¹ Sven Matthiesen² and Gisela Lanza¹

¹ wbk Institute of Production Science, Karlsruhe Institute of Technology (KIT), 76131 Karlsruhe, Germany

² IPEK Institute of Product Engineering, Karlsruhe Institute of Technology (KIT), 76131 Karlsruhe, Germany

*Corresponding author. E-Mail address: carolin.lange@kit.edu; Tel.:01523/9502649

Abstract: Remanufacturing requires reliable assessment methods to evaluate the reuse potential of especially mechanically loaded components. Bolted joints are widely used in mechanical systems and play a critical role in assembly, disassembly, and remanufacturing processes. This study investigates the influence of preload, corrosion exposure on joint performance. Using M10 bolts, loosening torque and angle are measured after controlled tightening, salt spray corrosion, and reassembly with either reused or new bolts. Results show higher preload increases post-remanufacturing loosening torque and variability, while corrosion affects friction and torque distributions. Bolt replacement consistently improves torque retention. These experimental insights support data-driven remanufacturing decisions.

Keywords:

bolted joints, corrosion, loosening torque, remanufacturing, mechanical variability

1 Introduction

Under increasing pressure from resource scarcity, stringent environmental regulations, and the demand for economic efficiency, many industries are transitioning toward circular economy strategies [1]. The circular economy represents a fundamental shift from a linear “take–make–dispose” model to a closed-loop system, aiming to extend product lifetimes, retain material value, and reduce environmental impact. Within this framework, remanufacturing has emerged as a key enabler of sustainability and resource efficiency. [2]

Remanufacturing involves the systematic disassembly, inspection, cleaning, repair, and reassembly of end-of-life products in order to restore them to like-new or even enhanced functional performance. This approach is widely applied in sectors such as automotive engineering, heavy machinery, and industrial equipment, where components are often reused with warranties comparable to those of new products [3]. Despite its economic and ecological benefits, remanufacturing poses significant technical challenges, particularly with respect to ensuring the reliability and structural integrity of reused components [2], [4]. Among these, bolted joints play a critical role, as they must reliably withstand repeated assembly and disassembly cycles throughout a product’s lifecycle [5].

Bolted joints are especially susceptible to corrosion-induced degradation during service life. Corrosion alters surface conditions, friction coefficients, and material properties, thereby directly affecting the mechanical behavior of threaded connections. These changes have a direct impact on the torque–preload relationship, which is a key parameter for the functional integrity of bolted joints. One key manifestation of this degradation is the change in loosening behavior, commonly quantified by the breakaway (loosening) torque in comparison to the initial tightening torque. The ratio between breakaway torque and initial tightening torque is therefore a sensitive indicator of corrosion-related effects such as increased friction, thread adhesion, and surface damage caused by rust formation. Variations in this

torque ratio can indicate increased friction due to corrosion products, thread damage, or adhesion effects, all of which introduce uncertainty in joint preload and structural safety. In remanufacturing contexts, this uncertainty complicates decisions regarding component reuse, replacement, or requalification. [6]

Despite the practical relevance of this issue, there is a lack of systematic analysis linking corrosion progression in bolted joints to measurable changes in loosening torque behavior. Existing approaches often focus on visual inspection or qualitative corrosion assessment, which are insufficient for reliable mechanical evaluation. In particular, a quantitative comparison between initial tightening torque and measured breakaway torque under defined corrosion conditions is rarely addressed systematically. Consequently, a quantitative comparison of initial tightening torque and loosening torque under defined corrosion conditions is needed to better understand the mechanical implications of corrosion and to support robust decision-making in remanufacturing processes. [7]

The knowledge gained from this study can be directly fed back into product development. By understanding the conditions under which bolts can be safely reused, engineers can optimize joint designs for remanufacturing, improve maintenance protocols, and select materials and coatings that enhance long-term durability. This creates a data-driven basis for design decisions that support both sustainability and circular economy objectives. [8]

2 Methodology

This study employs a structured Design of Experiments (DOE) approach to systematically analyze the influence of corrosion on the loosening behavior of bolted joints. The experimental design is based on a full factorial $3 \times 3 \times 2$ layout with three primary controlled variables: (i) preload force (12 kN, 20 kN, 28 kN), (ii) corrosion exposure time (2 h, 4 h, 8 h), and (iii) configuration (new bolt, all old). Preload force (i) was varied at three levels (12 kN, 20 kN, 28 kN), representing under-tightened, nominal, and over-tightened assembly conditions [9]. Corrosion exposure time (ii) was likewise varied at three levels (2 h, 4 h, 8 h). These variables were selected because preload force directly governs the initial clamping condition and frictional state of the joint, corrosion exposure time represents the progression of surface degradation during service life, and the reassembly configuration reflects typical remanufacturing decisions regarding component reuse or replacement.

In addition to preload force (i) and corrosion time (ii), a third controlled variable was introduced during reassembly, namely the configuration (iii). Two remanufacturing scenarios are distinguished:

- **Replacement with new bolts:** This is the most common scenario, as bolts are widely available and inexpensive to replace. In contrast, the component into which the bolt is fastened often has a higher value, requires more CO₂-eq. and skilled labor to manufacture, and is less easily transported as a spare part. Therefore, preserving and reusing this component is of high interest.
- **Reuse of corroded bolts:** This scenario becomes particularly relevant when spare parts are unavailable or when the bolts are of a specific type that cannot be easily replaced

The experimental procedure follows a sequential workflow consisting of initial tightening, corrosion exposure, disassembly with loosening torque measurement, reassembly under defined remanufacturing conditions, and final loosening torque measurement. The loosening torque (LT) measured after the reassembly phase is defined as the target variable of this study, as it serves as an indirect indicator of the residual load-bearing capability of the bolted joint. Changes in loosening torque reflect alterations in frictional conditions, thread engagement, and preload retention, all of which are closely related to the joint's ability to maintain clamping force and mechanical integrity after remanufacturing. [9-11] By

comparing LT values across different preload forces, corrosion durations, and remanufacturing scenarios, the reuse potential of bolted joints can be quantitatively assessed.

The experimental framework describes a sequential procedure starting with initial assembly and controlled corrosion exposure, followed by disassembly and reassembly under defined remanufacturing scenarios, and concluding with final measurements. Throughout the process, experimental data are systematically collected and transformed into process and product data, providing the foundation for modeling and functional model development [8].

3 Experimental Setup

Standard M10 × 1.5 mm bolts in accordance with ISO 898-1, property class 8.8, were used as test specimens due to their widespread industrial application. Each bolted joint consisted of one bolt, one DIN 934 M10 nut, two DIN 436 square washers (11 × 30 × 3 mm), and one DIN 125 round washer (10.5 × 20 × 2 mm). An overview of the assembled joint prior to corrosion exposure is shown in Figure 1.



Figure 1: Experimental bolt assembly before corrosion: M10×1.5 mm ISO 898-1 Class 8.8 bolt, two DIN 436 square washers (11×30×3 mm), one DIN 125 round washer (10.5×20×2 mm), and a DIN 934 M10 nut

The overall experimental workflow for assembly, corrosion exposure, and remanufacturing assessment of bolted joints is illustrated in Figure 2. Prior to assembly, all bolts, nuts, and washers were cleaned for two minutes in an ultrasonic bath to remove surface contaminants that could influence friction and corrosion behavior. The components were then assembled and tightened using a Kistler INSPECTOR 5413-5103 electronic torque wrench. Tightening was performed to achieve the predefined preload forces of 12 kN, 20 kN, or 28 kN. Torque–angle data were recorded during tightening to ensure consistent assembly conditions. After initial tightening, the specimens were stored for 24h for settling. Later they were subjected to accelerated corrosion in a salt spray chamber for 2 h, 4 h, or 8 h. Corrosion was induced using a 15% hydrochloric acid solution. Custom-designed 3D-printed holders were employed to ensure uniform orientation and full exposure of the bolted joints to the corrosive mist. Following corrosion exposure, the specimens were dried and disassembled. The loosening torque was measured using the same Kistler torque wrench. During each disassembly process, the electronic torque wrench continuously recorded the torque–angle curve $M(\varphi)$. From this curve, the peak torque value corresponding to the breakaway event was identified and extracted. This peak value was then assigned to the respective bolt ID to ensure traceability throughout the dataset. The joints were then reassembled according to one of the two defined remanufacturing scenarios. In both cases, the joints were retightened to the same preload torque used during the initial assembly (12 kN, 20 kN, or 28 kN).

After reassembly, all specimens were stored for a minimum of 24 hours in a controlled environment at 20 °C and 55% relative humidity to allow for settling effects. Subsequently, the joints were disassembled again, and the final loosening torque and angle was recorded.

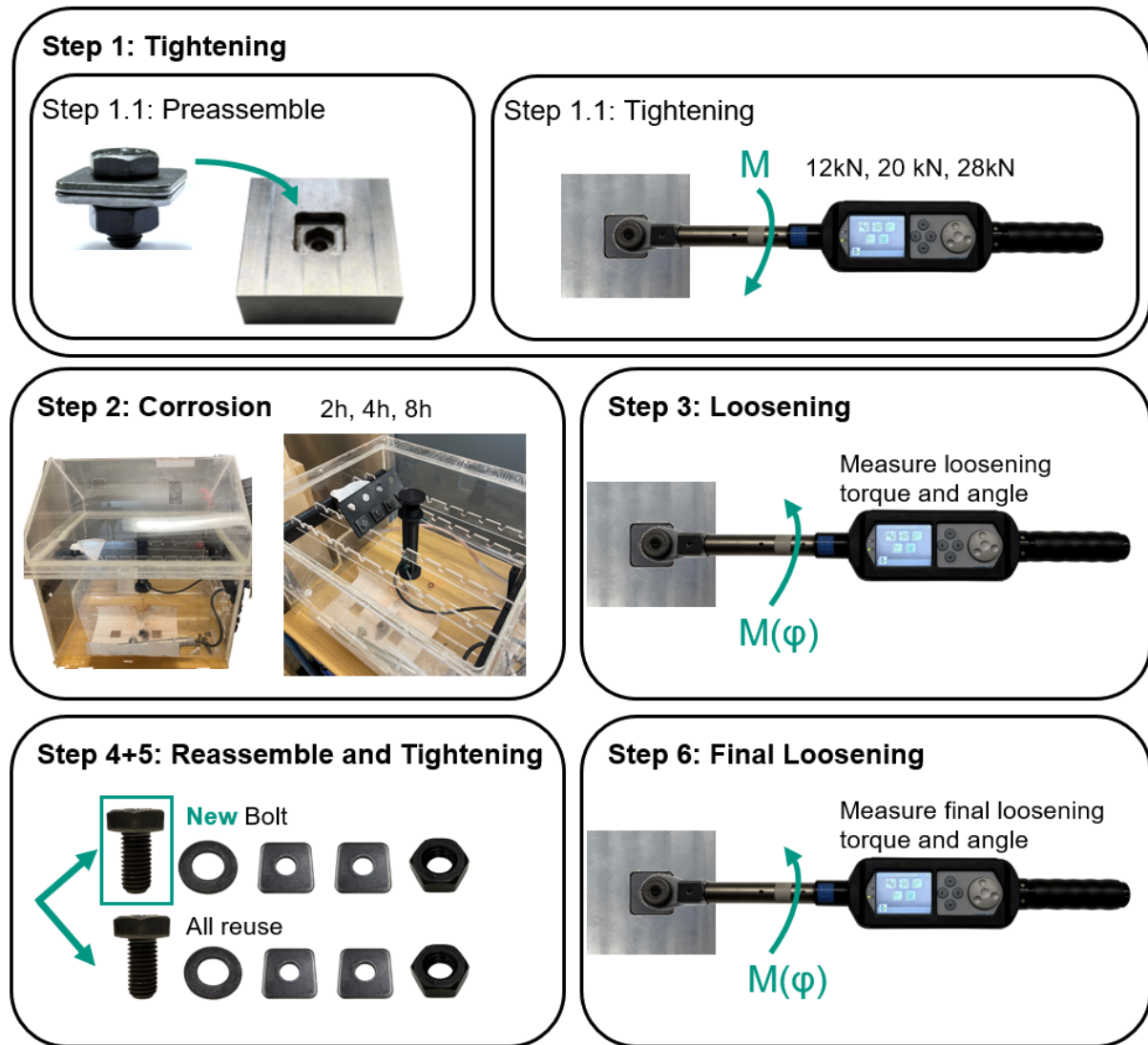


Figure 2: Experimental workflow for assembly, corrosion and remanufacturing assessment of bolted joints

Table 1: Experimental Design: Overview of experimental steps, controlled variables, recorded measurements, and sample sizes for corrosion and remanufacturing assessment of bolted joints.

Step	Controlled variable	Recorded value	Sample Size
1. Tightening	Preload force (12 kN, 20 kN, 28 kN)	Tightening angle (peak) Tightening torque (peak)	51 x 3 =153
2. Corrosion	Duration (2h, 4h, 8h)		(17x3) x3=153
3. Loosening		Loosening angle (peak) Loosening torque (peak)	153
4. Reassemble	New bolt All old		(10 [new bolt] + 7 [all old]) x3x3=153
5. Tightening		Tightening angle (peak) Tightening torque (peak)	153
6. Final Loosening		Loosening angle (peak) Loosening torque (peak)	153

The experimental workflow was structured into sequential steps, each associated with specific controlled variables and recorded measurements. Table 1 provides a detailed overview of these steps, indicating the variables that were systematically varied, the corresponding measurements collected at each stage, and the total number of samples for each configuration. This structured approach ensures that the influence of preload force, corrosion exposure, and reassembly condition on loosening torque and other mechanical responses can be quantitatively assessed across the full set of specimens.

4 Results

This study investigated the influence of preload force, corrosion exposure time, and remanufacturing method on the average percentage increase in loosening torque after reassembly. Data were grouped into 18 experimental conditions (3 preload levels x 3 corrosion durations x 2 reassembly configurations), allowing clear trends to be identified.

4.1 Correlation Between Loosening Torque, Tightening Torque and Angle

To analyze the relationship between applied tightening torque and loosening torque, Figure 3 presents a comparison of the peak tightening torque applied in Step 1 to achieve the three target preload levels with the corresponding initial loosening torque measured in Step 3. This visualization allows assessment of how different preload forces influence the initial mechanical behavior of the bolted joints and highlights the variability introduced by corrosion exposure during Step 2. The statistical analysis confirms the robustness of this trend, as all preload groups exhibited statistically significant differences (12 kN: $p = 0.0068$; 20 kN: $p = 0.0056$; 28 kN: $p = 0.00005$).

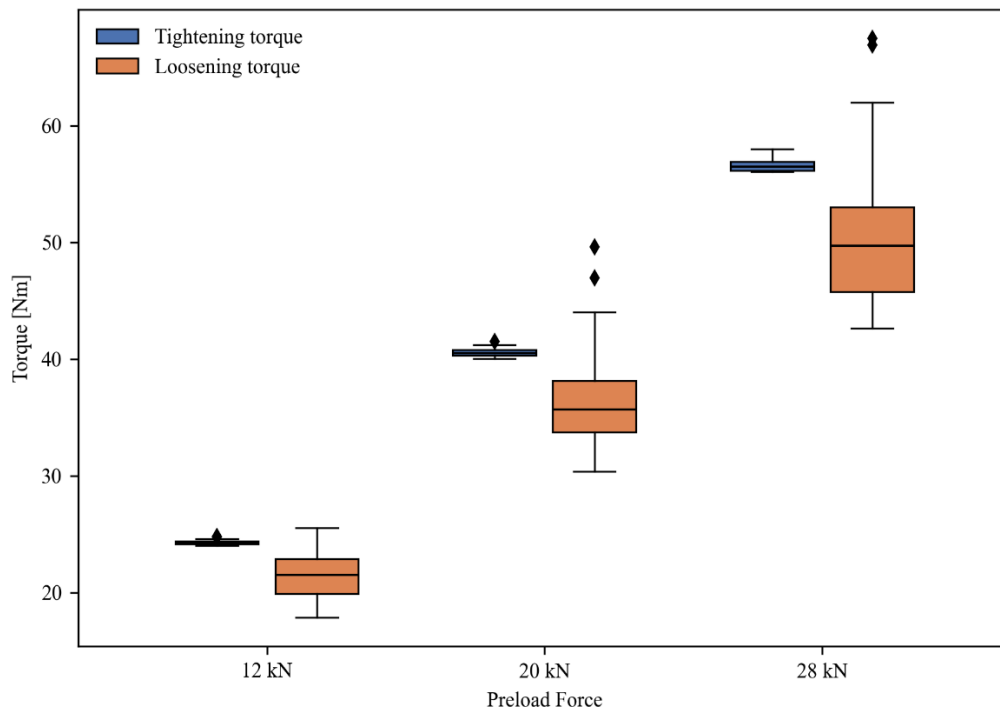


Figure 3: Boxplot of Tightening Torque (peak) of Step 1 and Initial Loosening Torque of Step 3 under three preload forces

Figure 4 presents a boxplot comparison of the peak tightening angle applied in Step 1 to achieve the three target preload levels with the corresponding initial loosening angle measured in Step 3. As expected, the median tightening angle increases with increasing preload, reflecting the additional rotation required to overcome higher thread friction and the friction between the bolt head and the washers. The distribution of tightening angles also becomes wider as the preload force increases,

indicating greater variability in the required rotation to reach higher clamping forces. This broader spread reflects the combined effects of friction variability and differences in thread engagement at elevated preload levels.

In contrast, the loosening angle is consistently lower than the corresponding tightening angle and shows a more compact distribution across all preload levels. The smaller variability reflects minor plastic thread deformations and stabilized friction after corrosion. Overall, while higher preload requires greater tightening rotations with a wider distribution, the initial loosening behavior remains relatively stable and less variable due to mechanical and frictional adjustments during corrosion and assembly.

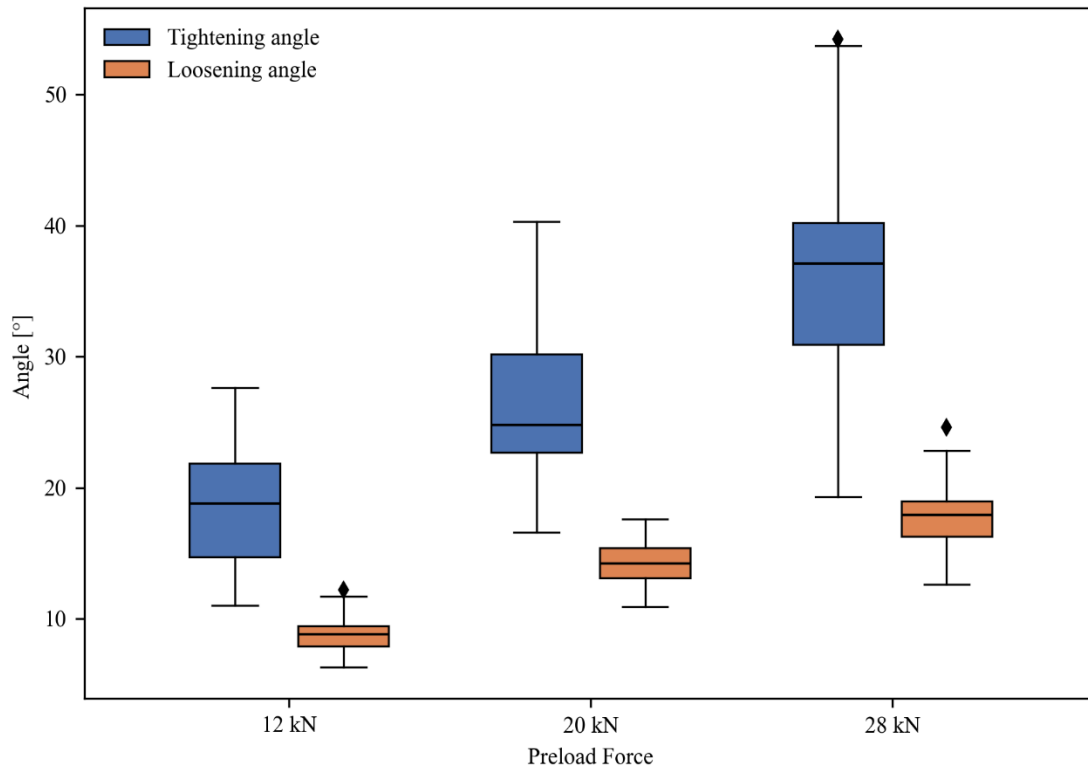


Figure 4: Boxplot of Tightening Angle (peak) of Step 1 and Initial Loosening Angle of Step 3 under three preload forces

Figure 5 shows the initial loosening torque measured in Step 3 following Step 1 tightening and Step 2 corrosion exposure for three different durations. Across all corrosion conditions, the main distribution of the initial loosening torque is wider than that of the applied tightening torque. The widest distribution was observed for the 4 h corrosion condition, indicating increased variability in joint behavior at this intermediate stage. This trend can be attributed to the evolving effects of corrosion at the bolted interface. Corrosion alters the surface condition of the threads and produces adhesion of corrosion products, which changes the local friction characteristics. After 2 h of exposure, corrosion is in its early stage, and its effect on all samples is relatively uniform, resulting in a narrower distribution. At 4 h, however, corrosion enters a transitional stage: the corrosion products are unevenly distributed across the bolt and nut surfaces, leading to greater variability in friction and, consequently, a wider spread in loosening torque values. After 8 h, the corrosion layer becomes more homogeneous and friction stabilizes, reducing the variability and narrowing the distribution of loosening torque compared to the 4 h condition.

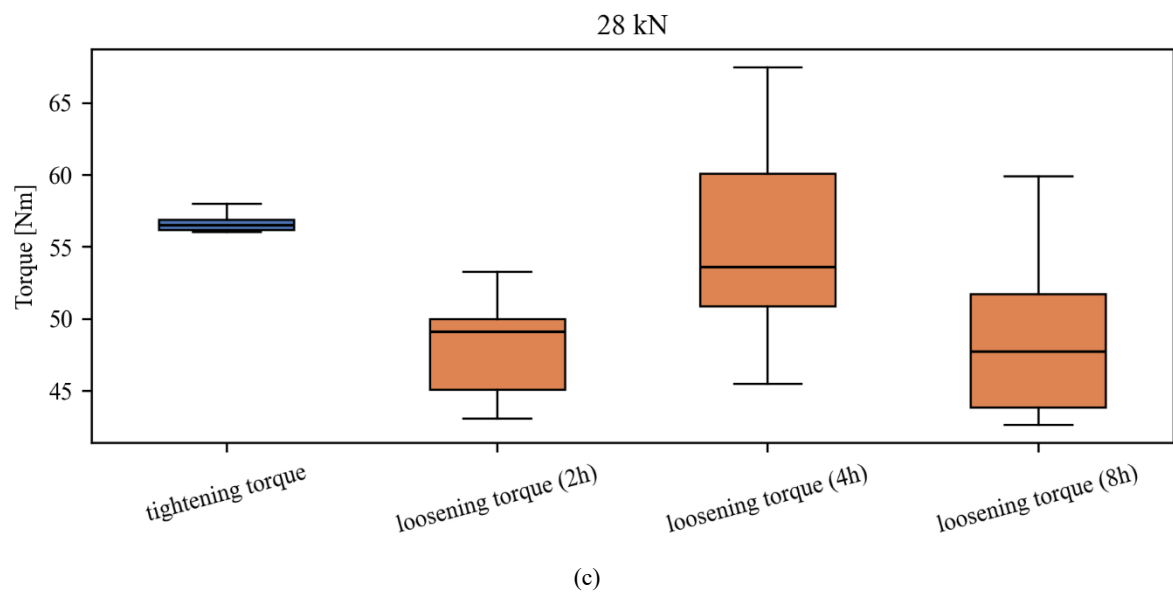
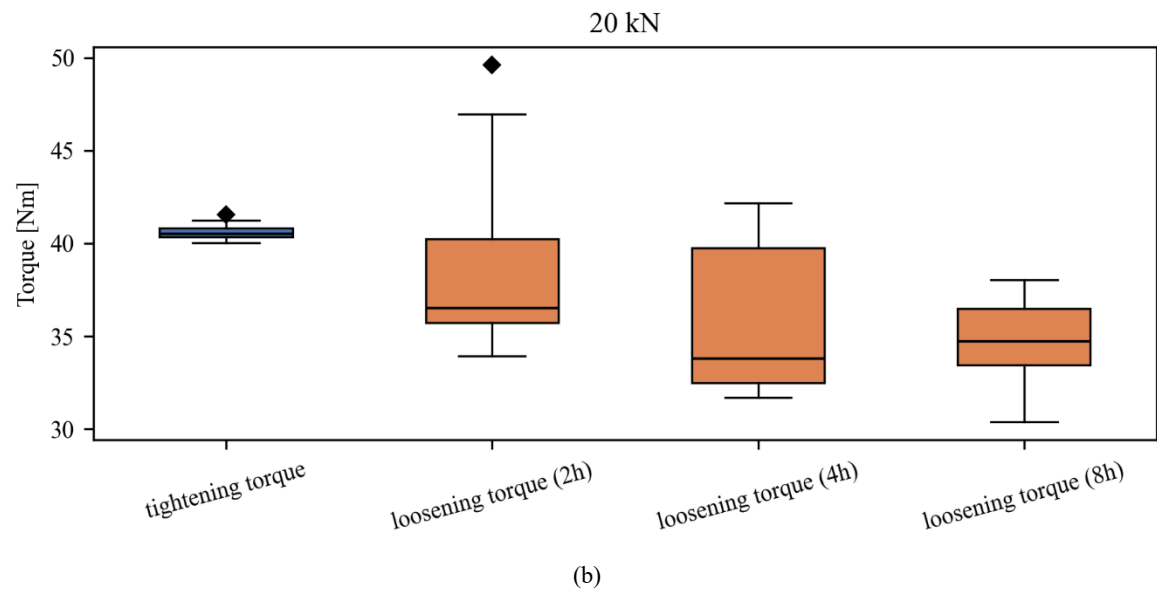
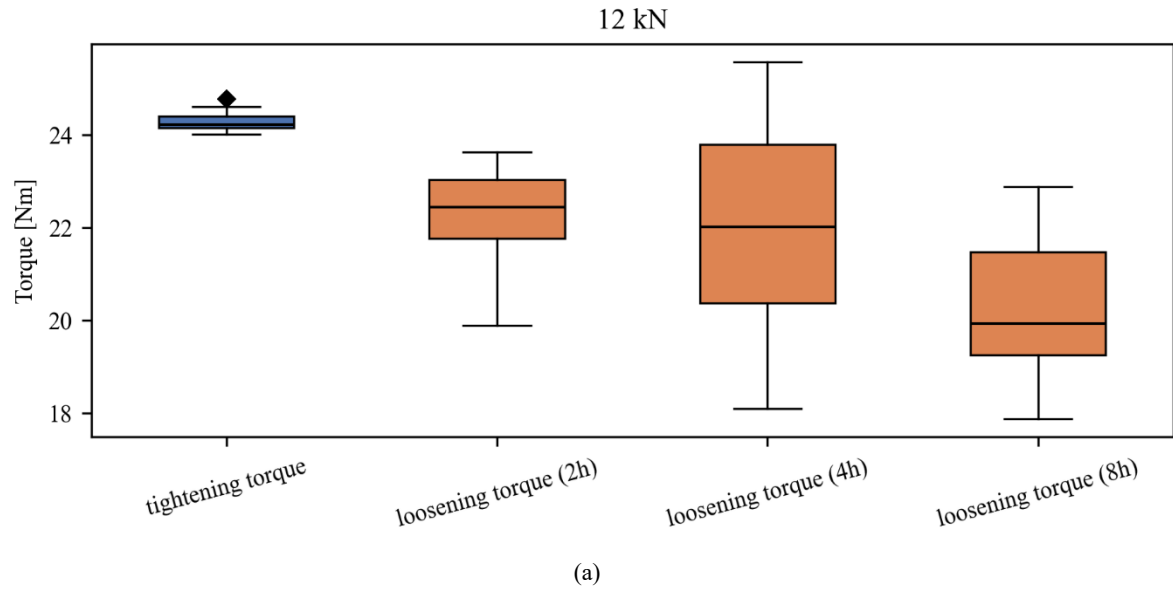


Figure 5: Boxplot of Tightening Torque (peak) of Step 1 and Initial Loosening Torque of Step 3 under Three Corrosion Durations and three Preload Forces (a) 12kN (b) 20 kN (c) 28 kN

4.2 Correlation between Loosening Torque and Final Loosening Torque

Of the 153 experimental samples, 145 exhibited an increase in final loosening torque (FLT) after reassembly compared to the loosening torque measured in Step 3. Only eight samples showed a decrease, with five occurring at a preload of 12 kN, two at 20 kN, and one at 28 kN. For each of the 145 samples showing an increase, the percentage increase in final loosening torque relative to the initial value was calculated. The average percentage increase for each experimental group is summarized in Table 2.

As preload force increased from 12 kN to 20 kN and 28 kN, the overall average percentage increase in final loosening torque rose correspondingly from 11.91 %, to 20.80 %, and 22.05 %, respectively. This trend suggests that higher preload forces promote greater torque retention after remanufacturing, likely due to improved initial clamping conditions and reduced micro-slipping at the interface under corrosion-induced stress.

Within each preload level, the effect of corrosion exposure time was also observed. Under 12 kN, the average percentage increase grew steadily with corrosion duration, from 6.26 % (2 h) to 13.07 % (4 h) and 16.38 % (8 h), indicating a positive correlation between corrosion exposure and torque increase. This is likely attributable to interface changes caused by corrosion, which increase friction during disassembly. In contrast, at higher preload forces (20 kN and 28 kN), the influence of corrosion time was less pronounced or even slightly diminishing. For instance, at 28 kN, the average increases were 25.44 %, 20.69 %, and 20.01 % for 2 h, 4 h, and 8 h, respectively.

Table 2: Average percentage increase in final loosening torque compared to initial loosening torque of step 3 - Loosening in each experimental group

Step 1	Step 2	Step 3		Step 4	Step 6 – Final Loosening Torque (FLT)				
Preload force	Corrosion level	Loosening Torque (LT)		Retightening configurations	average percentage increase $X = \frac{FLT-LT}{LT}$				
		$\overline{LT}$ [Nm]	Var(LT)		$\overline{FLT}$ [Nm]	Var(FLT)	X [%]		
12 kN	2h	22.21	1.82	New bolt	23.90	1.95	7.58	6.26	11.91
				All old	23.11	1.45	4.37		
	4h	21.85	5.23	New bolt	24.32	4.60	11.52	13.07	
				All old	24.91	4.82	15.30		
	8h	20.36	2.45	New bolt	23.99	2.94	16.72	16.38	
				All old	23.17	4.36	15.90		
20 kN	2h	38.76	19.77	New bolt	47.24	12.11	22.45	21.31	20.80
				All old	45.44	48.15	19.69		
	4h	35.62	15.45	New bolt	44.12	21.11	24.11	20.25	
				All old	40.45	3.19	14.73		
	8h	34.62	5.95	New bolt	43.30	11.80	26.13	20.83	
				All old	39.35	6.29	13.26		
28 kN	2h	47.99	10.09	New bolt	60.78	25.43	27.04	25.44	22.05
				All old	59.24	33.13	23.16		
	4h	55.45	42.84	New bolt	69.67	103.43	25.59	20.69	
				All old	62.05	24.27	13.69		
	8h	48.22	21.97	New bolt	57.41	18.13	19.58	20.01	
				All old	57.72	3.32	20.63		

Across all conditions, replacement of the bolt during remanufacturing resulted in a higher average percentage increase than reusing all original components. For example, at 20 kN and 4 h, bolt replacement yielded an average increase of 24.11 %, compared to 14.73 % when all original components were reused. This pattern was consistently observed across most experimental groups, confirming that bolt replacement enhances post-remanufacturing joint performance, likely due to improved thread engagement and reduced surface degradation. Overall, a greater increase in loosening torque indicates improved residual clamping strength and frictional resistance, both of which contribute positively to the durability and reliability of remanufactured bolted joints

5 Discussion

The results of this study provide clear insights into how preload force, corrosion exposure, and remanufacturing method influence the mechanical behavior of bolted joints. The observed trends in loosening torque and angle distributions reveal the combined effects of thread friction, surface degradation, and interface changes due to corrosion.

Higher preload forces were found to increase both the average loosening torque after reassembly and the variability of tightening torque and angle. Mechanically, this can be attributed to improved initial clamping conditions that reduce micro-slipping under corrosion-induced stress. However, the wider spread in tightening torque and angle at higher preload forces indicates that achieving target preload requires overcoming greater frictional resistance, including friction between the bolt head and washers, as well as along the thread surfaces. This variability highlights the importance of precise torque application in both original assembly and remanufacturing contexts. Corrosion significantly affects loosening behavior by altering surface conditions at the bolt–nut interface. These findings confirm that the stage of corrosion development critically influences both the mean and the scatter of loosening torque, emphasizing the need to consider corrosion history when evaluating the reuse potential of bolts.

The comparison between remanufacturing scenarios demonstrates that bolt replacement generally provides higher residual torque retention than reusing corroded bolts. This effect is likely due to the restoration of optimal thread engagement and the elimination of localized surface degradation, which mitigates frictional inconsistencies. In contrast, reused bolts carry accumulated wear and corrosion, which can limit torque recovery and increase variability. Therefore, while reusing corroded bolts may be feasible under controlled conditions, bolt replacement offers a more reliable approach for maintaining joint performance.

It is important to note the limitations of the current study. Only a finite number of samples were tested, which limits the statistical generalizability of the results. Furthermore, no direct measurements of mechanical strength or load-bearing capacity were performed; future studies could include tensile or fatigue tests to evaluate the actual structural performance of the joints. In addition, the experimental setup did not subject the bolts to vibrations, which are common in real-world applications. Vibrations could promote bolt loosening and allow deeper penetration of corrosion products, potentially altering torque behavior and variability. These factors should be addressed in follow-up experiments to better simulate in-service conditions.

Overall, these findings provide valuable insights into the influence of preload, corrosion, and remanufacturing method on bolted joint behavior. They also highlight the need for further testing under more realistic operational conditions to fully assess joint reliability and to develop predictive models for remanufacturing and maintenance strategies.

6 Summary

The experimental results demonstrate that both preload force and corrosion exposure significantly affect the loosening behavior of bolted joints. Higher preload forces generally lead to increased post-remanufacturing loosening torque and wider distributions of tightening angle and torque, reflecting greater frictional resistance and variability in thread engagement. Corrosion alters the surface condition of the threads, producing adhesion of corrosion products and modifying friction, which in turn influences both torque and angle measurements. The variability of initial loosening torque is greatest at intermediate corrosion durations. Bolt replacement during remanufacturing consistently results in higher average percentage increases in loosening torque compared to reusing all original components, highlighting the benefit of improved thread engagement and reduced surface degradation. Overall, these findings indicate that careful consideration of preload, corrosion exposure, and remanufacturing method is essential for predicting residual clamping strength and ensuring the reliability of remanufactured bolted joints.

Acknowledgments

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – "Functional modeling of bolted joints under uncertain product conditions for remanufacturing" - with the project number 525034540.

References

- [1] P. Majumder und H. Groenevelt, „Competition in Remanufacturing“, *Production and Operations Management*, Bd. 10, Nr. 2, S. 125–141, 2001, doi: 10.1111/j.1937-5956.2001.tb00074.x.
- [2] J. P. Jensen, S. M. Prendeville, N. M. P. Bocken, und D. Peck, „Creating sustainable value through remanufacturing: Three industry cases“, *Journal of Cleaner Production*, Bd. 218, S. 304–314, Mai 2019, doi: 10.1016/j.jclepro.2019.01.301.
- [3] M. Matsumoto, S. Yang, K. Martinsen, und Y. Kainuma, „Trends and research challenges in remanufacturing“, *Int. J. of Precis. Eng. and Manuf.-Green Tech.*, Bd. 3, Nr. 1, S. 129–142, Jan. 2016, doi: 10.1007/s40684-016-0016-4.
- [4] S. Sitcharangsie, W. Ijomah, und T. C. Wong, „Decision makings in key remanufacturing activities to optimise remanufacturing outcomes: A review“, *Journal of Cleaner Production*, Bd. 232, Mai 2019, doi: 10.1016/j.jclepro.2019.05.204.
- [5] J. Kurilova-Palisaitiene, E. Sundin, und B. Poksinska, „Remanufacturing challenges and possible lean improvements“, *Journal of Cleaner Production*, Bd. 172, S. 3225–3236, Jan. 2018, doi: 10.1016/j.jclepro.2017.11.023.
- [6] S. Mangold, E. Gerlitz, S. Zimprich, M. Friedmann, S. Matthiesen, und J. Fleischer, „Systematic quantitative investigation of the unscrewing process with regard to breakaway torque“, *Jnl Remanufactur*, Bd. 13, Nr. 1, S. 53–66, März 2023, doi: 10.1007/s13243-022-00120-x.
- [7] C. Jiang, W. Xiong, C. S. Cai, Y. Zhu, und J. Wang, „Preload loss of high-strength bolts in friction connections considering corrosion damage and fatigue loading“, *Engineering Failure Analysis*, Bd. 137, S. 106416, Juli 2022, doi: 10.1016/j.engfailanal.2022.106416.
- [8] N. Afifi, J.-P. Kaiser, A. Wettstein, G. Lanza, und S. Matthiesen, *Data-Driven Functional Modeling of Corroded Bolted Joints: A Framework for Remanufacturing*. 2025. doi: 10.1115/IMECE2024-145271.
- [9] *VDI 2230 Blatt 1 - Systematische Berechnung hochbeanspruchter Schraubenverbindungen - Zylindrische Einschraubenverbindungen*. 2015. Zugriffen: 20. Januar 2026. [Online]. Verfügbar unter: <https://www.vdi.de/mitgliedschaft/vdi-richtlinien/details/vdi-2230-blatt-1-systematische-berechnung-hochbeanspruchter-schraubenverbindungen-zylindrische-einschraubenverbindungen>
- [10] J. Shah, H. Wang, A. Mukherjee, J. Shah, H. Wang, und A. Mukherjee, „Investigating the Correlation Between Corrosion-Induced Bolt Head Damage and Preload Loss Using Ultrasonic Testing“, *Sensors*, Bd. 25, Nr. 14, Juli 2025, doi: 10.3390/s25144491.
- [11] R. G. Budynas, K. J. Nisbett, J. K. Nisbett, und J. E. Shigley, *Shigley's mechanical engineering design*, 10. ed. in SI units. in McGraw-Hill series in mechanical engineering. New York, NY: McGraw-Hill Education, 2015.

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 Koziolk ², Tobias Düser ¹, Albert Albers ¹

¹Karlsruhe Institute of Technology (KIT), IPEK – Institute of Product Engineering, Kaiserstr. 10, 76131 Karlsruhe

²Karlsruhe Institute of Technology (KIT), KASTEL – Institute of Information Security and Dependability, Am
Fasanengarten 5, 76131 Karlsruhe

³Karlsruhe 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

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 too 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	Precisely 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	Precisely 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].



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. Koziolok, 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. Koziolk 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. Koziolk 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. Koziolk 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. Koziolk, 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 Entscheidungen* Frankfurt, 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/>.

CAD-as-Code and Agentic AI for Function-Aware Engineering Design

Elias Berger^{1,2*}, Jan Mehlstäubl², Bernhard Saske¹, Kristin Paetzold-Byhain¹

¹ Dresden University of Technology, 01062 Dresden

² MAN Truck & Bus SE, 80995 Munich

*Corresponding author. E-Mail address: elias.berger@tu-dresden.de

Mechanical Computer-Aided Design (CAD) remains a challenge in product development and engineering design due to steep learning curves. Hence, large language models (LLM) gain popularity to generate parametric geometry programmatically (CAD-as-Code). This paper examines CAD-as-Code as a methodological paradigm with implications for engineering design practice. We present a hybrid agentic framework, that uses CAD-as-Code with physics-in-the-loop and human validation to generate structurally sound designs. We prove our concept by generating physically valid designs based on load bearing requirements. Finally, we derive implications for engineering design practice from our prototype observations.

Keywords:

engineering design automation, generative ai, agentic ai, computer-aided design, cad-as-code

1 Introduction

In engineering design, CAD represents a persistent bottleneck. Translating functional requirements into geometric models requires specialized expertise and substantial time investment, restricting design capabilities to trained specialists. As product complexity increases and development cycles compress, this expertise barrier increasingly constrains innovation velocity and scalability.

Therefore, applying generative artificial intelligence (AI) to CAD is gaining momentum. Recent research has established technical feasibility: LLMs generate geometric models optimized for dataset similarity [1–3], while Vision-Language Models provide visual feedback in iterative loops [4, 5]. These approaches converge on, what we term “*CAD-as-Code*”, by defining CAD designs as executable scripts. They treat it instrumentally as an interface convenience rather than examining it as a paradigm with practical implications. This reveals two gaps. Architecturally, current work optimizes for geometric similarity to training data rather than functional correctness. Second, existing work does not examine how CAD-as-Code impacts engineering design conceptually.

This work is a concept paper that examines CAD-as-Code as a methodological paradigm with implications for engineering design. We argue that CAD-as-Code represents a fundamental development, not merely a technical convenience. By formulating engineering design as executable code, it transforms CAD from a human-operated graphical tool into a computational platform with bidirectional programmatic access: AI agents generate parametric models as scripts, and physics-based tools evaluate them deterministically. However, realizing this potential requires addressing several challenges such as limitations in spatial reasoning of LLMs, interoperability with human operators, and questions of trust and professional responsibility.

We provide of proof of concept through a prototype implementation and derive opportunities and challenges from this exploratory development. Our prototype makes use of established knowledge-

based engineering tools for assessing structural integrity, such as Finite Element Analysis (FEA). CAD-as-Code enables closed-loop validation where, rather than training LLMs to approximate physical laws from data, we integrate knowledge-based simulation directly into agentic decision-making made possible by the CAD-as-Code paradigm. This allows us to advance from visual inspection to physics-grounded validation, increasing trust in AI systems.

We discuss the proposed paradigm shift critically by answering the following research questions:

1. Why is CAD-as-Code as a methodological paradigm for AI-assisted mechanical CAD in engineering a promising direction of research?
2. How can CAD-as-Code be applied in a hybrid agentic framework for design automation?
3. What opportunities and challenges emerge from using programmatic CAD together with agentic AI?
4. What are the implications for engineering design practice and professional responsibility?

To answer these questions, we contribute: (1) conceptual positioning CAD-as-Code as a paradigm enabling agentic AI and hybrid design automation, (2) prototype implementation demonstrating technical feasibility through five diverse structural scenarios, (3) empirically grounded analysis of opportunities and challenges derived from the prototype results considering the broader engineering design discipline.

2 Background

2.1 CAD Software as an Engineering Design Tool

Modern CAD systems are built on geometric modeling kernels that represent three-dimensional objects through boundary representation (B-rep) or constructive solid geometry (CSG) approaches [6]. Users interact with these kernels through graphical user interfaces (GUI) that provide visual feedback. However, this GUI-centric interaction presents fundamental challenges for AI automation. CAD GUIs require spatial reasoning about visual layouts, coordinate transformations for mouse positioning, context-dependent menu navigation, and interpretation of graphical feedback. These capabilities remain largely beyond current AI systems [7]. This interface mismatch between AI capabilities and CAD interaction motivates programmatic alternatives like CAD-as-Code.

2.2 Data-based CAD Generation Methods

A comprehensive overview of this development is provided by [8] who systematically categorizes different model types, modalities, and representations. In recent months, the application of generative methods based on LLMs for CAD has gained momentum. Several studies demonstrate the ability of LLMs to generate parametric CAD models from textual [2, 9–12], image [13] or multimodal [1, 3, 14] instructions.

Agentic and iterative generation processes have been recently proposed. Preintner et al. (2025) integrates an evolutionary optimization loop to iteratively improve generated designs based on visual feedback. Similarly, Ocker et al. (2025) propose a Multi-Agent System driven by a VLM to automate CAD model generation by mirroring the structure of human engineering teams with specialized agents. Alrashedy et al. (2025) presents a process that employs commercial VLMs and prompting to generate CAD code and uses visual tests to verify if the CAD object matches the intended shape.

However, the evaluation of these generative CAD models has so far been primarily based on geometric similarity metrics such as Chamfer Distance, Intersection-over-Union (IoU), or Normal Consistency.

These metrics quantify the shape similarity between the generated and reference models but fail to capture functional properties or manufacturability. Studies such as [15] and [3] explicitly confirm this limitation: while geometric consistency improves, the evaluation lacks consideration of functional requirements and design intent like load-bearing capacity, cost, or sustainability.

2.3 Knowledge-Based and Hybrid Systems for Geometry Generation

Knowledge-based generative design systems use explicit engineering rules, physical models, and constraint formulations to create valid, context-sensitive CAD geometries. Unlike data-driven approaches, they ensure physical plausibility, manufacturability, and functional correctness from the outset. Key methods include topology optimization and parametric or rule-based CAD. These techniques are transparent and reliable, especially when data is scarce, but require substantial effort for knowledge acquisition and maintenance [16, 17].

To address the limited physical grounding of purely data-driven generative models, recent work has emphasized the renewed relevance of Knowledge-Based Engineering (KBE) [18]. In contrast to models that infer patterns from data, KBE systems derive valid geometries from explicit engineering knowledge, physical laws, and formalized constraints, enabling context-sensitive solutions aligned with functional requirements [19]. While KBE provides transparency, validity, and robustness it is resource-intensive, requiring substantial upfront knowledge acquisition and ongoing maintenance. Emerging research suggests that the future of generative CAD lies in hybrid systems that couple the creative capabilities of LLMs with the rigor of knowledge-based mechanisms such as simulation-in-the-loop, rule systems, and physics-based boundary conditions, thereby aligning generative performance with engineering objectives [20].

3 Method

This work follows a Design Science Research (DSR) methodology [21] to motivate why CAD-as-Code, combined with the increasing capabilities of agentic AI, may initiate a paradigm shift when applied to engineering design automation. We implemented a prototype to proof the concept across five structural scenarios to demonstrate feasibility and surface key opportunities and challenges. Based on empirical evidence and observations from prototype development, we derive and critically discuss emerging challenges and opportunities and contrast them against existing literature. The study is exploratory: it does not claim statistical validation, with quantitative evaluation deferred to future work.

4 Current state of generative CAD using CAD-as-Code

4.1 Coding approaches for CAD

Currently, CAD software is a visual tool, but modern software libraries enable the creation of CAD models not through GUIs, but by providing code-based interfaces to geometric modeling kernels. Code-based CAD libraries such as CadQuery [22], build123d [23], and RapidCADPy [24] enable a shift toward programmatic, parametric, and reproducible geometry generation in mechanical design. Built on mature solid modelling kernels such as OpenCascade, these systems allow users to define sketches, constraints, features, and Boolean operations through high-level Python abstractions. Further, coding approaches allow for direct integration of other engineering tools such as FEA. Previously, different tools were air-gapped for agentic AI, because of the GUI-focused data flows [7].

Advantages include explicit parameterization, version control compatibility, and seamless integration with computational pipelines—capabilities increasingly exploited in design automation and generative engineering. However, our experience developing the framework presented in this work revealed

practical limitations. Scripted CAD environments often expose only a subset of kernel functionality, making advanced operations such as robust filleting, complex blends, and surface modeling more difficult than in commercial parametric systems. Robustness issues arise from geometric degeneracies and kernel failures that can be challenging to diagnose programmatically. Furthermore, code-based workflows lack the interactive visual feedback and constraint-management of professional CAD environments, limiting their usability for detailed production design requiring frequent visual inspection and manual refinement.

4.2 Code-based CAD Generation using LLMs

Recent studies have increasingly applied LLMs and VLMs to CAD. These approaches follow a common workflow: the DeepCAD dataset [25] or derivatives thereof are used for training. These CAD models are first converted into executable Python code, which enable parametric modeling through code [1]. For Image-To-CAD 2D renderings are produced using a CAD kernel [3, 13] and for Text-To-CAD corresponding textual shape descriptions based on the renderings are generated using a VLM [2, 3]. The results are pairs or triplets of CAD code, image, text that form synthetic training datasets.

Open-source LLMs or VLMs are then fine-tuned on these datasets, effectively learning to reconstruct CAD operations from visual or textual input. The use of these three modalities, code, text, and image, aligns with the capabilities of existing multimodal models like Llama-3, simplifying the training process.

Other approaches directly leverage the multi-task in-context learning ability of LLMs [26]. They leverage commercial LLMs through few-shot prompting, providing example CAD objects as context to generate new parametric models as code [4, 5]. This approach omits the expensive fine-tuning process and enables rapid experimentation with frontier models.

4.3 Optimization Targets of CAD Generation

In AI experiments the algorithm is optimized towards a desired goal. Both fine-tuned and few-shot methods predominantly evaluate outputs using geometric similarity metrics such as Chamfer Distance, Intersection-over-Union (IoU), or visual comparisons using VLMs [4, 5, 15]. These metrics assess shape similarity to training data but do not evaluate functional correctness such as structural integrity, manufacturing feasibility, or performance under load. Evaluation remains geometry-centric rather than function-centric, leaving unexplored the integration of deterministic simulation into generative design loops.

5 A framework for agentic engineering design

We propose an agentic framework that makes use of CAD-as-Code by integrating generative AI, simulation-driven reasoning, and knowledge-based engineering (KBE) into a closed-loop design system (Figure 1). The architecture consists of four core components—Requirements Agent, Engineer Agent, Quality Assurance Agent, and a Knowledge-Based System—all orchestrated by a central workflow controller. The framework combines exploratory, data-driven generation with deterministic engineering validation, enabling scalable and context-aware CAD automation.

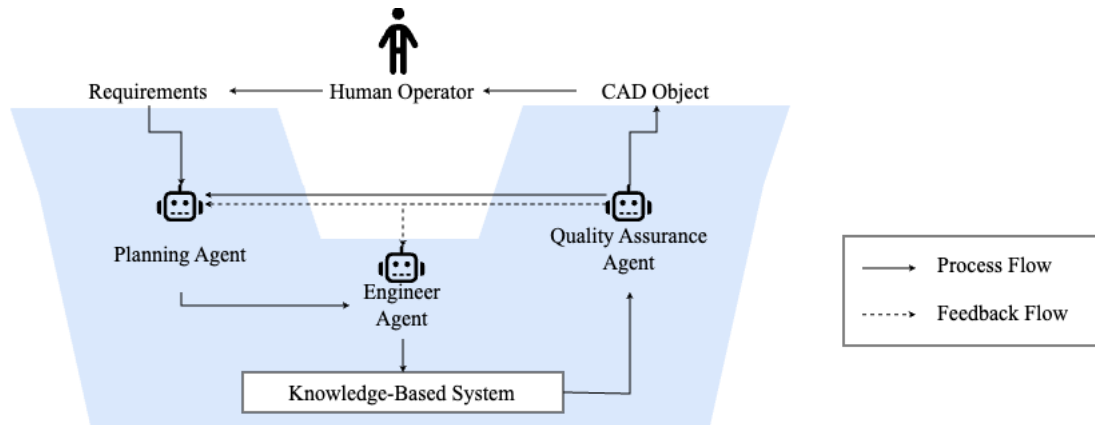


Figure 1: A framework for human-agentic-AI collaboration for engineering design. Under the CAD-as-Code paradigm, agents based on LLMs can create geometries using Python libraries and interact with knowledge-based systems (e.g. Finite-Element Analysis). The quality assurance agent evaluates the CAD design visually and structurally, providing feedback to the planner and engineer agents for iterative refinements. Graphic created by Authors, adapted from [5].

5.1 Planning Agent: Specification Interpretation and Task Decomposition

The Planning Agent interfaces with the human operator and translates high-level requirements into structured design plan, constraints, and parameter ranges. By leveraging natural language understanding and domain-specific rule sets, it decomposes ambiguous or under-specified requirements into actionable engineering tasks. This ensures that early-stage concept generation remains aligned with functional intent and boundary conditions.

5.2 CAD Engineer Agent: Generative Geometry and Iterative Refinement

The Engineer Agent serves as the primary generative component. Using a code-driven CAD backend it produces parametric models that can be validated, edited, and integrated into downstream engineering workflows. Through interaction with deterministic oracles (FEA, CAM, LCA), the system can discover design strategies that are not captured in static datasets, bridging data-driven creativity with physics-based rigor.

5.3 Quality Assurance Agent: Engineering Validation and Multi-Criteria Assessment

The Quality Assurance (QA) Agent evaluates candidate designs using simulation tools and knowledge-based rules, ensuring compliance with constraints such as structural, manufacturability, sustainability, and cost. Based on this assessment, the QA agent also generates feedback for the Engineer Agent, flagging issues such as geometric invalidity, constraint violations, meshing failures, or manufacturing infeasibility. Feedback to the planning agent is raised if the proposed plan fails to meet functional requirements.

5.4 Knowledge-Based System: Physics, Rules, and Domain Logic

At the foundation of the framework lies a deterministic knowledge-based system encompassing engineering rules, ontologies, constraint solvers, manufacturability checks, and physics-based models. It acts as a reliable validation oracle, ensuring safety, consistency, and domain compliance. The KBE layer provides grounding for AI-generated concepts, which remain physically plausible and manufacturable even when training data is scarce.

5.5 Orchestration Layer and Workflow Management

A central orchestrator and workflow engine manages agent communication, task routing, and error handling. It ensures that requirements flow toward generation, evaluation flows back as feedback, and validated designs propagate to the human operator.

6 Proof of Concept

6.1 Setup and Implementation Details

To validate the proposed concept, we implemented the multi-agent system illustrated in Fig. 1. The setup was built using the LangGraph framework, which provides structured orchestration of agent interactions and tool calls. The agent architecture was powered by an LLM backbone based on Anthropic Claude Sonnet 4.5 which handled code generation, reasoning, and dynamic tool utilization. CAD creation and modification were executed programmatically via the RapidCADPy¹ library, allowing agents to construct and iteratively update parametric geometry. For physics-based evaluation, structural simulations were performed using torch-fem², which supplied quantitative feedback to the agents and enabled closed-loop refinement of the generated designs.

6.2 Functional Requirements

For our prototype the inputted functions are defined as load bearing requirements. We evaluated the framework across five structural design scenarios with varying geometries, loading conditions, and boundary constraints. The system's objective is to minimize weight, while fulfilling the load bearing requirements. The design specifications were processed by the Requirements Agent, which translated them into a formalized set of engineering constraints for the downstream agents.

6.3 Observations from Prototype Development

Each scenario began with functional requirements specified as load-bearing constraints with applied forces and boundary conditions. The agentic system generated initial designs, evaluated them via FEA given the load bearing requirements, and iteratively refined geometry based on quantitative structural feedback.

The results demonstrate progressive design improvement across iterations (see Figure 2). For example, the bridge scenario (Row 1) evolved from a solid rectangular profile to an optimized beam structure with strategic material removal in low-stress regions, reducing weight while maintaining structural integrity. The tower scenario (Row 2) progressively added internal bracing as the system identified stress concentrations requiring reinforcement. Across all scenarios, the system produced fully parametric, editable CAD models—unlike topology optimization's voxelized density. These models maintain geometric intent, preserve design history, and integrate directly into standard engineering workflows.

7 Discussion

In line with DSR methodology, we treat the prototype as a research artifact and derive opportunities and challenges through reflective abstraction of empirical observations. The resulting implications are then triangulated within existing scientific literature.

7.1 Opportunities

7.1.1 Programmatic CAD for Design Automation

CAD-as-Code enables LLMs to apply their encoded engineering knowledge directly to design generation, bypassing human CAD operators. Beyond leveraging existing VLMs, CAD-as-Code offers structural advantages. Treating models as source code one can apply software development techniques to design: version control, reuse across projects, and modularization into standardized feature libraries. It enhances design reproducibility and adaptability, as parameterized scripts can regenerate variants

¹ <https://github.com/rapidcad/rapidcadpy>

² <https://github.com/meyer-nils/torch-fem>

automatically. Moreover, code-based models integrate seamlessly with simulation and manufacturing pipelines through application programming interfaces (APIs), supporting automated and reproducible engineering workflows.

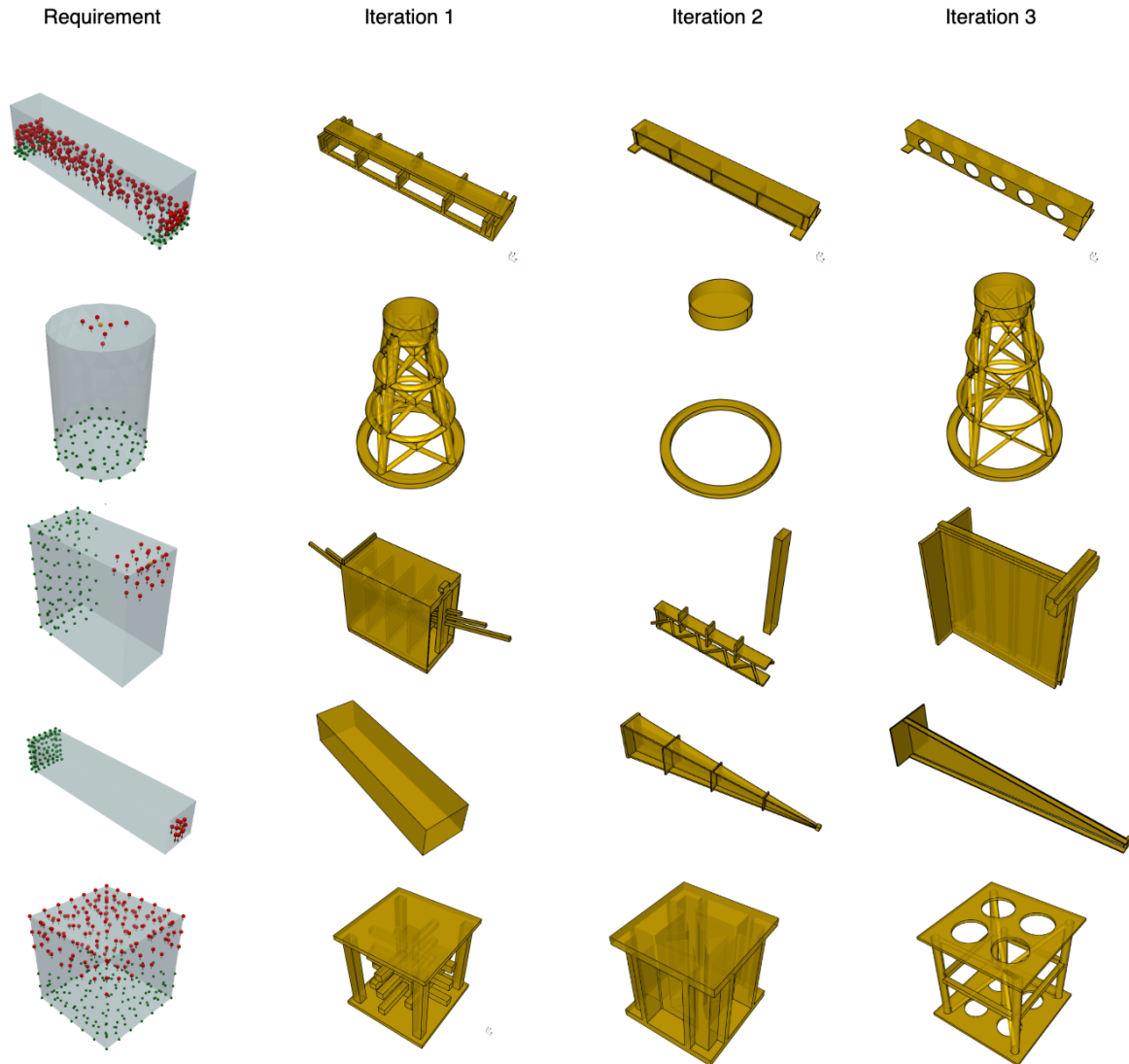


Figure 2: Left column: visualization of the load bearing requirements with applied forces in red and fixed constraints in green. This is the input for our system. Right columns: rendered outputs of each iteration from the hybrid agentic prototype, optimizing for least volume while fulfilling the load bearing requirements.

7.1.2 Agent-Based Frameworks for Autonomous CAD Generation

A particularly promising direction in AI-assisted design is the use of multiple specialized agents working collaboratively within a coordinated framework. In such a system, distinct agents assume different roles. For example, one agent may generate a concept shape based on input requirements, another may evaluate the design against engineering constraints using simulation tools or rule-based systems, and a third may perform optimization. These agents operate within a data-driven and knowledge-driven process, with communication and task coordination managed by a central orchestrator. Recent advances such as Model Context Protocols (MCP, Anthropic, 2024) and tool calling improve the ability of AI systems to interact with external software and tools, enabling agents not only to plan design workflows but also to execute them in real-world engineering environments.

Agent-based frameworks offer key advantages beyond automation. Agents can run in parallel, enabling simultaneous design exploration, accelerated optimization, and continuous design-simulation cycles

without human intervention. By abstracting domain knowledge into modular behaviors, these systems lower barriers for non-experts—system engineers, product managers, or simulation specialists can generate prototypes or request modifications without deep CAD expertise. This broadens participation in early-phase design and accelerates iterative development. Robust AI-CAD interfaces are therefore critical for enabling autonomous, scalable, and accessible design automation.

7.1.3 Comparative Advantages of Agent-Driven Design over Topology Optimization

The interaction between engineering agent, quality assurance agent, and FEA resembles topology optimization—both iteratively refine designs based on performance feedback. However, the agent-based approach offers key advantages. First, it generates parametric, fully editable CAD models rather than voxelized density fields requiring extensive post-processing, maintaining compatibility with standard engineering workflows. Second, agents can accumulate knowledge across iterations, improving efficiency and enabling generalization—capabilities topology optimization lacks. Third, multi-agent architecture enables explicit multi-criteria reasoning, incorporating manufacturability, cost, sustainability, and safety alongside structural performance. These properties position agent-based systems as a more flexible alternative to topology optimization for industrial contexts requiring interpretable, modifiable, workflow-integrated outputs.

7.2 Challenges

7.2.1 Implementation challenges

A key limitation in current AI-driven CAD systems lies in data availability since research datasets lack the complexity found in industrial-grade CAD models [28]. This limits the training and generalization capabilities of AI models. The data-related findings align with prior reviews that identify data scarcity and heterogeneity as key bottlenecks in data-driven design and manufacturing [29, 30]. Further, LLM-driven reasoning is computationally intensive and introduces substantial execution overhead. However, the future of agentic AI lies in small, specialized LLMs that offer faster inference [31].

7.2.2 Ethical and Trust Considerations

The explainability of AI agent decision is key for the system to explain why a certain design was generated. Without such transparency, human designers may be rightly hesitant to trust AI suggestions. Initial user studies and workshops will be valuable to gauge how designers interact with these tools and what level of autonomy is acceptable for *trust* and *usability*, which mirror broader findings in human-AI interaction research emphasizing transparency, accountability, and calibrated trust as prerequisites for successful adoption [32–34]. The consensus in the field is that near-term systems will be hybrid, where AI agents assist but do not replace humans, combining the creativity and intuition of humans with the speed and computational power of AI [35, 36]. Achieving the right balance will be both a technical and a professional challenge.

7.2.3 Engineering Knowledge Limitations

Despite recent advances, current AI-driven design systems exhibit limitations in their ability to replicate the domain-specific reasoning and contextual understanding that human engineers routinely apply. Although large language models can articulate engineering best practices when prompted, they often struggle to operationalize these principles within the design process itself [37]. This discrepancy suggests that LLMs lack robust mechanisms for integrating abstract engineering knowledge with spatial reasoning [38, 39]. We hypothesize that these shortcomings stem from the inherently text-based training of LLMs, which provides limited exposure to geometric data. Consequently, LLMs may fail to internalize the causal and spatial dependencies required for consistent, functionally grounded design decisions.

8 Conclusion

This work positions CAD-as-Code as a methodological paradigm for engineering design, moving beyond its instrumental use as an CAD interface. It demonstrates that the CAD-as-Code paradigm enables agentic AI architectures to operate professional CAD software and solve engineering tasks. By integrating a multi-agent workflow with FEA, the proposed system moves beyond the limitations of geometric similarity metrics to produce structurally valid designs. Our experiments validate the feasibility of our concept. However, challenges regarding computational overhead and trust remain. Ultimately, Agentic AI is best positioned not as a replacement for human expertise, but as a sophisticated co-pilot that democratizes access to complex engineering tools and accelerates the design process.

References

- [1] A. C. Doris, M. F. Alam, A. H. Nobari, and F. Ahmed, “CAD-Coder: An Open-Source Vision-Language Model for Computer-Aided Design Code Generation,” May 20, 2025, *arXiv*: arXiv:2505.14646. doi: 10.48550/arXiv.2505.14646.
- [2] M. S. Khan, S. Sinha, T. U. Sheikh, D. Stricker, S. A. Ali, and M. Z. Afzal, “Text2CAD: Generating Sequential CAD Models from Beginner-to-Expert Level Text Prompts,” Sep. 25, 2024, *arXiv*: arXiv:2409.17106. doi: 10.48550/arXiv.2409.17106.
- [3] J. Xu, Z. Zhao, C. Wang, W. Liu, Y. Ma, and S. Gao, “CAD-MLLM: Unifying Multimodality-Conditioned CAD Generation With MLLM,” Mar. 13, 2025, *arXiv*: arXiv:2411.04954. doi: 10.48550/arXiv.2411.04954.
- [4] K. Alrashedy, P. Tambwekar, Z. Zaidi, M. Langwasser, W. Xu, and M. Gombolay, “Generating CAD Code with Vision-Language Models for 3D Designs,” Feb. 28, 2025, *arXiv*: arXiv:2410.05340. doi: 10.48550/arXiv.2410.05340.
- [5] F. Ocker, S. Menzel, A. Sadik, and T. Rios, “From Idea to CAD: A Language Model-Driven Multi-Agent System for Collaborative Design,” Mar. 06, 2025, *arXiv*: arXiv:2503.04417. doi: 10.48550/arXiv.2503.04417.
- [6] Ian. Stroud, *Boundary representation modelling techniques*, 1st ed. 2006. London: Springer, 2006. doi: 10.1007/978-1-84628-616-2.
- [7] K. Cheng et al., “SeeClick: Harnessing GUI grounding for advanced visual GUI agents,” in *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: Long papers)*, Bangkok, Thailand: Association for Computational Linguistics, Aug. 2024, pp. 9313–9332. [Online]. Available: <https://aclanthology.org/2024.acl-long.505>
- [8] W. Zhang, W. Chen, and L. Zhao, “A survey on generative AI for CAD and mechanical design,” *Comput.-Aided Des.*, 2025.
- [9] P. Govindarajan, D. Baldelli, J. Pathak, Q. Fournier, and S. Chandar, “CADmium: Fine-Tuning Code Language Models for Text-Driven Sequential CAD Design,” Jan. 08, 2026, *arXiv*: arXiv:2507.09792. doi: 10.48550/arXiv.2507.09792.
- [10] C. Lv and J. Bao, “CADInstruct: A multimodal dataset for natural language-guided CAD program synthesis,” *Comput.-Aided Des.*, vol. 188, p. 103926, 2025, doi: 10.1016/j.cad.2025.103926.
- [11] M. Usama, M. S. Khan, D. Stricker, and M. Z. Afzal, “NURBGen: High-fidelity text-to-CAD generation through LLM-driven NURBS modeling.” 2025. [Online]. Available: <https://arxiv.org/abs/2511.06194>
- [12] S. Wang et al., “CAD-GPT: Synthesising CAD Construction Sequence with Spatial Reasoning-Enhanced Multimodal LLMs,” *Proc. AAAI Conf. Artif. Intell.*, vol. 39, no. 8, pp. 7880–7888, Apr. 2025, doi: 10.1609/aaai.v39i8.32849.
- [13] M. F. Alam and F. Ahmed, “GenCAD: Image-Conditioned Computer-Aided Design Generation with Transformer-Based Contrastive Representation and Diffusion Priors,” Apr. 08, 2025, *arXiv*: arXiv:2409.16294. doi: 10.48550/arXiv.2409.16294.
- [14] M. Kolodiazhnyi et al., “cadrille: Multi-modal CAD Reconstruction with Online Reinforcement Learning,” Sep. 19, 2025, *arXiv*: arXiv:2505.22914. doi: 10.48550/arXiv.2505.22914.
- [15] T. Preintner, W. Yuan, A. König, T. Bäck, E. Raponi, and N. van Stein, “EvoCAD: Evolutionary CAD code generation with vision language models.” 2025. [Online]. Available: <https://arxiv.org/abs/2510.11631>
- [16] K. Albrecht and R. Anderl, “Information model for the integration of manufacturing restrictions into the algorithm based product development process,” *Procedia CIRP*, vol. 50, pp. 819–824, 2016, doi: 10.1016/j.procir.2016.04.136.
- [17] K. Herrmann, O. Altun, P. Wolniak, I. Mozgova, and R. Lachmayer, “Methodischer aufbau von entwicklungsumgebungen nach dem generative parametric design approach,” in *Proceedings of the 32nd symposium design for X, DFX 2021*, 2021. doi: 10.35199/dfx2021.14.
- [18] K. Herrmann, B. Bode, J. Wurst, P. C. Gembariski, I. Mozgova, and R. Lachmayer, “Quantification of the material-related environmental impact of topology-optimized multi-material components,” in *Proceedings of the 33rd symposium design for X, DFX 2022*, 2022. doi: 10.35199/dfx2022.01.
- [19] G. Pahl, W. Beitz, J. Feldhusen, and K.-H. Grote, *Engineering design: a systematic approach*. Springer, 2007.
- [20] E. Berger, K. Herrmann, F. Pusch, J. Mehlstäubl, R. Lachmayer, and K. Paetzold-Byhain, “From geometry to function: Towards context-aware generative AI for engineering design,” in *Proceedings of the DESIGN conference*, Dubrovnik, Croatia, 2026.
- [21] A. R. Hevner, S. T. March, J. Park, and S. Ram, “Design Science in Information Systems Research,” 2004.
- [22] AU et al., *CadQuery/cadquery: CadQuery 2.4.0*. (Jan. 2024). Zenodo. doi: 10.5281/zenodo.10513848.

- [23] Gumyr and build123d contributors, *build123d: A python CAD programming library*. (2024). GitHub. doi: 10.5281/zenodo.XXXXXXX.
- [24] RapidCAD Contributors, *RapidCADPy: Python interface for Autodesk Inventor*. (2025). [Online]. Available: <https://github.com/rapidcad/rapidcadpy>
- [25] R. Wu, C. Xiao, and C. Zheng, “DeepCAD: A Deep Generative Network for Computer-Aided Design Models,” Aug. 16, 2021, *arXiv*: arXiv:2105.09492. doi: 10.48550/arXiv.2105.09492.
- [26] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, “Language models are unsupervised multitask learners,” 2019.
- [27] Anthropic, “Introducing the model context protocol.” Nov. 25, 2024.
- [28] E. Berger, M. P. Dammann, J. Mehlstäubl, B. Saske, F. Braun, and K. Paetzold-Byhain, “Challenges and opportunities in the integration of generative AI with computer-aided design,” *Proc. Des. Soc.*, vol. 5, pp. 881–890, 2025, doi: 10.1017/pds.2025.10102.
- [29] J. Luo, L. Zhao, and W. Chen, “A review of deep learning applications in engineering design: Data, models, and opportunities,” *J. Mech. Des.*, vol. 144, no. 12, 2022.
- [30] X. Zha and X. Du, “Challenges and opportunities of AI-driven design in manufacturing systems,” *Comput. Ind.*, vol. 148, p. 103844, 2023.
- [31] P. Belcak et al., “Small Language Models are the Future of Agentic AI,” Sep. 15, 2025, *arXiv*: arXiv:2506.02153. doi: 10.48550/arXiv.2506.02153.
- [32] S. Amershi et al., “Guidelines for human-AI interaction,” in *Proceedings of the 2019 CHI conference on human factors in computing systems*, 2019, pp. 1–13.
- [33] M. Eiband, H. Schneider, M. Bilandzic, D. Fazekas-Con, M. Haug, and H. Hussmann, “Bringing transparency design into practice,” in *Proceedings of the 23rd international conference on intelligent user interfaces*, 2018, pp. 211–223.
- [34] D. Wang, Q. Yang, A. Abdul, and B. Y. Lim, “Designing for calibrated trust in human–AI teams,” *Proc. ACM Hum.-Comput. Interact.*, vol. 4, no. CSCW1, pp. 1–26, 2020.
- [35] M. H. Jarrahi, C. Lutz, and G. Newlands, “Artificial intelligence, human intelligence and hybrid intelligence based on mutual augmentation,” *Big Data Soc.*, vol. 9, no. 2, p. 20539517221142824, Jul. 2022, doi: 10.1177/20539517221142824.
- [36] B. Shneiderman, “Human-Centered Artificial Intelligence: Reliable, Safe & Trustworthy,” *Int. J. Human–Computer Interact.*, vol. 36, no. 6, pp. 495–504, Apr. 2020, doi: 10.1080/10447318.2020.1741118.
- [37] F. Doshi-Velez and B. Kim, “Towards a rigorous science of interpretable machine learning,” *ArXiv Prepr. ArXiv170208608*, 2017.
- [38] S. Ahmed and Y. Reich, “Human-AI collaboration in engineering design: Opportunities and challenges,” *J. Eng. Des.*, vol. 34, no. 3, pp. 179–199, 2023.
- [39] C. McComb, J. Linsey, and J. Cagan, “Limitations of data-driven generative design in engineering practice,” *Des. Stud.*, vol. 79, p. 101091, 2022.

Influence of Tokenization Strategies on the Prediction of CAD Model Descriptions

Sayeda Hadisa Habibi¹, Julia Bergelt¹, Michael Teichmann¹, Fred H. Hamker^{1*}

¹Chemnitz University of Technology, 09107 Chemnitz

*Corresponding author. E-Mail address: fred.hamker@informatik.tu-chemnitz.de; Tel.: 0371/53137875

Abstract

Computer-aided design (CAD) plays a critical role in the development of innovative 3D models and tackles many different industrial branches ranging from small enterprises to large corporations. In recent years, growing interest has emerged in automating CAD workflows, using deep learning, to reduce repetitive manual tasks and allow engineers to focus on higher-level design and innovation. However, despite major advances in deep learning, especially in Natural Language Processing (NLP), applying these methods to CAD data remains challenging. One major challenge is an efficient representation of CAD data. Traditional methods such as voxel-based and point-cloud representations suffer from limitations, such as high memory consumption and loss of geometric detail. On the other side, standard tokenization techniques for NLP such as WordPiece, originally developed for text, may not handle the discrete and structured nature of CAD data. We focus on advanced text-based representations for CAD, building upon the DeepCAD framework, which is well suited for transformer-based architectures. We compare three representation strategies: a common method from NLP, a method inspired by DeepCAD, and a hybrid approach that combines elements of both. Our study aims at evaluating their effectiveness and at identifying pathways toward more robust and scalable CAD representations for design automation.

Keywords:

Sequential CAD Representation, Tokenization, DeepCAD Tokenization, WordPiece Tokenization, Representation Efficiency

1 Introduction

Mechanical design must incorporate factors such as cost, maintenance and manufacturability. At its core, it involves shaping parts and assemblies, which directly influence product performance and compliance. While computer-aided design (CAD) software is widely used to create 3D models, it primarily digitizes the engineer's ideas, leaving the integration of requirements to human expertise. Since many designs involve repeated geometrical elements, automating CAD modeling tasks can streamline product creation, reduce redundancy, improve quality, and allow engineers to focus on innovation [1].

A promising approach is using generative AI trained on existing CAD models, which already contain best practice knowledge, for predicting the next necessary design steps to support an engineer during his design process. In this work, we focus on text-based CAD representations which allow to apply highly successful methods from natural language processing (NLP). The challenge is that conventional tokenization techniques for creating the required input sequences, such as WordPiece [2], may not optimally handle the discrete and structured nature of CAD data. Thus, we investigate the efficiency of CAD-specific tokenization methods and compare the widely used WordPiece tokenizer with the sketch-extrude-based technique introduced by DeepCAD [3]. As dataset we use the DeepCAD dataset [3],

which includes the design history and is derived from the large ABC dataset [4]. The design history is important because it provides sequential information about how a model was constructed, allowing AI models to learn patterns in the step-by-step design process rather than just static final geometries. We seek for an information efficient representation and base our comparison of the different tokenization methods on the metrics token fertility and prediction performance on masked sequence parts, using a BERT-style transformer model [5], to measure the suitability for generative AI.

2 State of the Art

Computer-aided design (CAD) models may be represented either as explicit geometric data, such as polygon meshes or surface and solid descriptions, or as text-based, parametric definitions that encode human-readable primitives and their associated parameters. Geometry-based representations are detailed descriptions of a 3D object that capture the physical form of an object in a spatial context [6]. Although geometry-based representations are used for different deep learning approaches, such as recognition, due to their complex structure, they are not suitable for our intended application of making design step suggestions. Text-based representations, however, have richer information related to metadata, part specification, and relationships between components. For example, FeatureScript, a domain-specific language developed by Onshape¹, enables the representation of CAD models as a sequence of parametric operations. This provides fine-grained control, interpretability, and a clear understanding of the design inputs. Moreover, the design steps performed by the designer are captured in the corresponding order. A collection of over 1 million designs from Onshape in the FeatureScript format is stored in the ABC dataset [4]. The DeepCAD dataset is a subset of the ABC dataset and also retains the full design history for each CAD model based on data from Onshape's public repository [7]. However, it limits the models to designs, that only includes *Sketches* and *Extrude* operations. Another dataset, SketchGraphs [8], consists of 15 million real-world parametric sketches, also from Onshape. As the name suggests, the dataset contains solely sketches, which are represented as geometric constraint graphs. All three datasets, ABC, DeepCAD, as well as SketchGraphs, are publicly available and free for academic usage.

To utilize deep learning methods, such as transformer architectures, the textual input that should be processed must undergo a tokenization. One common tokenization approach for textual data is WordPiece [2]. It first breaks the input text based on whitespaces. Then, each word is further broken down into so-called tokens. It uses a greedy longest-match-first strategy, where the algorithm iteratively selects the longest possible sub-word from a predefined vocabulary that matches the beginning of the remaining word. If no full match is found, the word is decomposed into smaller sub-word units. Sub-words that appear as continuations of previous tokens are typically prefixed with “##”, indicating that they are not standalone tokens but part of a larger word. Recent advanced text-based tokenization for 3D data can be divided into two categories: sketch-based and 3D-based. For instance, the sketch-based model Vitruvion [9] is a generative model designed for synthesizing structured 2D sketches. It decomposes sketch generation into two stages: First, it autoregressively generates geometric primitives (e.g., lines, arcs, and circles), and then it predicts geometric constraints (e.g., tangency, coincidence) between these primitives. In the tokenization process, each geometric shape is parametrized according to its geometric properties and is encoded into a structured format. The model uses a transformer-based architecture and is trained on the SketchGraphs dataset. Another sketch-based generative model designed for producing structured 2D sketches is TurtleGen [10]. It produces sketches that already meet the geometrical rules and follow the topological conditions for creating a sketch design, which makes

¹ <https://www.onshape.com>

the workflow easier for generating 3D models. Sketches are represented using Turtle Graphics, where each sketch is described as sequences of drawing commands. A sketch begins with a *LoopStart* token, followed by line, arc, or circle commands defined by relative displacement vectors (e.g., Δx , Δy). This discrete, interpretable tokenization captures both geometry and topology, enabling efficient training of autoregressive models such as transformers without requiring constraint solvers or explicit coordinate systems [10]. Contrarily, DeepCAD [3] is a transformer-based 3D model generator that comprises the two operations *Sketches* and *Extrusion*. A sketch is composed of one or more closed curves (e.g., lines, arcs, and circles) defined on a 2D plane. These serve as the foundational elements for solid modeling techniques, such as extrusion, which transforms a 2D sketch into a 3D object. The tokenization process involves encoding the parameters of each command into a vector, where each element is an integer value in the range of 0 to 255. SkexGen is another 3D-based generative model for computer aided design that builds upon TurtleGen and DeepCAD by explicitly modeling faces (2D regions defined by closed loops) as the basis for valid 3D geometry generation via extrusion. It represents geometry, topology, and extrusion parameters using separate codebooks and employs specialized sketch and extrusion decoders. While this structured design enables interpretable and reusable CAD generation, SkexGen is slower than DeepCAD due to autoregressive decoding [11].

3 Methods

We present the dataset used in our experiments, followed by the proposed tokenization methods. We then present the model architectures corresponding to each tokenization approach and conclude with details of the training procedures.

3.1 Data Collection

We used the DeepCAD dataset as data source for our research. While several other CAD datasets exist, such as the ABC dataset [4], which contains a larger number of samples and greater geometric variation, DeepCAD offers a unique advantage due to its structured sequential format. It captures the design process as a sequence of parametric modelling operations, specifically sketch and extrude commands. This format reduces geometric complexity while preserving essential information about the design process [3]. This makes DeepCAD particularly compatible with transformer-based architectures, which require structured, tokenized input sequences. Given the scope of this work, which focuses on investigating representation efficiency by using, among other measures, a lightweight transformer model, the DeepCAD dataset is sufficient.

The DeepCAD dataset includes a total of 179,133 CAD designs, following the same structural format as the Fusion360 Gallery dataset [12]. The designs are represented through a sequential representation stored in JSON files, organized under three main top-level keys *entities*, *properties*, and *sequence*, describing the structure, metadata, and timeline of operations for a given CAD model. *Entities* store all individual modeling operations such as sketches, extrusions, and other parametric features. A sketch entity is used to define 2D profiles. It contains a unique name, a transformation matrix, the reference plane as well as identifiable profiles, which define closed shapes through *Lines*, *Arcs*, and *Circles*. An extrusion entity is defined by several attributes, including the associated profiles (defined in *Sketches*), and the extent type (e.g. one-sided extrusion, equal extrusion on both sides of the sketch plane, or asymmetric extrusion with different lengths on each side). Finally, the *sequence* field defines the ordering of modeling operations within a design. While the design order may not be apparent from the file structure, the sequence list provides this information explicitly. Each entry contains an index, a type (e.g., *Sketch* or *Extrude*), and a reference to the corresponding entity ID. This allows for the reconstruction of the modeling timeline, identifying which operation occurs first and which follows in the design process.

3.2 Tokenization methods

We investigate three different tokenization methods: WordPiece-naïve, WordPiece-DC, and DeepCAD. All tokenization pipelines (Figure 1) begin with the raw JSON files from the DeepCAD dataset. From this common starting point, each method applies its own sequence of processing steps to transform CAD construction information into a text-based representation, suitable for a BERT-style transformer. These steps include extracting or constructing the CAD sequence, normalizing numerical values, and converting sketches and extrudes into structured text rows when applicable. The differences between the pipelines reflect the distinct philosophies behind each representation: the WordPiece-naïve method focuses on flattening entities and sequence, the WordPiece-DC approach relies on structured sketch-extrude descriptions while preserving command information as text, and the DeepCAD method incorporates quantization and vectorization.

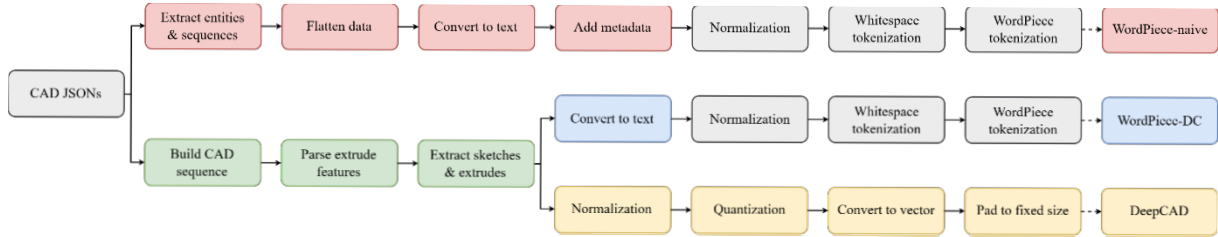


Figure 1: Pipeline of preprocessing for the three tokenization methods WordPiece-naïve (red), WordPiece-DC (blue), and DeepCAD (yellow). Common steps are depicted in gray.

We first determined suitable vocabulary sizes for the two WordPiece methods. Our analysis is based on token fertility within our CAD dataset and aims at identifying an optimal vocabulary size for training each tokenizer. Fertility is defined as the ratio between the number of output tokens (sub-words generated by the tokenizer) and the number of input words [13]. In general, lower fertility values indicate better efficiency, since information is distributed among less tokens. Our results are as follows:

3.2.1 WordPiece-naïve

Before training the WordPiece-naïve tokenizer, we converted the DeepCAD dataset into a flat, linear format suitable for token-based modeling. There, each CAD design is represented as a linear sequence of commands, analogous to a sentence in natural language. The flattening process iterates through each JSON file and recursively flattens each nested dictionary and list into key-value pairs, using dot notation and index markers to preserve hierarchy. Then we put every word in a list of sequence, e.g., ID, entity, and numerical values (integers and decimals). The tokenized sequence for an example is given in Figure 3a, step 2.

We determined the vocabulary for our tokenizer by constructing a corpus from the entire (flattened) dataset. We first applied whitespace tokenization to remove unnecessary spacing, then determined and built the vocabulary of the tokenizer on this cleaned corpus. We chose the vocabulary size through statistical analysis, based on token frequency distributions and the resulting fertility rates. Token frequency distribution was extracted from the full dataset, and all tokens were ranked in descending order based on their occurrence frequency. We estimated the minimum vocabulary size by computing the cumulative sum of token frequencies and identifying the smallest number of tokens needed to cover a predefined percentage of all token instances. In this work, we selected a 95% coverage threshold, meaning that the vocabulary should include the most frequent tokens that collectively account for 95% of the corpus. This approach reduces model complexity by reducing rare words in our vocabulary that contribute little to the overall data structure. Using this criterion, the resulting minimum vocabulary size was found to be 162. We computed the corresponding fertility rate of 5.88 (see Table 1), which however indicates poor tokenizer performance. Consequently, we further analyzed the efficiency of different

vocabulary sizes aiming at a fertility rate close to 1. We determined that a vocabulary size of 10,000 provides a good balance between vocabulary size and fertility. Although increasing the vocabulary size up to 15,000 slightly reduced the fertility rate, the improvement was negligible and did not yield any measurable benefits in our final analysis of model performance. Thus, to avoid unnecessary model complexity, we selected a vocabulary size of 10,000.

Table 1: Total number of tokens and fertility rate across different vocabulary size for WordPiece-naive

Vocabulary size	162	2000	4000	5000	10000	15000
Total tokens (after tokenization)	1,613,936,272	306,279,707	300,254,836	298,662,500	294,669,840	292,459,964
Fertility rate	5.878	1.115	1.093	1.088	1.073	1.065

After determining the vocabulary size, we tokenized the dataset using the WordPiece tokenizer. We refer to this approach as WordPiece-naive. In this naive method, all command tokens, such as entity and entity's IDs in the sequence, are preserved to examine their effect on tokenization and token prediction performance. One motivation for retaining all terms is that maintaining the full structure may help the model capturing meaningful relationships between elements, thereby improving its understanding of contextual patterns in the data.

3.2.2 DeepCAD

This tokenization method was first introduced in [3]. The process begins by parsing the JSON file into a hierarchical object representation by iterating over all operations within a design. For each operation of type *ExtrudeFeature*, the associated sketch and 2D profile are processed. A local coordinate system is then computed using a transformation matrix. The sketch geometry is converted into a profile object, and key extrusion parameters, such as the direction, are calculated. All *ExtrudeFeature* operations are collected into a list for further processing. Each CAD command is tokenized with a fixed-length vector consisting of a command type and its associated arguments. The supported commands include *Line*, *Arc*, *Circle*, and *Ext* (for extrusion), as well as two structural tokens: *SOL* (Start of Loop) and *EOS* (End of Sequence).

In contrast to standard tokenization methods for natural language, the DeepCAD tokenizer reserves for each tokenized command exactly 16 arguments, representing all possible numerical parameters. To handle the variable presence of parameters across command types, a tokenization mask is used. This mask is defined as a binary array and indicates which parameters are active for each command. For example, sketch-based commands such as *Line* activate only the sketch-related fields (e.g., the coordinates x and y), while all other fields are padded with a masking value (e.g., -1).

After tokenization, each CAD model undergoes a normalization process to ensure geometric consistency across the dataset. First, each sketch profile is independently normalized within its own bounding box. The corresponding scaling factor is stored as part of the associated extrusion command, enabling accurate recovery of the original sketch dimensions during downstream processing or visualization. In addition to sketch-level normalization, a global CAD normalization is applied to the entire model. Specifically, all geometric entities are uniformly scaled such that the maximum absolute coordinate value lies within the range $[-1, 1]$. As a result, all designs are consistently normalized to fit within a canonical $2 \times 2 \times 2$ cube. Following normalization, all continuous values, integers, and floating-point parameters are discretized into the integer range $[0, 255]$. The masked tokens and padding values are assigned to -2 and -1, respectively. When converting these integers into token indices for embedding,

the indices are shifted such that masked tokens map to index 0, padding tokens map to index 1, and valid parameter values 0-255 map to indices 2-257 (see Figure 3c, steps 2 and 3).

This discretization step transforms all geometric and operational parameters into a fixed set of discrete tokens, enabling efficient encoding and compatibility with the transformer-based architecture. The result is a sequence of tokenized commands, each consisting of a command type and its corresponding discrete arguments. These sequences are serialized and stored, serving as the model-ready input for the subsequent BERT training.

3.2.3 WordPiece-DC

We developed a new tokenization method inspired by WordPiece and DeepCAD. We refer to this method as WordPiece-DC. Instead of using a fixed vector length as for DeepCAD, which reserves space for all possible parameters leading to inefficiency and inflexibility, we keep parameters as separate elements. Our approach allows the inclusion of additional data elements without adaptation of the tokenization pipeline. Rather than converting commands into purely numerical representations, we also preserve them in text form. For each JSON file, the full CAD sequence is reconstructed by converting every *ExtrudeFeature* entry into a list of *Extrude* objects. Bounding box information is also extracted to maintain geometric context. Each *Extrude* operation contains one or more sketch profiles composed of curves (*Lines*, *Arcs*, *Circles*). WordPiece-DC converts these curves into a text-preserved, structured sequence. All numerical values are normalized to the range $[-1, 1]$ (min-max normalization over the dataset) to stabilize training. An example is given in Figure 3b, step 2.

Similar to the WordPiece-naive tokenization, we have to determine a suitable vocabulary size. Therefore, we trained the tokenizer with different vocabulary sizes. Although a vocabulary size of 15,000 achieves an ideal fertility rate of 1.0 (see Table 2), it did not improve the final model performance in the subsequent analyses. Therefore, we avoided introducing unnecessary complexity to the model and use a vocabulary size of 5,000 which provided the best performance.

Table 2: Total number of tokens and fertility rate across different vocabulary size for WordPiece-DC

Vocabulary size	965	2000	4000	5000	10000	15000
Total tokens (after tokenization)	10,368,158	9,813,855	9,622,580	9,585,626	9,508,852	9,482,328
Fertility rate	5.878	1.115	1.093	1.088	1.073	1.065

3.3 Model Architecture of Transformer

The model architecture used in this research is a lightweight variant of the original BERT model [5], but consisting of only two encoder blocks. The model was intentionally designed for its reduced parameter count and computational efficiency, making it ideal for rapid experimentation and consistent comparison across different tokenization strategies. While more complex models could potentially yield higher performance, the aim of this research is not to have an optimized model architecture, but to isolate the effect of tokenization methods under a fixed setup. Using the same model structure and trainable parameters across experiments ensures a fair comparison of each tokenization approach.

WordPiece-naive and WordPiece-DC share the same model architecture, as they differ only in their preprocessing steps but ultimately use WordPiece tokenization. Although DeepCAD adopts an embedding strategy similar to that used in natural language processing, where each token is projected into a continuous embedding space, the construction of CAD command embeddings differs in its

composition. Following the approach introduced in DeepCAD [3] (see Figure 2 for an illustration and Table 3 for the configuration), the embedding of each command is computed as the sum of three components. The first component is the command embedding, which represents the semantic category of the operation. Six command types are considered: *Line*, *Arc*, *Circle*, *SOL*, *EOS*, and *Ext*. Each command type is mapped to a learnable embedding vector. The second component is the parameter embedding. Each command is associated with a fixed set of 16 parameters, where each parameter is quantized into an 8-bit integer. These integers are converted into one-hot vectors of dimension 256. Finally, a (learned) positional embedding is added to encode the order of commands within the sequence. The final command representation is obtained by summing the command embedding, the argument embedding, and the positional embedding before being passed to the transformer encoder. The encoder blocks follow the standard transformer architecture, comprising a multi-head self-attention mechanism, identical to that used in the original transformer [14] followed by a feed-forward network, with residual connections, layer normalization, and dropout.

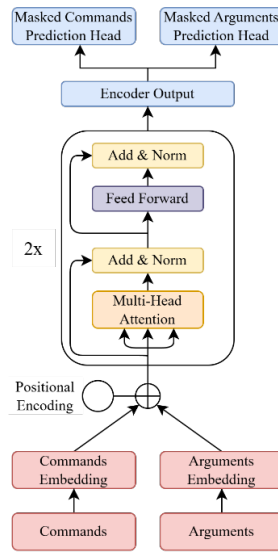


Figure 2: Network architecture of the used transformer for the DeepCAD method. From bottom to top: embedding, added positional encoding, two transformer blocks, two output heads for command prediction and argument prediction.

For the WordPiece models, a masked language modeling (MLM) head is added. It consists of a linear layer that projects the encoder outputs back to the vocabulary size. During training, the predictions are passed through a softmax, and the model is optimized using cross-entropy loss to predict randomly masked tokens. For the DeepCAD representation, a structured prediction head is used to jointly model command types and their associated arguments (Figure 2). This head includes two distinct linear projections: one produces logits over CAD commands, the other predicts arguments, structured according to the expected number and dimensionality of arguments per command. During training, both outputs are passed through a softmax, and the model is optimized using cross-entropy loss. The cross-entropy loss quantifies the difference between the predicted probability distribution and the true distribution of token classes. Cross-entropy loss is especially effective in classification tasks, as it penalizes confident yet incorrect predictions more heavily.

Table 3: Hyperparameters used for building and training the transformer models shared across all three tokenization methods

Hidden size	Feed-forward hidden size	#Blocks	#Attention heads	Dropout	Max learning rate	Min learning rate	Batch size	Warmup epochs
128	512	2	2	0.1	1e-4	1e-5	100	10

3.4 Training

Our lightweight BERT model was trained on an NVIDIA RTX 4090 GPU and optimized using AdamW with a weight decay of 0.01 [15]. To improve training stability, we used a warmup period to ramp up the learning rate to its maximum of $1e-4$ over the first 10 epochs and then reduced the rate following a cosine schedule to a minimum learning rate of $1e-5$ for the remainder of training. Additionally, early stopping was used during training in the sense that if the validation loss did not improve by at least 0.001 after 10 epochs, the training process is terminated. We applied early stopping to allow each model to train as long as required for convergence. For WordPiece-naive, the best model was obtained at epoch 288. In contrast, for WordPiece-DC and DeepCAD, the best models had converged at epochs 281 and 181, respectively. Consequently, the training times for WordPiece-DC (5h) and DeepCAD (4h) were shorter than for WordPiece-naive (29h), also due to the differences in sequence lengths. WordPiece-DC and DeepCAD generate more compact token representations, leading to shorter input sequences and lower attention overhead during training. This factor resulted in faster convergence and more efficient training.

As common, we split the used dataset (that is the DeepCAD dataset) into three parts: 80% of the models for training, 10% for validation, and 10% for testing [16]. We ensured that the CAD designs in each split (training, validation, and test) remained consistent across all models. During training of the WordPiece-naive model, we found that the input sequence length after tokenization posed challenges. With a vocabulary size of 10,000, our statistical analysis showed a maximum sequence length of 573,367 tokens. Standard transformer models exhibit quadratic time and memory complexity with respect to sequence length due to the self-attention mechanism. This leads to GPU memory constraints and extended training time. To address this challenge, we apply a sequence length cutoff strategy. Our statistical analysis show, that the data samples have a wide range of sequence length and contain many outliers. Thus, for practical reasons, we limited the maximum sequence length to 1024 tokens. In contrast, the WordPiece-DC and DeepCAD tokenization methods represent more information by a single token. Thus, these methods were not affected by the sequence length limit resulting in a maximum sequence length of 60 tokens for DeepCAD and 308 tokens for WordPiece-DC, respectively. For WordPiece-DC, each CAD design was represented as a single sequence containing all commands and their corresponding arguments, without structured or fixed-sized arrays. This approach, unlike DeepCAD, avoids padding and allows us to evaluate whether omitting pad values affects model accuracy.

4 Evaluation and Results

We found for WordPiece-naive an optimal vocabulary size of 10,000 and 5,000 for WordPiece-DC. This difference in size is due to the different preprocessing, where the input sequence for WordPiece-DC is reduced to a clean stream of a command with its arguments. Similarly, the DeepCAD tokenization represents commands along with their arguments, however embeds them in a single token. Thus, WordPiece-DC and DeepCAD tokenization produce less tokens than WordPiece-naive (see Table 4), which indicates better efficiency. Notably, the DeepCAD method achieves a significant reduction in fertility, an important advance for processing large CAD models.

We further investigated the information prediction performance for the resulting different data representations. The prediction task for commands and arguments differs in its difficulty. There is a limited number of commands, while the arguments can be various numerical values. Thus, we report the prediction accuracy (perfect match) for commands and arguments separately. For the arguments, we additionally report the normalized mean squared error (NMSE) to indicate how far the predicted values deviate from the data (Table 4).

Table 4: Comparison of the tokenization efficiency and information representation quality

Tokenization Method	Fertility	Total words (input)	Total tokens (after tokenization)	Accuracy (commands)	Accuracy (arguments)	NMSE (arguments)
WordPiece (naive)	1.0731	274,585,076	294,669,840	95.84 %	94.45 %	0.02
WordPiece-DC	1.0164	9,475,439	9,630,567	93.47 %	74.95 %	0.18
DeepCAD	0.2683	9,475,439	2,542,247	100 %	80.89 %	0.10

Although WordPiece-naive showed very good performance for argument prediction, both in terms of accuracy and NMSE, it is not practical for industrial use. The sequence length cutoff led to the loss of a significant amount of information. Furthermore, WordPiece-naive cannot preserve the hierarchical structure of the data, making reverse engineering after prediction both challenging and time consuming (Figure 3a). Thus, reconstructing the CAD model after prediction is challenging.

In contrast, DeepCAD retains all design steps, including numerical values. While the predicted numbers are not always exact, they remain close to the ground truth (see Figure 3c, compare steps 5 and 6), and its structured representation preserves the hierarchy of the design. WordPiece-DC addresses the issue of long sequences of WordPiece-naive by producing a more compact representation. However, our findings show that it underperforms in both classification and regression tasks. For instance, when it is supposed to predict a numerical value, it sometimes predicts a command instead (see Figure 3b, compare steps 5 and 6).

1.	entity Ff10ixV43HBZz0_1 entity transform origin y -0.4408 entity transform origin x -0.8157																			
2.	entity	Ff	##11	##01	##xV	##43	##HB	##Zz	##0_1	entity	transform	origin	y	-0.44	##08	entity	transform	origin	x	-8157
3.	155	885	Mask	2121	2753	683	1440	3315	1430	155	211	314	70	Mask	264	155	211	314	69	Mask
4.	155	885	3951	2121	2753	683	1440	3315	1430	155	211	314	70	155	264	155	211	314	69	314
5.	entity	Ff	##11	##01	##xV	##43	##HB	##Zz	##0_1	entity	transform	origin	y	entity	##08	entity	transform	origin	x	origin
6.	entity	Ff	##11	##01	##xV	##43	##HB	##Zz	##0_1	entity	transform	origin	y	-0.44	##08	entity	transform	origin	x	-8157

a) WordPiece-naive

1.	Extrude 0.3333 -0.3333 0.3333 -0.3333 0.4537 -0.0915 -0.2503 -0.3227 -0.3333 NewBodyFeatureOperation SymmetricFeatureExtentType															
2.	Extrude	0.3333	-0.3333	0.3333	-0.3333	0.45	##37	-0.091	##5	-0.2503	-0.3227	-0.3333	NewBodyFeatureOperation	SymmetricFeatureExtentType		
3.	117	102	85	102	Mask	1324	235	4936	52	2669	Mask	85	139	Mask		
4.	117	102	85	102	85	1324	235	4936	52	2669	129	85	139	129		
5.	Extrude	0.3333	-0.3333	0.3333	-0.3333	0.45	##37	-0.091	##5	-0.2503	OneSideFeatureExtentType	-0.3333	NewBodyFeatureOperation	OneSideFeatureExtentType		
6.	Extrude	0.3333	-0.3333	0.3333	-0.3333	0.45	##37	-0.091	##5	-0.2503	-0.3227	-0.3333	NewBodyFeatureOperation	SymmetricFeatureExtentType		

b) WordPiece-DC

1.	Sketch-Extrude pair: origin: [192 128 192], normal: [0.2459 -0.2559 -0.9349], x_axis: [-0.3497 0.8761 -0.3318], y_axis: [0. 0. 1.], sketch position: [128 216 155], sketch size: 8, operation: 1, type: 1, extent_one: 129, extent_two: 128															
2.	5	-1	-1	-1	-1	192	128	192	128	206	152	8	129	128	0	1
3.	7	1	1	1	1	194	130	194	Mask	208	154	10	Mask	130	2	3
4.	7	1	1	1	1	194	130	194	130	208	154	10	108	130	2	3
5.	Ext	-1	-1	-1	-1	0.5	0.0	0.5	0.0	0.6094	0.1875	-0.9375	-0.1719	0.0	-1	-0.9922
6.	Ext	-1	-1	-1	-1	0.5	0.0	0.5	0.0	0.6094	0.1875	-0.9375	-0.0078	0.0	-1	-0.9922

c) DeepCAD (Note: Padding values are indicated with -1)

Figure 3: Example of masked token prediction results produced by a) WordPiece-naive, b) WordPiece-DC, and c) DeepCAD model. 1. processed input sequence; 2. tokenized text; 3. token IDs, including mask for MLM; 4. predicted token IDs; 5. final detokenized predictions; 6. ground-truth text. Bold entries refer to the masked tokens, also recognizable by the mask token.

5 Conclusion

This study compared three tokenization methods: “naive” WordPiece, WordPiece with DeepCAD inspired CAD-specific preprocessing, and the CAD-specific approach from DeepCAD. We found that WordPiece-based tokenization can achieve few sub-tokens per word, which is seen as an efficient

representation. When being advanced with domain specific preprocessing, as done in WordPiece-DC, the total token amount is largely lowered, however, this approach yields the lowest performance in terms of both command accuracy and argument accuracy. In contrast, WordPiece-naive tokenization achieves the highest accuracy. Despite this advantage, it is impractical for industrial use due to its complex, inefficient, and excessively long token sequences. Although the DeepCAD approach exhibits lower accuracy than WordPiece-naive, we attribute this difference to the increased complexity of the tokens it processes, which presumably demands a more scaled network architecture. Advantageous is that, unlike the WordPiece-naive method, DeepCAD does not require data truncation, allowing it to retain the full sequence information. However, DeepCAD has several limitations: It relies on fixed vector positions for arguments, with unused positions padded, which reduces efficiency. Additionally, it lacks the ability to generalize to extensions of the CAD representation language, such as primitives not included during the development phase. In summary, our results demonstrate that standard tokenization methods already achieve good performance. Domain specific tokenization can additionally reduce sequence length substantially, which enables the use of larger neural architectures and facilitates the processing of long-range dependencies in more complex, and more realistic, CAD constructions. These results underscore the need for further research into CAD-specific tokenization methods that effectively balance representational compactness and the preservation of predictive performance.

References

- [1] L. Mandelli and S. Berretti, "CAD 3D Model classification by Graph Neural Networks: A new approach based on STEP format," Oct. 30, 2022, *arXiv*: arXiv:2210.16815. doi: 10.48550/arXiv.2210.16815.
- [2] X. Song, A. Salcianu, Y. Song, D. Dopson, and D. Zhou, "Fast WordPiece Tokenization," Oct. 05, 2021, *arXiv*: arXiv:2012.15524. doi: 10.48550/arXiv.2012.15524.
- [3] R. Wu, C. Xiao, and C. Zheng, "DeepCAD: A Deep Generative Network for Computer-Aided Design Models," Aug. 16, 2021, *arXiv*: arXiv:2105.09492. Accessed: Oct. 21, 2024. [Online]. Available: <http://arxiv.org/abs/2105.09492>
- [4] S. Koch et al., "ABC: A Big CAD Model Dataset For Geometric Deep Learning," Apr. 30, 2019, *arXiv*: arXiv:1812.06216. Accessed: Oct. 21, 2024. [Online]. Available: <http://arxiv.org/abs/1812.06216>
- [5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," May 24, 2019, *arXiv*: arXiv:1810.04805. doi: 10.48550/arXiv.1810.04805.
- [6] S. Feist, L. Jacques De Sousa, L. Sanhudo, and J. Poças Martins, "Automatic Reconstruction of 3D Models from 2D Drawings: A State-of-the-Art Review," *Eng*, vol. 5, no. 2, pp. 784–800, May 2024, doi: 10.3390/eng5020042.
- [7] M. F. Alam and F. Ahmed, "GenCAD: Image-Conditioned Computer-Aided Design Generation with Transformer-Based Contrastive Representation and Diffusion Priors," Apr. 08, 2025, *arXiv*: arXiv:2409.16294. doi: 10.48550/arXiv.2409.16294.
- [8] A. Seff, Y. Ovadia, W. Zhou, and R. P. Adams, "SketchGraphs: A Large-Scale Dataset for Modeling Relational Geometry in Computer-Aided Design," Jul. 16, 2020, *arXiv*: arXiv:2007.08506. doi: 10.48550/arXiv.2007.08506.
- [9] A. Seff, W. Zhou, N. Richardson, and R. P. Adams, "Vitrivion: A Generative Model of Parametric CAD Sketches," Apr. 28, 2022, *arXiv*: arXiv:2109.14124. doi: 10.48550/arXiv.2109.14124.
- [10] K. D. D. Willis, P. K. Jayaraman, J. G. Lambourne, H. Chu, and Y. Pu, "Engineering Sketch Generation for Computer-Aided Design," Apr. 19, 2021, *arXiv*: arXiv:2104.09621. doi: 10.48550/arXiv.2104.09621.
- [11] X. Xu, K. D. D. Willis, J. G. Lambourne, C.-Y. Cheng, P. K. Jayaraman, and Y. Furukawa, "SkeXGen: Autoregressive Generation of CAD Construction Sequences with Disentangled Codebooks," Jul. 11, 2022, *arXiv*: arXiv:2207.04632. Accessed: Oct. 21, 2024. [Online]. Available: <http://arxiv.org/abs/2207.04632>
- [12] K. D. D. Willis et al., "Fusion 360 Gallery: A Dataset and Environment for Programmatic CAD Construction from Human Design Sequences," May 17, 2021, *arXiv*: arXiv:2010.02392. doi: 10.48550/arXiv.2010.02392.
- [13] M. Ali et al., "Tokenizer Choice For LLM Training: Negligible or Crucial?," in *Findings of the Association for Computational Linguistics: NAACL 2024*, Mexico City, Mexico: Association for Computational Linguistics, 2024, pp. 3907–3924. doi: 10.18653/v1/2024.findings-naacl.247.
- [14] A. Vaswani et al., "Attention Is All You Need," Aug. 02, 2023, *arXiv*: arXiv:1706.03762. Accessed: Oct. 21, 2024. [Online]. Available: <http://arxiv.org/abs/1706.03762>
- [15] I. Loshchilov and F. Hutter, "Decoupled Weight Decay Regularization," Jan. 04, 2019, *arXiv*: arXiv:1711.05101. doi: 10.48550/arXiv.1711.05101.
- [16] H. Babaei, M. Zamani, and S. Mohammadi, "The impact of data splitting methods on machine learning models: A case study for predicting concrete workability," *Mach. Learn. Comput. Sci. Eng.*, vol. 1, no. 1, p. 21, May 2025, doi: 10.1007/s44379-025-00021-3.

Comparing Approaches for the Automatic Generation of System Models in Generative Systems Engineering

Martin Becker^{1*}, Damun Mollahassani¹, Jens C. Göbel¹

¹Rheinland-Pfälzische Technische Universität Kaiserslautern, Lehrstuhl für Virtuelle Produktentwicklung,
67663 Kaiserslautern

*Corresponding author. E-Mail address: martin.becker@mv.rptu.de; Tel.:0631 205 3787

Abstract: The reuse and adaptation of SysML modules and model fragments is central to efficient model-based systems engineering, yet it doesn't remain easy in practice due to heterogeneous documentation, fragmented repositories, and the high effort required to produce syntactically valid and tool-agnostic models. Large language (LLM) models offer automation potential, but unconstrained generation produces hallucinations and syntactically invalid SysML v2 that cannot be parsed or visualized. This paper proposes an approach to automatically selecting generation assistance in LLM-assisted SysML v2 modeling based on prompt complexity and legacy model hints. For this approach, retrieval augmented generation (RAG) that injects task-specific SysML v2 fragments and domain hints into the prompt, grammar constrained decoding (GCD) based on a lightweight SysML v2 grammar subset, and rule-based post processing (PP) with repair loops are used and compared. The result is a robust approach to engineering assistance that leverages the LLM's existing knowledge, aiming to generate reasonable, syntactically valid SysML v2 Notation. Results indicate that RAG is best used with extensive prompts and detailed product knowledge. At the same time, PP improves syntactic correctness for output close to correct and GCD for short prompts and limited knowledge.

Keywords: AI, LLM, SysMLv2

1 Introduction

The transition from traditional, domain-specific products to highly complex, interdisciplinary smart products demands development processes that are both agile and robust [1]. This shift substantially increases the multidisciplinary complexity of product development and cross-domain collaboration, while also necessitating shorter development cycles and increased cost efficiency [1]. In this dynamic environment, innovation is typically not characterized by radical new developments, but rather by iterative improvements based on pre-existing products [2]. Consequently, the ability to effectively manage and reuse existing engineering knowledge becomes a critical competitive advantage. However, as the complexity of these products increases, the manual adaptation of existing solutions becomes progressively more resource-intensive, requiring deep interdisciplinary expertise, which many organizations struggle to maintain.

MBSE addresses these challenges by representing requirements and architectural relationships in formal, human, and machine-readable models that support consistency across the product lifecycle. However, for iterative development, adapting existing model structures for new projects remains manual mainly and is aggravated by communication barriers and heterogeneous nomenclature across organizations and value creation networks. LLMs can reduce the effort required to create model artifacts

from natural-language input [3], but their introduction presents technical hurdles. Existing strategies for reusing and reintegrating model fragments into new system models often lack sufficient automation [4], depend on manual search mechanisms [5], or adhere to rigid formalisms that limit the flexibility required in early design phases [6]. Furthermore, while generative AI has shown potential for generating initial model fragments [7], it currently lacks the semantic grounding required to reliably modify specific legacy data without the risk of syntactic errors. However, multiple approaches exist that can improve the reliability of syntactically correct code generation. These need to be carefully selected, depending on the required task, and provided with additional information

This paper proposes an approach that automatically selects a mode of adapting the LLM output generation, reusing and adapting SysML v2 modules and model fragments. The approach generates SysML v2 textual notation that is directly parseable by engineering tools. Based on the prompt characteristics and available legacy hints, the selector selects one of four output adaptation techniques. The LLM is treated as a proposal generator, while a systems engineer performs final verification and decides whether the fragment can be reintegrated into the larger model.

The Design Research Methodology [8], was used to provide the critical context of the approach, therefore the paper follows the following structure. Chapter 2 details the current state of the art and the specific technical challenges associated with LLM-based modeling. Chapter 3 presents the investigative setup and the different strategies applied to enhance LLM performance, including the technical integration and grammar constraints. Chapter 4 provides a detailed validation of these capabilities through a descriptive study involving the incremental development of an electric mountain bike based on an existing electric city bike model. Finally, Chapter 5 concludes with a summary of the findings and an outlook on future research directions concerning the automation of systems engineering tasks.

2 State of the Art

Smart products integrate various engineering disciplines with software development and service design, challenging established product-development practices by increasing integration and coordination demands across heterogeneous stakeholders and subsystems. Abramovici et al. [9] note that this interconnectivity significantly increases development time and costs, as traditional methods require managing seamless integration across heterogeneous disciplines within various subsystems. In addition to these challenges, Albers et al. [2] emphasize that the majority of product innovations are iterations of established products and, therefore, product generations, rather than radically new developments without any prior product knowledge to build upon. However, the manual adaptation of this legacy knowledge is often hindered because the required interdisciplinary expertise is frequently unavailable or siloed within specific departments [4], or because individuals do not share their knowledge due to personal relations, culture, or identification as professional colleagues [10].

This siloing is a direct consequence of the complexity of smart products, which compels stakeholders to collaborate in Value Creation Networks (VCNs) to access complementary capabilities [11]. While this collaboration enables the development of advanced systems, it simultaneously fragments engineering knowledge across different organizations and stakeholders. This can result in the loss of access to engineering knowledge due to communication issues, differences in discipline-specific nomenclatures and understanding, and stakeholder-specific perspectives on challenges [12].

MBSE offers a solution to centralize this knowledge and enable holistic system understanding among various stakeholders, even when different disciplines are involved, but Chami and Bruel [13] highlight that steep learning curves and specialized nomenclature reduce acceptance in industry and frequently hinder its adoption despite MBSE's potential. Akundi et al. [14] add that organizational resistance and

technical barriers prevent its widespread acceptance among non-experts. Henderson and Salado [15] therefore show that reuse remains inconsistent across industries because most organizations lack the infrastructure to manage model libraries effectively. Nonetheless, Madni and Sievers [16] indicate that effective model reuse can be a primary driver for cost reduction in MBSE and, therefore, iterative product development.

Consequently, attempts have been made to structure models for greater reusability, resulting in less effort required to generate new system models. Mahboob and Husung's [17] framework for describing SysML model structures as an example focuses on describing reusable elements but lacks a mechanism to automate the actual reuse process. This leaves the systems engineers with an increased workload and therefore does not sufficiently help with the adoption of the process. Focusing on retrieval, Mendieta et al. [18] developed an approach to identify and find past model fragments relevant to current projects. While this delivers potential solutions, the retrieved artifacts are not adapted or modified for the new use case, limiting the level of automation, in an attempt to automate modeling efforts and deliver significant automation and relief to the systems engineers, Biase et al. [19] utilized Behavior Driven Development (BDD) to automate the generation of state machines. Although precise, the reliance on strict formalism limits the approach's applicability solely to state machines, rendering it unsuitable for holistic system adaptation. The emergence of LLMs as a readily available technology has increased their adoption in almost all aspects of life and introduced new possibilities and challenges for engineers. Ariffud [20] shows broad adoption of LLMs across engineering disciplines. Despite this potential, Kretzschmar et al. [3] warn that hallucinations restrict LLM usage in critical paths. To apply AI to MBSE, Akundi et al. [21] introduced a text-to-model framework generating SysML parts from natural language and therefore making SysML and systems engineering more approachable. However, this approach does not utilize LLMs or the flexibility that modern LLMs offer users. Additionally, SysML v2, as the coming standard, has not been used. Similarly, Schleifer et al. [7] presented an LLM-based pipeline that effectively generates use case diagrams directly from requirements. Yet, this solution is limited to the initial requirements phase and does not support modifications to complex architectural models later in the lifecycle. Furthermore, the use case diagrams are not generated directly in SysMLv2 textual notation for immediate rendering, requiring additional steps to generate the diagrams. Addressing code generation, Kourani [22] proposed a framework using LLMs and RAG to generate executable code for process modeling. While effective for code generation, the strict adherence to the code generation, code execution, and code repair pattern is applied only to process engineering tasks, not to model-based engineering tasks. Furthermore, the approach is not compared to alternative approaches of ensuring code validity. DeHart [23] developed a methodology enabling LLM interaction with SysML v2 via Python scripts, while using human feedback to ensure valid output in an engineering context. However, this approach relies on advanced LLMs and customized LLM assistants, not on automatic code validation. This limits the degree of automation achieved in the modeling support of systems engineers. Addressing the circularity of engineering assets, Mollahassani et al. [24] introduced a framework for AI-driven MBSE to enhance sustainability and model reuse. While this framework establishes the theoretical basis for circularity, it does not yet fully detail the interactive, agentic workflows required to mitigate LLM hallucinations during the specific adaptation process. Finally, Johns et al. [25] developed the AI Systems Modeling Enhancer, a plug-in allowing an LLM to model directly in Catia Magic. This allows a partial automation of modeling tasks using AI. The approach, however, does not use the more modern SysMLv2 language and, as a plug-in for Catia Magic, is vendor-specific. This severely limits the number of applications supported.

Therefore, this paper discusses the application of LLMs and especially small open-source LLMs in the role of active modeling assistance in MBSE using SysMLv2, during the iterative product development. To make the assistance independent from specific vendors, SysML v2 textual notation must be generated

directly. Since the required textual notation can be generated using various approaches with varying results, this paper presents an automated selection to support engineers with the best option for generating valid SysML v2 textual notation under the given circumstances. This assumes the existence of established product knowledge, which the LLM can reuse to generate new, adapted system models and thereby relieve systems engineers during product development. This results in the following research questions:

- How can SysML v2 code be automatically generated, and how do the different approaches affect the model quality?
- Under which circumstances should which generation method be used?

3 Approach for Ensuring Syntactical Validity of SysML v2 Models

This paper focuses on code generation for model reuse in MBSE using SysML v2. As identified in the previous chapter, LLMs can generally support modeling efforts by serving as a translation layer between natural-language requirements and formal engineering specifications. This capability significantly lowers the entry barrier for the engineers involved in the product development process, who may not be fluent in formal modeling languages. This can, therefore, increase acceptance of MBSE and SysML. However, SysML v2 presents a unique challenge to LLMs because it is a relatively new standard. Consequently, the amount of public training data available for SysML v2 is significantly lower compared to established programming languages, such as Python. This scarcity often leads to LLMs failing to produce syntactically correct code despite understanding the semantic intent.

Iterative product development typically relies on heterogeneous legacy knowledge describing the current system, including documents, requirements specifications, and prior model versions. In this work, the development of an electric bike represents such a setting in which existing bicycle documentation is reused and adapted to develop a new product. Previously discussed research proposed workflows and tool interactions to integrate this knowledge into MBSE activities and to capture additional expert knowledge in natural language. When LLMs are provided with both documents and expert input, they can support the elicitation and transformation of engineering information into candidate model structures. However, SysML v2 text notation imposes strict syntactic constraints, and unconstrained generation frequently violates them, limiting direct tool import and visualization. As a result, a plain large language model is often insufficient for standalone modeling assistance when automation and reliable tool compatibility are required.

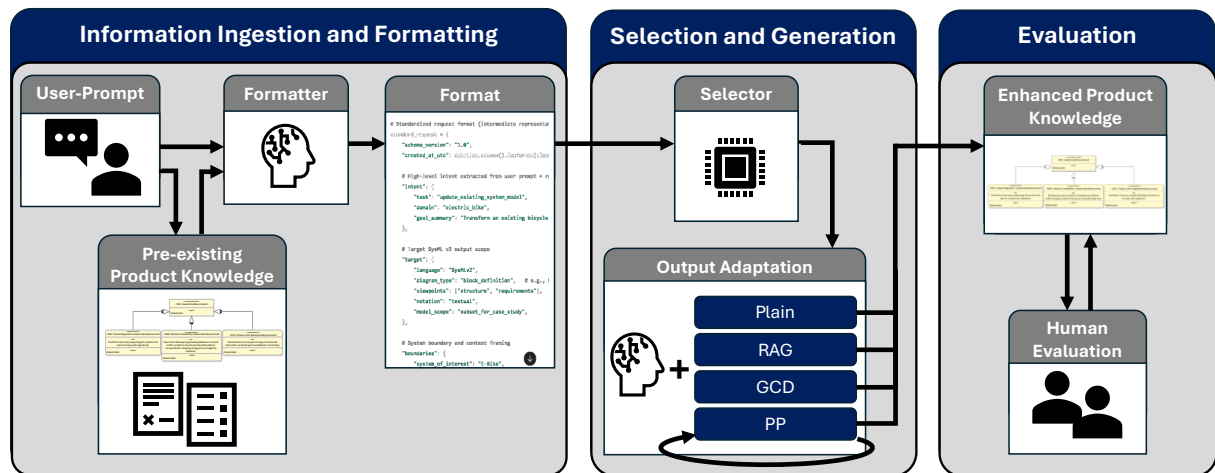


Figure 1: Process for ensuring syntactical validity of SysML v2 models

The approach is implemented as an orchestration pipeline around a local LLM to generate SysML v2 textual notation. Figure 1 summarizes the workflow for iterative product development, where user input and legacy engineering assets are transformed into an updated system model. The primary requirement is that the generated SysML v2 text is syntactically valid and can be parsed and visualized in engineering tools. To improve reliability, the pipeline first standardizes the combined prompt and legacy knowledge into a structured intermediate representation and then applies an algorithmic selector to choose an output adaptation strategy. The considered adaptations are RAG, GCD, and PP, with a plain LLM serving as a baseline and fallback.

3.1 Information Ingestion and Formatting

The first phase of the approach implements a data preprocessing step that unifies heterogeneous inputs into a uniform, machine-readable representation. It combines the natural language user prompt with accessible knowledge from the development of previous product generations, such as documents and model fragments. It converts both sources into a standardized intermediate format. This step is required because the subsequent selector depends on comparable input features across use cases. At the same time, raw user prompts and retrieved text passages can differ strongly in structure, terminology, and level of detail. The transformation is performed by an LLM, shown as the Formatter in Figure 1, which is configured to produce structured output in a fixed schema rather than free text. Therefore, this phase focuses on acquisition, restructuring, and presentation of information in a consistent schema that can be reused across projects and modeling tasks, to reduce surface variability and to expose modeling intent.

The text is checked for system boundaries, named subsystems and components, stated interfaces, and the intended viewpoints and diagram types. In parallel, the attached documents are processed to extract relevant passages that can serve as additional knowledge required in the current task. Retrieved content is condensed into short statements that preserve engineering meaning while reducing token load and redundancy. The formatter reduces the content into a single request object. The intermediate format is a structured representation with defined fields, as “Format” in Figure 1. This structure follows the structured output principle and encodes, at minimum, boundaries describing the explicit system of interest plus in scope and out of scope statements, viewpoints and requested diagram types, extracted requirements in the style of bullet points, candidate components or subsystems named in the prompt, as well as interfaces or flows mentioned in the prompt that can be mapped to SysML v2 elements in later steps. Without this phase, the selector would operate on inconsistent input, and the downstream generation prompt would be less reproducible and more prone to missing constraints or injecting irrelevant context.

3.2 Selection and Generation

After normalizing the user prompt and structuring the request as a dictionary containing bullet points of requirements, information, boundaries, and keywords, the request is algorithmically evaluated to determine which model generation method is most promising. This is done by using a weighted summarization of the amount of information and keywords given. Therefore, system boundaries are lowest in weight, with the required viewpoints being second to last in weight. The more important parameters for this evaluation are the number of keywords extracted from the user prompt, and the number of keywords found in additional code snippets, attached files, and retrieved documents. This estimation then results in a mode selection where the decision between a keyword-based RAG, GCD or PP is made, or whether no additional method is used and the plain LLM should be utilized for code generation.

RAG injects curated SysML v2 notation fragments into the prompt context before inference, shifting generation from unconstrained synthesis toward reuse of known valid patterns. Retrieval is implemented

as a deterministic keyword stem match over the normalized user prompt. For each detected keyword, the corresponding curated SysML v2 fragments are appended as contextual hints that illustrate the intended construct. To ensure a minimal baseline of usable notation, a compact SysML v2 reference package is included when RAG is active, so that generation remains guided even if no keyword matches occur.

GCD restricts token selection during inference to continuations that satisfy a predefined context-free grammar, which prevents syntactic violations within the covered language fragment. In this work, the constraint is implemented using Backus-Naur Form grammar rules, as supported by llama.cpp, in the local inference setup. This improves the likelihood that the generated SysML v2 text is syntactically correct. Still, it can reduce content coverage because the model can only express constructs admitted by the grammar, which does not ensure that names, allocations, and relations match the intended architecture. The applied SysML v2 grammar is therefore scoped to the constructions required by the instantiations.

PP converts near-valid outputs into syntactically correct SysML v2 notation via a validation-and-repair loop. Typical failures include minor formatting violations, missing semicolons, inconsistent keyword choices, or the insertion of extraneous lines that are plausible in other modeling languages but invalid in SysML v2 textual notation. PP targets these high-frequency, low-complexity defects after generation, with the objective of converting near-valid outputs into strictly valid artifacts without requiring repeated full regeneration, with an optional fallback LLM correction if deterministic repairs fail. This method cannot correct significant deviations from the target SysML v2 notation and cannot improve the content covered by the LLM.

3.3 Evaluation

The final phase applies the four-eyes principle by assigning the final decision to the systems engineer responsible. The purpose is to ensure that the generated SysML v2 textual notation is both formally correct and practically usable within engineering tools, and that the content matches the intended system modification rather than merely syntactically correct code. This positions LLMs as supporting tools, with structured output and explicit roles guiding generation, while acceptance and quality assurance remain human responsibilities.

The evaluation starts with a formal verification of tool compatibility. The engineer imports the generated SysML v2 text into the tool of his choice, such as the SysML v2 pilot implementation, and checks that it can be visualized without errors. This step validates syntactical correctness and guards against defects that may not be apparent in plain text, such as missing delimiters, invalid keywords, or incomplete constructs. If the import fails, the engineer determines whether the issue is a minor formatting defect that can be corrected locally or a structural defect that requires rerunning the pipeline with a different output adaptation. After syntactical correctness is established, the engineer evaluates the semantic and structural value of the output. The focus here is on whether the model contains the expected elements and relationships implied by the prompt and the retrieved legacy evidence, and whether the chosen viewpoint and diagram type were realized in the intended scope. Typical checks include whether system boundaries were respected and whether key components and interfaces are present with consistent naming. The engineer also checks for overgeneration, where the model introduces elements that conflict with the existing system model and associated artifacts.

If the result is acceptable, the engineer integrates the generated systems model into the ongoing development work and continues modeling in the authoring tool. If the result is incomplete but structurally sound, the engineer adds the missing information manually and treats the output as an initial model fragment. If the result is invalid or systematically misaligned with the intent, the engineer updates

the input knowledge and restarts the pipeline so that the ingestion, selection, and generation steps can incorporate the new information.

4 Evaluation of Output Adaptation Techniques

This chapter validates the approach from Chapter 3 by instantiating it in an electric bike development scenario using mistral-small 3.1 24b and by assessing whether it reliably produces SysML v2 textual notation that is syntactically valid and therefore usable in engineering tools. The validation target, therefore, includes both the decision logic that maps prompt and knowledge availability to a generation method and the resulting generated SysML v2 code, which is evaluated for formal correctness and usability. Nevertheless, each method needs to be validated individually to ensure the required quality. The development setting is iterative product generation, where an existing system model and associated artifacts from a previous bike development serve as prior knowledge, while the user prompt expresses the intent to develop an e-bike. The approach assumes that information availability varies widely between development situations, ranging from short, underspecified user prompts to detailed prompts with concrete specifications. To compare results across the different approaches and validate the importance of method selection, the validation uses two prompt granularities that represent extremes of information availability.

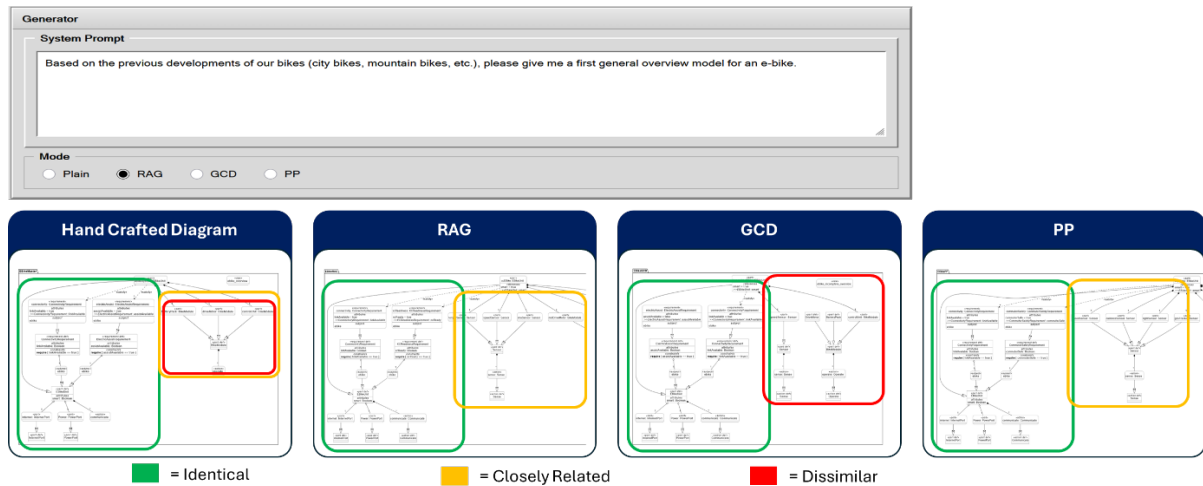


Figure 2: Comparison of outputs to the manually generated master model

These granularities are an abstract prompt and a detailed prompt, while keeping inference settings and the reference system intent constant. Because LLMs output is stochastic, each configuration is prompted repeatedly with slight variations. Model quality is evaluated along the dimensions content coverage against a manually defined reference architecture and syntactical correctness, which is assessed by attempting to render the generated SysML v2 artifact in the SysML v2 pilot implementation. Both the content and syntactic correctness are scored based on the presence of expected elements. Figure 2 compares the SysML v2 diagrams produced with RAG, GCD, and PP to a hand-crafted master model for the same prompt, illustrating how each method preserves or alters structural patterns. Highlighted regions denote identical, closely related, and dissimilar substructures in the rendered views.

The idea for the selector is that the failure mode changes with information availability. When prompts are extensive, and legacy knowledge is high, the main risk is the loss of domain-specific structure and SysML notation patterns, not the absence of concepts. In such cases, the approach prioritizes keyword-based RAG, which is deterministic and defined by the user prompt as it is scanned for predefined keywords. The validation results shown in Figure 3 support this method, especially for long and

extensive user prompts that contain numerous keywords. The reported percentages are obtained by comparing each generated model against a manually created reference architecture for the electric bike scenario. This reference is used as a checklist of expected SysML v2 elements and relations derived from the hand-crafted reference model, and each generated artifact is inspected for the presence of these expected items. The content score corresponds to the proportion of expected items that are present in the generated SysML v2 text, while the parsing score corresponds to the proportion of expected items that remain usable after parsing and rendering the artifact in the SysML v2 pilot implementation. With a detailed prompt, RAG achieves the highest mean content score at 97 percent and the highest mean representability score at 94 percent, effectively replicating the reference architecture for the electric bike case.

	Run		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	Average
Content Score	Abstract	Plain	55	80	25	35	35	50	40	0	10	5	35	0	20	10	0	0	75	27,9
		RAG	90	0	0	0	0	0	0	0	0	0	0	0	100	95	80	80	0	26,2
		GCD	100	80	55	40	80	40	50	0	10	0	60	15	70	65	40	55	80	49,4
		PP	60	90	10	30	40	30	30	5	5	15	45	0	10	5	0	0	50	25,0
	Extended	Plain	70	90	15	65	100	100	80	70	100	90	100	90	65	50	10	20	100	71,5
		RAG	70	100	100	100	100	100	85	100	100	100	100	100	100	100	100	100	100	97,4
		GCD	90	90	70	80	85	90	80	35	45	100	90	100	100	100	70	60	100	81,5
		PP	45	100	15	85	100	70	60	95	85	100	90	85	55	90	0	0	95	68,8
Parsing Score	Abstract	Plain	0	0	0	0	30	0	0	0	0	0	0	0	0	0	0	0	0	1,8
		RAG	80	0	0	10	0	0	5	0	5	0	0	0	90	70	0	10	0	15,9
		GCD	80	85	20	30	65	35	5	15	15	0	60	20	50	80	40	65	60	42,6
		PP	65	70	15	20	10	5	0	0	5	0	5	0	0	10	0	0	5	12,4
	Extended	Plain	0	35	0	15	30	10	5	15	5	10	10	10	0	45	0	0	20	12,4
		RAG	60	100	95	90	100	100	100	100	100	100	100	100	100	100	75	70	100	93,5
		GCD	60	100	60	90	80	100	75	20	35	100	90	90	100	95	35	25	85	72,9
		PP	50	85	20	0	70	40	10	50	75	75	65	65	25	10	0	0	60	41,2

Figure 3: Relative content score of 17 prompt runs, with percentage values

Under the abstract prompt, RAG drops to a mean content coverage of 26 percent and exhibits high variance, consistent with a keyword-triggered retrieval mechanism that requires explicit cues for stable activation and effective adaptation.

When prompt completeness and accessible product knowledge are low, the failure mode shifts from semantic incompleteness to reduced formal correctness, because unconstrained decoding frequently emits textual notation that cannot be visualized as SysML v2. In such cases, the approach prioritizes GCD, restricting token selection during decoding and thereby preventing syntax errors within the covered subset. The electric bike instantiation supports this choice. Under the abstract prompt, GCD increases the mean syntactic correctness and the visualization score to 43 percent compared to the plain LLM baseline, while under the detailed prompt it reaches a 73 percent mean parsing score, second only to RAG. The same evidence shows that grammar constraints do not inject missing engineering content, and in some runs, the restriction reduces content coverage by limiting the model's expressive freedom to what the grammar admits.

For prompts of intermediate length and information content, and for cases where neither retrieval effectiveness nor grammar coverage can be confidently assumed, the approach prioritizes rule-based PP as a repair mechanism. The electric bike instantiation shows that PP yields intermediate improvements that are strongest when the base output is already close to valid. Under the detailed prompt, PP raises the mean parsing score to 41 percent while achieving 69 percent content coverage, whereas under the abstract prompt, it remains limited to 12 percent representability and 25 percent content coverage, indicating that PP cannot compensate for significant deviations from SysML v2 notation and cannot add missing architecture information. The significant advantage of PP is that it does not modify the output

Graph-based product prototype optimization using multi-objective reinforcement learning: A framework concept

Sven Munker^{1*}, Hans Aoyang Zhou¹, Anas Abdelrazeq¹, Julian Haller¹ and Robert H. Schmitt¹

¹RWTH Aachen University, Chair of Intelligence in Quality Sensing, 52074 Aachen

*Corresponding author. E-Mail address: sven.muenker@wzl-iqs.rwth-aachen.de; Tel.: 0241/80 25843

Abstract: Circular-economy targets and emerging regulatory constraints increase the need to assess manufacturability and end-of-life disassembly already in early design, where late discovery of feasibility issues can trigger costly redesign loops. For this purpose, we represent product designs as AND/OR assembly graphs, that encode feasible assembly alternatives. We ask whether constraining product-level edits on an AND/OR assembly graph and evaluation in a production simulation can target this issue. To answer the question, this paper specifies a framework concept for optimizing product configurations with respect to assembly and disassembly. Design edits are restricted to graph-valid operators and further pruned by a manufacturability knowledge graph that encodes admissible substitutions and hard constraints. Candidate designs are evaluated by a production simulation across assembly and disassembly scenarios, returning a vector of performance indicators. Training uses scalarization of this vector, while full vectors are retained for post-hoc trade-off analysis. We also describe a minimum viable prototype that instantiates a restricted subset of the concept: binary joining-method selection on editable operations with a scheduling-based evaluator and a reinforcement-learning loop.

Keywords:

AND/OR graphs, design for disassembly, multi-objective reinforcement learning

1 Introduction

Sustainability targets and regulatory constraints increase the need to consider manufacturability and circularity during early design stages [1]. Decisions made constrain assembly and disassembly options downstream and can induce expensive iteration cycles especially when issues are discovered late [2]. Increasing product complexity, growing numbers of feasible assembly alternatives, and the higher routing flexibility in matrix assembly systems increase the coupling between design and production decisions. As a result, feasibility and performance issues emerge from interactions between product structure, joining choices, and resource constraints and are difficult to anticipate without explicit models [3]. Model-based evaluation can reduce iteration by enabling earlier assessment of manufacturability and (dis-)assembly consequences, but it requires both a representation that supports structural edits and alternatives without violating the product's requirements and an evaluation method that quantifies production-level impacts.

Feasible product-level changes such as joint selection, part substitution, and alternative assembly structures are combinatorial. Unconstrained exploration wastes computation on infeasible candidates and makes systematic trade-off analysis difficult when objectives conflict (e.g., assembly time vs. planning flexibility). Because each design edit changes the feasible action set and the resulting production performance can only be assessed through expensive simulation, the search problem becomes a sequential decision problem under black-box evaluation.

A framework for optimizing early-stage product variants under production-level evaluation must fulfill the following requirements:

- Represent alternative product structures compactly and expose well-defined editable sites
- Apply validity preserving edit operations that maintain structural consistency and enforce hard disassembly-feasibility constraints
- Model disassembly explicitly as a feasibility- and method-dependent process
- Multi-objective handling that is able to capture the trade-off between assembly and disassembly

Our hypothesis is that reinforcement learning (RL) can solve this problem, by learning edit policies that adapt exploration to observed outcomes and improve sample-efficiency relative to unguided heuristics. This motivates the following research question:

Research Question: *Does constraining product-level design edits on an AND/OR assembly graph and evaluating candidates in a multi-objective production simulation enable a reinforcement-learning agent to find manufacturable designs more efficiently and with better trade-offs than heuristic baselines?*

This paper contributes (1) a framework concept with explicit modules, interfaces, and validity-preserving edit operators; (2) a test protocol including a toy use-case and baseline heuristics; and (3) a prototype instantiation that implements a restricted subset of the framework to validate key integration points (representation, action encoding, and evaluation loop) and identify engineering constraints for scaling. Methodologically, the work follows a design-science where the artifact (framework) is defined from requirements and validated through a computational experiment.

In Section 2, the state of the art of decision frameworks in the context of assembly graphs, joining methods and optimization over constrained design spaces is illustrated. In Section 3, the framework concept including product representation, product constraints, change operators, assembly evaluation and optimization loop is presented. In Section 4, a toy use-case training dataset and a minimum viable prototype (MVP) of the framework is presented. In Section 5, the content of the paper is summarized and an outlook for future research is given.

2 State of the Art

In the following section, we present a short overview of data-driven systems in the context of product design and assembly and disassembly, illustrating frameworks and components that are relevant to the proposed framework of this paper. We close the section by highlighting the current research gap that this paper aims to fill.

Assembly sequence planning has long used explicit and implicit plan representations to capture precedence constraints and alternative subassemblies. AND/OR assembly graphs (AOG) were introduced as a compact representation of families of assembly plans and have been used to represent alternative assembly decompositions and sequences without enumerating a single fixed plan [4,5]. In manufacturing and remanufacturing contexts, AOGs are also used to represent feasible (dis)assembly sequences derived from geometric and technical precedence constraints, supporting alternative subassemblies and parallelizable operations [6]. Beyond precedence-only modeling, recent work targets physically feasible sequence planning from CAD (Computer-Aided Design) -derived geometry, incorporating collision-free motion and gravitational stability constraints. ASAP (Automated Sequence Planning for Complex Robotic Assembly with Physical Feasibility) is a physics-based planner that generates physically feasible assembly sequences for complex multi-part assemblies and demonstrates performance across a large dataset of product assemblies [7]. While graph representations have been

established as a product representation in the context of assembly planning, they are rarely used in early design stages to improve the product quality regarding assembly and disassembly.

Design for Disassembly (DfD) in circular-economy contexts emphasizes that disassembly time is not a simple inverse of assembly time and is driven by connector/join choice and accessibility. For end-of-life scenarios it can further depend on product condition. Accordingly, DfD methods use systematic time- or effort-based metrics and target-oriented disassembly planning, favoring non-destructive, reversible connections, where feasible [8,9]. Empirical studies show that disassembly time and reversibility depend strongly on the joining method, motivating simplified but method-dependent models (e.g. reversible fasteners vs. non-reversible bonds) for early-stage assessment [10]. This line of work supports modelling disassembly as target-state attainment and treating undoability as a joining-method-dependent feasibility predicate rather than assuming universal reversibility of every joining operation.

In process planning, manufacturing process knowledge (e.g. tooling and fixturing capabilities, resource models) is commonly represented using structured rule bases and ontology/graph-based formalisms. Recent surveys emphasize knowledge-graph-based approaches for representing and operationalizing process knowledge, covering extraction, graph construction, validation, and knowledge graph -driven process generation with planning pipelines [11]. Complementary work uses ontology-based manufacturability analysis to encode capability and constraint models that map design features to feasible manufacturing operations [12]. Within assembly contexts, knowledge-graph approaches have also been proposed to organize assembly resources and link product, process, and resource knowledge for retrieval and decision support [13]. These directions motivate a framework interface in which admissible substitutions and hard constraints are provided by a dedicated knowledge module, independent of the optimization method.

Optimizing product or process choices when evaluation requires simulation or scheduling is commonly framed as simulation optimization, where objective and constraints are observable only through computational evaluation. Surveys cover algorithmic families and application patterns, including settings with expensive evaluation and multiple outputs [14]. When precedence structure includes AND/OR conditions, dedicated scheduling formulations exist to capture alternative precedence requirements, providing a basis for integrating product graph structure into production-level evaluation [15, 16].

For multi-objective decision-making in sequential settings, multi-objective reinforcement learning (MORL) surveys formalize vector-valued objectives, scalarization choices, and solution concepts beyond single-objective policies, motivating the use of scalarization for training while retaining vectors for explicit decision support across objectives [17]. In constrained action spaces, invalid action masking is a practical mechanism to restrict exploration to admissible actions. Analyses show that masking can be important when invalid actions are prevalent and formalize implications for policy-gradient learning [18].

Existing work provides (i) graph representations that compactly encode alternative (dis)assembly sequences, (ii) DfD insights that disassembly feasibility and effort are method-dependent and target-state oriented, (iii) structured knowledge representations for manufacturability constraints and admissible process choices, and (iv) optimization paradigms that combine simulation/scheduling evaluation with multi-objective sequential decision-making. However, these elements are rarely integrated into a modular loop that (a) restricts design edits to graph-valid operator instances inferred from a product graph, (b) enforces manufacturability and disassembly feasibility through an explicit knowledge interface, and (c) evaluates design candidates as scenario-indexed makespan vectors to expose assembly-disassembly trade-offs under a shared production model.

3 Framework Concept

In this chapter, we illustrate the concept of the framework to improve the product's assemblability and disassemblability in the early design stage. The framework consists of three modules: The *knowledge module* that encodes manufacturability knowledge and is used to obtain alternative joining operations for the product, the *evaluator module*, where the product and all candidate alternatives are evaluated in multiple scenarios and the *optimizer module*, where changes on the product are being performed, based on the results from the evaluator module. The product is represented by an AOG over its components and joining operations. This graph is optimized in the following way (Figure 1):

Start by inserting the initial product AOG G_0 into the evaluator module and set it as the current AOG G_t (the subscript t denotes the t -th iteration of evaluation and optimization) (0). The initial product AOG will be evaluated once, and its results will be stored as the baseline. Using the disassembly AOG generator, create the disassembly AOG G_t^{-1} (1). Create a set of assembly and disassembly scenarios S in the scenario generator (2). Simulate the set of scenarios in the production simulation yielding a vector of simulation results $\mathbf{y}(G_t, S)$ (e.g., makespans) (3). Aggregate the simulation results into a scalar reward $r(G_t)$ (4). In the enhanced AOG generator, create the enhanced AOG G_t^e using the alternatives encoded in the knowledge module (5). Pass the simulation result vector, the reward and the enhanced AOG to the optimizer module (6). Extract the available actions from the enhanced AOG using the action generator (7). Use an optimizer algorithm to select the next action given the reward and vector of simulation results (8). Apply the chosen action to create the (next) candidate AOG (9). Pass the candidate AOG to the evaluator module for evaluation where it is set as the current AOG (10). Repeat steps (1)-(10) until the edit budget is exhausted or optimality is reached. Output the final product AOG G_f (11).

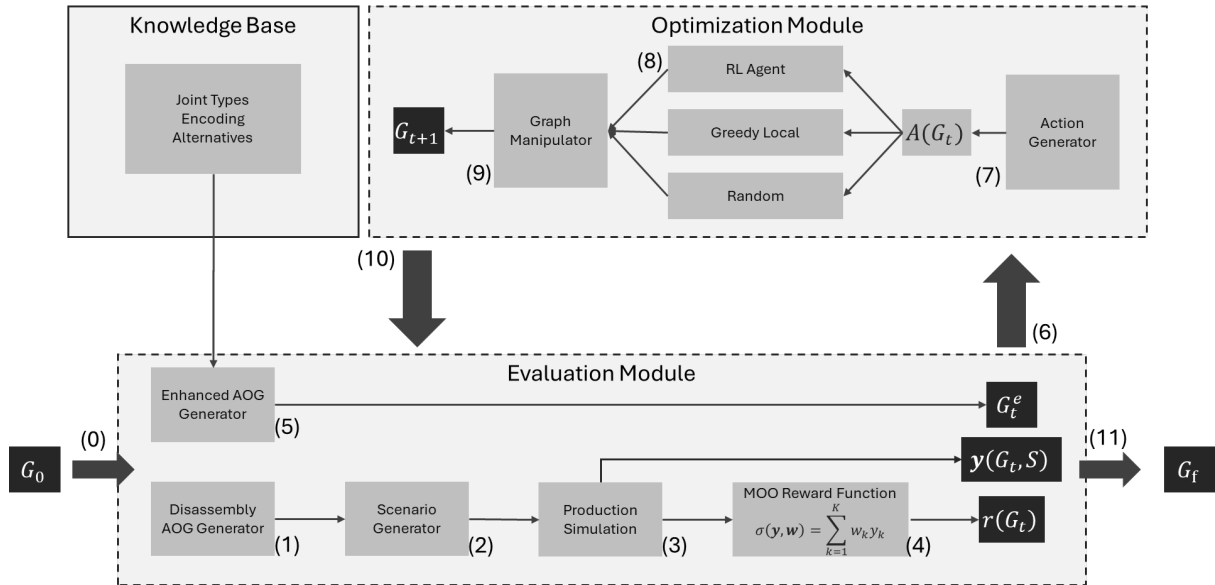


Figure 1: Conceptualization of the framework. G_t is the AOG after step t , G_t^{-1} is the disassembly AOG to G_t , G_t^e is the enhanced AOG to G_t , $\mathbf{y}(G_t, S)$ is the vector of performance indicators for the set of scenarios S and the AOG G_t and $r(G_t)$ is the reward function computed from $\mathbf{y}(G_t, S)$

In the following sections, the different components of the framework are discussed in detail. We start by introducing AOGs as the main data structure to represent the product. Afterwards, we describe the purpose and possible structure of the knowledge module, before illustrating the different components of the evaluator module. We close this chapter with the discussion of the optimization module and its position within an RL framework, while describing the two naïve baselines that will later be used for comparison.

3.1 Product Representation: AND/OR Assembly Graphs

The product is encoded as an AOG assembly graph G . AND relations encode mandatory composition and precedence relations, while OR relations encode alternative realizations at defined choice points. Node and edge types cover at least: parts, joints and operations. As an extension to the usual AOG definition, edges corresponding to joints are marked as either editable or uneditable. For further clarification, we use the term joint to refer to an edge in the AOG, while the term joining operation refers to the underlying operation. Editable joints might be changed during the optimization process, while uneditable joints remain fixed. A candidate design is defined by changing the joining-operation behind at least one joint.

The AOG is treated as the primary structural representation. Disassembly is not assumed to be a mere reversal of assembly. Instead, disassembly feasibility and required disassembly actions depend on selected joining processes (e.g. reversible vs. destructive operations). Therefore, disassembly-relevant structure is derived from the AOG together with joining-method selections and feasibility predicates. This avoids the implication that every joining operation has an inverse disassembly operation.

The representation must support two operations: (i) extraction of editable sites (joints where the underlying joining operation might be changed) and (ii) application of edits that preserve the feasibility of the corresponding disassembly. The AOG is not required to fully enumerate assembly sequences. It encodes enough structure to ensure edits remain consistent with the product's assembly logic and target part removal constraints.

3.2 Manufacturability Knowledge Module

To obtain alternatives for the editable joints, external knowledge on alternative joining methods is needed. The manufacturability knowledge module provides the rules and data required to infer admissible alternatives at a given editable site. It can be implemented as a knowledge graph or a structured knowledge base with predicates, depending on the complexity of the encoded information.

The module must encode the alternative definitions of joining operations, including process families and admissible substitutions per site type and compatibility constraints linking joining processes and part properties (e.g. materials, geometry classes). The alternative joining operations $j_1, \dots, j_k$ are assessed through the alternatives function

$$alts(G, l) \rightarrow \{j_1, \dots, j_k\}$$

That returns admissible alternatives at site l for AOG G . In this concept, operator instantiation uses *alts* to construct the constrained action space.

3.3 Evaluator Module

The evaluator module has two main functions: On the one hand, it evaluates AOG in the context of its assemblability and disassemblability for a given production system. On the other hand, it is used to enhance the AOG with the possible changes through simulations that could be made in the next step, while ensuring feasibility of the disassembly of the target part.

3.3.1 Disassembly AOG Generator

Before the product can be evaluated, the disassembly-relevant structure of the AOG has to be computed (see Figure 2 for two examples of AOGs and their disassembly counterparts). This disassembly AOG is derived from the AOG together with joining-method selections and feasibility predicates. This avoids the implication that every joining operation has an inverse disassembly operation. The disassembly AOG generation component of the evaluator module serves two functions. On the one hand it performs a hard feasibility check:

$$feasible(G, p) \rightarrow \{true, false\},$$

which determines whether for a given AOG G and a target product p , there exists a sequence to disassembly that product. On the other hand, it calculates the full disassembly AOG:

$$disassemble(G, p) \rightarrow G_p^{-1}(G, p),$$

encoding all sequences to remove target part p from the final product in G .

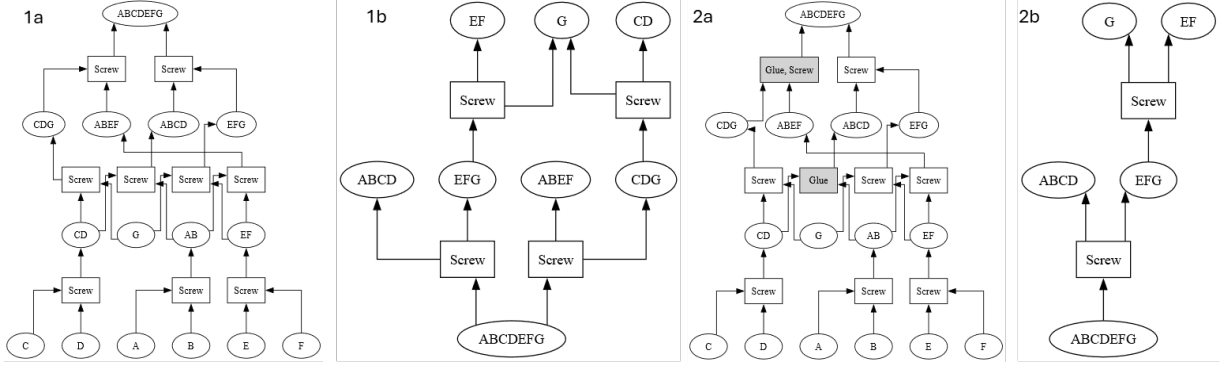


Figure 2: Comparison of a fully undoable AOG (1a) and the associated disassembly AOG for target part G (1b) with an alternative candidate (2a) where one of the disassembly paths of G is blocked by a non-undoable operation (grey) and the associated disassembly AOG (2b)

3.3.2 Scenario Generator

To access the differences of assembly and disassembly, we evaluate each candidate AOG under a set of scenarios that systematically varies the production context. The scenario generation component maps the current AOG G_t and the disassembly AOG $G_t^{-1}(G, p)$ of a candidate design to a set of scenarios in $S = \{s_1, \dots, s_n\}$, where each scenario s_i encodes a specific setting of the production system configuration, the number of target products to be produced (assemblies) and target parts to be removed (disassemblies) and the background products (AOGs and number of different products to be produced).

While the production system configuration and the background products remain constant and the total product load $n_{assembly} + n_{disassembly}$ remain constant, the ratio of disassembly to assembly tasks is varied to reveal trade-offs between the two tasks. A joining method that reduces assembly time may increase makespan in disassembly-heavy scenarios due to destructive operations, longer removal steps, or reduced accessibility.

3.3.3 Production Simulation

The production simulation component is used for the main analysis of the product. It takes the set of scenarios S and simulates each scenario returning a vector

$$\mathbf{y}(G, S) = [M(G, G_p^{-1}, s_1), \dots, M(G, G_p^{-1}, s_K)]^T$$

of makespans $M(G, G_p^{-1}, s_k)$, one for each scenario s_k . Inputs include: the candidate design state (G, G_p^{-1}) , a production system model (resources, capabilities, setup rules), and the scenario specification (workload including assembly and disassembly tasks). The scenario-indexed makespans form the multi-objective vector used for training and analysis.

3.3.4 MOO Reward Function

To enable training with standard single-objective RL algorithms while preserving the multi-objective character of the problem, the vector $\mathbf{y}(G)$ of scenario results is scalarized. The MOO reward function computes the reward as the scalar weighting of the different scenario results

$$r(G) = -\sigma(\mathbf{y}(G), \mathbf{w}), \quad \sigma(\mathbf{y}, \mathbf{w}) = \sum_{k=1}^K w_k y_k$$

with a non-negative weight-vector $\mathbf{w} \in \mathbb{R}^K$.

3.3.5 Enhanced AOG Generator

The enhanced AOG generation component computes the different alternative designs that could be obtained within the next step. It iterates through all the editable joints in G using $alts(G, l, s)$ to obtain different alternatives. For each joint, it is determined whether the alternatives are feasible or not. Alternatives that do not change the undoability of a joint or change it from not undoable to undoable are directly included, while alternatives that remove the undoability of a joint the corresponding AOG is computed and its disassembly feasibility is computed using $feasible(G, p)$. Afterwards, the enhanced AOG G^e is computed by including the alternatives in the metadata of the joints.

3.4 Optimization Module

In the optimization module, the next change on the product AOG is selected. It consists of three components. On the one hand, the action generation extracts the joint changes as actions from the enhanced AOG. On the other hand, the optimization algorithms select the next action to be evaluated. Lastly, the manipulation component implements this change on the AOG.

3.4.1 Action Generation

The action generation component parses the enhanced AOG G^e to extract the possible actions $a_1, \dots, a_n$ that can be made in the next step of the optimization

$$actions(G^e) \rightarrow A = \{a_1, \dots, a_n\}$$

The set of actions is then passed to the optimization algorithms, to select the next action.

3.4.2 Optimization Algorithms

In the next step, the next change of the AOG is selected. While in the main framework, this selection is done by a reinforcement learning agent, we also use two different baselines, a random and a greedy local optimization algorithm for comparison. The optimization is carried out under a given edit budget, that defines how many optimization-evaluation iterations can be performed in total.

Reinforcement-learning implementation: The optimization problem is formulated as a finite-horizon Markov decision process. A state encodes the current design G_t , the results from the production simulation and the remaining edit budget. An action selects a graph-valid edit operator instance from the constrained set inferred from the enhanced AOG. Training uses the scalar reward $r(G_t)$ derived from the evaluation of the current design. Candidate evaluation is performed after each edit step. Invalid edits with respect to structural or manufacturability hard constraints are prevented by the given set of actions.

Random feasible edits: Starting from G_0 repeatedly sample an admissible edit from $alts(G_t, l)$ and apply it if invariants hold. Evaluate each resulting candidate across all scenarios and track the best candidate under the scalarized objective. This baseline serves as a reference to whether constraint pruning alone yields improvements without informed search.

Greedy local improvement: At each step, enumerate a neighborhood of feasible single-step edits at the current design, evaluate candidates across all scenarios under the scalarized objective, and apply the best improving edit. Terminate when no improvement of edit exists or when the edit budget is exhausted. This baseline tests whether myopic optimization suffices in the constrained space.

3.4.3 Graph Manipulator

After an action is selected by the chosen optimization algorithm, the Graph Manipulator includes this action in a new candidate AOG by changing the associated joint. Afterwards the candidate AOG is passed to the Evaluation Module for evaluation.

4 Toy use-case dataset and prototype instantiation

This section describes a current prototype that instantiates a restricted subset of the framework to verify core integration points and identify constraints and the associated toy use-case and training dataset. The prototype optimizes joining process selections in a product represented as an AOG, where each editable joining operation has two alternative methods that alter processing time and thus schedule quality. The evaluator is a scheduling solver that computes a multi-order schedule on a fixed assembly system, and the RL agent proposes method selections to reduce makespan.

4.1 Toy use-case and training dataset creation

We use the ASAP training dataset [7] as a basis to build the toy use-case and training dataset. We use all 4-to-6-part assemblies contained in the dataset. For each assembly, ASAP was used to enumerate the full geometrically feasible assembly tree and the corresponding set of feasible assembly sequences. These sequences were merged into a product-level AOG.

For the creation of the toy use-case we defined two operations: Glueing that represents a fast operation with a timespan of 2 time units, that cannot be undone later and screwing, that is slower method with a time span of 5 time units but can be undone. The training dataset generation yields 50 training instances. For each of these instances, four products are sampled from the product pool and joining methods for their joining operations are assigned randomly. One of the four sampled products is selected as the target product, and one of its components is selected uniformly at random as the disassembly target. This target component is fixed within the instance. Disassembly is defined as the removal of this target component from the assembled product. To ensure that disassembly is feasible in the toy benchmark, all joining operations that directly affect the target component are set to the undoable method and it is checked whether at least one valid disassembly sequence exists to remove the target component. For each of the three non-target products in the scenario set, an assembly demand is sampled uniformly between 1 and 10 units.

4.2 Prototype Architecture

To verify the feasibility of the framework concept, we realized it in a prototype. The prototype instantiates each of the framework components in a minimal or close-to minimal setting. Its purpose is to verify the feasibility of the architecture.

Along the lines of the AOG creation in 4.2 the knowledge module encodes the two different operations glueing and screwing that are assumed to be exchangeable if the joint in the AOG is marked as editable. The components of the disassembly and enhanced AOG generators follow along, by naively reversing the AOG, only with respect to the undoability of joining operations and by the only restriction to an editable joint being if the resulting AOG remains feasible for disassembly. In the scenario generation, four different combinations of assembly and disassembly are created (Table 1). Afterwards, the scenarios are passed to the production simulation represented by a CP-SAT-based multi-order scheduler [19], that

computes a schedule for each of the scenarios. The corresponding makespans are combined, by averaging over the entries of the makespan vector relative to the baseline makespan of the initial AOG. The enhanced AOG, the makespan vector and the reward are passed to the optimization module. The action generator extracts the actions from the enhanced AOG by scanning its joints metadata and enumerating one candidate for each editable joint and each feasible alternative method. Each action is encoded as a pair of joint and alternative method and accessed by the RL agent via a stable key. The RL agent is implemented as a tabular Q-learning agent. At each step, the stored values are updated based on the reward obtained from the last evaluation. Afterwards, the next substitution is selected from the currently available actions using an ε -greedy policy. In the graph manipulator, the enhanced AOG is reduced to a usual AOG, by changing the joint associated with the chosen action and removing the encoded alternatives. The resulting candidate AOG is passed back to the evaluator module.

Table 1: The parameters for the set of scenarios

Scenario	Assemblies	Disassemblies
1	10	0
2	8	2
3	6	4
4	4	6

The prototype was tested on the training dataset, with a limited edit budget of 10 edits and an evaluation budget of 30 seconds. While the described training dataset in combination with the prototype implementation yielded feasible candidate AOGs and suitable changes being selected by the optimizer modules, no improvements over the baselines or learning behavior could be observed during training. This is most likely due to the complex interplay between joints and makespans in the combination with a high variation of the simulated makespans under the limited budget. For a naïve setting with a trivial optimum (both operations being undoable while one being faster than the other), a learning behavior could be observed.

5 Conclusion and Outlook

This paper specified a framework concept for product prototype optimization, that constrains design exploration to graph-valid edits on an AND/OR assembly graph and evaluates candidates under assembly and disassembly scenario-based workloads. Multi-objective evaluation is realized as a vector of scenario makespans, which is scalarized during training. An MVP was presented that implements this concept for joining-method selection for two alternative joining methods, with method-dependent undoability, disassembly modeling via an inverse AOG with blocked undo operations, and scheduling-based evaluation. While the MVP produced feasible solutions, it could be seen that the state representation and limited scheduling budget were insufficient to learn a reliable policy for systematic product improvement.

Future work will build up on this basis. On the one hand, we plan to improve the current MVPs performance on the simple benchmark problem by further evaluation with longer simulation times to reduce the variability of the scheduler results and increase the quality of evaluation, whilst also improving the state space representation of the scheduler results and the associated action generation to enable the RL agent to find better policies. On the other hand, the problem and scenario definition will be enhanced and pushed closer to reality, e.g. by encoding a more realistic set of alternatives in the knowledge module and by widening the disassembly definition from a naïve binary to a more elaborate view, separating assembly and disassembly times, or by including more complex products.

In reference to the research question, we have shown that the presented framework succeeds in performing manufacturable design edits, however, the implemented prototype is not yet able to outperform heuristic baselines. To achieve this goal, further improvements both in training effort, as well as the different concept modules is needed.

Acknowledgement: Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – HUMAN-DECISION (543081196), part of SPP 2443: Hybride Entscheidungsunterstützung in der Produktentstehung (521148212)

CRedit Author Statement: Sven Münker (Conceptualization, Writing), Hans Aoyang Zhou (Review, Supervision, Funding Acquisition), Anas Abdelrazeq (Review, Supervision, Funding Acquisition), Julian Haller (Review), Robert H. Schmitt (Review, Supervision, Funding Acquisition, Resources)

References

- [1] Regulation (EU) 2024/1781 Establishing a Framework for the Setting of Ecodesign Requirements for Sustainable Products, https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=OJ%3AL_202401781, Accessed 26.08.2025
- [2] Michael, J., Cleophas, L., Zschaler, S., Clark, T., Combemale, B., Godfrey, T., ... & Vangheluwe, H. Model-Driven Engineering for Digital Twins: Opportunities and Challenges. *Systems Engineering*.(2025).
- [3] Nielsen, Christian Petersson. "Matrix-structured manufacturing systems: From design to operations." (2023).
- [4] Sanderson, Arthur C., Luiz S. Homem de Mello, and Hui Zhang. "Assembly sequence planning." *AI Magazine* 11.1: 62-62. (1990).
- [5] de Mello, Luiz S. Homem, and Arthur C. Sanderson. "And or graph representation of assembly plans." (1986).
- [6] Münker, Sören, and Robert H. Schmitt. "CAD-based and/or graph generation algorithms in (dis) assembly sequence planning of complex products." *Procedia CIRP* 106: 144-149.(2022)
- [7] Tian, Yunsheng, et al. "Asap: Automated sequence planning for complex robotic assembly with physical feasibility." *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, (2024).
- [8] Mandolini, Marco, et al. "Time-based disassembly method: how to assess the best disassembly sequence and time of target components in complex products." *The International Journal of Advanced Manufacturing Technology* 95.1: 409-430.(2018).
- [9] Vanegas Peña, Paul Fernando, et al. "Ease of disassembly of products to support circular economy strategies." (2018).
- [10] Kondo, Yasuo, et al. "Reversibility and disassembly time of part connection." *Resources, Conservation and Recycling* 38.3: 175-184. (2003).
- [11] Xiao, Youzi, et al. "Knowledge graph-based manufacturing process planning: A state-of-the-art review." *Journal of Manufacturing Systems* 70: 417-435.(2023).
- [12] Cao, Jianpeng, et al. "Ontology-based manufacturability analysis automation for industrialized construction." *Automation in Construction* 139: 104277.(2022).
- [13] Shi, Xiaolin, et al. "Knowledge graph-based assembly resource knowledge reuse towards complex product assembly process." *Sustainability* 14.23: 15541.(2022).
- [14] Amaran, Satyajith, et al. "Simulation optimization: a review of algorithms and applications." *Annals of Operations Research* 240.1: 351-380.(2016).
- [15] Möhring, Rolf H., Martin Skutella, and Frederik Stork. "Scheduling with AND/OR precedence constraints." *A 2-page abstract appeared in Proceedings of the Eleventh Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'00)*, 235-236.. (2000).
- [16] Guo, Daqiang, et al. "A roadmap for Assembly 4.0: self-configuration of fixed-position assembly islands under Graduation Intelligent Manufacturing System." *International Journal of Production Research* 58.15: 4631-4646.(2020).
- [17] Roijers, Diederik M., et al. "A survey of multi-objective sequential decision-making." *Journal of Artificial Intelligence Research* 48 (2013): 67-113.(2013).
- [18] Huang, Shengyi, and Santiago Ontañón. "A closer look at invalid action masking in policy gradient algorithms." *arXiv preprint arXiv:2006.14171* (2020).
- [19] Davies, Toby O., Frédéric Didier, and Laurent Perron. "Violations: constraint-based local search in cp-sat." *International Conference on the Integration of Constraint Programming, Artificial Intelligence, and Operations Research*. Cham: Springer Nature Switzerland, (2024).

Towards Structured Knowledge Reuse to Support Design for Manufacturability: A Case-Based Reasoning Approach

Kokulan Thanabalan^{1,3,*}, Lukas Malburg^{1,3}, Leonhard Kunz², Ralph Bergmann^{1,3}

¹University Trier, Business Information Systems II – Artificial Intelligence and Intelligent Information Systems, 54296 Trier

²University of Kaiserslautern-Landau (RPTU), 67663 Kaiserslautern

³German Research Center for Artificial Intelligence (DFKI), 54296 Trier

*Corresponding author. E-Mail address: thanabalan@uni-trier.de

Abstract: Design for Manufacturability (DfM) relies heavily on experiential knowledge to identify and resolve manufacturability issues early in product development. However, this knowledge is often tacit, inconsistently documented, and at risk of being lost due to demographic changes and workforce fluctuation. Existing DfM support approaches are predominantly rule-based or data-driven and therefore struggle to reflect experience-based design reasoning and evolving manufacturing contexts. This paper proposes a structured Case-Based Reasoning (CBR) approach to support systematic reuse of DfM knowledge in early design phases. The core contribution is a case representation that captures manufacturability problems, applied solutions, and outcomes within the relevant functional, structural, and manufacturing context. Building on the Function–Behavior–Structure paradigm, the representation integrates structured attributes with explanatory textual information to balance retrieval performance and practical applicability. Following the Design Science Research methodology, the approach is implemented in a prototypical CBR system and evaluated using an industrially motivated use case. Initial results indicate that the proposed representation enables meaningful retrieval of similar DfM cases and supports experience-based design decision-making. The findings demonstrate the potential of CBR to complement existing DfM tools by enabling transparent, flexible, and reusable manufacturability knowledge support.

Keywords:

Case-Based Reasoning, Design for Manufacturability, Knowledge Reuse, Design Decision Support

1 Introduction

Global megatrends are fundamentally reshaping manufacturing and, consequently, the way new products are designed and engineered within New Product Development (NPD). These trends include societal shifts, environmental sustainability requirements, rapid technological advances, and increasing political and economic uncertainty [1]. As NPD is inherently knowledge- and experience-intensive, manufacturing companies are particularly exposed to the risk of workforce-related knowledge loss. Demographic change, especially in industrialized regions, further intensifies this challenge by accelerating the retirement of highly experienced engineers and designers [2]. To mitigate the resulting knowledge drain, organizations must establish effective mechanisms to capture, formalize, and transfer experiential design and manufacturing knowledge, either by structured training programs or by decision support systems that provide guidance during the design process [3].

Design for Excellence (DfX) methodologies have emerged as a systematic approach to address these challenges by embedding lifecycle-oriented considerations directly into product development. The integration of DfX assessments into digital development environments enables

designers, particularly those with limited experience, to identify potential design deficiencies early, thereby reducing costly late-stage changes and accelerating design iterations [4]. Among the various DfX dimensions, Design for Manufacturability (DfM) plays a particularly critical role, as it focuses on ensuring efficient, reliable, and cost-effective manufacturing. Despite its importance, DfM knowledge is still predominantly applied in a manual and time-consuming task. Designers typically rely on iterative feedback loops with manufacturing and quality departments to resolve manufacturability issues. Commercial DfM software solutions support this process through automated, rule-based checks that identify geometric violations directly within CAD models [5,6]. While these systems are robust and deterministic, they are limited to predefined rule sets and require substantial effort to adapt to changing manufacturing conditions. More recent Artificial Intelligence (AI)-based approaches, particularly those based on machine learning, promise more flexible and data-driven manufacturability assessments. However, their industrial applicability is often constrained by the need for large, curated, and labeled datasets and by limited transparency and reliability of the generated results [7,8].

Despite significant progress, many existing AI based decision support systems fail to incorporate experiential knowledge, understood as insights gained through practical, hands on experience. This gap is particularly evident in the early stages of design, where decision making is often intuitive and strongly influenced by context. In both academic research and tool development, dominant approaches tend to prioritize formal methods, abstract models, and data intensive techniques. However, these approaches are rarely adopted in industrial practice due to their limited practical relevance, substantial data requirements, and poor alignment with the ways designers typically work, which rely heavily on experience, intuition, and tacit knowledge [9,10].

Case-Based Reasoning (CBR) offers a promising AI problem-solving methodology to address this gap by explicitly enabling the reuse of experiential knowledge. CBR stores past problem-solving situations as cases and retrieves relevant cases when new problems arise [11]. However, despite its potential in related fields [19] there is currently no structured framework that defines how DfM-related problem situations should be represented as cases to practically support design engineers in identifying and reusing solutions to similar manufacturability problems. In the design and development of CBR systems, defining an appropriate case representation together with suitable similarity measures for accurately retrieving relevant cases remains a challenging task [18].

This paper addresses this shortcoming by proposing a CBR-based approach for DfM knowledge support with a specific focus on the development of an effective case representation. The central research question investigates how DfM problem situations can be structured and represented to enable meaningful retrieval of similar cases for design support within a CBR application. Following the Design Science Research methodology [12], a CBR-based DfM decision support system is developed as a research artifact and iteratively refined to explore and validate the proposed case representation in practice. A conceptual model for DfM case representation is derived from the literature and expert interviews and implemented in a prototype system, where it is evaluated in terms of retrieval accuracy, ability to identify similar DfM problems, and practical applicability in manufacturing-aware design activities. The paper concludes with a discussion of results and implications, followed by directions for future research.

2 Foundations & Related Work

This chapter introduces the fundamental principles of DfM and CBR, along with their key concepts. The related work section discusses existing research on the use of AI as a method for supporting design decisions in production.

2.1 Design for Manufacturability and Case-Based Reasoning

DfM is a core discipline within the broader DfX framework and aims to ensure that products are designed for efficient, reliable, and cost-effective manufacturing. Its fundamental principle is the early integration of manufacturing constraints and capabilities into the product development process, thereby reducing downstream rework, production delays, and quality issues [13]. Numerous studies indicate that decisions made in early development phases have a disproportionate impact on manufacturing cost and product quality, underscoring the strategic importance of DfM integration [14]. DfM therefore encompasses a wide range of considerations, including geometric simplicity, tolerance specification, material selection, process compatibility, and assembly effort [15]. Today, DfM is supported by digital tools that enable objective design evaluation, cost simulation, and the generation of optimization recommendations. The approach is well established in both industry and academia and continues to evolve with increasing automation and digitalization [16]. Typical DfM tools include design axioms and guidelines, manufacturing process-specific design rules, and computer-aided DfM software packages [14].

CBR is a well-established analogy-based problem-solving methodology from AI that addresses new problems by reusing knowledge from similar, previously solved situations stored as cases [17]. Its fundamental assumption is that similar problems tend to have similar solutions, which closely reflects engineering practice in DfM, where recurring manufacturability issues often stem from comparable design features, process constraints, and functional requirements. A widely accepted conceptualization of the CBR process is the four-step cycle proposed by Aamodt and Plaza [18]. In a DfM context, this cycle comprises the retrieval of similar manufacturability cases, the reuse with adapting prior design or process solutions, the revision of proposed solutions through evaluation or feedback, and the retention of validated solutions as new cases for future, upcoming problems. CBR systems structure knowledge using distinct knowledge containers, including a defined vocabulary, a case base following this vocabulary, similarity measures, and adaptation knowledge. Central to this defined structure is the suitable case representation, which determines how experiential knowledge is stored and accessed. In addition, the selection and modeling of this case representation directly influence retrieval quality and solution adaptation [3]. Over time, CBR has matured into a robust methodology with successful applications across diverse domains such as technical diagnosis, design, planning, and experience management, all of which share characteristics with DfM, including recurring problem patterns, evolving knowledge, and reliance on analogical reasoning [19].

2.2 Case-Based Reasoning for Design Decision Support

Several studies have investigated the application of CBR to decision support in product design and engineering. Early work on CBR in concurrent engineering and NPD environments demonstrated its suitability for supporting designers and managers in situations characterized by incomplete, unstructured information. These studies indicate that CBR enables informed decision-making by providing access to comparable past situations while maintaining flexibility that is difficult to achieve with traditional rule-based expert systems. However, they also emphasized that the effectiveness of CBR strongly depends on systematic case collection, maintenance, consistent terminology, and user training, and that early implementations primarily relied on unstructured textual case descriptions [9,25].

CBR research has focused on enhancing CBR systems through the integration of formal knowledge representation techniques. Ontology-enabled CBR approaches have been proposed for manufacturing-related decision support tasks, such as manufacturing process selection. By combining similarity-based retrieval with ontology-based reasoning, these systems establish semantic relationships between product features, materials, and manufacturing processes, enabling more context-aware recommendations [22]. While these approaches demonstrate the potential of combining CBR with semantic technologies, their

primary focus lies on process-level decisions rather than on design-centric manufacturability knowledge.

A foundational contribution to knowledge-intensive CBR is the CREEK architecture, which tightly integrates cases with general domain knowledge by embedding them within a unified semantic network. In this approach, cases are treated as first-class knowledge entities rather than isolated data records, enabling similarity assessment to be grounded in semantic and causal explanations instead of purely syntactic feature matching. This paradigm highlights the potential of explanation-driven, knowledge-intensive CBR for supporting complex design reasoning tasks beyond process-level decision making. [23].

Advances in knowledge graph technologies have further expanded the representational capabilities of CBR-based systems. Recent work proposes representing design cases as knowledge graphs structured by domain ontologies, allowing both explicit and implicit design knowledge to be captured in a semantically rich and machine-readable form. Graph-based similarity measures enable retrieval based on structural and semantic similarity rather than purely textual matching [24]. However, these approaches introduce new challenges related to modeling effort, scalability, and the appropriate level of semantic granularity required for practical design support.

CBR has also been combined with formal design methodologies, such as Axiomatic Design, to enable systematic reuse of design knowledge across different abstraction levels. In these approaches, cases are represented through hierarchical structures of customer requirements, functional requirements, constraints, and design parameters, enabling modular retrieval and adaptation [27]. Nevertheless, their applicability is limited by the need for rigorous functional decomposition and well-structured case libraries, which can be difficult to maintain in heterogeneous and evolving manufacturing environments.

Despite these advances, there remains a lack of case representations specifically tailored to DfM problem solving, in which manufacturability issues emerge from the interaction between design features, manufacturing constraints, and contextual production factors. In particular, the structured representation and retrieval of DfM knowledge at the intersection of product geometry, manufacturing context, and observed manufacturing outcomes remain largely unexplored. This research addresses this gap by developing and evaluating reusable CBR representations for DfM design problems. These representations balance structural expressiveness with practical applicability, enabling easy capture of design knowledge and systematic reuse of manufacturability knowledge during manufacturing-aware design activities.

3 Requirements for Case Representation

The requirements for the case representation are derived from the research project XDP-Opt [11] and grounded in established literature on CBR in design and product development, as well as expert interviews. Consistent with established CBR approaches in concurrent engineering and new product development, each case must capture the problem, the applied solution, and the resulting outcome [25].

Design cases represent complex experiences that cannot be adequately modeled using flat data structures. Prior work therefore emphasizes aggregated and hierarchical case representations that reflect multiple design aspects, such as function, behavior, and structure [28]. This motivates the use of composite case objects composed of semantically distinct attributes.

Furthermore, the case representation must support both structured and unstructured information. While structured data improves consistency and retrieval performance, many elements of

design knowledge, such as problem descriptions, constraints, and decision rationales, remain difficult to formalize and require textual representation. This is particularly relevant in early product and manufacturing design phases, which are characterized by uncertainty, incomplete information, and ill-structured problems [9,28]. Based on these findings, the following requirements are identified:

(R1) Explainability of design cases: Cases must support human understanding of past design decisions, including their context and consequences, to compare different cases.

(R2) Problem-solving documentation: Each case must describe how a DfM problem has been addressed and which solution strategy has been applied.

(R3) Outcome representation: The effects of the solution must be recorded to support learning from both successful and unsuccessful decisions.

(R4) DfM context coverage: A case must represent the essential context of a DfM problem, including:

- the manufacturing process,
- product or component characteristics,
- the function of the component, and
- the manifestation and consequences of the design problem.
- Support for hard-to-structure knowledge: Information that cannot be reliably structured must still be captured within the case representation.

4 Case Representation

Case representation is a critical success factor in CBR, as retrieval effectiveness directly depends on the structure and content of stored cases. In line with established CBR literature, cases are understood as structured descriptions of past problem-solving episodes, typically comprising a problem description, solution, and outcome, complemented by explanatory context and lessons learned [29].

Building on these principles and established approaches for capturing DfM knowledge, the proposed case representation is designed to support experience-based retrieval of manufacturability knowledge in early product design. Each case describes a concrete design situation in which a manufacturability issue occurred, how it manifested during manufacturing, which design or process changes are applied, and the resulting outcomes. The case structure is developed by analysis of the product development decision-making environment, iterative refinement, and alignment with literature and identified requirements [9].

To reflect designers' reasoning about manufacturability problems, the representation follows the Function–Behavior–Structure (FBS) paradigm [30]. This enables explicit linking of the functional intent of a component or product, the observed manufacturing behavior indicating a problem, and the underlying structural design causes. Contextual metadata and outcome information are additionally included to support case filtering, interpretation, and informed reuse. Formally, a case is represented as:

$$\text{Case} = \langle \text{Meta Data}, F, B, S, \text{Solution}, \text{Outcome} \rangle$$

The Meta Data provides contextual information for retrieval and comparison, including the manufacturing process (e.g., classified according to DIN 8580), component category, material class, and the product development phase in which the issue occurred. The Function (F) element captures the functional intent of the component or product, including function class and relevant material, energy, or signal flows [31]. The Behavior (B) element describes the manufacturability problem, consisting of a DfM problem class and a free-text symptom description of observed manufacturing issues. The Structure (S) element represents the underlying design causes using relevant feature types defined

by a controlled vocabulary. The Solution element documents the applied corrective action, including a controlled solution type and a textual description of the implemented change. Finally, the Outcome element records whether the issue has been resolved and qualitatively assesses the impact on cost, time, and quality.

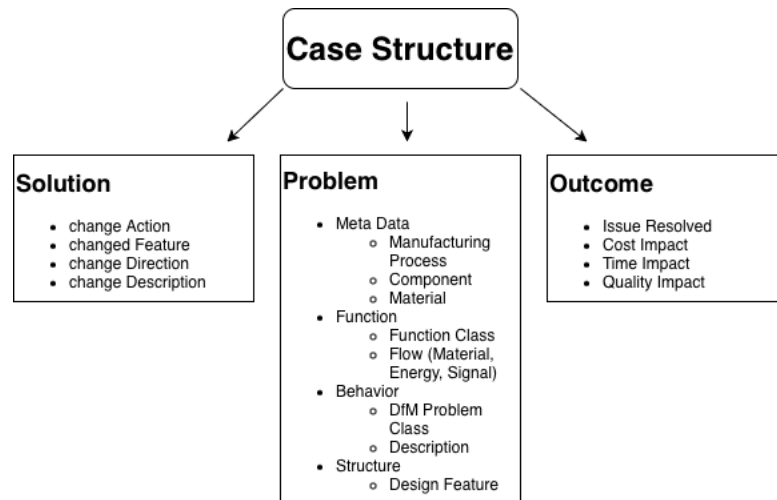


Figure 1: Case Representation Structure

5 Proof-of-Concept Implementation

To demonstrate the feasibility of the proposed CBR approach for supporting DfM in product development, a Proof-of-Concept implementation is realized.

The Proof-of-Concept is implemented using the state-of-the-art CBR system named ProCAKE, which provides native support for case-based reasoning, taxonomy-based knowledge representation, and similarity-based retrieval [32]. Case representation follows a structured, object-oriented modeling approach. Each case is modeled as an aggregate object composed of multiple attributes, such as manufacturing process, material, component, function, behavior, structure, solution, and outcome. Attribute values are either references to other aggregate objects or taxonomy-constrained string values defined by hierarchical vocabularies. This ensures consistent and semantically meaningful classification while still allowing limited free-text descriptions where required. In this way, the case base captures practical design knowledge in the form of problem–solution–outcome tuples.

For case retrieval, existing similarity measures provided by ProCAKE are employed. Local similarities are computed for individual attributes and combined into a global similarity score using a weighted aggregate similarity measure. Taxonomy-based attributes are compared using path-based similarity within the respective hierarchies, while textual attributes are compared using string-based similarity. Standard similarity configurations and default parameter settings provided by the ProCAKE framework are used. Further details on the underlying similarity measures can be found in the ProCAKE documentation [33]. To support transparency and reproducibility, the implementation of the Proof-of-Concept, including the case structure and example cases, is made publicly available online via a GitHub repository.¹

¹ https://github.com/XDP-Opt/XDP_Opt_Truckdesign_Demo

6 Evaluation

The evaluation aims to assess the suitability of the developed case representation for supporting design decision-making in a DfM context. The evaluation is based on an industrially motivated use case derived from the research project XPT-Opt [11]. In this project, case data has been collected by expert interviews and subsequently transformed into individual cases for the case base.

The selected use case comprises 20 truck design cases, each representing a concrete design experience related to a specific truck component. Each case describes a design or manufacturing problem with relevant contextual information (e.g., component type, material, and manufacturing process), a proposed solution addressing the identified DfM issue, and the resulting outcome. The developed case base has been integrated into the Proof-of-Concept implementation in ProCAKE. This integration aims to examine whether the proposed case representation is suitable for providing design decision support in a DfM context.

For the experiment, three representative query cases are defined, each reflecting a realistic design situation within the scope of the industrial use case. As a baseline (ground truth), an experienced domain expert analyzes the case base and identifies, for each query, the cases considered most useful w.r.t. solving the given design problem. These expert assessments are based on professional judgment and domain experience and serve as the benchmark against which the retrieval results are evaluated.

Retrieval performance is evaluated using the retrieval error metric, a methodology commonly used in CBR research [34]. The metric assesses whether the expert identified most useful case is retrieved within the top- k most similar cases returned by the developed CBR system.

For a given query, the retrieval error is defined as follows:

- The retrieval error is 0 if the most useful case identified by the expert is contained within the set of the k most similar retrieved cases.
- Otherwise, the error is 1.

The overall retrieval error is computed as the mean percentage of queries for which the retrieval error equals 1. A lower retrieval error indicates better retrieval performance and, consequently, a higher suitability of the retrieval results. To reflect realistic system usage in an interactive design support scenario, the number of retrieved cases k is varied. In practice, users are typically presented with multiple similar cases rather than a single best match, and increasing k provides a more realistic view of whether the system retrieves the expert-relevant case among the top suggestions. In this evaluation, k is varied ($k = 1, 2$, and 3).

Case retrieval is based on a weighted similarity measure, where individual case attributes contribute to the overall similarity score according to their relevance for DfM problem similarity. For this initial evaluation, uniform attribute weights have been applied, such that all attributes contributed equally to the similarity computation. This design choice is made intentionally to keep the experiment simple and to focus on assessing the feasibility of the proposed case representation itself. In addition, determining and validating attribute weights typically requires substantial effort from domain experts. Using equal weights avoids biases introduced by potentially inappropriate or suboptimal weight configurations, which could otherwise distort the evaluation outcomes.

An example query case and the corresponding retrieval results are illustrated in Figure 2. The figure shows how the individual similarity measures are computed during retrieval. The yellow and black arrows indicate the local similarities between the attributes of the two cases, while the yellow

arrows additionally represent the aggregated similarities at the object level. The green arrow denotes the global similarity between the two cases, i.e., the final retrieval score.

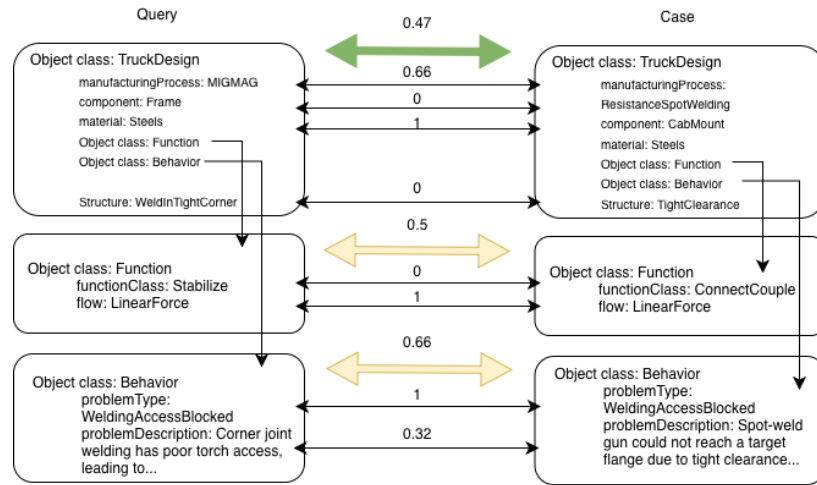


Figure 2: Example Retrieval with Similarity Results

7 Results & Discussion

This section presents the retrieval results for the three query cases and evaluates system performance using the retrieval error metric defined in the previous section. The number of retrieved cases is varied ($k = 1, 2, 3$) to reflect an interactive decision-support setting in which users typically inspect several similar cases rather than a single best match. For each query, retrieval is considered successful if the expert-identified most useful case appears within the top- k results returned by the system.

Table 1: Evaluation Results

Query	$k = 3$	$k = 2$	$k = 1$	Ø Retrieval Error
Q1	0	0	0	0 %
Q2	0	1	1	66,7 %
Q3	0	0	0	0 %
Ø Retrieval Error	0 %	33,3 %	33,3 %	

Table 1 summarizes the results. For Q1 and Q3, the retrieval error is 0 for all values of k , demonstrating that the expert's most useful case is consistently retrieved even when only the top-1 result is considered. For Q2, the retrieval error depends on k : the system succeeds for $k = 3$ (error 0), but fails for $k = 2$ and $k = 1$ (error 1). This leads to an average retrieval error of 33.3% for $k = 1$ and $k = 2$, and 0% for $k = 3$ across the three queries. These results indicate that the developed case representation and similarity computation reliably retrieve the expert-relevant case for two out of three queries, and that presenting multiple candidates improves robustness: when three results are shown ($k = 3$), the expert's preferred case is always included. The failures for Q2 at $k = 1$ and $k = 2$ suggest that, in this particular query, the expert's target case is ranked slightly lower than the top two results, pointing to limitations in the current similarity model (e.g., uniform attribute weights and a small case base) and the case representation.

8 Conclusion & Future Work

This paper presented a case representation for DfM-specific knowledge and demonstrated the feasibility of a CBR system using this representation by a first experimental evaluation. The results indicate that

the proposed case structure is suitable for capturing manufacturability-related design experiences and for retrieving similar cases to support design decision-making in early product development.

However, the evaluation is limited to a small, industrially motivated dataset. Further experiments with larger case bases, ideally using real-world data from industrial practice, are required to assess the robustness, scalability, and general applicability of the proposed approach. In addition, this work assumed that all case attributes are equally important. Future work could therefore investigate adaptive attribute weighting and learning-based similarity refinement to improve retrieval quality and increase coverage of additional relevant cases.

Future research should focus in particular on the adaptation phase of the CBR cycle to investigate how previously applied solutions can be systematically adapted to new DfM problems. In addition, further research could explore methods for integrating information from non-textual sources, such as technical drawings or CAD models, to enable reuse of manufacturability knowledge embedded in legacy design artifacts.

Acknowledgements

This research was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 543073350.

References

- [1] D. Kalaitzi et al., “Megatrends and Trends Shaping Supply Chain Innovation,” in *Next Generation Supply Chains*, R. Fornasiero, S. Sardesai, A. C. Barros and A. Matopoulos, Eds., Lecture Notes in Management and Industrial Engineering. Cham: Springer, 2021. doi: 10.1007/978-3-030-63505-3_1.
- [2] M.-L. Germain, “The impact of changing workforce demographics and dependency on technology on employers’ need for expert skills,” in *Expertise at Work: Current and Emerging Trends*, pp. 177–195, 202.
- [3] A. De Bem Machado, S. Secinaro, D. Calandra and F. Lanzalonga, “Knowledge management and digital transformation for industry 4.0: a structured literature review,” *Knowledge Management Research & Practice*, vol. 20, pp. 320–338, 2022.
- [4] A. E. Patterson, Y. H. Lee and J. T. Allison, “Generation and Enforcement of Process-Driven Manufacturability Constraints: A Survey of Methods and Perspectives for Product Design,” *Journal of Mechanical Design*, vol. 143, no. 11, Art. no. 110801, Nov. 2021. doi: 10.1115/1.4050740.
- [5] HCL Software, “Powerful Design for Manufacturing Software | HCL DFMPPro.” Available: <https://dfmpro.com/>. Accessed: Jul. 11, 2025.
- [6] Boothroyd Dewhurst, “DFMA® Software: Design for Manufacture and Assembly.” Available: <https://www.dfma.com/>. Accessed: Jul. 11, 2025.
- [7] Y. Zhang, S. Yang, G. Dong and Y. F. Zhao, “Predictive manufacturability assessment system for laser powder bed fusion based on a hybrid machine learning model,” *Additive Manufacturing*, vol. 41, Art. no. 101946, 2021. doi: 10.1016/j.addma.2021.101946.
- [8] L. Makatura, M. Foshey, B. Wang, F. Hähnlein, P. Ma, B. Deng, M. Tjandrasuwita et al., “Large Language Models for Design and Manufacturing,” *An MIT Exploration of Generative AI*, Mar. 2024. doi: 10.21428/e4baedd9.745b62fa.
- [9] R. Belecheanu, K. Pawar, R. Barson and B. Bredehorst, “The Application of Case Based Reasoning to Decision Support in New Product Development,” *Integrated Manufacturing Systems*, vol. 14, 2003. doi: 10.1108/09576060310453335.
- [10] L. Qian and J. S. Gero, “Function–behavior–structure paths and their role in analogy-based design,” *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, vol. 10, no. 4, pp. 289–312, 1996. doi: 10.1017/S0890060400001633.
- [11] Kunz, L., Thanabalan, K., Legler, T., Malburg, L., Bergweiler, S., Ruskowski, M., & Bergmann, R. (2025). XDP-Opt: Experience-Based Design Process Optimization for Industrial Manufacturing. *Proceedings of the KI2025 Workshop of the SIG Knowledge Management (FG WM)*, Potsdam, Germany, 2025, In Press.
- [12] H. Österle, J. Becker, U. Frank et al., “Memorandum zur gestaltungsorientierten Wirtschaftsinformatik,” *Schmalenbachs Zeitschrift für betriebswirtschaftliche Forschung*, vol. 62, pp. 664–672, 2010. doi: 10.1007/BF03372838.

- [13] D. M. Anderson, *Design for Manufacturability: Optimizing Cost, Quality, and Time to Market*. Cambria, CA, USA: CIM Press, 1990.
- [14] G. Paul and R. Al-Dirini, "Design for Manufacturing – A Review and Case Study," 2009.
- [15] L. Kunz, M. Klostermeier, K. Thanabalan, T. Legler and M. Ruskowski, "Structural Analysis, Simulation & Testing," in *Proceedings of the 36th Symposium Design for X (DFX 2025)*, D. Krause, K. Paetzold and S. Wartack, Eds., DfX Series, 2025, pp. 179–188. doi: 10.35199/dfx2025.19.
- [16] J. Bralla, *Design for Manufacturability Handbook*, 2nd ed. New York, NY, USA: McGraw-Hill Professional, 1998.
- [17] M. M. Richter and R. O. Weber, *Case-Based Reasoning: A Textbook*. Heidelberg: Springer, 2013.
- [18] R. Bergmann, K.-D. Althoff, M. Minor, M. Reichle and K. Bach, "Case-Based Reasoning – Introduction and Recent Developments," *Künstliche Intelligenz*, no. 1, pp. 5–11, 2009.
- [19] A. Aamodt and E. Plaza, "Case-based reasoning: foundational issues, methodological variations, and system approaches," *AI Communications*, vol. 7, no. 1, pp. 39–59, Mar. 1994.
- [20] W. Wang and J. Zhao, "Case-based reasoning in product innovation design," in *Proceedings of the 2006 International Technology and Innovation Conference (ITIC 2006)*, Hangzhou, China, 2006, pp. 267–272. doi: 10.1049/cp:20060768.
- [21] M. L. Maher and A. Gomez de Silva Garza, "Developing case-based reasoning for structural design," *IEEE Expert*, vol. 11, no. 3, pp. 42–52, Jun. 1996. doi: 10.1109/64.506753.
- [22] M. Mabkhot, A. Al-Samhan and L. Hidri, "An Ontology-Enabled Case-Based Reasoning Decision Support System for Manufacturing Process Selection," *Advances in Materials Science and Engineering*, vol. 2019, pp. 1–18, 2019. doi: 10.1155/2019/2505183.
- [23] A. Aamodt, "Knowledge-Intensive Case-Based Reasoning in CREEK," in *Advances in Case-Based Reasoning*, P. Funk and P. A. González Calero, Eds., *Lecture Notes in Computer Science*, vol. 3155. Berlin, Heidelberg: Springer, 2004. doi: 10.1007/978-3-540-28631-8_1.
- [24] Y. Zhang, X. Liu, J. Jia and X. Luo, "Knowledge Representation Framework Combining Case-Based Reasoning with Knowledge Graphs for Product Design," *Computer-Aided Design and Applications*, vol. 17, pp. 763–782, 2019. doi: 10.14733/cadaps.2020.763-782.
- [25] B. U. Haque, R. A. Belecheanu, R. J. Barson and K. S. Pawar, "Towards the application of case based reasoning to decision-making in concurrent product development (concurrent engineering)," *Knowledge-Based Systems*, vol. 13, nos. 2–3, pp. 101–112, 2000. doi: 10.1016/S0950-7051(00)00051-4.
- [26] Combining knowledge-based engineering and case-based reasoning for design and manufacturing iteration
- [27] N. Janthong, D. Brissaud and S. Butdee, "Combining axiomatic design and case-based reasoning in an innovative design methodology of mechatronics products," *CIRP Journal of Manufacturing Science and Technology*, vol. 2, no. 4, pp. 226–239, 2010. doi: 10.1016/j.cirpj.2010.05.005.
- [28] M. L. Maher and A. Gomez de Silva Garza, "Case-based reasoning in design," in *IEEE Expert*, vol. 12, no. 2, pp. 34–41, March-April 1997, doi: 10.1109/64.585102.
- [29] S. El-Sappagh and M. Elmogy, "Case Based Reasoning: Case Representation Methodologies," *International Journal of Advanced Computer Science and Applications*, vol. 6, pp. 192–208, 2015. doi: 10.14569/IJACSA.2015.061126.
- [30] J. S. Gero and U. Kannengiesser, "The Function-Behaviour-Structure Ontology of Design," in *An Anthology of Theories and Models of Design*, A. Chakrabarti and L. Blessing, Eds. London: Springer, 2014. doi: 10.1007/978-1-4471-6338-1_13.
- [31] C. Bryant, R. Stone, J. Greer, D. McAdams, T. Kurtoglu and M. Campbell, "A Function-Based Component Ontology for Systems Design," in *Proceedings of the International Conference on Engineering Design (ICED'07)*, Paris, France, Aug. 28–31, 2007.
- [32] R. Bergmann, L. Grumbach, M. Hoffmann, M. Hotz, L. Jung, M. Lenz, L. Malburg, A. Schultheis and C. Zeyen, *ProCAKE Framework*, version 7.0.0. Zenodo, Oct. 2024. doi: 10.5281/zenodo.14020042.
- [33] ProCAKE Wiki, "ProCAKE – The Process-Oriented Case-Based Knowledge Engine," *GitLab pages*. Accessed: Jan. 24, 2026. Available: <https://procake.pages.gitlab.rlp.net/procake-wiki/>
- [34] P. Klein, L. Malburg and R. Bergmann, "Learning Workflow Embeddings to Improve the Performance of Similarity-Based Retrieval for Process-Oriented Case-Based Reasoning," in *Case-Based Reasoning Research and Development*, *Proc. of the 27th Int. Conf. on Case-Based Reasoning (ICCBR 2019)*, Otzenhausen, Germany, Sept. 8–12, 2019, K. Bach and C. Marling, Eds., *Lecture Notes in Computer Science*, vol. 11680. Cham: Springer, 2019, pp. 188–203.

AI- and Data Science-Driven Decision Support: Transforming Early Product Creation and Collaboration in Product Development

Michael Tatschl^{1*}, Gerolf Schalk^{1,2}, Hannes Hick¹

¹Graz University of Technology, Institute of Machine Components and Methods of Development, Austria

²Siemens Industry Software GmbH, Science, Academics & Transfer

*Corresponding author. E-Mail address: michael.tatschl@tugraz.at

Abstract:

Early product creation involves decision-making under uncertainty, driven by heterogeneous customer inputs, multidisciplinary dependencies, and iterative stakeholder coordination. Translating informal customer needs into precise system definitions remains a significant challenge. This paper proposes an AI- and data science (DS)-driven automation framework that integrates customer-interactive, human-in-the-loop requirements engineering with automated system concept development and evaluation. The approach is validated using an unmanned aerial vehicle use case in an industrial-grade digital engineering environment. The results demonstrate how integrated AI- and DS-supported workflows can improve efficiency, decision-making quality, coordination, and traceability in early product creation.

Keywords:

Engineering Automation, AI- and Data Science- Supported Product Development, Human-Machine Interaction, Decision-Support, Systems Engineering

1 Introduction and Problem Statement

The development of complex systems is characterized by a vast solution space in which numerous design variants may satisfy a given set of system requirements. Generating initial concepts, comparing them, and determining the most promising solutions within this space is particularly challenging due to complex dynamics, high uncertainty, adaptive decision-making behavior, and collaborative design iterations [1]. As a result, the definition of initial system concepts remains a largely manual and time-consuming activity, despite rapid advances in digital engineering.

A major source of uncertainty in early product creation lies in requirements engineering (RE) as a starting point of development. Customers typically express their needs in incorrect, ambiguous, and incomplete requirements in natural language, making it problematic for deriving technically precise and machine-interpretable requirements [2]. Engineers must therefore invest significant manual effort in interpreting, refining and validating these inputs before they can be used for downstream engineering activities such as system modeling and system design conception. This disconnection between customer intent and unambiguous, formal system representation limits the extent to which RE activities can be automated in early product creation [3]. Recent advances in data science (DS) and artificial intelligence (AI) offer substantial potential to support engineering activities. In particular, natural language processing (NLP) and AI techniques have been increasingly applied to support RE tasks. However, many existing approaches focus on isolated tasks and are not embedded into integrated engineering workflows. As a result, the overall impact of AI-based techniques often remains limited due to technical and organizational challenges, hindering the establishment of coherent automation workflows. [4]

This paper addresses this gap by proposing an AI- and DS-supported automation framework that seamlessly connects customer-driven RE with automated system design conception. This approach employs an AI-assisted requirements processing pipeline to transform customer-provided inputs into structured, consistent, and partially parameterized system requirements. These requirements serve as the foundation for the automated generation and evaluation of system design concepts, enabling rapid exploration of feasible solutions and providing a structured basis for coordination and decision-making between customers and engineers. The contribution of this work lies in demonstrating how DS- and AI-based approaches can be orchestrated within an automation framework for early product creation, with a particular emphasis on RE and automated system concept development. By focusing on technical workflow integration rather than isolated technical implementations, the proposed approach supports hybrid intelligence in product creation, where human expertise as well as DS- and AI-based methods jointly contribute to informed decision-making. Accordingly, this paper is positioned within the HIPPE track “*DS/AI Interaction on Product Creation*.”

2 Related Work and Background

This chapter explores the state of the art in engineering, particularly in the automation of RE and product design concept development. Research in these fields has gained significant momentum in recent years, driven by advances in NLP, DS, AI, and data-driven analysis techniques, as discussed subsequently.

In current RE literature, most concepts focus on automating isolated tasks such as requirements elicitation, classification, consistency checking, and ambiguity detection [5]. While these approaches improve the efficiency and quality of individual activities, they typically operate as standalone solutions and are not embedded into a broader automation framework spanning early system conceptualization. In parallel, only few research efforts address the interaction of intuitive, interactive user interfaces (UI) [6] that support human-machine interaction during requirements formulation, analysis, and refinement. To address this gap, a dedicated multi-step RE agent is introduced in this paper. This agent, similar to suggestions from recent work [7], combines DS and AI capabilities to ensure structured input for subsequent design activities. Moreover, direct interaction with customers’ requirement repositories in Application Lifecycle Management (ALM) systems is rarely considered [8], limiting the reuse of existing requirements data as well as hindering seamless transitions and data flow during system conceptualization. For the automation of system concept development, structured inputs must be represented and managed in a formalized manner to ensure continuity and traceability across the individual process stages. In this context, system models as employed in Model-Based Systems Engineering (MBSE) offer a rigorous means to formally capture system structure, behavior, and parametric relationships, and are therefore chosen as an information backbone for automated design workflows in this paper [9]. Within the system concept design phase, the resulting design space is systematically explored to identify feasible system configurations that satisfy requirements while adhering to technical and disciplinary constraints. Prior research has proposed a range of approaches that integrate MBSE with multidisciplinary design analysis and optimization methods to address the inherent complexity of such tasks [10, 11]. These contributions emphasize holistic frameworks for system concept development, enabling the coordinated consideration of multiple engineering domains and supporting the identification of optimized design solutions.

As a result, a central research gap emerges at the intersection of these research streams. While AI- and DS-based approaches in RE primarily focus on the linguistic and semantic processing of individual requirements, and model-based approaches in system design and optimization address the quantitative evaluation of predefined solution spaces, their integration into a coherent, end-to-end automation framework remains insufficiently addressed. Recent review work highlights that AI-based methods are often applied in a fragmented manner across different phases of product development, leading to siloed

insights and a lack of continuity across product creation phases [12]. Against this background, existing work provides limited insight into how human-centered requirements definition can be systematically combined with AI-based analysis and automated system conceptualization within a single, continuous workflow. This gap is especially relevant for early product creation phases, where effective coordination and decision-making between customers and engineers are critical [1]. Without an integrated automation logic, requirements remain detached from system concept generation, limiting the applicability of DS- and AI-based methods to support tools rather than comprehensive product conception solutions.

This paper addresses this gap by proposing an AI- and DS-supported automation framework that integrates interactive RE and automated system design concept generation, rather than introducing new isolated algorithms. With the objective of developing a universally applicable AI- and DS-supported automation framework incorporating human-machine interaction via a UI enabling customer inputs to be iteratively refined, structured, and transformed into actionable input for automated system conceptualization, the following research question is investigated: *“How can RE and system conceptualization processes be arranged interactively and automated to support the coordination and decision-making in early product creation, and how can AI and DS be leveraged to support these processes?”*

3 Research Methodology

This paper addresses the research question by proposing a universally applicable AI- and DS-supported automation framework for interactive RE and automated system design concept generation. The proposed framework, illustrated in Figure 1, is structured into three main steps: 1) Requirements engineering, 2) System concept development, and 3) System concept communication.

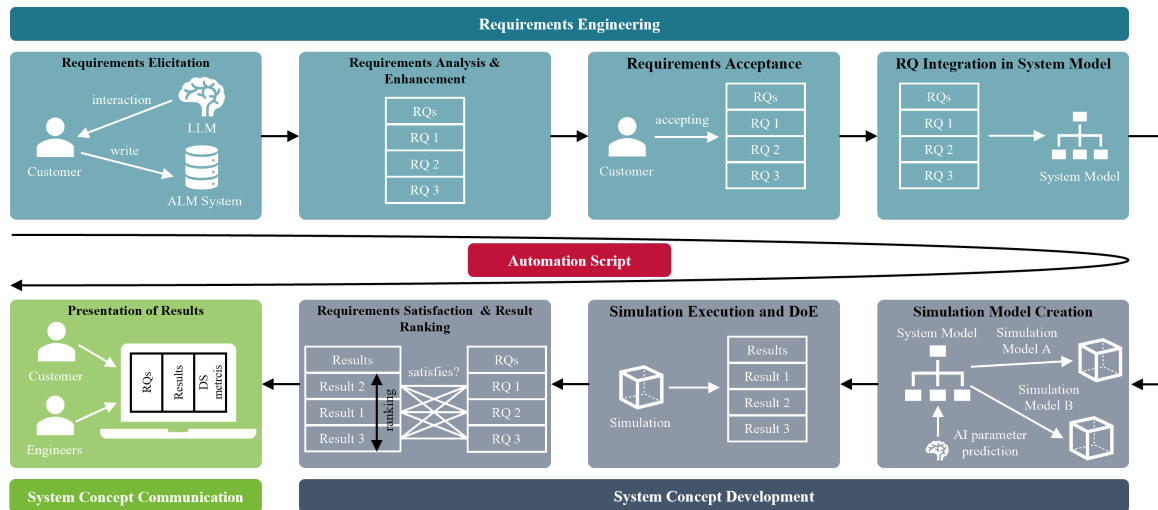


Figure 1: Automation Framework

The methodology follows a design-oriented research approach, in which a structured automation framework is developed, implemented, and validated against clearly defined objectives derived from the research question. To ensure traceability between the research question and the methodological contribution, the proposed framework is assessed based on the fulfillment of the following objectives:

- *Obj1:* The customer defines and refines requirements interactively using a machine-UI.
- *Obj2:* Customer requirements are automatically transferred into structured, correct, and unambiguous technical requirements.
- *Obj3:* System concepts covering the whole solution space are generated and evaluated against the requirements automatically.

- *Obj4*: The system concepts are ranked based on various criteria.
- *Obj5*: The system concepts are presented to the stakeholders by providing expressive metrics and visualizations highlighting trade-offs.
- *Obj6*: Activities where AI and DS approaches can be beneficially applied are identified.

Leveraging MBSE system models as a data backbone and a model-based definition of the workflow, the automation logic within a unified script gains access to all relevant engineering data in a formalized and machine-readable form. In MBSE, state-of-the-art descriptive system models created using semantically rich modeling standards provide abstraction and data traceability across system and product realization levels [13]. By using MBSE, execution of automated analysis, generation, and evaluation steps in a defined and reproducible sequence is enabled. Basis of the proposed automation is a process model, representing the workflow from Figure 1, but in more detail with several sub-workflows and input as well as output definitions. On the lowest hierarchical level, each workflow step is equipped with a script enabling its execution. The workflow is modeled in a model-based environment to make the information accessible and useable to the superior automation script, which identifies all workflow steps, applied methods, and executes the linked scripts in the defined sequence. This model-based approach enables a clear definition of the inputs and outputs needed to execute a workflow step and, additionally, allows a high variability by facilitating the substitution and complementation of the workflow or the applied methods. The data exchanged between workflow steps is equipped with metadata that defines how it should be stored and how to query it in the database.

For each workflow step, the methodology is described in detail in the following sections and assessed with respect to the fulfillment of the objectives listed above. To demonstrate practical applicability, the proposed framework is exemplified using a mechatronic system introduced in Chapter 4. While this use case provides a concrete validation scenario, the methodology itself is not limited to a specific product and is therefore applicable to a wide range of mechatronic systems.

3.1 Requirements Engineering

The goal of automated RE is to enable interactive, AI-supported definition, refinement, and structuring of customer requirements, while ensuring that the resulting requirements are consistent, machine-interpretable, and suitable as input for automated system concept development. This step primarily addresses *Obj1* and *Obj2*.

The automated RE approach combines human-machine interaction, rule-based analysis, DS, and state-of-the-art large language model (LLM)-based refinement within a single automation script. Customers interact with the system via a UI that allows them to input, review, and iteratively refine their requirements, as illustrated in Figure 2. In line with a recent human-in-the-loop RE initiative, this interaction helps to keep human intent and contextual knowledge central to the RE process, while AI- and DS-based methods support requirements formalization, validation, and traceability [14]. Customer requirements can be provided either through structured files (e.g., spreadsheet-based inputs) or via direct interfaces to a requirements repository in an ALM system, which are accessed through standardized application programming interfaces (APIs). This enables seamless reuse of existing requirements data and ensures compatibility with industrial requirements management environments. The automated requirements processing pipeline is triggered by the customer via the UI and operates on a combination of linguistic and semantic analysis techniques. Initially, requirement descriptions are analyzed using NLP methods to extract syntactic and semantic features. A DS-based classifier utilizing sentence embeddings is employed to assess the consistency and correctness of requirement descriptions by assigning the labels *consistent* or *not consistent*. A requirement is classified as consistent if it fulfills established RE principles, such as SMART criteria (specific, measurable, achievable, relevant, and traceable) and the requirements boilerplate [15].

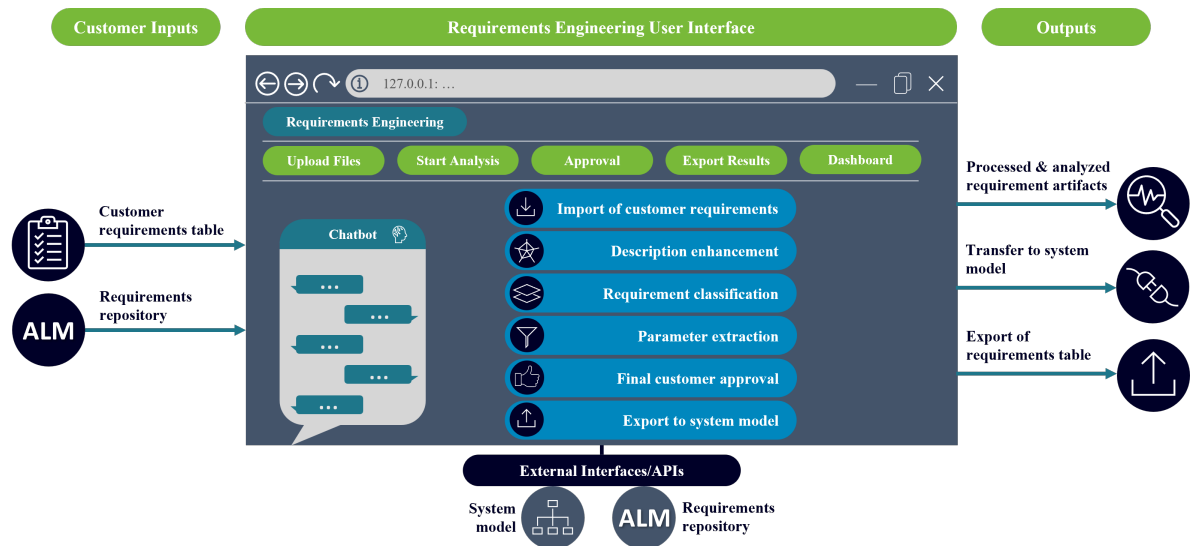


Figure 2: Requirements Engineering User Interface

To increase robustness, consistency classification results are verified by combining multiple analysis mechanisms – namely an NLP language library and a DS classifier trained on consistently and inconsistently formulated requirements – before assigning a final label. Requirements identified as inconsistent are subsequently refined using an LLM-based approach, which reformulates the requirement description in accordance with the requirements boilerplate via API-based access. However, LLM-based text generation still suffers from limited controllability, inexplicable decision-making processes, and hallucinations [16]. To address this challenge, the proposed approach applies constrained, rule-based utilization of the boilerplate to reduce generative behavior and undesired output. Subsequently, after description enhancement, the requirements are re-evaluated for consistency and, if validated, forwarded to the next processing step. In addition, the unambiguity of requirement descriptions is assessed with a correct reference requirements set which captures the context to ensure unambiguity and apply it to the customer requirements set. This context-capturing is executed via LLM-support and DS-measures such as cosine similarity to ensure high-quality customer requirements. In the next step, requirements are classified into quantitative and qualitative categories. This classification is based on predefined rule-based patterns (e.g., unit identifiers and domain-specific keywords) in combination with NLP-based analysis. For quantitative requirements, numerical values, units of measurement, and relational operators (e.g., less than, equal to, between, greater) are extracted and mapped to semantically meaningful parameter names. Finally, the customer can review the processed and analyzed requirements directly within the UI or export the results as a structured table. Upon customer approval, the requirements – together with their associated parameters – are transferred into a system model via the modeling tool’s APIs. This structured and uniform transformation procedure enables the direct, model-based use of requirements and parameters as input for downstream engineering activities. Consequently, the automated RE pipeline clearly fulfills *Obj1* by enabling interactive customer involvement through a UI and *Obj2* by transforming informal requirements into structured and high-quality technical requirements by utilizing AI and DS. The resulting requirements form a consistent and machine-interpretable basis for system concept development and evaluation in subsequent workflow steps.

In addition to the core requirements processing pipeline, the UI provides analytical and interaction-oriented features that support transparency, validation, and decision-making during RE. A dedicated dashboard visualizes key indicators of the requirements analysis, such as the ratio of consistent and inconsistent requirements, the distribution of qualitative and quantitative requirements, and the criticality of requirements. These visualizations enable customers to assess the current quality,

completeness, and reliability of the requirements set immediately and they can be exported as a structured report for documentation processes. Furthermore, the UI integrates an in-UI conversational assistant based on an LLM. This assistant supports users by answering questions related to the UI, general RE principles, and the analyzed requirements themselves. Rather than replacing expert judgment, this functionality serves as an interactive decision-support mechanism that enhances customer understanding and facilitates informed interaction within the automated workflow.

3.2 System Concept Development

The goal of this step is to evaluate the whole solution space and find optimized solutions while considering all requirements and technical constraints, aiming at fulfilling *Obj3* and *Obj4*. The system requirements elaborated in the step before are used as a trigger to start the automated system concept development workflow, providing - depending on the computation time - nearly instantaneous feedback on the feasibility of fulfilling the requirements and on key performance indicators of possible system concepts.

A data backbone providing semantically linked information about the requirements, the technical system and available quantitative models together with a high degree of formalization is essential to enable automatized data handling. MBSE models fulfil the requirements of formalization and are, therefore, chosen for describing the structure and behavior of the system. In this paper, the system model is a pre-created superset model, representing all potential variants of the system and its subsystems as well as the associated technical constraints within one model. The model comprises structural as well as functional descriptions of the system, extended with variation points indicating the allocation of structural blocks, behavioral blocks and parameters to a certain design variant. [9] All realizable variants of system are, therefore, defined by the content and degree of detail of the pre-created system model, establishing a bottleneck limiting the considered solution space. Requirements and quantitative models are linked to the data artifacts from the system model by integrating and relating all engineering data into an enterprise knowledge graph. In that way, the semantic linkage of data from heterogeneous sources is enabled, positioning the system model as a data backbone to make data actionable in a wider context, similar to the architecture proposed in [17]. The relationships between data points in the knowledge graph rely on a data meta model that defines how data must be structured to support the desired functionalities needed for the automation workflow. For example, system functions fulfil system requirements and can be decomposed into subfunctions. Simulation models are allocated to respective subfunctions, drawing the link from the descriptive system model to quantified analysis. If multiple alternatives exist to realize these functions, such as different variants of a subsystem, different simulation models may need to be considered. In these cases, the simulation models are further described with conditions on their applicability. Parameters, serving as input and/or output of functions, are described by value ranges and step sizes to define the possible solution space. Certain parameters cannot be described meaningfully with value ranges, such as the system mass, as they are highly dependent on other parameters. These parameters are estimated via calculations, which can either be rule-based formulas or predictive models. Technical constraints are linked to certain parameters and apply conditions on their parameter values.

Providing the customer with reliable results necessitates the full exploration of the solution space stretched by the parameters formalized in the system model. In literature, many possibilities of doing that are discussed [18]. For demonstration purposes and the sake of simplicity, full factorial design of experiments (DoE) is used in this paper. After applying full factorial DoE to generate parameter sets representing the whole solution space, these parameter sets are used as an input to system simulations quantifying the behavior of all the investigated concepts. The appropriate simulation model representing each function of a certain concept is selected based on the underlying conditions, integrated into a

simulation environment and, if necessary, connected to other simulation models there. Load profiles necessary as an input for the system simulation must be provided initially or derived from the system requirements using a dedicated model. The simulation results of each concept are evaluated against the requirements, and underperforming parameter sets are dismissed. The remaining concepts are further evaluated and ranked regarding certain criteria, such as cost or mass.

The methodology proposed for the system concept development successfully achieves both objectives *Obj3* and *Obj4* by leveraging formalized data storage and modelled automation workflows to trigger the execution of quantitative models, evaluate the results against the requirements and a rule-based ranking approach. Regarding *Obj6*, neural networks in the form of predictive models to estimate high-level parameters in early stages of product development are identified to be a beneficial application of AI in this development stage.

3.3 System Concept Communication

As a final step, the automatically generated and ranked concepts are communicated to the customer and engineers via a UI, providing a clear differentiation between the suggested concepts utilizing metrics regarding multiple ranking criteria. Furthermore, many different other expressive metrics, which are beneficial for decision making, can be presented - for example, an index quantifying the extent to which requirement-defined goals are exceeded, or a similarity index identifying concepts similar to previously developed systems, for which high confidence in development success can be assumed. An example of such a similarity index is the DS-based metric cosine similarity, showing the beneficial application of DS in this step for *Obj6*. Pareto fronts are used for visualizing the differences between concepts and potential trade-offs regarding key parameters, facilitating comparison. By providing stakeholders ranked results, DS-based metrics and pareto fronts, the UI facilitates comparison of solutions and provides a basis for discussions between customers and engineers, fulfilling *Obj5*.

4 Framework Application on Use Case

This section presents the application of the proposed automation framework to a multirotor unmanned aerial vehicle (UAV) use case and demonstrates its execution within an industrial-grade environment. In doing so, the feasibility of the framework in the context of early system conceptualization is illustrated. The UAV use case was selected due to its representative characteristics for early product creation, including its multidisciplinary nature and the presence of performance-related trade-offs.

For the application of the framework, a representative set of customer-defined requirements for a UAV system is provided via the RE-UI. The input comprises high-level expectations related to flight performance, handling, safety, cost, and system configuration. As illustrated in Figure 3, these requirements reflect realistic customer input encountered in early product creation and therefore include heterogeneous levels of detail as well as underspecified formulations. After the customer initiates the RE process, the UI automatically analyzes the textual descriptions and identifies requirements that violate consistency. In the presented use case, requirements related to handling and safety are flagged as *inconsistent* and are explicitly highlighted in the UI. For these requirements, the UI generates revised, boilerplate- and SMART criteria compliant formulations using an LLM-based refinement step. The proposed consistent descriptions are presented transparently to the customer, who can review, edit if necessary, and explicitly accept the updated formulations. This interaction demonstrates the human-in-the-loop character of the approach, where automated reasoning is combined with user validation rather than replacing it. Following acceptance, the workflow generates a consistent, structured set of system requirements and automatically extracts quantitative parameters in machine-readable form, such as maximum allowable mass and minimum flight duration. These requirements are transferred to the system model, where automated scripts impose the corresponding parameter constraints.

Requirements Engineering UI

Customer Requirements Preview

Requirements Type	ID	Requirements Titel	Requirement Description	source	Consistency
Drone Requirement	REQ-01	Flight Duration	The drone must reach a minimum flight time of 15 minutes at full payload.	customer	consistent
Drone Requirement	REQ-02	System Handling	The drone shall enable easy handling in normal flight conditions.	customer	not consistent
Drone Requirement	REQ-03	System Controllability	The signal range must be at least 2.5 km.	customer	consistent
Drone Requirement	REQ-04	System Costs	The total cost of the system must not exceed 120 €.	customer	consistent
Drone Requirement	REQ-05	System Configuration	The system must be able to support a quadcopter configuration.	customer	consistent
Drone Requirement	REQ-06	System Safety	The drone shall provide a high level of operational safety.	customer	not consistent
Drone Requirement	REQ-07	System Mass	The overall drone mass must not exceed 2 kg.	customer	consistent
Drone Requirement	REQ-08	System Thrust	The system must be able to reach a speed of 100 km/h at full payload.	customer	consistent

☒ Show Inconsistencies

Inconsistent Customer Requirements

Requirements Type	ID	Requirements Titel	Requirement Description	source	Consistency
Drone Requirement	REQ-02	System Handling	The drone shall enable easy handling in normal flight conditions.	customer	not consistent
Drone Requirement	REQ-06	System Safety	The drone shall provide a high level of operational safety.	customer	not consistent

☒ Create Consistent Formulations

Refined Descriptions by LLM-API

ID	Old Description	New Description (LLM)
REQ-02	The drone shall enable easy handling in normal flight conditions.	The system must be controllable by the operator during windless conditions with a control delay of less than 200 ms.
REQ-06	The drone shall provide a high level of operational safety.	The system shall achieve a mean time between failures of at least 100 flight hours in standard laboratory conditions.

Figure 3: Excerpt of Requirements Engineering User Interface – UAV System Use Case

Together with predefined technical constraints and all design variants and parameter ranges, a full-factorial DoE discretizes the feasible solution space. Drone mass, which cannot be meaningfully bounded by simple ranges due to strong parameter interdependencies, is estimated for each design point using a neural network trained on data from previously developed drones, aiming at improving early-stage accuracy over rule-based methods. Flight duration is evaluated via system simulation using an automatically generated flight profile derived from requirement values. All relevant models and data are managed in a knowledge graph, which is queried to assemble and execute the appropriate simulations for each parameter set. Designs that fail to meet requirements are discarded, while the remaining concepts satisfy all constraints but exhibit different performance trade-offs. For example, one concept achieves a long flight duration but at the cost of a relatively high mass, whereas another concept only marginally satisfies the flight duration requirement while exhibiting significantly lower mass. To support the identification of the most suitable solution for the customer, an UI is introduced, which can be seen in Figure 4. This UI enables a multi-criteria ranking, e.g., regarding mass and flight duration, directly by the customer or by the engineer. However, also additional metrics can be incorporated, which are evaluated separately using dedicated quantitative models. Examples for such metrics are cost, environmental impact, or DS-based metrics, including cosine similarity to previously developed products. Pareto fronts can be used as another way to visualize the trade-offs between different concepts, like the trade-off between system mass and flight duration, which is displayed in Figure 4.



Figure 4: Concept Analysis User Interface – UAV System Use Case

The proposed automation framework was implemented and executed in a controlled engineering environment, namely the Digital Lifecycle Lab (DLL) located at Graz University of Technology [19]. The DLL provides interconnected access to industrial-grade digital engineering solutions, including ALM systems, system modeling environments, and simulation tools. This integrated setup enables the reproducible execution of AI- and DS-based workflow steps under realistic engineering conditions. Within this environment, the proposed framework was validated with respect to its technical feasibility and its capability to support interactive RE and downstream automation steps. The DLL thus serves as a neutral validation environment focusing on workflow integration and methodological soundness.

5 Results and Discussion

The UAV use case demonstrates the feasibility and effectiveness of the proposed automation framework in early system conceptualization. The results show that interactive RE supported by a machine–UI enables customers to define and iteratively refine requirements in a transparent human-in-the-loop process (*Obj1*). Automated detection and LLM-based refinement of inconsistent or underspecified requirements led to boilerplate-compliant and technically interpretable formulations, while retaining user authority over final decisions. Following validation, customer requirements were automatically transformed into structured, unambiguous technical requirements and machine-readable parameters, which were consistently transferred into the system model (*Obj2*). This enabled seamless downstream automation while maintaining traceability between requirements, parameters, and generated system concepts. Based on these inputs, the framework automatically generated and evaluated system concepts covering the discretized solution space using a full factorial DoE (*Obj3*). Hybrid evaluation models were applied, combining system simulation for flight duration with a neural network–based mass prediction to enable realistic assessments in early design stages. All concepts violating requirements or technical constraints were automatically filtered out. The remaining feasible concepts exhibited trade-offs between competing criteria, such as mass and flight duration. These concepts were ranked using a dedicated UI that allows stakeholders to prioritize evaluation criteria according to their preferences (*Obj4*). To support informed decision-making, expressive metrics and visualizations, including Pareto front representations, were provided to highlight trade-offs between non-dominated solutions (*Obj5*). Finally, the use case highlights beneficial applications of AI and DS methods within the framework, including LLM-based requirement refinement, data-driven mass prediction, and similarity-based metrics leveraging a knowledge graph (*Obj6*). Overall, the results illustrate that the proposed framework effectively integrates RE, automated concept generation, AI-supported evaluation, and decision support into a coherent workflow suitable for early-stage system design.

At the same time, several limitations and opportunities for further improvement can be identified. As the current implementation constitutes a prototype, the emphasis lies on demonstrating methodological feasibility rather than achieving completeness, robustness, or industrial-scale performance. Therefore, several aspects must be further developed to ensure its applicability in an industrial context. In particular, the level of detail of both the system model and the data meta model needs to be increased to capture additional behavioral phenomena and a broader range of structural variants during early concept development. Moreover, the reliance on full factorial design of experiments is not scalable for large design spaces and should therefore be replaced or complemented by more efficient solution-space exploration strategies that balance computational effort and result fidelity. In addition, industrial applicability requires robust strategies for model selection and model coupling, as well as a systematic quantification of the uncertainties introduced by simulation-based and AI-driven models. Equally critical is the establishment of comprehensive data and model governance mechanisms, including quality assurance, version control, and traceability, to ensure reliability and reproducibility across the engineering lifecycle. From an infrastructure perspective, modeling and simulation environments must

be extended to handle large volumes of heterogeneous data, and a fully integrated, API-based toolchain is necessary to support seamless automation across tools and disciplines. Finally, advanced preference modeling approaches that go beyond Pareto-front analysis and rule-based ranking, together with role-specific UIs for different stakeholder groups, are required to make results accessible, interpretable, and actionable for industrial decision-makers.

Future work is therefore required to enhance scalability, usability, and computational efficiency, and to assess the transferability of the proposed workflow to a broader range of mechatronic systems beyond the presented UAV use case. Nonetheless, the presented framework already enables an accelerated, consistent, and traceable development process, emphasizing structured communication and demonstrating the potential of AI- and DS-supported workflows to enhance early product creation.

References

- [1] Fernandes, J. V., Henriques, E., Silva, A., and Pimentel, C. 2017. Modelling the dynamics of complex early design processes: An agent-based approach. *Design Science* 3, e19.
- [2] Rosado da Cruz, A. M., and Cruz, E. F. 2025. Machine Learning Techniques for Requirements Engineering: A Comprehensive Literature Review. *Software* 4, 3, 14.
- [3] Femmer, H., Unterkalmsteiner, M., and Gorschek, T. 2017. Which requirements artifact quality defects are automatically detectable? A case study. In *2017 IEEE 25th International Requirements Engineering Conference Workshops (REW)*. IEEE, 400–406.
- [4] Necula, S.-C., Dumitriu, F., and Greavu-Şerban, V. 2024. A systematic literature review on using natural language processing in software requirements engineering. *Electronics* 13, 11, 2055.
- [5] Umar, M. A., and Lano, K. 2024. Advances in automated support for requirements engineering: A systematic literature review. *Requirements Engineering* 29, 2, 177–207.
- [6] Dąbrowski, J., Letier, E., Perini, A., and Susi, A. 2023. Mining and searching app reviews for requirements engineering: Evaluation and replication studies. *Information Systems* 114, 102181.
- [7] Jin, D., Jin, Z., Chen, X., and Wang, C. 2024. MARE: Multi Agents Collaboration Framework for Requirements Engineering. *arXiv preprint arXiv:2405.03256*.
- [8] Dąbrowski, J., Bennaceur, A., Rajbahadur, G. K., Nuseibeh, B., and Alrimawi, F. 2025. Prompt Me: Intelligent Software Agent for Requirements Engineering – A Vision Paper. In *Requirements Engineering: Foundation for Software Quality*. Hess, A. and Susi, A., Eds. Springer Nature Switzerland, Cham, Switzerland, 235–243.
- [9] Object Management Group. 2023. *OMG Systems Modeling Language (SysML®)*, Version 2.0. <https://www.omg.org/spec/SysML/2.0>.
- [10] Bussemaker, J., Boggero, L., and Ciampa, P. D. 2022. From System Architecting to System Design and Optimization: A Link Between MBSE and MDAO. In: *32nd Annual INCOSE International Symposium*. 32nd Annual INCOSE International Symposium, 2022-06-25 - 2022-06-30, Detroit, MI, USA.
- [11] Habermehl, C., Höpfner, G., Berroth, J., Neumann, S., and Jacobs, G. 2022. Optimization Workflows for Linking Model-Based Systems Engineering (MBSE) and Multidisciplinary Analysis and Optimization (MDAO). *Appl. Sci.* 2022, 12, 5316.
- [12] Witkowski, A., and Wodecki, A. 2025. Where does AI play a major role in the new product development and product management process? *Management Review Quarterly*.
- [13] INCOSE 2021. *SE Vision 2035 – Engineering Solutions for a Better World*. INCOSE.
- [14] Beg, A., O'Donoghue, D., and Monahan, R. 2025. Leveraging LLMs for formal software requirements – challenges and prospects. *arXiv preprint arXiv:2507.14330*.
- [15] Maletz, M., Bajzek, M., and Hick, H. 2021. Systems Engineering Methods for Automotive Powertrain Development. In *Systems Engineering for Automotive Powertrain Development*. Hick, H., Küpper, K., and Sorger, H., Eds. Springer, Cham, Switzerland, pp. 402–441.
- [16] Qiu, J., Fang, Q., and Kang, W. 2025. Towards controllable and explainable text generation via causal intervention in LLMs. *Electronics* 14, 16, 3279.
- [17] Bajzek, M., Krems, D., Kollegger, S., Traußnig, T., Dörr, S., Tatschl, M., Kniewallner, J. and Hick, H. 2025. Accelerated Automotive Battery Development to meet Market Opportunities. *INCOSE International Symposium*, 35: 760-777.
- [18] Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., and Tarantola, S. 2008. *Global sensitivity analysis: The primer*. John Wiley & Sons, Ltd.
- [19] Siemens Digital Industries Software, “TU Graz uses Siemens Xcelerator solutions to help industries shift to interdisciplinary, sustainable product development,”. accessed on Jan. 9, 2026. [Online]. Available: <https://resources.sw.siemens.com/en-US/case-study-tu-graz/>.

Literature analysis and development of a metadata model for human capability description in production

Tim Meinecke^{1*}, Thomas Hesse², Iris Gräßler², Matthias Schmidt¹

¹Leibniz University Hannover, Institute of Production Systems and Logistics, 30823 Garbsen

²Paderborn University, Heinz Nixdorf Institute, 33102 Paderborn

*Corresponding author. E-Mail address: meinecke@ifa.uni-hannover.de; Tel.: +49511/762-18129

Abstract: This research article addresses the critical gap in describing human capabilities for assembly, disassembly and reassembly processes in production systems at a granular level. As circular economy principles become essential in manufacturing, understanding production-centered capability requirements is increasingly vital. While most metadata models focus on technical resources, existing human capability models either focus narrowly on human-robot collaboration or lack integration across diverse capability dimensions. This study conducts a systematic literature review following the PRISMA scheme, examining different scientific databases to develop a comprehensive, production-centered metadata model. The resulting model formalizes three fundamental classes as well as two supplemental classes for describing human resources while incorporating knowledge and possible legal requirements essential for production environments. The present article proposes a holistic integration of capabilities from a human factor perspective, with the objective of facilitating data science and artificial intelligence approaches to inform decisions in the product creation process. The developed metadata model provides the foundation for decision support systems that match human capabilities to specific production tasks and requirements, advancing circular manufacturing practices and sustainable production system design.

Keywords:

Human capability, Capability description, Metadata model, Systematic literature review

1 Motivation

In today's production environments, it is becoming increasingly important to anticipate products and their creation at an early stage [1]. Due to dynamic markets, stricter regulations, and intensified competition, companies are faced with the challenge of operating in an economically and ecologically sustainable manner [2,3]. One approach to face these issues lies in the establishment of circular cycles [4,5]. Circular economy (CE) is a model of production and consumption, which involves reusing, repairing, refurbishing and recycling of existing materials and products as long as possible [6,7]. It is particularly important to consider the resources required for manufacturing products in advance to enable material circularity. Despite global efforts to promote a CE, global circularity rate has fallen from 7.2% to 6.9%, as overall material use and primary raw material extraction are increasing faster than the use of secondary materials [8]. According to the Circularity Gap Report 2025, almost one-fifth of global material input ends up in landfill [8]. The European Union can be seen as a pioneer in this regard: In a study conducted in 2022 by the Statistical Office of the EU (Eurostat), 40.8% of the total waste volume in the EU was recycled, and 6.4% was used for energy recovery [9]. There are different strategies to enhance circularity, often referred to as the 9R framework based on POTTING et al. [10]. While there are some low-barrier strategies like the energetic recovery or recycling of material, the higher objective of CE is to extend the lifespan of products for their intended use (e.g., through repair or reuse) or directly influence usage, for example by rethinking usage [7,10]. To reduce the share of linearly used materials

right from the start, measures can be taken into account early on in the product creation process [11]. The potential for increasing the reusability of products or their components is an important aspect of product design. However, achieving this higher goal requires more than just design; implementation in production must also be taken into account.

The reuse of components or products usually requires disassembly with subsequent reassembly. The whole process of assembly, disassembly and reassembly (ADR) needs thorough planning in advance [11]. For example, special design guidelines must be followed and manufacturing techniques must meet certain requirements [1,11]. These requirements correspond to necessary capabilities of a production system. To exemplify, the required capabilities must be described prior to assembly, but this consideration must also be made throughout the entire process of ADR. For instance, to assemble a newly developed module, there is a need for new manufacturing technologies. At the same time, consideration must also be given to the return of the module: Do new processes also need to be implemented or further technologies developed for this purpose? For this reason, digital methods are particularly necessary, which enable early-stage simulation and suggest alternative strategies for ensuring manufacturability and subsequent disassembly [12]. A first approach is to formalize available possibilities for capability description in a digital model so that they can be processed using data science/artificial intelligence (DS/AI) methods [11].

The assessment of ADR capabilities can be related both to the technical level, with machines and manufacturing technologies, and to human capabilities with different skills and characteristics. As a result of ongoing digitalization, the representation of such models in the form of metadata models (MDMs) is beneficial. Thanks to their digital representation, these models can be modified with little effort and can be expanded at a later date. In recent years, the focus of the required capability description has been on the technical level. Examples are the Manufacturing Resource Capability Ontology (MaRCO) [13] or a Semantic Web model for automated robots [14]. Both of them primarily deal with the description of machine capabilities on shopfloor level. In the encroachment on the human domain, research has already been conducted for integrating so-called human factors into assembly systems [15]. However, most of these models only describe the capabilities required of employees to collaborate with robots [16]. There is also a research gap in models that specifically include all ADR processes. The modeling of competencies for all processes of the ADR is still in its early stages. MDMs for describing human resources and capabilities usually target specific areas of application that are not related to production. It is easier to record known parameters such as constitutional characteristics or model them in classes. The main issue stems from the lack of integration of the various approaches into a holistic description of all human capabilities. This leads to the following research question (RQ):

RQ: What does a production-centered MDM look like that makes human resources holistically describable for ADR processes?

This paper aims to provide a comprehensive description of human capabilities in circular production systems. To do this, the paper follows the principles of a systematic literature review (SLR) according to the PRISMA scheme [17]. The following structure of the paper was chosen: Section 2 introduces the concept of human competencies in a general way and the principles of building a MDM. Section 3 presents the methodological approach to conduct a SLR and transparently reflects the decisions that have been made. In Section 4, the MDM for human capability description is developed. The literature identified in the SLR is used for this purpose. Section 5 opens a discussion on the scientific findings and limitations of the model, while Section 6 concludes with a summary and suggestions for future research.

2 Theoretical foundation

2.1 Concept of competence

There are many terms used to describe an employee's overall knowledge and skills. Even though some terms are used interchangeably, a trend can be extracted from the majority of academic literature regarding the dependencies between these terms [18]. This section provides an overview of the general hierarchy. For this purpose, the taxonomy according to ERPENBECK et al. [19] in Figure 1 is used, which shows the relationships between the different contents in the concept of competence.

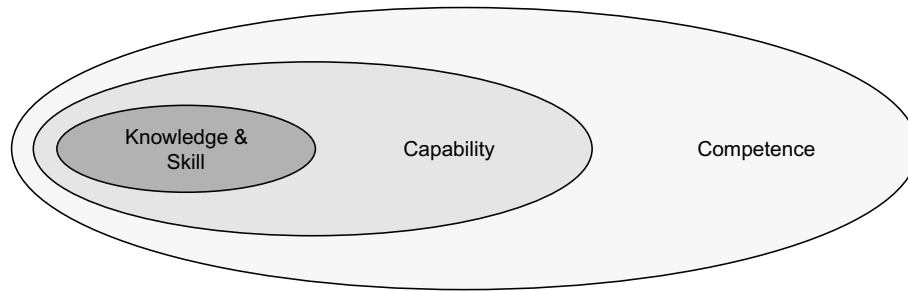


Figure 1: Relationships between knowledge, skills, capabilities and competence (based on [19])

The knowledge of an employee is core to the business. *Knowledge* is described as a stock of facts, theories, and rules available to individuals or groups, embedded in a social and contextual dimension. It is characterized by a high degree of certainty, so that its validity can be assumed [20]. Knowledge can be divided into implicit and explicit knowledge [21]. Explicit knowledge is consciously acquired and thus, it can be communicated verbally or graphically [22]. Implicit knowledge is expressed in simple terms in practical actions, without being able to explain the underlying principles [21]. It is formed in particular through the accumulation of experience in the course of work and can arise from both conscious learning or unconscious and unreflective absorption [20,22]. A *skill* is action-oriented knowledge acquired through practice or training [19]. STANLEY and WILLIAMSON [23] state that knowledge and skill are closely interlinked. Skills relate to specific tasks and require “knowing how” in order to perform these tasks [23]. They can be composed of several sub-skills, e.g. broken down into smallest units like movements according to the Methods-Time Measurement (MTM) method [24]. Skills can usually be learned more quickly than capabilities [19]. They can usually be tested or measured. Typical examples of practical skills are welding, screwing, or inspecting. *Capability* describes a meta-level. Capabilities (or abilities) are often equated with skills. However, this broader concept corresponds to natural talents or those developed through experience [25]. These talents enable employees to cope with unforeseen task contexts, and therefore, they can be learned comparatively slower than skills. Capabilities can include coordination, creativity, learning ability, etc. In addition, a distinction must be made between the visible and invisible nature of skills and capabilities: While constitutional traits like dexterity can be measured directly, mental traits like spatial thinking can only be recorded by means of predictors and thus provide a classification of how the employee performs in comparison to other people. The broadest concept of *competence* includes all previously mentioned categories. VITELLO et al. state in their Cambridge report that “competence is the ability to integrate and apply contextually-appropriate knowledge, skills and [...] [capabilities] to consistently perform successfully within a specified domain.” [26]. Competence can therefore be viewed as the overarching concept. As pictured in Figure 1, there are several inclusions between the aforementioned concepts. Competencies include skills, but cannot be reduced or limited to them. This also applies to knowledge and capabilities and vice versa.

2.2 Introduction to metadata models

A MDM describes the relationships between data elements and metadata artifacts at a conceptual and logical level [27]. Metadata represents structured information about entities such as data, processes or systems. The model itself consists mainly of four components [28]: (1) Resources or the data itself; (2) Attributes to describe the datasets further and give them specific features; (3) Relationships to describe how the resources interact with each other; (4) Categories to classify the different types of metadata. Metadata can be divided into three categories: descriptive, structural, and administrative. Descriptive metadata provides information about the content of a digital object, e.g., title, author, subject. Structural metadata documents how component pieces of complex information resources fit together. For example, it might describe the relationships between different entities. Typical are parent-child-relationships where properties are passed on and specialized. Administrative metadata is used to support the management of resources, including details about creation, file type, access conditions, rights management, and preservation [29].

To develop a MDM, the purpose of the model must be defined initially [30]. After that, the user group has to be specified. For easier, automated processing, the model implementation should use a standardized modeling language like Web Ontology Language (OWL).

3 Research methodology

In order to answer the RQ, a SLR was conducted. The SLR follows the PRISMA scheme from PAGE et al. [17]. The objective of the search is to identify existing models or parts that deal with the description of human capabilities. An emphasis hereby is placed on the production area, i.e., all processes of the ADR. The inclusion and exclusion criteria were the following:

1. The paper has no relation to human capabilities.
2. The paper has no production context.

Due to the RQ and the focus on human resources, the first exclusion criterion was stated. The second one was intentionally formulated to be more general, covering the entire production process. This meant that results for individual ADR processes, e.g., only assembly processes, were also taken into account. The further development of an MDM for a holistic description in the ADR is therefore the aim of this work.

The SLR was initiated on December 7, 2025, on which the exports of the widely recognized databases Scopus, Web of Science and IEEE Xplore were downloaded. The search period has been set until the end of 2024, with only papers published in English or German language and accessibility as open access. All search strings consist of a combination of the terms “capability”, and “model” or “metadata model”. The search was conducted based on the author keywords. Therefore, the following strings were used:

Table 1: Search strings within the databases

Database	Search string
Scopus	KEY(capability) AND (KEY(model) OR KEY(metadata model)) AND PUBYEAR < 2025 AND (LIMIT-TO (OA,"all")) AND (LIMIT-TO (LANGUAGE,"English") OR LIMIT-TO (LANGUAGE,"German"))
Web of Science	(AK=capability AND (AK=model OR AK=metadata model))
IEEE Xplore	("Author Keywords": capability) AND (("Author Keywords": model) OR ("Author Keywords": metadata model))

As can be seen from the strings provided in Table 1, the search was initially as broad as possible. There was no initial focus on the production context and all possible capability description models should be found. By applying the exclusion criteria, the results were gradually narrowed down to human capability description in production environments. In addition, the SLR within the searched databases was supplemented by a manual search to include sources that have introduced specific submodels of human capability description that were needed for the subsequent model development in Section 4. A screening was then carried out to pinpoint the pertinent articles from the entire collection. To maintain the study's rigor and objectivity, working papers, preprints, and other non-peer-reviewed publications were omitted from the analysis. A summary of the complete process is provided in Figure 2.

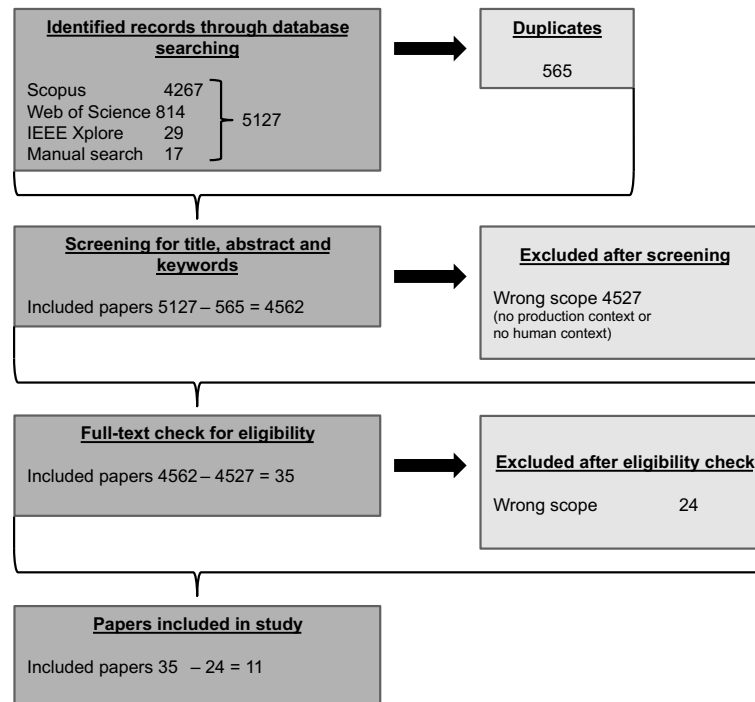


Figure 2: Flow chart of the SLR according to the PRISMA method (based on [17])

The search yielded a total of 5127 entries. Of these, 565 papers were disregarded as duplicates of existing entries. The remaining 4562 papers were meticulously evaluated based on their titles, abstracts, and keywords. By applying the previously defined exclusion criteria, 4527 papers were discarded as irrelevant. The subsequent full-text check for eligibility excluded another 24 papers due to incorrect scopes, resulting in a total of eleven included papers. Analysis of the identified papers revealed that both internal thought processes and externally observable characteristics are essential for describing human capabilities. In addition to internal processes, knowledge about the application context is also relevant, as explained in the following section.

4 Capability description model

The previous findings lead to the development of the MDM for human capabilities in production. With a concentration on the human component in accordance with the RQ, classes describing technical resources were omitted. In order to describe human capabilities at an operational level, the category of competence will be not included in the model. Necessary competencies are often used in a generalized manner in job descriptions, resulting in a lack of clarity regarding the capabilities required for ADR. The narrower concepts of knowledge, skill, and capability still apply. The analysis of the papers led the development and resulted in the formalization of the MDM. At the highest level, five aggregated classes are introduced. Each class will be explained in detail and stated in the respective figures below.

Knowledge of ADR processes and production are essential elements for a worker. Therefore, all concepts of knowledge are summarized in the class *domain knowledge*. In a production context, employees must be able to direct themselves toward completing a task and perform both physical and mental processes. Beyond that, each person has senses that enable them to perceive their environment and react accordingly. So, to describe a human worker fundamentally in his mental and physical attributes, the three classes *sensory capability*, *mental capability* and *physical capability* are introduced. Apart from normal activities carried out in everyday production, there are also special requirements for training or proof that the employee is capable of carrying out a specific ADR process. This leads to the introduction of the class *certification*. In order to describe human capabilities in the production environment holistically, subclasses were introduced for each class. The concept of classes and subclasses was originally derived from JÄRVENPÄÄ [31], who divided capabilities further into simple and combined ones. The aggregation of classes and subclasses for this MDM is based on production requirements. This means that specific skills are referred to as explicit subclasses in certain places, whereas (mostly in the mental area) entire capabilities are more often named. On top, the semantic structuring was introduced based on BAUERNHANS� et al. [32].

The following Figure 3 explains the further composition of the cognitive processes in production. First, domain knowledge contains specific knowledge about the sequence of *product/process*. The worker needs to know for example different product variants or how to (dis-)assemble a product. *Material knowledge* consists mainly of knowing how certain materials behave, as well as what makes them hazardous. To exemplify, a worker knows the probability of breaking between a plastic clip or a metal screw when disassembling a returned product. At least, the human needs *tool knowledge* about the deployed ones in ADR processes – things that are indispensable for achieving a good process quality.

The class certification has no specific subclasses. The justification for its existence is of a regulatory nature. Many companies use certifications to explicitly authorize workers to perform tasks that they would otherwise not be allowed to perform.

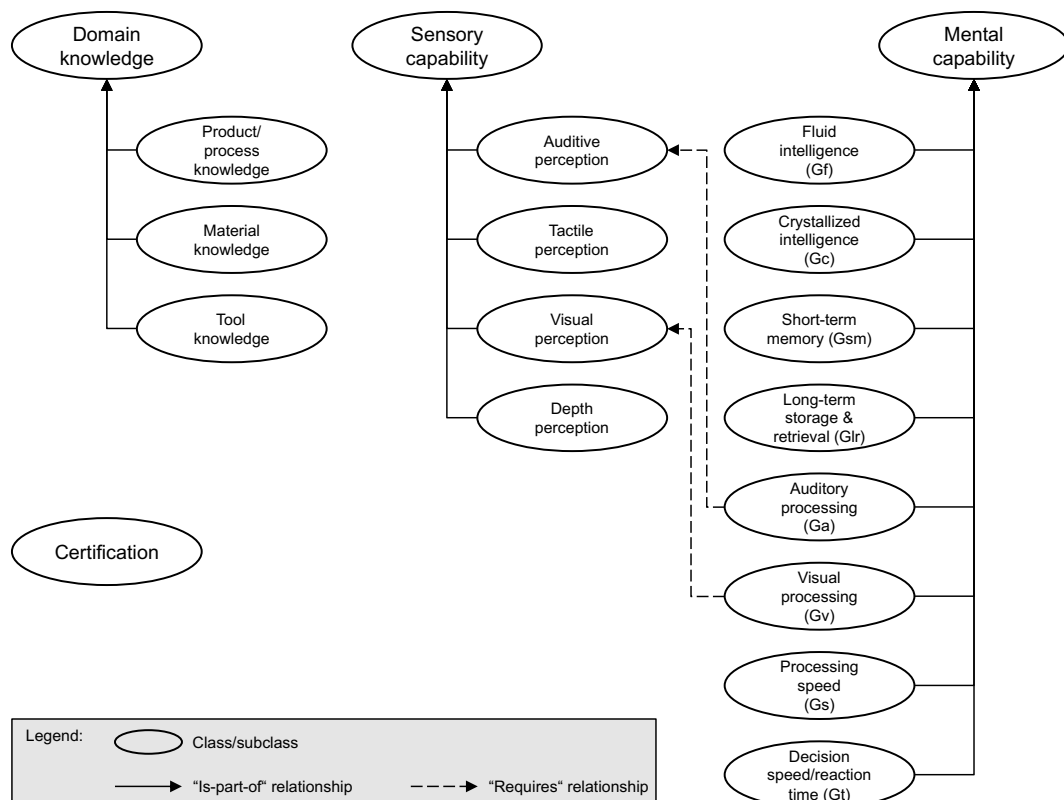


Figure 3: Classes domain knowledge, certification, sensory capability and mental capability

In terms of sensory capabilities, the senses that are regularly required in a production context were identified as particularly relevant for performing an ADR task. *Auditive*, *tactile* and *visual perception* are the most important subclasses here, supported by *depth perception* as an extension of vision [33]. Most ADR processes can already be covered by the first three sensory subclasses mentioned above. The addition of depth perception for spatial imagination is necessary because complex products or disassembly processes in particular often involve difficult or restricted access.

Regarding cognitive processing, a basic approach is required: Different types of mental processes are used to solve ADR tasks. For example, loosening a corroded joint requires a mental problem-solving process that must be remembered and then continuously checked and adjusted during execution. The extraction of mental capability subclasses was carried out based on the Cattell-Horn-Carroll (CHC) theory [34,35]. It is one of the leading psychological models for structuring cognitive capabilities, developed in the mid-1990s and constantly refined since then [36]. The core of the model is the hypothesis that human intelligence consists of different factors, called G-factors. At the bottom of the CHC theory are specific factors or rather capabilities, which are tied to a specific task or test [37]. Originally, there were two factors, Gf and Gc. The *fluid intelligence (Gf)* is used for inductive and deductive reasoning and enables employees to adapt to new environments and develop solving strategies for novel problems [34]. In a production environment, Gf is crucial for unexpected events, defects or new product variants during ADR processes. In contrast, *crystallized intelligence (Gc)* describes the worker's capability of problem solving with automatisms and established strategies. Areas such as general knowledge and vocabulary fall within the scope of Gc. For example, understanding (dis-)assembly instructions, technical terms, or apply learned procedures requires Gc. The CHC theory was then divided further, with two subclasses describing intelligence factors concerning the human memory. *Short-term memory (Gsm)* is used for temporarily storing and remembering information in working memory. For example, it is important for tracking process steps and managing parts. *Long-term storage & retrieval (Glr)* is the counterpart and can be used, for instance, to retrieve and compare defects with previous signs of wear on a product. The subclasses *auditory processing (Ga)* and *visual processing (Gv)* exist based on the worker's senses. Ga is the capability for analyzing and distinguishing auditory stimuli (excluding language); it is relevant for noise localization (e.g., error sources) and the recognition of patterns or deviations such as rattling noise from a broken component. Gv can be defined as "the ability to make use of simulated mental imagery (often in conjunction with currently perceived images)" [36]. In production environments, Gv first requires visual perception in order to generate, analyze, and mentally manipulate visual patterns. It is essential for spatial understanding of components and connections in products. The last two subclasses are more concerned with the time needed to take an action. The *processing speed (Gs)* can be described as the capability to perform simple, repetitive cognitive tasks quickly and fluently [36]. *Decision speed/reaction time (Gt)* refers, in comparison, to the worker's capability to react quickly to situations and making simple decisions. Gt capabilities reflect the immediacy with which a human can react to stimuli (e.g., number of seconds before moving a hand towards a dropping part), whereas Gs capabilities reflect the ability to work quickly over a longer period of time (e.g., loosening ten screw connections).

Alongside the mental aspects of human individuals, there are also their physical capabilities. As a rough preliminary classification for the production context, there are specific *body characteristics* that constitute a worker and different *movements* that can be executed with the body. The detailed classification can be seen below in Figure 4. To distinguish between the different characteristics and their movements, a partition of the body was made. The anatomy from the waist upward was assigned to the upper musculoskeletal system (UMS), for example, the torso and arms. Complementary, all areas below the waist were assigned the lower musculoskeletal system (LMS). Adapted from GRÄBLER et al. [38], different parameters can be assigned to each body characteristic and movement. The number of

parameters can be expanded as needed. Parameters were assigned according to the principle that unchangeable or quasi-static characteristics were attributed to body characteristics, while attributes equivalent to a momentary state were attributed to movements. For example, reach distance and arm length are unchangeable characteristics of a worker. When specific ADR tasks require a given arm span, then it is possible to match technical requirement profiles with the given worker's capability profiles.

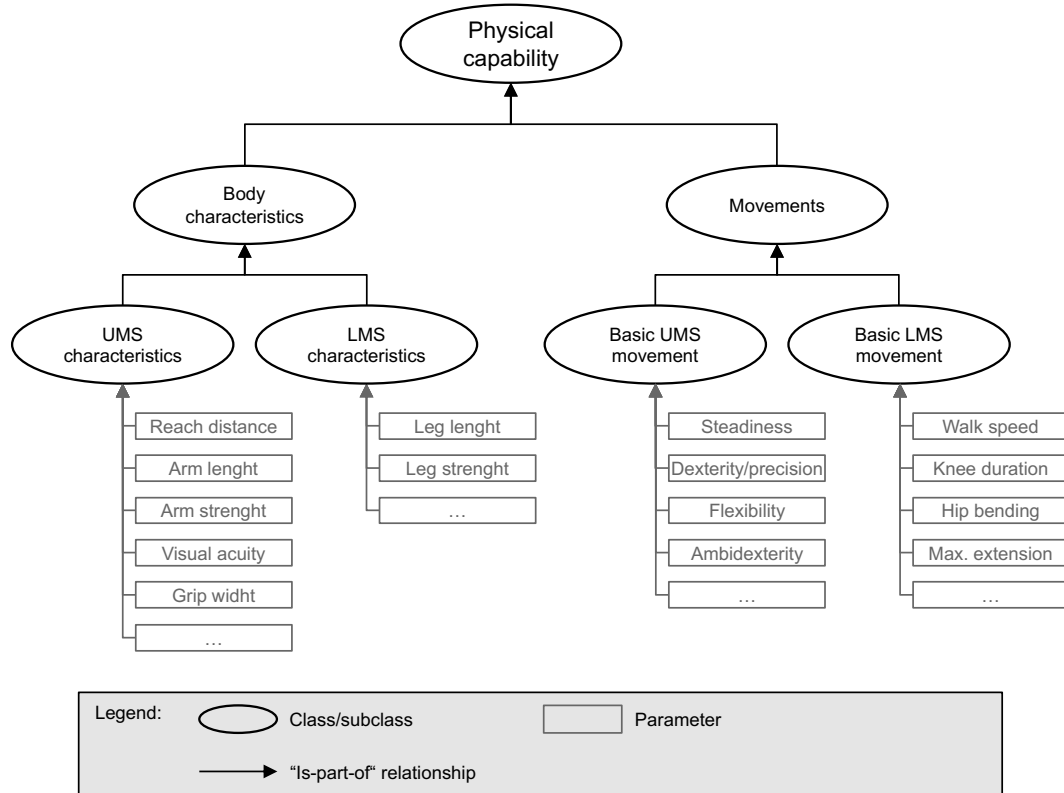


Figure 4: Class physical capability

5 Limitations

The scientific findings presented above are subject to certain limitations that should be taken into account when interpreting the results. An open search was enabled by the broad formulation of the search strings within the scope of the methodological approach selected, without excessive restriction of the results. However, this approach also necessitated greater effort during the review. Because a significant number of papers were outside the field of production, more than 4500 entries were already excluded during the initial screening. Notably, several seminal works were not identified through the SLR and had to be included manually. One reason for this could be that their application context is not limited to production and that they did not originate from a MDM. In addition, the search was limited to open accessible literature in the databases Scopus, Web of Science, and IEEE Xplore. Thus, papers that were not open access or listed in the aforementioned databases could not be screened. The search was also limited to keywords; hence, contributions that list human capabilities only in their title, abstract or full text were not found.

Besides that, the MDM is designed to be expandable if needed. Most classes describe human workers holistically in production environments. However, emotional capabilities such as empathy or conflict resolution capability were not considered. Hence, the focus of this MDM lies in the human-task interaction and not between several workers in production. In addition, human learning and forgetting effects were not part of this study. Employees who are initially unsuitable for certain tasks may later become qualified by undergoing targeted training.

6 Conclusion and outlook

For enabling circular production systems, a holistic understanding of the production process including disassembly and reassembly is important. The paper at hand addresses this critical research gap and offers an initial approach for describing human resources. Existing models often neglect the human influence on task completion and emphasize the technical aspect of production systems. A SLR based on the PRISMA scheme identified eleven relevant publications which formed the basis for the development of a production-centered MDM. The developed MDM formalizes five aggregated classes for the holistic description of human resources: domain knowledge, certifications, sensory abilities, mental abilities (based on CHC theory), and physical abilities (differentiated according to movements and physical characteristics of the upper and lower musculoskeletal system).

The significance of this model lies in its application as a foundation for data-driven decision-making systems that match human capabilities with specific production tasks. This is particularly essential for the circular economy, as repair, overhaul, and recycling processes require specialized human resources.

Future developments should expand the MDM to include technical skills and product specifications in order to create a complete knowledge graph for hybrid decision support systems. To model product requirements and the corresponding capabilities, it is necessary to establish a connection between these two elements. In addition, the MDM should be validated using actual data, and the interfaces to data systems must be specified. At the same time, learning effects, emotional capabilities and collaborative aspects could be taken into account to make the model operationalizable for more complex scenarios.

Acknowledgements

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – project no. 543074930 – as part of the SPP 2443 (Hybrid Decision Support in Product Creation, project no. 521148212, www.spp2443.de).

References

- [1] T. Hunger, M. G. Arnold, S. Engesser and K. van den Gerald Boogaart, “Integrating sustainability facets into the early stages of new product development using multi-criteria tools - a critical review.” *Journal of Cleaner Production*, vol. 517, p. 145834, 2025. DOI: <https://doi.org/10.1016/j.jclepro.2025.145834>.
- [2] O. Bernhard, D. Schirling and M. F. Zaeh, “Interdependencies within sustainable production systems.” *Production and Manufacturing Research*, vol. 13, no. 1, 2025. DOI: <https://doi.org/10.1080/21693277.2025.2544977>.
- [3] L. Kortus and T. Gutmann, “How do firms build dynamic capabilities to develop sustainable products? A multiple case study in the manufacturing industry.” *Journal of Cleaner Production*, vol. 415, p. 137887, 2023. DOI: <https://doi.org/10.1016/j.jclepro.2023.137887>.
- [4] World Economic Forum, “Circular Transformation of Industries. Unlocking Economic Value.”, 2025.
- [5] H. Rahnama, “Driving Sustainability in Manufacturing through Digital Transformation and Collaboration.”, Doctoral thesis, Luleå University of Technology, 2025.
- [6] L. Marrucci, T. Daddi and F. Iraldo, “The integration of circular economy with sustainable consumption and production tools: Systematic review and future research agenda.” *Journal of Cleaner Production*, vol. 240, p. 118268, 2019. DOI: <https://doi.org/10.1016/j.jclepro.2019.118268>.
- [7] J. Kirchherr, D. Reike and M. Hekkert, “Conceptualizing the circular economy: An analysis of 114 definitions.” *Resources, Conservation and Recycling*, vol. 127, pp. 221-232, 2017. DOI: <https://doi.org/10.1016/j.resconrec.2017.09.005>.
- [8] Circle Economy, “The Circularity Gap Report 2025.”, Circle Economy, 2025.
- [9] Eurostat, “Waste statistics.”, [Online]. Available: https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Waste_statistics. [Accessed 22 December 2025].
- [10] J. Potting, M. Hekkert, E. Worrell and A. Hanemaaijer, “Circular Economy: Measuring Innovation in the Product Chain. Policy Report.”, 2017.
- [11] I. Gräßler, J. Pottebaum, P. Nyhuis, M. Schmidt, B. Grewe, F. Vollenkemper, T. Hesse and T. Meinecke, “Evolving Design for Assembly, Disassembly and Reassembly into a new paradigm. Design-for-Capabilities with Hybrid Decision Support as an enabler for circular products.” in *Stuttgarter Symposium für Produktentwicklung SSP 2025*, K. Hölzle, M. Kreimeyer, D. Roth, T. Maier and O. Riedel, eds.: Stuttgart: Fraunhofer IAO, pp. 508–519.

- [12] R. Stark, "Virtual Product Creation (VPC) Explained." in *Virtual product creation in industry. The difficult transformation from IT enabler technology to core engineering competence*, R. Stark, ed.: Berlin, Heidelberg: Springer, pp. 43–56, 2022.
- [13] E. Järvenpää, M. Lanz and N. Siltala, "Formal Resource and Capability Models supporting Re-use of Manufacturing Resources." *Procedia Manufacturing*, vol. 19, pp. 87-94, 2018. DOI: <https://doi.org/10.1016/j.promfg.2018.01.013>.
- [14] M. Weser, J. Bock, S. Schmitt, A. Perzylo and K. Evers, "An Ontology-based Metamodel for Capability Descriptions." in *2020 IEEE 25th International Conference on Emerging Technologies and Factory Automation (ETFA)*: Piscataway, NJ: IEEE, 2020, pp. 1679–1686.
- [15] D. Roesmann and I. Gräßler, "Integration of Human Factors for Assembly Systems of the Future." in *The Digital Twin of Humans: An Interdisciplinary Concept of Digital Working Environments in Industry 4.0*, I. Gräßler, G. W. Maier, E. Steffen and D. Roesmann, eds.: Springer International Publishing, pp. 187–203, 2023.
- [16] S. Hall, M. Dhanda and V. Dhokia, "Towards an ontology to capture human attributes in human-robot collaboration." *Proceedings of the Design Society*, vol. 4, pp. 2585-2594, 2024. DOI: <https://doi.org/10.1017/pds.2024.261>.
- [17] M. J. Page, J. E. McKenzie, P. M. Bossuyt and I. Boutron, "The PRISMA 2020 statement: an updated guideline for reporting systematic reviews." *BMJ*, vol. 372, n71, 2021. DOI: <https://doi.org/10.1136/bmj.n71>.
- [18] F. Green, "What is Skill? An Inter-Disciplinary Synthesis.", Centre for Learning and Life Chances in Knowledge Economies and Societies, 2011.
- [19] J. Erpenbeck, L. Rosenstiel, S. Grote and W. Sauter, eds., "Handbuch Kompetenzmessung." Stuttgart: Schäffer-Poeschel, 2017.
- [20] D. Pritchard, "What is this thing called knowledge?", 3rd ed. London, New York: Routledge Taylor & Francis Group, 2013.
- [21] A. Schelten, "Begriffe und Konzepte der berufspädagogischen Fachsprache. Eine Auswahl." Stuttgart: Steiner, 2000.
- [22] S. Zelewski, ed., "Ontologiebasierte Kompetenzmanagementsysteme. Grundlagen, Konzepte, Anwendungen." Berlin: Logos-Verlag, 2005.
- [23] J. Stanley and T. Williamson, "Skill." *Noûs*, vol. 51, no. 4, pp. 713-726, 2017.
- [24] R. Bokranz and K. Landau, "Handbuch Industrial Engineering. Produktivitätsmanagement mit MTM.", 2nd ed. Stuttgart: Schäffer-Poeschel, 2012.
- [25] M. Shekarriz, A. Mousavi and P. Broomhead, "On the definition of capability in workplace a new perspective - Part 1." *Engineering Management*, 2013.
- [26] S. Vitello, J. Greateorex and S. Shaw, "What is competence? A shared interpretation of competence to support teaching, learning and assessment.", Cambridge University Press & Assessment, 2021.
- [27] International Organization for Standardization, Information technology - Metadata registries (MDR). Part 1: Framework. (ISO/IEC 11179-1:2023), 2023.
- [28] R. Eichler, C. Giebler, C. Gröger, H. Schwarz and B. Mitschang, "Modeling metadata in data lakes - A generic model." *Data and Knowledge Engineering*, vol. 136, p. 101931, 2021. DOI: <https://doi.org/10.1016/j.datak.2021.101931>.
- [29] H. Ulrich, A.-K. Kock-Schoppenhauer, N. Deppenwiese, R. Gött, J. Kern, M. Lablans, R. W. Majeed, M. R. Stöhr, J. Stausberg, J. Varghese, M. Dugas and J. Ingenerf, "Understanding the Nature of Metadata: Systematic Review." *Journal of Medical Internet Research*, vol. 24, no. 1, pp. 1-14, 2022. DOI: <https://doi.org/10.2196/25440>.
- [30] M. Baca, "Introduction to metadata. Third Edition.", 3rd ed. Los Angeles: The Getty Research Institute, 2016.
- [31] E. Järvenpää, "Capability-based adaptation of production systems in a changing environment.", Doctoral thesis, Tampere University of Technology Tampere, 2012.
- [32] T. Bauernhansl, M. Weyrich, L. Zarco, T. Muller, P. Marks, T. Schlegel and J. Siegert, "Semantic structuring of elements and capabilities in ultra-flexible factories." in *Procedia CIRP CMS*, Vol. 93: Elsevier B.V, 2020, pp. 335–340.
- [33] P. Hadjiphilippou, "The contribution of the five human senses towards the perception of space." Cyprus: University of Nicosia, 2013.
- [34] D. P. Flanagan and S. G. Dixon, "The Cattell-Horn-Carroll Theory of Cognitive Abilities." in *Encyclopedia of Special Education*, C. R. Reynolds, K. J. Vannest and E. Fletcher-Janzen, eds.: Bognor: wiley, pp. 1–13, 2013.
- [35] T. G. Baudson, "Der Aufbau der Intelligenz. Die CHC-Theorie als Strukturmodell kognitiver Fähigkeiten." *MinD-Magazin*, vol. 91, pp. 8-10, 2012.
- [36] W. J. Schneider and K. S. McGrew, "The Cattell-Horn-Carroll Model of Intelligence.", 2012.
- [37] W. J. Schneider and K. S. McGrew, "The Cattell-Horn-Carroll Theory of Cognitive Abilities." in *Contemporary intellectual assessment. Theories, tests, and issues*, D. P. Flanagan, E. M. McDonough and A. S. Kaufman, eds.: New York: Guilford Publications, pp. 73–163, 2018.
- [38] I. Gräßler, D. Roesmann and J. Pottebaum, "Model based integration of human characteristics in production systems: a literature survey." *Procedia CIRP*, vol. 99, pp. 57-62, 2021. DOI: <https://doi.org/10.1016/j.procir.2021.03.010>.

Hybrid Decision Support in Strategic Product Planning: Leveraging Extreme Data for Early-Phase Innovation

Iris Graessler¹, Julius Goldstein¹

¹Universität Paderborn, Lehrstuhl für Produktentstehung, 33102 Paderborn
Heinz Nixdorf Institut, 33102 Paderborn
E-Mail address: julius.hendrik.goldstein@hni.upb.de

Abstract: Strategic product planning defines foresighted directions for product engineering by determining which ideas, technologies and concepts are pursued further. Decisions made in this phase strongly influence subsequent life cycle phases but are characterized by high uncertainty, fragmented information, and predominantly qualitative assessment practices. At the same time, increasing digitalization generates large, heterogeneous, and dynamic data from markets, engineering, usage, and sustainability domains. In this work, such information is understood as extreme data, as its heterogeneity, variety, and dynamics exceed the capabilities of traditional planning approaches. This paper investigates the potential of leveraging extreme data to strengthen hybrid decision support in strategic product planning. A structured literature review is conducted to identify key challenges. These challenges are analysed along the generic product life cycle to capture decision points, information flows, and information circularity. Relevant data sources are identified with respect to their relevance for strategic decisions. Based on this analysis, seventeen potentials for hybrid decision support are derived and clustered into five fields of action. The results indicate that extreme data can enhance strategic foresight, idea evaluation and portfolio decisions, lifecycle-spanning knowledge integration, engineering support for strategic decisions, and the adoption of AI-based methods.

Keywords:

Strategic Product Planning, Hybrid Decision Support, Extreme Data

1 Introduction

Strategic product planning (SPP) is the first phase of product creation and the entire product life cycle and sets foresighted engineering directions by deciding which ideas, technologies, and product concepts will be transferred to engineering [1]. Decisions made in this phase shape product architectures, market positioning, and engineering paths, thereby influencing all downstream life cycle phases [1,2]. Despite this central role, SPP is characterized by uncertainties, volatile environmental conditions, and fragmented information from heterogeneous life cycle domains [3]. As a result, decisions often remain qualitative, experience-based, and only partially traceable [4].

SPP operates in the area between market and engineering. This area includes activities such as strategic foresight, trend and technology analysis, idea generation, idea evaluation, and portfolio processes [1]. These activities require the integration of a wide range of information, including customer needs, technological maturity levels, environmental performance data, competitive dynamics, supply chain conditions, and internal capabilities. The increasing digitalization along the product life cycle expands the available information base and at the same time increases the complexity of decision-making [2,5].

This complexity poses a fundamental challenge: the information relevant to SPP is not only extensive, but also heterogeneous, dynamic, and spread across the entire life cycle. These forms of information

represent extreme data [5]. This kind of data exceeds the capabilities of traditional planning methods, which commonly rely on static reports, expert judgments, or periodic analyses. Such data, even though it is characterised as extreme, therefore offers the opportunity to strengthen SPP's decision-making basis by supporting common methods with reliable data [3].

Hybrid decision support addresses this challenge by combining human expertise with AI-supported methods [5]. Existing approaches already demonstrate the potential benefits of AI in trend analysis, forecasting, scenario engineering, and generative design, but they remain narrowly defined and are not used holistically across tasks in SPP [5,6]. In addition, these approaches hardly exploit the potential of extreme data and do not sufficiently consider the information flows along the generic product life cycle (gPLC) [2]. Therefore, a systematic synthesis on how data can strengthen hybrid decision support in spite of extremities in SPP is missing, limiting the ability to structure decision-relevant information, integrate lifecycle perspectives, and transparently support SPP decisions.

To address this gap, this article builds on a structured literature review, an analysis of the SPP process, and the examination of relevant data sources. The contribution aims to structure and conceptualize how data-driven and hybrid approaches can support strategic decision-making in early product planning. The paper thereby provides a foundation for systematically identifying promising directions for future research and methodological engineering in hybrid decision support for SPP.

2 State of the Art

SPP constitutes the interface between market, corporate strategy and product engineering by translating long-term strategic goals into concrete engineering directions and product ideas. Established models of SPP describe a structured narrowing of an initially open problem space toward a feasible engineering order, supported by portfolio considerations, foresight activities, and early concept evaluations [1,7]. Recent contributions emphasize that SPP increasingly needs to account for interdisciplinary products, product-service systems, and sustainability objectives, which significantly extend the scope and complexity of early planning decisions [1,8]. Beyond its positioning in the product lifecycle, SPP is characterized by its strong influence on downstream phases such as engineering, realization, and operation, as early decisions determine major cost, performance, and sustainability trajectories. At the same time, decision-making in SPP remains exploratory and knowledge-intensive, relying on expert judgment, workshops, and qualitative assessments rather than formalized data-driven methods [1].

A central challenge of SPP is coping with high uncertainty and fragmented information. Relevant inputs originate from diverse domains such as market engineering, technology trends, lifecycle considerations, and organizational constraints, which are rarely available in a consistent or structured form [7]. While foresight methods such as Scenario-Technique are widely used to support SPP, their application is often resource-intensive and dependent on expert knowledge, limiting their scalability and reproducibility [9]. Furthermore, existing planning approaches typically focus on isolated subprocesses, such as market analysis or technology assessment, without systematically integrating information across the entire product lifecycle. As a result, feedback from later phases such as usage, operation, or end-of-life is only partially considered in early planning decisions, despite its relevance for sustainability and circularity-oriented strategies [7]. This lack of information circularity reinforces subjectivity and hampers transparency in strategic decision-making.

In parallel to these challenges, ongoing digitalization leads to the generation of large, heterogeneous, and dynamic datasets across the product lifecycle. Along the gPLC[1], these include market data, engineering data, lifecycle assessment results, sensor data from usage phases, and sustainability-related indicators. Thus, comprehensive data models and the integration of processes form the basis for holistic

insights into SPP [10,11]. In the context of SPP, such data can be understood as extreme data, as their volume, variety, and dynamics exceed the processing capabilities of traditional planning methods and manual expert-based evaluation [12]. Extreme data in SPP are not specifically extreme due to volume, but because of distributed origin and access, semantic heterogeneity, and temporal dynamics. Relevant information is scattered across different lifecycle phases and organizational units, making it difficult to identify which data is decision-relevant and how they should be interpreted in early planning contexts [7]. While digital twins and lifecycle data infrastructures promise improved information availability, their systematic use in SPP remains limited [1].

To address increasing complexity, hybrid decision support approaches combining human expertise with AI methods are gaining attention in research and practice. These approaches aim to leverage computational capabilities for data processing and pattern recognition while preserving human judgment, creativity, and contextual understanding [13]. Initial conceptual work demonstrates the suitability of hybrid approaches for selected planning tasks, particularly in strategic foresight and scenario analysis [9]. AI-based methods have been applied to automate consistency checks, cluster scenarios, or support trend analysis, thereby improving efficiency and transparency [9]. However, existing approaches predominantly address isolated tasks and remain domain-specific, focusing on foresight or evaluation activities rather than the entirety of SPP. Moreover, current hybrid decision support systems rather exclude extreme data instead of tackling extremities as a central data processing element. While large datasets are implicitly processed, little attention is paid to systematically identifying decision-relevant extreme data sources across the lifecycle and integrating them into SPP processes [12]. This reveals a research gap at the intersection of SPP, hybrid decision support, and extreme data utilization.

3 Method / Approach

Overall, existing approaches to decision support in SPP can be categorized into foresight-oriented methods, portfolio evaluation approaches, and lifecycle-related data models. However, these approaches predominantly address isolated sub-processes and do not provide a systematic integration of extreme data across the entire product lifecycle. This raises the following research question: *What potential for offering hybrid decision support in SPP is currently locked in data due its extreme characteristics?* To answer this question, this research follows a four-step conceptual approach, as shown in Figure 1.

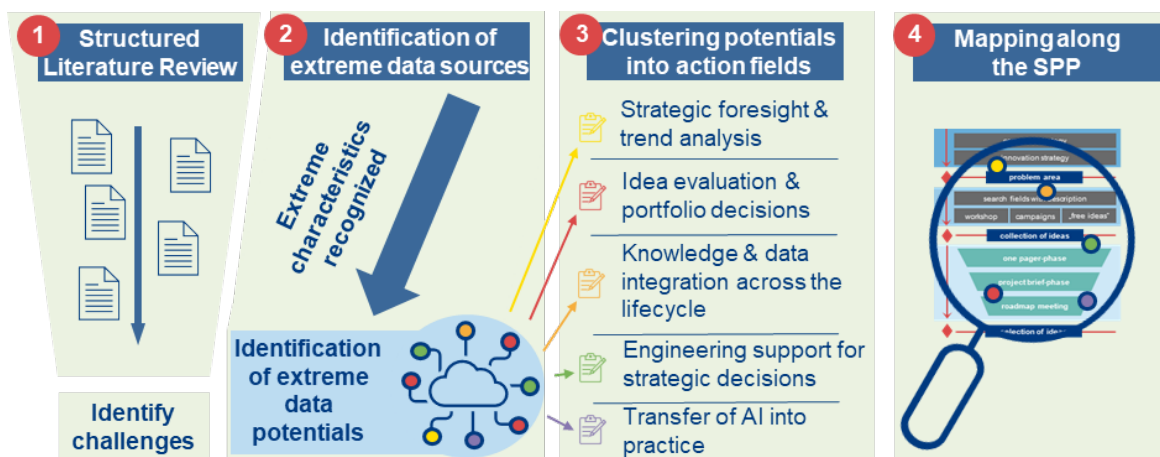


Figure 1: Methodological approach

The process starts with a structured literature review to identify challenges and decision-critical gaps in SPP. The underlying literature analysis is guided by the framework proposed by Machi and McEvoy,

which is particularly suited for theory-oriented and concept-building research. Based on these findings, relevant data sources are identified and analysed with respect to their decision relevance along the SPP process. The recognized data characteristics serve as the basis for deriving potentials for hybrid decision support. Subsequently, the identified potentials are clustered into five fields of action reflecting similarities in decision logic, data characteristics, and the role of AI-supported analytical methods. Finally, the action fields and associated potentials are mapped along the phases and activities of SPP to illustrate where hybrid decision support can systematically contribute to strategic decision-making.

4 Results

This section presents the results of the literature-based analysis conducted in this study. The identified findings are structured along recurring challenges in SPP and their implications for the use of extreme data and hybrid decision support.

4.1 Challenges in SPP

To identify the key challenges in dealing with extreme data in SPP, a structured literature analysis was conducted following the conceptual synthesis approach by Machi and McEvoy. The aim was not to exhaustively catalogue all available publications, but to systematically identify, compare, and synthesize recurring challenges and argumentative patterns relevant to early-stage decision-making and decision support in SPP. To ensure transparency and reproducibility, the literature search followed a systematic and traceable procedure. Scientific databases including Scopus, Web of Science, and IEEE Xplore were searched using predefined combinations of keywords related to SPP, decision-making, extreme data, and AI- or model-based decision support. Titles, abstracts, and full texts were screened iteratively, allowing the refinement of search terms and thematic focus in line with emerging concepts, as suggested by Machi and McEvoy. Only publications that provided substantive contributions to at least one of the following aspects were included: (1) decision-making processes in SPP (2) data-related challenges in early engineering phases, and (3) AI- or model-based decision support processes. The evaluation reveals four recurring, cross-domain challenges that currently limit the systematic use of extreme data in SPP:

(1) Uncertainty and incomplete information in early strategic decisions

Several studies emphasize that strategic decisions in the early phase of the product life cycle are made under particularly high uncertainty, as reliable information on markets, technologies, costs, or usage scenarios is only available to a limited extent. At the same time, these decisions shape long-term engineering and product paths [14,15]. Although extreme data from market analyses, technology roadmaps, and sustainability assessments are generally available, they have so far been integrated into search fields, scenarios, and early evaluation steps only to a limited extent [3,16].

(2) Fragmented data landscapes and lack of information circularity

The literature shows a pronounced fragmentation of the available data across the entire life cycle. Information from engineering, simulation, production, operation, service, recycling, and the market environment is stored in separate systems, differs in structure, quality, and timeliness, and is rarely synchronized [2,5,17]. Due to its extreme characteristics, such as volume, heterogeneity, and dynamics, this data is difficult to consolidate into a consistent flow of information for SPP [5,17]. As a result, key relationships between market requirements, engineering, life cycle assessment, and circular feedback loops are insufficiently visible.

(3) Qualitative and opaque idea and portfolio decisions

In practice, idea and portfolio decisions remain largely qualitative, for example through expert panels, scoring models, or stage-gate processes [14,18]. Although these methods make use of expert experiences and enable structured discussions, they make only limited use of existing data and are susceptible to subjective bias. This leads to reduced transparency, limited traceability, and heavy reliance on implicit expert knowledge [7,16].

(4) Domain-specific AI applications without an overarching framework

The literature shows a variety of methodological approaches, such as generative design, graph-based concept evaluation, AI-supported innovation metrics, or automated trend analyses, which are promising for specific SPP issues [1,19]. Yet, these approaches support only such specific tasks, without a broader scope in SPP. However, these approaches remain largely domain-specific and method-driven, typically addressing isolated subtasks of SPP rather than being anchored across the entire SPP process. As a result, they provide only limited support for the systematic integration of heterogeneous and dynamic data sources and rarely consider extreme data as a central design element. Consequently, an integrative framework that combines extreme data, AI methods, and human expertise into a hybrid decision support system spanning the entire SPP process does not yet exist.

Tabelle 1: Data domains and sources in SPP and their characteristics of extremeness

Data Domain	Exemplary Data Sources	Relevance for SPP	Characteristics of Extremeness
Market Data	Customer feedback (reviews, social media), market studies, sales forecasts, competitor analyses, technology and trend reports [20]	Identification of search fields, market opportunities, and strategic risks	High heterogeneity of unstructured data, dynamic evolution of market signals, varying data quality
Engineering Data	CAD models, simulation results, variant and modular structures, textual requirements, design constraints, Life Cycle Assessment (LCA) results [21]	Assessment of technical feasibility, complexity, and architectural implications	Strong semantic complexity, implicit design constraints and interdependencies, limited interpretability at a strategic level
Product & Realisation Data	Lifecycle-spanning sustainability, usage, and supply-network data, including continuously updated lifecycle assessment data streams, material and energy flow data across supplier interfaces, recyclability indicators[16]	Evaluation of long-term sustainability impacts and regulatory compliance	Lifecycle-spanning and cross-organizational data streams, continuous updates driven by external boundary conditions, high dependency on supplier interfaces
Use & Operation Data	Sensor data from product usage, maintenance and service records, failure statistics, usage profiles, customer feedback [22]	Feedback of real-world usage behaviour into early planning decisions	Large volumes of time-dependent sensor and service data, heterogeneous formats, context-dependent interpretation
Organizational & Process Data	Project histories, portfolio decisions, resource allocations, engineering timelines [15]	Transparency and traceability of strategic planning and decision processes	Incomplete documentation of decisions and lessons learned, fragmented information across organizational units

In the second step, the analysis shifts from challenges in SPP to the data perspective required to address them. To operationalize the notion of extreme data in SPP, Table 1 summarizes exemplary data domains and sources across market, engineering, product lifecycle, and use phases, and describes the characteristics of extremeness. Rather than reflecting data volume alone, the degree of extremeness captures the specific challenges these data pose for SPP, particularly with respect to characteristics like heterogeneous formats, high volume, unknown quality, errors, gaps, or different languages in early planning stages [7]. Market, Engineering, lifecycle, and use-phase data are therefore classified as highly extreme, as they evolve dynamically and are often only partially accessible during SPP. Organizational

data, while typically available at earlier stages, are comparatively more structured and are thus rated as moderately extreme. This classification provides a systematic basis for identifying where extreme data becomes decision-relevant in SPP and for deriving subsequent potentials for hybrid decision support.

4.2 Identified potential of hybrid decision support in SPP

The analysis identified 17 different areas of potential for hybrid decision support in SPP. These include, for example, cluster algorithms for structuring large quantities of idea inputs [16] and hybrid dashboards for providing transparent evaluation metrics [14]. All identified potentials were consolidated into five fields of action that structure their relevance along the planning process (Figure 1).

Strategic foresight & trend analysis	Idea evaluation & portfolio decisions	Knowledge & Data integration across the lifecycle	Engineering support for strategic decision	Transfer of AI into practise
- AI-supported strategic foresight [6] - Scenario enrichment using extreme data [16] - Patent text mining for early technology intelligence [5] - AI-based patent and technology analysis [18] - Weak signals & trend mining [12]	- NLP-supported analysis of requirements and idea texts [19] - AI-based clustering and quality assessment of requirements and ideas [15] - AI-supported portfolio decision-making [8] - Feedback of usage and monitoring data into early-stage planning [7]	- Lifecycle data integration (engineering / service / PLM) [1] - AI-based decision support systems [17] - Machine learning-based patent intelligence [5] - Lifecycle data integration for intelligent planning processes [16]	- Generative design for solution space expansion [19] - Digital twins for decision support [22] - Ontology- and knowledge graph-based decision support [19]	- Circularity and sustainability assessment using extreme Data [16]

Figure 2: Clustering potentials into fields of action

Strategic foresight & trend analysis in SPP benefits greatly from AI methods that can analyse large, heterogeneous, and dynamic data sets. Extreme data, including market information, usage data, scientific publications, patents, social media signals, and technological maturity indicators, enables more precise identification of trends and search fields. AI-supported trend analyses reveal potential by performing pattern recognition on unstructured data and making engineering visible at an early stage [17]. In addition, the Scenario-Technique can be systematically expanded through the integration of data-based evidence, with AI-supported analysis methods identifying weak signals and structured trend engineering, thereby reducing uncertainties in strategic orientation [9]. Similarly, patent- and technology-related text mining approaches enable the efficient derivation of technology trends and innovation potential from global databases. These potentials strengthen strategic advance planning by systematically using extreme data sources to derive robust search fields and long-term technology paths.

The **evaluation of ideas** and concepts in SPP has traditionally been highly subjective. Extreme data and AI methods offer potential to make evaluation processes more objective, comprehensible, and evidence-based. Natural language processing (NLP) enables the automated analysis of requirements and idea texts and supports the identification of semantic similarities, conflicts, or quality characteristics [2]. In addition, AI-based clustering allows large quantities of ideas or requirements to be organized in a more structured manner and provides decision-makers with transparent groupings, which is particularly helpful in early screening processes [19]. Explainable AI approaches increase the traceability of data-based portfolio decisions and enable sensitivity analyses that clearly reveal the influence of specific criteria [18]. Similarly, usage and monitoring data from operations or services can be fed back into early evaluation via extreme data pipelines, enabling a more realistic assessment of market potential, risks,

and life cycle impacts [23]. This opens up substantial potential for evidence-based prioritization of strategic projects.

A key potential of hybrid decision support exists in the **integration of data** across the entire product life cycle. Extreme data is generated in engineering, production, service, operation, return, and recycling but has so far hardly been used for strategic decisions. The concept of information circularity within the gPLC shows that available data streams contain valuable indicators about product use, failure behavior, maintenance costs, and sustainability impacts [16]. The Information Circularity Assistance Framework demonstrates how AI methods, digital twins, and semantic information models can be linked to support strategic decisions with cross-lifecycle information. Another potential arises from the use of knowledge graphs to link heterogeneous data sources, revealing relationships between requirements, technologies, components, and usage scenarios [6]. Machine learning-based patent analyses also enable systematic early technology reconnaissance and can enrich strategic roadmaps with data-based evidence [2]. These potentials strengthen the quality of decision-making by making extreme data sets structurally accessible and strategically usable.

Engineering methods such as simulation, digital twins, or early transfer of AI approaches like ontology-based knowledge models open up new potential for SPP, especially when combined with extreme data. Generative design processes use AI to generate alternative concept solutions and significantly expand the solution space in the early stages. In this context, digital twins provide a data-rich representation of physical products and systems, which can be coupled with simulation models to systematically support strategic decisions on sustainability, performance, and costs [17]. Ontology- and knowledge graph-based decision support systems enable the linking of requirements, functions, technologies, and usage scenarios in consistent information structures, thereby transparently displaying strategic dependencies. Machine learning methods extend this by recognizing patterns in engineering data and automating or improving strategic analyses, such as concept evaluation. Overall, these potentials enable stronger feedback from engineering into SPP and thus a more fact-based derivation of future product architectures.

In order of hybrid decision support and extreme data having a **practical impact in SPP**, organizational, methodological, and cultural prerequisites must be taken into account. Studies show that companies often have difficulty with anchoring data-driven methods in decision-making processes, even though their benefits are proven [17]. Therefore, approaches such as explainable AI methods that provide decision-makers with transparency and confidence are essential for the effective adoption of hybrid decision support in SPP [13]. Further potential arises from circularity and sustainability assessments that use extreme data from operations and services to inform strategic decisions in line with ecological and regulatory requirements [1]. In addition, organization-wide concepts for collaboration between data science, engineering, and management can promote the successful use of hybrid systems. The potential of this field of action thus addresses the organizational transformation necessary to establish databased decision support in a sustainable manner.

Figure 3 relates the previously derived fields of action and their associated potentials to the SPP process. Building on the clustering of potentials presented in the preceding section, the figure illustrates how these action fields manifest across the phases and core activities of SPP. Based on the gPLC, lifecycle-related information flows are transferred to the SPP process, making decision-relevant interfaces and feedback loops visible. The analysis focuses on the early phase of SPP, represented by the first three control points shown in Figure 3. These control points encompass the problem framing, idea generation, and idea selection stages, which are characterized by high uncertainty and strategic relevance. [1]

The individual potentials, shown as discrete markers, indicate concrete points within the SPP process where extreme data and AI-supported methods can contribute to hybrid decision support. Their

positioning highlights that potentials identified within the same field of action occur at multiple stages of SPP, ranging from early strategic problem framing to idea evaluation and selection decisions. This emphasizes that the areas of action are not tied to individual process steps but are spread across several steps. The visual aggregation of potentials along the process reveals cross-cutting patterns: some fields of action predominantly support early strategic reasoning, while others connect early planning decisions with downstream lifecycle and engineering considerations. Rather than introducing new clusters, the figure demonstrates how the previously identified fields of action interact with existing SPP structures and where multiple fields converge at key decision points. Overall, Figure 3 emphasizes the process-wide relevance of the action fields and illustrates how hybrid decision support enriched by extreme data can be systematically embedded into early phase SPP processes. The figure thereby supports the identification of priority intervention points without duplicating the content-based clustering of potentials.

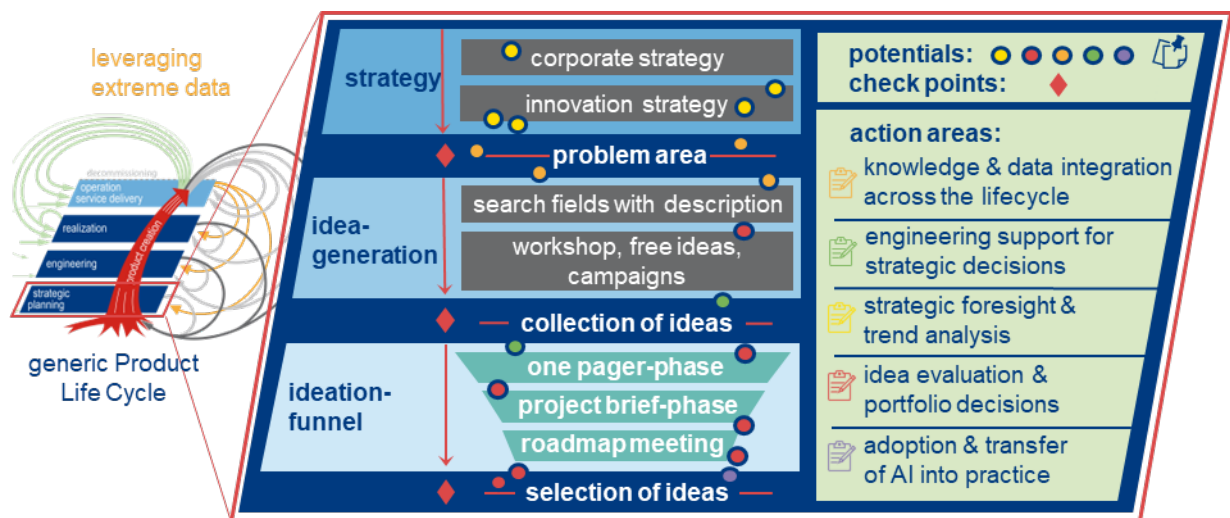


Figure 3: Areas of action for hybrid decision support in strategic product planning

4.3 Application and Discussion

This paper addressed the question of what potential the use of extreme data offers for hybrid decision support in SPP. The results indicate that extreme data can systematically strengthen hybrid decision support in early planning phases by improving strategic foresight, increasing transparency in idea and portfolio decisions, enabling lifecycle-spanning knowledge integration, and supporting the consideration of engineering-related implications at a strategic level. These potentials are consolidated into five fields of action, which together provide a structured framework for integrating data-driven insights and human expertise in SPP decision-making. Across the identified action fields, a central insight is that extreme data primarily shifts the nature of strategic reasoning rather than replacing established practices. In strategic foresight and trend analysis, for example, data-driven approaches enhance weak signal detection and scenario robustness, yet they remain dependent on expert interpretation to ensure plausibility and relevance. Similarly, in idea evaluation and portfolio decisions, AI-based analyses increase transparency and consistency but do not eliminate normative judgments related to strategic priorities, risk tolerance, or organizational constraints and decision contexts.

The analysis further reveals that knowledge and data integration across the lifecycle represents both a major opportunity and a critical limitation. While linking heterogeneous data sources enables improved traceability and continuity of strategic decisions, it also introduces significant methodological and organizational complexity. Engineering support for strategic decisions demonstrates how generative design and digital twins can broaden solution spaces beyond human cognitive limits; however, these approaches rely on structured input data and validated modelling assumptions, which are often

unavailable in early strategic phases. Adoption and transfer considerations highlight that hybrid decision support is not solely a technical challenge. Organizational readiness, user acceptance, and methodological transparency emerge as decisive factors for realizing the identified potentials. Without addressing these aspects, even technically advanced approaches risk remaining isolated applications rather than becoming integral elements of SPP.

The proposed framework has been exemplarily applied in context of a German manufacturer of kitchen appliances during early-phase SPP. In conducted workshops, strategic search fields and product ideas for future appliance generations were assessed under conditions of high uncertainty, by considering market requirements, technological developments, and sustainability objectives. To support these decisions, heterogeneous data sources were taken into account, including customer feedback from service and usage data, engineering related information on existing product architectures, and lifecycle-related data on energy consumption and regulatory boundary conditions. Structuring the identified data according to the identified fields of action has supported transparent discussions of trade-offs between functionality, energy efficiency, and product strategy, and has laid the foundation for further hybrid decision support. This example illustrates how extreme data can be used to complement expert judgment in early SPP without implying empirical validation of the framework's effectiveness. Overall, the analysis shows that extreme data does not constitute a standalone solution. Instead, value is gained by supporting hybrid decision-making by complementing human expertise with data-driven insights. At the same time, the analysis confirms that existing research remains fragmented and largely domain specific. This lack of methodological continuity across the entire SPP process underscores the need for integrative frameworks that systematically combine extreme data, AI methods, and human judgment.

5 Conclusion and Outlook

This research examines how data even with extreme characteristics can contribute to hybrid decision support in SPP. By combining a structured literature review with an analysis along the gPLC, seventeen distinct potentials were identified and consolidated into five action fields covering strategic foresight, idea evaluation and portfolio decisions, lifecycle-wide knowledge integration, engineering support for strategic decisions, and the transfer of AI-based methods. Together, these results provide a structured overview of where and how extreme data can support decision-making in SPP. The findings indicate that the value of extreme data does not lie in data availability alone, but in its systematic integration into hybrid decision support approaches. Extending inputs by data that is extreme deepens the informational basis of strategic reasoning by connecting heterogeneous and dynamic data sources across lifecycle phases. When combined with human expertise, this enables more transparent, traceable, and evidence-informed decisions in SPP, while preserving the exploratory and interpretative character of early SPP. The results further show that current approaches remain fragmented and predominantly focused on isolated tasks. The identified action fields therefore represent starting points for structuring future research rather than ready-to-use solutions. Advancing these fields requires the engineering of methodological frameworks that explicitly address information circularity, decision-relevant data selection, and the interaction between analytical models and human judgment. Future research will focus on operationalizing the identified potentials into concrete methods and tools for SPP. This includes developing integrated data pipelines across lifecycle phases, designing hybrid decision support concepts that are robust under uncertainty, and empirically evaluating their impact on planning quality and decision transparency in SPP contexts. By addressing these challenges, extreme data enriched hybrid decision support can contribute to a more systematic and resilient discipline of SPP.

6 Acknowledgements

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – project no. 543074930 – as part of the SPP 2443 (Hybrid Decision Support in Product Creation, project no. 543190685, www.spp2443.de).

References

- [1] Iris Gräßler, Sven Rarbach, Benedikt Grewe, “Modell der Strategischen Produktplanung – Digitale Zwillinge für kreislauffähige Produkte und Produktionsprozesse,” *Industry 4.0 Science*, vol. 2025, no. 3, 2025. DOI: <https://doi.org/10.30844/I4SD.25.3.24>.
- [2] A. Liu, S. Lu, F. Tao and N. Anwer, “Integration of data science with product design towards data-driven design,” *CIRP Annals*, vol. 73, no. 2, pp. 509-532, 2024. DOI: <https://doi.org/10.1016/j.cirp.2024.06.003>.
- [3] M. M. Mariani, I. Machado, V. Magrelli and Y. K. Dwivedi, “Artificial intelligence in innovation research: A systematic review, conceptual framework, and future research directions,” *Technovation*, vol. 122, p. 102623, 2023.
- [4] G. Giuggioli and M. M. Pellegrini, “Artificial intelligence as an enabler for entrepreneurs: a systematic literature review and an agenda for future research,” *International Journal of Entrepreneurial Behavior & Research*, vol. 29, no. 4, pp. 816-837, 2023. DOI: <https://doi.org/10.1108/IJEBR-05-2021-0426>.
- [5] H. Quan, S. Li, C. Zeng, H. Wei and J. Hu, “Big Data and AI-Driven Product Design: A Survey,” *Applied Sciences*, vol. 13, no. 16, p. 9433, 2023. DOI: <https://doi.org/10.3390/app13169433>.
- [6] F. Wang, “Microsoft Word - 20250130 - Wang - 2025 - Collaborative Foresight in the Age of AI.docx,”
- [7] Iris Gräßler, Jens Pottebaum, Peter Nyhuis, Rainer Stark, Klaus-Dieter Thoben, Petra Wiederkehr, “Hybrid Decision Support in Product Creation - Improving performance with data science and artificial intelligence,” *Industry 4.0 Science*, vol. 2025, no. 1, 2025. DOI: <https://doi.org/10.30844/I4SE.25.1.18>.
- [8] O. Sankowski, J. Küchenhof, F. M. Dambietz, M. Züfle, A. Wallisch, D. Krause and K. Paetzold, “Challenges in early phase of product family development processes,” *Procedia CIRP*, vol. 100, pp. 840-845, 2021. DOI: <https://doi.org/10.1016/j.procir.2021.05.034>.
- [9] Iris Graessler, Alena Marie Tusek, Henrik Thiele, Daniel Preuss, Benedikt Grewe, Michael Hieb, ed., *Literature study on the potential of Artificial Intelligence in Scenario-Technique* (Worsley, Manchester, United Kingdom: International Society for Professional Innovation Management Ltd, 2022).
- [10] I. Graessler, J. Pottebaum and P. Scholle, “Integrated Process and Data Model for Agile Strategic Planning,” 11th International Workshop on Integrated Design Engineering IDE Workshop, 2017.
- [11] I. Graessler, P. Scholle and J. Pottebaum “Integrated process and data model for applying scenario-technique in requirements engineering,” *Proceedings of the 21st International Conference on Engineering Design ICED 17*, 2017.
- [12] C. Vocke, C. Constantinescu and D. Popescu, “Application potentials of artificial intelligence for the design of innovation processes,” *Procedia CIRP*, vol. 84, pp. 810-813, 2019. DOI: <https://doi.org/10.1016/j.procir.2019.04.230>.
- [13] Rujonel F. Cariaga, “Human-AI Collaboration in Decision-Making, Creativity, and Productivity: Systematic Review,” vol. 2025. DOI: <https://doi.org/10.2139/ssrn.5378095>.
- [14] M. Falahat, S. C. Chong and C. Liew, “Navigating new product development: Uncovering factors and overcoming challenges for success,” *Heliyon*, vol. 10, no. 1, e23763, 2024. DOI: <https://doi.org/10.1016/j.heliyon.2023.e23763>.
- [15] T. Marion and S. Fixson, “Advancing in-process innovation metrics: a framework for navigating uncertainty,” *Proceedings of the Design Society*, vol. 5, pp. 339-348, 2025. DOI: <https://doi.org/10.1017/pds.2025.10048>.
- [16] I. Graessler, M. Weyrich, J. Pottebaum and S. Kamm, “Information Circularity Assistance based on extreme data,” at - *Automatisierungstechnik*, vol. 73, no. 1, pp. 3-21, 2025. DOI: <https://doi.org/10.1515/auto-2024-0039>.
- [17] N. Haefner, J. Wincent, V. Parida and O. Gassmann, “Artificial intelligence and innovation management: A review, framework, and research agenda☆,” *Technological Forecasting and Social Change*, vol. 162, p. 120392, 2021. DOI: <https://doi.org/10.1016/j.techfore.2023.122402>.
- [18] A. Witkowski and A. Wodecki, “Where does AI play a major role in the new product development and product management process?,” *Management Review Quarterly*, 2025. DOI: <https://doi.org/10.1007/s11301-025-00533-5>.
- [19] T. Blüher and R. Stark, “Systemic Evaluation of PSS in the Early Concept Phase Using Graph-Based Reasoning,” *Applied Sciences*, vol. 14, no. 23, p. 11241, 2024. DOI: <https://doi.org/10.3390/app142311241>.
- [20] C. Brewis, S. Dibb and M. Meadows, “Leveraging big data for strategic marketing: A dynamic capabilities model for incumbent firms,” *Technological Forecasting and Social Change*, vol. 190, p. 122402, 2023. DOI: <https://doi.org/10.1016/j.techfore.2023.122402>.
- [21] J. Krüger, L. Wang, A. Verl, T. Bauernhansl, E. Carpanzano, S. Makris, J. Fleischer, G. Reinhart, J. Franke and S. Pellegrinelli, “Innovative control of assembly systems and lines,” *CIRP Annals*, vol. 66, no. 2, pp. 707-730, 2017.
- [22] F. Tao, H. Zhang, A. Liu and A. Y. C. Nee, “Digital Twin in Industry: State-of-the-Art,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 2405-2415, 2019. DOI: <https://doi.org/10.1109/TII.2018.2873186>.
- [23] A.-S. Tombeil, D. Kremer, J. Neuhüttler, C. Dukino and W. Ganz, “Potenziale von Künstlicher Intelligenz in der Dienstleistungsarbeit,” in *Automatisierung und Personalisierung von Dienstleistungen*, M. Bruhn and K. Hadwich, eds.: Wiesbaden: Springer Fachmedien Wiesbaden, 2020, pp. 135–154.

A Product Meta Data Model of Assembly-Related Extreme Data to enable Hybrid Decision Support

Iris Graessler¹, Felix Vollenkemper^{1*}, Jens Pottebaum¹

¹Paderborn University, Heinz Nixdorf Institute, Chair for Product Creation, 33102 Paderborn

*Corresponding author. E-Mail address: felix.vollenkemper@hni.upb.de; Tel.: 05251 606237

Abstract: Models of Circular Economy introduce circularity strategies like reuse, repair or remanufacture. A significant level of complexity is added when these strategies are applied not only on integrated product level, but on lower levels of assemblies and parts. Therefore, approaches, like Design for Assembly, Disassembly and Reassembly (DfADR) become increasingly important including the consideration of ADR capabilities. To put DfADR into practice, it is necessary to structure ADR-related product data and make it usable for product engineers. For instance, assemblies resulting from disassembly might not be consistent with original assembly structures. Therefore, a Product Meta Data Model (MDM) is required to handle assembly-related data even when it has extreme characteristics. A literature review on related MDMs reveals a range of models for general product data and identifies a gap in consideration of assembly-related data. In an expert workshop, requirements are raised to evaluate existing models and develop ADR-related extensions. The MDM is conceptually modelled and validated by competence questions. The paper presents an approach that makes assembly-related data usable for hybrid decision support and enables engineers to think ahead of ADR processes.

Keywords:

meta data model, product data, assembly, disassembly, re-assembly, extreme data

1 Introduction and Research Question

From the perspective of product creation, Circular Economy means to plan and engineer products in a way that it is economically attractive to circulate all material into new products after decommissioning [1]. Concepts of Circular Economy introduce circularity strategies like reuse, repair, refurbish or remanufacture that keep the original structure of the product and its components [2]. A significant level of complexity is added when these strategies are applied not only on integrated product level, but also on lower levels of assemblies and parts. Enabling Circular Economy requires the economic closing of material flows on highest possible levels. High-level circularity strategy means re-assembly of usable assemblies of parts in complete. Therefore, approaches, like Design for Assembly, Disassembly and Reassembly (DfADR) become increasingly important as well as the consideration of ADR capabilities. Design for Capabilities (DfC) refers to DfADR by considering capabilities: Capabilities comprise machinery, plant, equipment and humans. They must be known and considered in product engineering to ensure later disassembly and re-assembly [3]. Currently, there is a lack of considering human capabilities. The required capabilities of a human highly depend on the product's features. Product specifications such as joining points define which ADR capability the employee needs. For this purpose, assembly-related data of a product must be made usable already during early stages of product creation. Such data concern both, joining and separating different components of a product.

However, disassembly of products leads to processes that differ from assembly and are more complex [4]. In industrial realization, combinations of disassembly processes with inspection and sorting of used

products in terms of reverse logistics need to be considered [5]. In the realization ADR-process data are directly affected by product data. The ambition to deduct actual capabilities from ADR-process data needs to overcome challenges due to extreme characteristics of such data: the actual ADR-process is only indirectly represented, gaps and errors in data collection cannot be avoided, sensor data needs to be merged with human-generated logs, observation data of human labor is mixed with machine-assisted processing, and granularity spans from MTM-related movements to work process level. Hybrid decision support systems open up potential to process such data efficiently and support early stage decisions based on ADR and capability-related extreme data. A hybrid decision support system combines established engineering methods with methods of Data Science and Artificial Intelligence (DS/AI) like machine learning or ontology-based reasoning. Therefore, it is important that data are machine-readable and stored in a structured way. For this reason, ADR-related product data is structured by an integrative Meta Data Model (MDM) to enable DfADR and support product engineers, coupling ADR-process data that is fed back from actual ADR-processes and engineering data that is created for new products.

The paper at hand proposes an MDM that forms the basis for hybrid decision support to assist engineers by making data interpretable based on knowledge-based DS/AI methods. The focus is on product-related data, while process-related data is focused in parallel work [6]. Therefore, to identify and evaluate models that consider ADR-related product data and can be mapped to ADR capabilities, this article follows the research question: *Which extensions to existing product MDMs enable the link to ADR-capability representations?*

The paper is structured as follows: This introduction and presentation of the research question is followed by a presentation of the methodological approach. In the following chapter, the fundamentals of the topic are presented. Chapter 4 is divided into three sections. First, the systematic literature review is introduced. Secondly, the developed product MDM is presented. And finally, this MDM is validated in the third section. The work concludes with a conclusion and an outlook.

2 Methodological Approach

This research follows a multi-stage methodological approach combining a literature-based scientific approach with the On-To-Knowledge Methodology [9] as an established and often cited ontology engineering methodology. First, an expert workshop with five experts from product creation is conducted to define ADR-related requirements for a product MDM and synthesize them by thematic clustering. Relevance of the range from engineering to ADR process data is investigated. Ten requirements of the MDM are defined, which are transformed into competence questions that the MDM shall answer. In parallel, a systematic literature review is conducted to validate expert statements and to gather related work. The databases Scopus, Web of Science and IEEE Xplore are searched for publications applying a developed search string combining keywords related to product MDMs. In addition, relevant publications from previous research are included into the study. In the next step, the requirements are compared with existing models of product representation identified in the literature review. Relevant partial models form the basis of the envisaged DfC-focused product MDM, which is completed by mapping concepts in terms of semantic similarities and associations. The validation is carried out based on the requirements and competence questions.

3 Fundamentals

Product data in general describe the technical characteristics of a product, such as dimensions or material. Meta data are “data about data”, i. e., structured data that provide descriptive contextual or

administrative information about data in focus. By that, meta data support the management and use of data. Product meta data in special provide structural information about the data of a product. Meta data thus enable the efficient management and retrieval of product data without representing primary product characteristics. MDMs form the foundation for managing and analyzing large and heterogeneous data sets by rendering them machine-readable and semantically consistent. They enable the systematic collection of meta data from diverse entities such as products, humans, processes or companies and make this information available in a standardized way. Furthermore, an MDM specifies the properties, constraints, and permissible values of the data it describes. [7]

Beyond data structuring, MDMs can be implemented in ontologies by using languages such as OWL, which provide formal semantics that enable machine-interpretable meaning and automated reasoning over data sets. MDM thus provides an essential prerequisite for knowledge-based DS/AI methods. [8] For validation of such ontologies, competence questions can be used. Competence questions enable to validate semantic models by ensuring that relationships within these models can answer the questions [9]. In addition, competence questions can be applied to meta data models to assess whether the defined metadata elements and their relationships are sufficient to support the intended use cases and information needs [10].

A fundamental, well-established standard to describe product data is the Standard for the Exchange of Product model data (STEP). STEP is a neutral standard for product data that also stores meta data. The standard is written in the data definition language EXPRESS [11]. In addition to geometric data, described by points, surfaces, and curves, STEP also stores meta data like, for example, part numbers, names, and tolerances [12]. The Core Product Model (CPM) and the Open Assembly Model (OAM) are often cited product MDMs. FENVES et al. used the CPM to describe a product using basic elements. The product, referred as an artifact, is the central element that is specified in more detail via function, form, and behavior [13]. The CPM is used to structure product information [13,14]. The OAM, developed by BAYSAL et al., is an extension of the CPM [15]. The OAM focuses on assemblies and enables the storage of information on, for example, assembly or tolerance specifications [15]. CPM as a basis contains functions and behavior of the product and thus describes the product itself. OAM describes ADR relationships and thus the product structure. The representation of ADR capabilities require process logics, which are not considered in CPM and OAM. [14,15]

Extreme data is defined by European Union program Horizon Europe as data whose e.g. volume, speed, variety or complexity reach levels, which make current technologies fail while processing them [16]. PETCU et al. categorize them into the three main dimensions volume, variety and velocity [17]. Human-related data like ADR capabilities are characterized as extreme data due to highly individual characteristics [3].

4 Approach

First, requirements of an ADR-related product MDM are defined. These requirements are identified in an expert workshop with five experts from product creation. The experts were asked to identify product features that are relevant for ADR processes and those are not. Answers were discussed and clustered in iterative rounds of questions followed by a final group discussion. As a result, ten requirements are identified (see table 1). The requirements are subdivided into two main groups. The first group comprises requirements that describe the product itself. These include requirements according to geometry, function, and structural properties of the product (R01–R06). The second group consists of requirements relating product data to the processes and handling of the product. This group deals with requirements according to ADR-processes like implementation of connection and separation processes at the products connection points (R07–R10).

Table 1: identified requirements for the Meta Data Model

ID	Category	Requirement
R01	describing the product	The MDM must represent the geometric shape of the product in a STEP-compliant manner.
R02		The MDM must represent the function of the product.
R03		The MDM must represent the behavior of the product.
R04		The MDM must distinguish between individual parts and assemblies.
R05		The MDM must represent the existing and possible connection points of assemblies.
R06		The MDM must represent the existing and possible separation points of assemblies.
R07	describing product related ADR-processes	The MDM must map the joining processes for all connection points.
R08		The MDM must map the separation processes for all connection points.
R09		The MDM must map product features relevant to assemble and disassemble.
R10		The MDM must map capability-related product features for ADR processes.

Based on the workshop results, competency questions are developed in accordance with the On-To-Knowledge Methodology [9]. To validate the MDM, it must be able to answer these competence questions. The expert's statements from the workshop are converted into such competence questions and it is ensured that these correspond to the product domain of the MDM. Five competence questions are derived:

- What is the {size, mass, form} of the product?
- What is the {surface texture, material} of the product?
- What joints does the product contain?
- Which human capabilities are necessary for (dis- / re-) assembling the product?
- What connection type does the product provide for ADR processes?

This approach ensures that the MDM and the ontology, implementing the MDM, can handle corresponding queries. The competence questions are reference for the validation of the approach.

4.1 Literature Review

A literature review is conducted to identify existing product data MDM. For this purpose, the online platforms *Scopus*, *Web of Science*, and *IEEE Xplore* are searched. In addition to fundamental models, four models are identified and included into further analysis.

OntoSTEP is an ontology that semantically represents STEP concepts as concepts and relationships. It focuses on meta-data elements of STEP, i. e., the semantic description and integration of product data. OntoSTEP describes how the product is specified and provides formal semantics that make the model expandable. With respect to ADR capabilities, an extension is possible. [18,19]

The **Ontology Intelligent Master Model (OntoIMM)** is a formal ontology model for the structured, consistent, and cross-system mapping of product and process knowledge [20]. In order to overcome the arising limitations of CPM at this point, YU ET AL. introduce CPM+ in context of OntoIMM. A significant extension of this approach is the addition of *Parameters* and *ParameterRelationships*, which enable parameterization of products more concretely [20]. The **Common Data Model (CDM)** follows a similar parameter approach. The CDM is a neutral parameter and data model that contains semantic design parameters that are necessary, for example, to link CAD models [21]. However, the CDM not only stores this geometric product data, but also non-geometric functional parameters and constraints [21]. This means that ADR-related capability data, for example, can also be modeled here.

The **Product Semantic Representation Language (PSRL)** by PATIL et al. is an ontology-based, formal representation language that enables semantic interoperability between different systems in product engineering. In PSRL, product information such as objects, assemblies, specifications, and their relationships are modeled in a structured and semantically explicit way. A key feature of PSRL is that product information is formally modeled using description logics. [22]

Further identified publications demonstrate the expandability or use as a basis for these fundamental models. For example, FIORENTINI et al. use CPM and OAM to develop an ontology for representing assemblies with the Web Ontology Language (OWL) [23]. The authors describe CPM as a particularly generic model that forms a basis for further developments [23]. In this context, OAM ensures extensibility to assemblies [23]. In another paper, FIORENTINI et al. also point out the good basis of CPM and emphasize that OAM encompasses both the assembly as a concept and the assembly as a data structure [24]. This multitude of mentions proves the reusability and extensibility of the models.

The models identified in the literature review are evaluated in terms of their requirements fulfillment. In particular, gaps are identified in terms of “describing product related processes”. The mapping of joining or separating processes and capabilities is an extension that is insufficiently covered in existing models. At the same time, however, the literature review also showed that a large number of models can be expanded. Existing models can be used to describe products using meta data models or ontologies.

The evaluation reveals that OAM provides the best foundation for the present purpose. It fulfills almost all requirements. The mapping of capability-related product features for ADR processes is fulfilled to a limited extent. At the same time, other approaches only provide partial solutions for R10. Because OAM is essentially based on CPM, these two models are used as the basis for further development. The essential elements, above all the “artifact” concept, are adapted. The following section presents the developed product MDM and explains the parts taken from OAM and CPM.

4.2 The Product MDM

The extended product MDM is based on fundamental literature revealed in the literature review. The model essentially incorporates elements of CPM and OAM. Due to their focus on product data and insufficient mapping of ADR-process-related capabilities, CPM and OAM need to be further developed and linked to other MDMs for process and capability description. Overall, the developed MDM provides answers to the research question. The following Figure 1 shows the entire product MDM using the graphical notation of a class diagram. The classes based on the existing models are highlighted according to the key. According to the CPM, the top-level element is the *artifact* [13]. This can be a complete product, as well as a *subassembly* or a single *part* of it. Various properties can be assigned to the *artifact*, such as *function*, *feature* or *form*. For instance, *form* can in turn be divided into classes such as *material* and *geometry*. Each class of the MDM is described with various attributes. For example, the *material* of the *artifact* is specified in more detail by its density and mass. With regard to linking product data to ADR capabilities, this data is important, for example, to ensure that a person is able to lift the product (artifact) with its mass. The same check is conceivable with regard to the *geometry* class. Here, it must be ensured that, for example, arm spans, i.e., physical body characteristics, are sufficient to grip the product securely. In general, the *artifact* also has additional meta data that is recorded in *specifications*. Assembly and disassembly instructions are particularly relevant with regard to ADR processes, which are subordinated to the *specification* class. The classes *part* (system element) and *assembly* (subassembly) are based on the OAM [15]. The *part* describes the lowest possible level on an *artifact* [15]. Each *assembly* consists of at least one *part*. In the MDM, we adopt the concept of the *OAMFeature* from OAM, which is a subclass of the *feature* from the *artifact*. A feature is a unit of an *artifact's* form that serves a specific function such as assembling [13]. Each *AssemblyFeature* is an *OAMFeature*. The *AssemblyFeature* is defined here as a property of the product (part, (sub-)assembly, or artifact) that is

deliberately created by the engineer to connect it to another product (part, (sub-) assembly, or artifact). Accordingly, the link to the *connections* class follows: An assembly feature enables a connection.

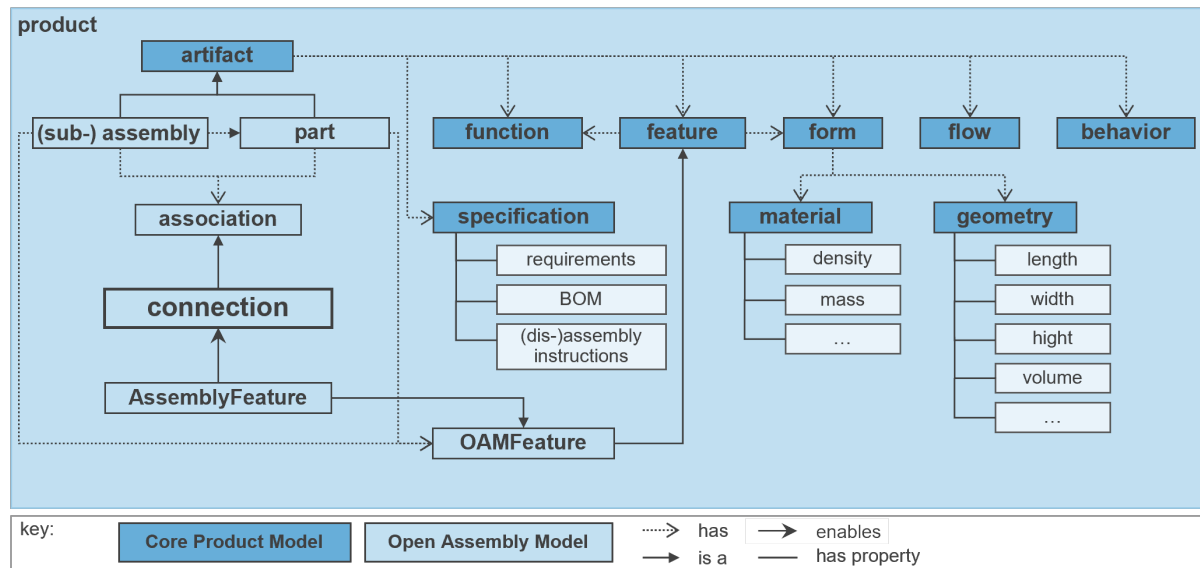


Figure 1: Product MDM based on elements of CPM and OAM with added “connection” concept

Figure 2 shows a detailed excerpt of the developed MDM presented in Figure 1. Exemplary human capabilities describing resources for ADR processes are focused. Different classes within the product MDM require access to resource-related information in order to support their interpretation. The exemplary resources described are various human characteristics from which human ADR capabilities are derived directly or indirectly. Various elements of the product MDM require capability information of resources. An *artifact* can have various *specifications*. In figure 1, for example, assembly and disassembly instructions are listed. In order to understand and implement these, humans need certain *mental capabilities*, which have to be linked to the product model. *OAMFeatures* and its concretization as *AssemblyFeatures* serve to connect two artifacts. When focusing on human capabilities, mental capabilities need to be represented to identify and understand required types of connections.

Accordingly, *mental capabilities* of human are required to understand the type of *connection* and to implement it using the correct joining process. This is especially true when such a connection needs to be properly resolved. Linking the classes of product MDM with *mental capabilities* makes it possible to query whether a person has these capabilities. The *connection* class is directly linked to the *mental capabilities*. The capability model must contain information about the processes that a resource, -including a human - must fulfill. These capabilities also include so-called *domain knowledge*, such as *material knowledge*, which ensures the correct handling of the product and the materials used in the process. In addition to capabilities in the traditional sense, certificates may also be relevant. In industrial field, for legal reasons, the statement “The employee is allowed to do this” must be equated with the statement “The employee is capable of doing this.” In other words, an employee can only do something if he or she is legally authorized to do so. The employee needs necessary certificates and training for machines and equipment for all required joining and separating processes. As an example, the *material* (attribute: mass) is linked to *certifications acquired*, to ensure that the employee is able to handle the mass of the product. Therefore, a capability model must contain such information about certifications of employees.

In addition to these mental and legal capabilities, physical conditions must also be verifiable. As already indicated in the description of the product MDM, this is exemplified by *body characteristics* which is a

subpart of the *physical capabilities*. A component with a specific geometry requires corresponding physical characteristics in humans to ensure that this component can be moved. As an example, it must be ensured that a person is able to handle the mass of an artifact. Here, a link to *certifications acquired* is also necessary to ensure that the person is allowed to lift the mass.

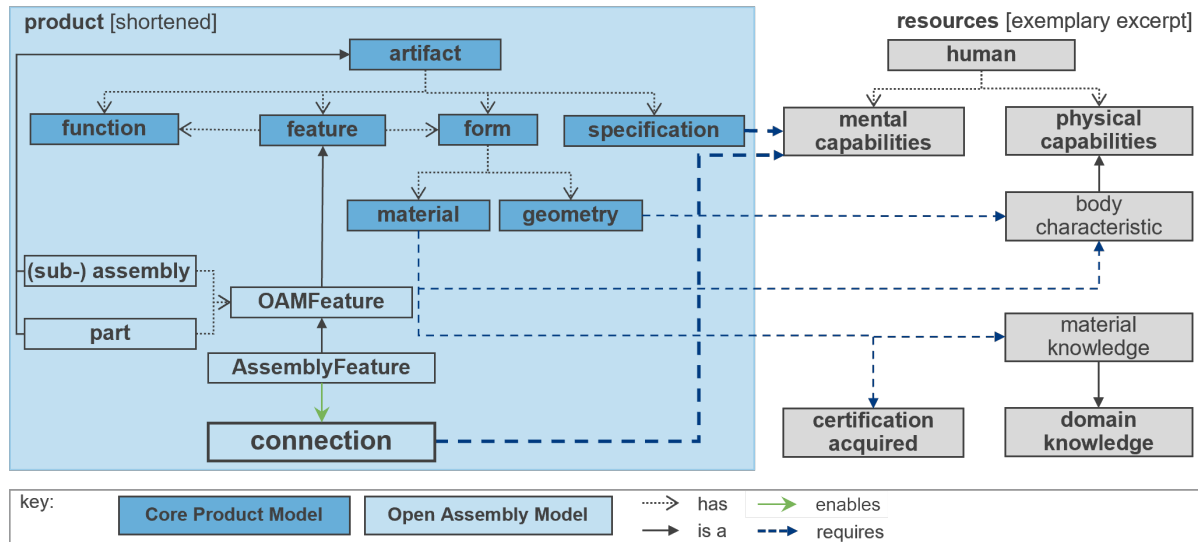


Figure 2: Excerpt from the product MDM, linked to high-level human capability concepts

According to BAYSAL et al., a connection represents a physical connection between two artifacts [15]. A person needs the capabilities and body characteristics to be able to use the connections or the underlying processes and tools. By that, the person is able to understand and handle the artifact. This statement makes it clear that, in addition to the precise description of the product, other aspects must also be considered. These include both the linking of meta data with available capabilities and the ADR-processes for fulfilling the connections within the product.

Figure 3 shows an excerpt of the developed product MDM with the focus on linking the connection to assembly or disassembly processes. There are various ways to join or separate connections. Therefore, joining and separating processes must be analyzed to determine boundary conditions that apply to the processes to be used. For example, the assembly and disassembly processes associated with each connection must be defined. In particular, disassembly cannot be assumed to be the reverse of the assembly process. Therefore, information about assembly, disassembly and re-assembly processes must be linked. The *connection* class enables this link. In line with the ADR capabilities, the process model indicates the *assembly process* and *disassembly process*. The link to a process model via the *connection* class is essential. A process must contain the specifications of assembly and disassembly processes. This includes, for example, the joining or separation process, the necessary tools, the process parameters and the safety requirements.

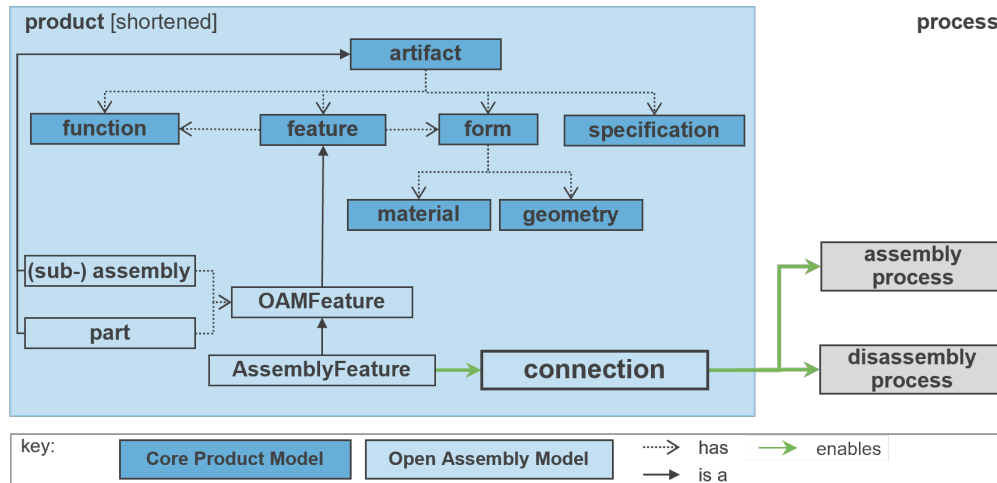


Figure 3: Excerpt from the product MDM, linked to ADR processes

4.3 Validation of the Approach

In an expert workshop, the foundations for requirements and competence questions relating to the MDM were established. The experts were five research associates from the field of product creation. The description of the product MDM and Figure 1 of the MDM shows that the MDM meets requirements R01 – R06. By describing the *connection* class and its ability to be linked to a process model, the requirements R07 and R08 are also verified. The descriptions of *specifications* and *features* showed that the MDM maps product features relevant for assembly, disassembly and capability-related ADR processes (R09, R10). Therefore, the MDM fulfills the prerequisites for validation. To validate the MDM, it must be able to answer these competence questions. Each of the five competence questions is answered based on the MDM:

What is the {size, mass, form} of the product? The size, mass and form of the product (artifact) are specified in the model. Each *artifact* has a *form*, which in turn has a *geometry* and a *material* (c.f. Figure 1). The *geometry* class includes attributes such as maximum length, width, and height. The *material* class has the attribute *mass*. In addition, each material also has the attribute density. Together with the volume, the mass can always be determined. This information is available for queries regarding handling.

What is the {surface texture, material} of the product? The surface texture and material can also be accessed via the *form* of an *artifact*. The *material* class is subordinate to the artifact. As described above, this class can be linked to, for example, human *material knowledge* with all its associated attributes. If information about surface texture is relevant in the context of handling, information from assembly/disassembly instructions is also recorded in the *specification* class of the *artifact*.

What joints does the product contain? Each part or (sub-)assembly is an *artifact* (product). A part or (sub-)assembly can in turn have an *OAMFeature*. An *OAMFeature* is a joint option purposefully created by the engineer and clarifies how this joint must be created. The *OAMFeature* in an assembly is an *AssemblyFeature* and enables a connection, the type of which is described.

Which human capabilities are required for (dis-) assemble the product? The required human capabilities for (dis-) assemble the product are defined by the joining and separating processes of the product. These are included in the *connections* of the *artifact* (product). In figure 2, for example, the *connection* is linked to the *mental capabilities* of a human.

What connection type does the product provide for ADR processes? Each *artifact* (product) has a *connection* which enables a physical connection to another artifact. This *connection* and their properties enable the link to ADR processes.

The workshop discussion revealed that all required information can be found in documents and models such as (dis-) assembly instructions, CAD models, or system models. This information and their meta data can be mapped to the product MDM that has been developed. For example, meta data from assembly instructions are mapped in the *specifications* class and are found by corresponding queries. This way, the information can be used to ensure the assembly and disassembly of the product and extreme data become usable.

5 Conclusion and Outlook

This paper presents an identification and evaluation of MDMs for product data that contain assembly-related data. The aim is to use this MDM to enable DfADR in product engineering by structuring assembly-related data and enabling engineers to think ahead of ADR tasks and handle extreme data. The literature review reveals a range of established models for general product data and identifies a gap in consideration of ADR process- and -capability-related data within product MDMs. Based on CPM and OAM, this paper proposes mapping concepts enabling retrieval of necessary capabilities to join or separate physical connections between artifacts. A fundamental condition is the existence of the required capabilities for ADR processes. Two fields of action are solved to enable ADR processes in product creation: First, the *connections* of the *artifact* must be linked to the mental and physical capabilities of the employees. These must be separated into assembly and disassembly. Second, each *connection* of the *artifact* must be linked to a (dis-) assembling process. These must be fulfilled to contain and use assembly-related extreme product data in the MDM.

The application-oriented structuring of product data is the first step to enable engineers for DfADR, support them through a hybrid decision support and make the Circular Economy feasible. The developed MDM offers advantages in managing extreme data by creating structured, consistent and contextualized information spaces. It also facilitates hybrid decision support and optimizes data management simultaneously. Looking ahead, the developed MDM will form the basis for the development of an ontology. This will be the foundation for hybrid decision support, which combines methods of DS/AI approaches and classic engineering methods.

6 Acknowledgement

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – project no. 543074930 – as part of the SPP 2443 (Hybrid Decision Support in Product Creation, project no. 521148212, www.spp2443.de).

References

- [1] I. Gräßler and J. Pottebaum, “Generic Product Lifecycle Model: A Holistic and Adaptable Approach for Multi-Disciplinary Product–Service Systems,” *Applied Sciences*, vol. 11, no. 10, p. 4516, 2021. DOI: <https://doi.org/10.3390/app11104516>.
- [2] José Potting, M.P. Hekkert, E. Worrell and Aldert Hanemaaijer, “Circular Economy: Measuring Innovation in the Product Chain,” 2017.
- [3] I. Gräßler, J. Pottebaum, P. Nyhuis, M. Schmidt, B. Grewe, F. Vollenkemper, T. Hesse and T. Meinecke, “Evolving Design for Assembly, Disassembly and Reassembly into a new paradigm: Design-for-Capabilities with Hybrid Decision Support as an enabler for circular products,” in *Stuttgarter Symposium für Produktentwicklung SSP 2025*: Stuttgart: Fraunhofer IAO, 2025, pp. 509–518.

- [4] K. Wang, X. Li, L. Gao and P. Li, "Modeling and Balancing for Disassembly Lines Considering Workers With Different Efficiencies," *IEEE transactions on cybernetics*, vol. 52, no. 11, pp. 11758-11771, 2022. DOI: <https://doi.org/10.1109/tyb.2021.3070122>.
- [5] K. A. Hribernik, M. von Stietencron, C. Hans and K.-D. Thoben, "Intelligent Products to Support Closed-Loop Reverse Logistics," in *Glocalized solutions for sustainability in manufacturing. Proceedings of the 18th CIRP International Conference on Life Cycle Engineering, Technische Universität Braunschweig, Braunschweig, Germany, May 2nd - 4th, 2011*, J. Hesselbach and C. Herrmann, eds.: Berlin, Heidelberg: Springer, 2011, pp. 486–491.
- [6] I. Graessler and T. Hesse, "Enabling Extreme Data by using DS/AI approaches: A Process Metadata Model to link Assembly Features and required Human Capabilities," in *HIPPE 2026 – International Symposium on Hybrid Intelligence in Product and Production Engineering*.
- [7] J. Stark, "Product Lifecycle Management (PLM)," *Product Lifecycle Management (Volume 1)*, pp. 1-32, 2022. DOI: https://doi.org/10.1007/978-3-030-98578-3_1.
- [8] L. Woolcott, "Understanding Metadata: What is Metadata, and What is it For?," *Cataloging & Classification Quarterly*, vol. 55, 7-8, pp. 669-670, 2017. DOI: <https://doi.org/10.1080/01639374.2017.1358232>.
- [9] Y. Sure, S. Staab and R. Studer, "On-To-Knowledge Methodology (OTKM)," in *Handbook on Ontologies*, P. Bernus, ed.: Berlin/Heidelberg: Springer Berlin Heidelberg, 2004, pp. 117–132.
- [10] Ontology development 101: A guide to creating your first ontology, 2001.
- [11] A. McKay, M. S. Bloor and A. de Pennington, "A framework for product data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 8, no. 5, pp. 825-838, 1996. DOI: <https://doi.org/10.1109/69.542033>.
- [12] International Organization for Standardization, *Industrial automation systems and integration — Product data representation and exchange. Part 1: Overview and fundamental principles*, vol. 25.040.40 (10303-1:2021) (New York: ISO).
- [13] S. J. Fenves, S. Foufou, C. Bock and R. D. Sriram, "CPM2: A Core Model for Product Data," *Journal of Computing and Information Science in Engineering*, vol. 8, no. 1, 2008. DOI: <https://doi.org/10.1115/1.2830842>.
- [14] S. J. Fenves, *A core product model for representing design information* Gaithersburg, MD: National Institute of Standards and Technology, 2001.
- [15] M. M. Baysal, U. Roy, R. Sudarsan, R. D. Sriram and K. W. Lyons, "The Open Assembly Model for the Exchange of Assembly and Tolerance Information: Overview and Example," *ASME 2004 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pp. 759-770, 2008. DOI: <https://doi.org/10.1115/DETC2004-57727>.
- [16] Horizon Europe, "Methods for exploiting data and knowledge for extremely precise outcomes (analysis, prediction, decision support), reducing complexity and presenting insights in understandable way (RIA).," [Online]. Available: <https://ec.europa.eu/info/funding-tenders/opportunities/portal/screen/opportunities/topic-details/horizon-cl4-2022-data-01-01>. [Accessed 30 July 2025].
- [17] D. Petcu, G. Iuhasz, D. Pop, D. Talia, J. Carretero, R. Prodan, T. Fahringer, I. Grasso, R. Doallo, M. J. Martin, B. B. Fraguera, R. Trobec, M. Depolli, F. A. Rodriguez, F. de Sande, G. Da Costa, J.-M. Pierson, S. Anastasiadis, A. Bartzokas, C. Lolis, P. Goncalves, F. Brito and N. Brown, "On Processing Extreme Data," *Scalable Computing: Practice and Experience*, vol. 16, no. 4, pp. 467-490, 2016. DOI: <https://doi.org/10.12694/scpe.v16i4.1134>.
- [18] R. Barbau, S. Krüma, S. Rachuri, A. Narayanan, X. Fiorentini, S. Foufou and R. D. Sriram, "OntoSTEP: Enriching product model data using ontologies," *Computer-Aided Design*, vol. 44, no. 6, pp. 575-590, 2012. DOI: <https://doi.org/10.1016/j.cad.2012.01.008>.
- [19] Ontostep: OWL-DL ontology for step, 2009.
- [20] C. Yu, F. Zhang, S. I. Butt, Y. Yan and W. Lv, "OntoIMM: An Ontology for Product Intelligent Master Model," *Applied Sciences*, vol. 9, no. 12, p. 2553, 2019. DOI: <https://doi.org/10.3390/app9122553>.
- [21] G. P. Gujarathi and Y.-S. Ma, "Parametric CAD/CAE integration using a common data model," *Journal of Manufacturing Systems*, vol. 30, no. 3, pp. 118-132, 2011. DOI: <https://doi.org/10.1016/j.jmsy.2011.01.002>.
- [22] L. Patil, D. Dutta and R. Sriram, "Ontology-Based Exchange of Product Data Semantics," *IEEE Transactions on Automation Science and Engineering*, vol. 2, no. 3, pp. 213-225, 2005. DOI: <https://doi.org/10.1109/TASE.2005.849087>.
- [23] Xenia Fiorentini, Iacopo Gambino, V.C. Liang, Sebti Foufou, Sudarsan Rachuri, Mahesh Mani and Conrad E. Bock, "An Ontology for Assembly Representation," 2007.
- [24] *Information models for product representation: core and assembly models*, 2005.

Enabling Extreme Data by using DS/AI approaches: A Process Metadata Model to link Assembly Features and required Human Capabilities

Iris Graessler¹, Thomas Hesse^{1*}, Jens Pottebaum¹

¹Paderborn University, Chair for Product Creation, 33100 Paderborn

*Corresponding author. E-Mail address: Thomas.hesse@hni.upb.de ; Tel.: +49 5251/606259

Abstract

Design for Assembly, Disassembly and Reassembly (DfADR) prepares for high levels of material circularity in terms of repair and remanufacturing. The intention of reassembling products and their parts implies that inspection and sorting need to be integrated with value-conserving objectives. Further, disassembly and reassembly require higher competence levels of workforce than assembly. To provide the required information during engineering, a metadata model (MDM) is required that links established product models to capability models. Since processes to be carried out for realizing a product largely determine required employees' capabilities, the proposed approach uses the ADR process domain to connect product and capability domains. The research is based on a systematic literature analysis to identify existing approaches regarding MDMs of ADR processes. Based on the results, an integrative MDM is developed that answers defined competence questions. The model enables queries which support engineers by feedback on which skills are required for a product. This builds up a basis to check whether required skills are available in the manufacturing company, either early in product engineering or later in ADR planning.

Keywords:

metadata model, capabilities, reassembly, disassembly

1 Introduction and research questions

Design for Assembly, Disassembly and Reassembly (DfADR) prepares for high levels of material circularity in terms of repair and remanufacturing by recycling products on a higher product hierarchy level. The intention of reassembling products implies that inspection and sorting need to be integrated with a value-conserving objective. Potentially, the originally planned disassembly processes cannot be carried out due to damage, for example, which is why further, partially even improvised processes are carried out. This means that employees need advanced skills to solve the problem. Therefore, it is necessary to know the required capabilities of employees and tools to enable disassembly and reassembly of products and their components [1]. Further the capability based task assignment improves the productivity during the ADR processes. [2,3] Data from engineering is evaluated to provide the necessary information during engineering. ADR relevant data may be extreme due to different sources and applications of data within the production environment. Extreme data is characterized by its origin from distributed sources, heterogeneous formats, high volume, unknown quality, errors and gaps, or different languages to a degree which makes previous technologies fail[4]. The processing of extreme data requires Data Science and Artificial Intelligence (DS/AI). The preprocessing of data to fit specific structures is required in order to analyze the data and extract information by means of DS/AI. In this case, a MDM is required which links an established product MDM based on the Open Assembly Model

(OAM) [5], to capability MDMs. A MDM for the ADR processes is useful to represent the necessary information, as it is dependent on the product which defines the boundary conditions like, for instance, the product structure or the type of joint. Special employees' capabilities are also required to carry out the processes, which are influenced by the parameters of the processes and the tools used. For this reason, the existing models are analyzed to develop a MDM which meets the required ADR-processes and characteristics. To develop a MDM that links assembly features with required human capabilities, the following research questions need to be answered: *Which granularity of details needs to be modeled to match assembly features to the capabilities of employees and tools (RQ1)? Which expansions are needed for existing process MDMs to enable the mapping between the product and the resource domain (RQ2)?*

The paper is structured as follows: starting with a description of the research methodology a literature analysis is performed to identify related work. Based on the findings of the literature analysis approaches are proposed. The concept is then verified and validated regarding the fulfillment of the requirements and the related competence question. The results are finally discussed regarding the research questions.

2 Research Methodology

The development process of the MDM is divided into six steps. First, requirements are defined regarding the linkage between capabilities and assembly, disassembly and reassembly (ADR) processes. Within an expert workshop with five experts in the research field of product creation, the defined requirements are validated. The Requirements were used to derive competence questions jointly in the workshop. Existing approaches within the subject area are identified by systematic literature analysis. The identified approaches are analyzed regarding the fulfillment of the defined requirements. Based on identified key elements from existing approaches, a new MDM is developed by analyzing existing approaches and creating the MDM which fulfills the defined requirements. The validation of the MDM is based on competence questions in reference to the validation of the Ontology Engineering Methodology (OEM) [6].

3 Literature analysis

The objective of systematic literature analysis is to identify related work, validate requirements, understand shortcomings and extract applicable contributions regarding RQs. To define targeted search strategy problems and opportunities, stakeholders as well as the focus of later MDM application are detailed in an expert workshop to define a targeted search strategy. Results are documented in the form of requirements (see table 1). The requirements cover the interdependencies of assembly, disassembly and reassembly processes to the required tools and resources. Furthermore, the tools used and the environment must be described to cover the needs of specific certifications or other requirements of the resources like human capabilities. The requirements were used to derive competence questions. The derived competence questions are used to validate the MDM and cover the main groups of requirements (see table 1)

Table 1: Requirements and competence questions of the MDM

Requirements	
ID	Description
R1	The MDM must map the joining processes required for assembly activities.
R2	The MDM must represent the interdependence between remanufacturing processes and reassembly processes.
R3	The MDM must map the separation operations required for the disassembly process.
R4	The MDM must map the dependencies between single joining operations.
R5	The MDM must map the constraints from joining operations for separation operations.
R6	The MDM must represent the process characteristics of the joining operations.
R7	The MDM must represent the process characteristics of the separation operations.
R8	The MDM must represent the boundary conditions dependent the assembly system.
R9	The MDM must represent the necessary tools and resources.
R10	The MDM must reflect the necessary workplace safety regulations.
R11	The MDM must represent the interdependence of quality assurance processes and reassembly processes
R12	The MDM must map inspection processes with the necessary checking parameters of the product.
R13	The MDM must map the assembly activities to the relevant product characteristics.
R14	The MDM must map the process characteristics to the capability characteristics.
R15	The MDM represents the interdependence between the processes and the required capabilities of the resources.
R16	The MDM represents the interdependence between processes and mental and physical capabilities.
Competence Questions	
ID	Question
CQ1	Which joining processes are required for the joint connection?
CQ2	Which separation process is required for the joint connection?
CQ3	What process parameters must be observed for the joining process?
CQ4	What process parameters must be observed for the separation process?
CQ5	What tools are needed for ADR processes?
CQ6	What quality control processes are needed?

Systematic research is conducted in scientific databases, including Web of Science, IEEE Xplore, and Scopus, to identify related approaches. Therefore, the following search string is derived which is based on keywords of the requirements and got refined using the keywords of publications of the initial analysis.

assembl AND product AND process AND (onto* OR metadata model OR MDM OR taxonomy)*

The search identified 1063 publications. After manual filtering by title and abstract, 35 publications are analyzed, resulting in 9 which are relevant for further evaluation. The analyzed approaches build three clusters. The first cluster contains approaches which represent assembly tasks and the required resources. The approach of LEE et al. [7] describes a MDM that represents the tasks required to assemble a product with regard to capabilities of a collaborative robot. Within the model, the individual assembly tasks are represented as combined tasks based on the assembly features. These are described at the lowest level by atomic actions that correspond to the basic movements of Methods Time Measurement (MTM). The publication of HASAN focuses on the representation of the assembly system to improve the assembly planning process [8]. The focus of the approach is on the representation of necessary manufacturing

processes and the required resources. In addition, the processes are differentiated into joining processes and handling processes. The handling processes are to be equated with secondary processes. DAS and SWAIN [9] present an ontology which describes a collection of joining processes. The information of the ontology consists of materials, products, joining processes and requirements of the joining processes. Based on the knowledge a 5-step process to analyze CAD models is described. The identification of the required joining process builds on automated rule-based systems. The approach of CLAEYS et al. [10] represents the manual assembly process by a context model using assembly instructions and further information about the context. The context information contains the locations, the people involved and further objects. The interdependencies between the objects are modeled to represent the influence of the object's condition to process and experience of the employees.

The second cluster contains approaches which represent a mapping of the product, process and resource domain. The focus of the approaches is the representation of required capabilities regarding assembly processes. The publication of GHANJAOUI et al. [12] describes reusable modeling elements from model-based Systems Engineering. The modeling elements are used to link design and assembly concepts. It aims to link the product architecture to required capabilities of the employees. The transfer of the modeling data is enabled using a MDM, which is used to develop ontology. In JÄRVENPÄÄ et al. [13] a approach is presented to model production resources and their capabilities. The capabilities consist of simple capabilities, combined capabilities and competences. The selection of resources is based on the comparison between the requirements and capabilities. Further the approach of FERRER et al. [14] aims to link data of different engineering disciplines. The linked data includes data on processes, products and resources. The developed database is used to configurate automated assembly systems. To configurate the required resources the product and process data is analyzed. Based on the findings the resources are configured to meet the requirements.

The third cluster contains approaches which represent assemblies and their assembly systems. In BAO et al. [11] an approach is described which addresses the development of a ontology to represent the digital twin of assembly workshops. BAO et al. focus on the required resources of assembly processes, for example tools, auxiliary materials and human resources. The developed ontology enables the processing of relevant data. The publication of KIM et al. [15] describes an approach to represent ontology-based assemblies. The ontology is built with a hierarchy of assembly and joining concepts. The approach aims to improve the collaboration during the design of assembly models. The findings are used to model the foundation for the interdependence between processes and the required information of the product structure as well as necessary properties.

The literature analysis reveals existing approaches for MDMs in subject areas like assembly automation, assembly priority graphs and the description of the capabilities of collaborative robots. However, there is a lack of MDMs that link the ADR processes in relation to product features and required capabilities. Both assembly and disassembly processes must be considered to determine whether all the human capabilities required for the necessary ADR processes are available. The evaluation matrix (table 2) reveals that the existing MDMs are not capable to represent the required information to identify the needed capabilities for ADR processes. The requirements R3, R5 and R7 which are in the process category are not fulfilled of any identified approach. As shown in R3 and R5 none of the identified approaches cover the separation operations and their process characteristics. In addition, the interdependencies between joining and separation operations (R5) are missing but are essential to identify required capabilities of the separation operations because there are multiple possible separation operations linked to one joining operation. Further the requirements R10-R12 are not fulfilled by the approaches which show that the required safety regulations of the workplace (R10) and the quality assurance processes are not represented. Therefore, fundamental literature and regulations are analyzed to develop a process MDM which satisfies the requirements.

Table 2: Evaluation matrix of the identified approaches

● - fully addressed ◐ - partially addressed ○ - not addressed		BZY+21 [11]	CHS+18 [10]	DS 22 [9]	FAV+16 [14]	GBB+24 [12]	Has 24 [8]	JLL+11 [13]	KMY 06 [15]	LZL 24 [7]
Processes	R1	◐	◐	●	●	●	●	◐	◐	○
	R2	○	○	○	○	○	●	○	○	●
	R3	○	○	○	○	○	○	○	○	○
	R4	○	○	○	○	●	●	○	○	○
	R5	○	○	○	○	○	○	○	○	○
	R6	●	○	●	○	●	●	●	●	○
	R7	○	○	○	○	○	○	○	○	○
Resources	R8	●	◐	○	◐	●	●	◐	◐	●
	R9	●	●	○	○	●	●	●	○	●
	R10	○	○	○	○	○	○	○	○	○
Quality	R11	○	○	○	○	○	○	○	○	○
	R12	○	○	○	○	○	○	○	○	○
Interface	R13	◐	●	●	◐	●	●	●	●	○
	R14	○	●	○	◐	●	○	●	○	●
	R15	○	◐	○	○	◐	○	◐	○	○
	R16	○	◐	○	○	◐	○	○	○	○

4 Concept

Connecting the product domain to the resource domain needs the required ADR processes during the assembly of the product: The process parameters as well as the type of process define the required domain knowledge as well as the physical and mental capability needed. The relationships are based on fundamental assumptions of human-centered assembly planning. [3] The product MDM represents the product structure, features, specifications and the form of the product [5]. The resource MDM represents the capabilities of the resource, in this case the human resource. It includes domain knowledge, physical capabilities and mental capabilities. The process MDM represents a concept which links the product to the resource model (see Figure 1). The matching of the MDMs is based on the required ADR processes which are determined by analyzing the assembly features of joining operations [16] and separations operations [17]. Relationships that are transferable to other classes are not shown in the figures.

On the first level of the mapping MDM, a distinction is made between ADR processes, remanufacturing processes and the environment. The environment describes, for example, the operating resources and the assembly workplace. The ADR process is detailed by the assembly process and the disassembly process. These types of processes are further subdivided to the level of the single operations (see figure 2). The remanufacturing processes [18] instead describe processes which are a subset of different classes like ADR-processes and quality control processes. The additional allocation is used because the processes must fulfill special requirements because of the variation of the initial product quality compared to unused components. For this reason, the class remanufacturing processes has the „contain“-relation to operations for example disassembly processes or inspection. On the lowest level of the ADR

processes the processes and the required capabilities are linked. The capabilities in this case are limited to the human capabilities and are differentiated in physical capabilities and mental capabilities. These capabilities must be further subdivided to get matched to specific processes. The processes contain the necessary data which must be queried in case of testing if the resource is capable to carry out the process. Besides the process parameters the processes must contain a description which implies information

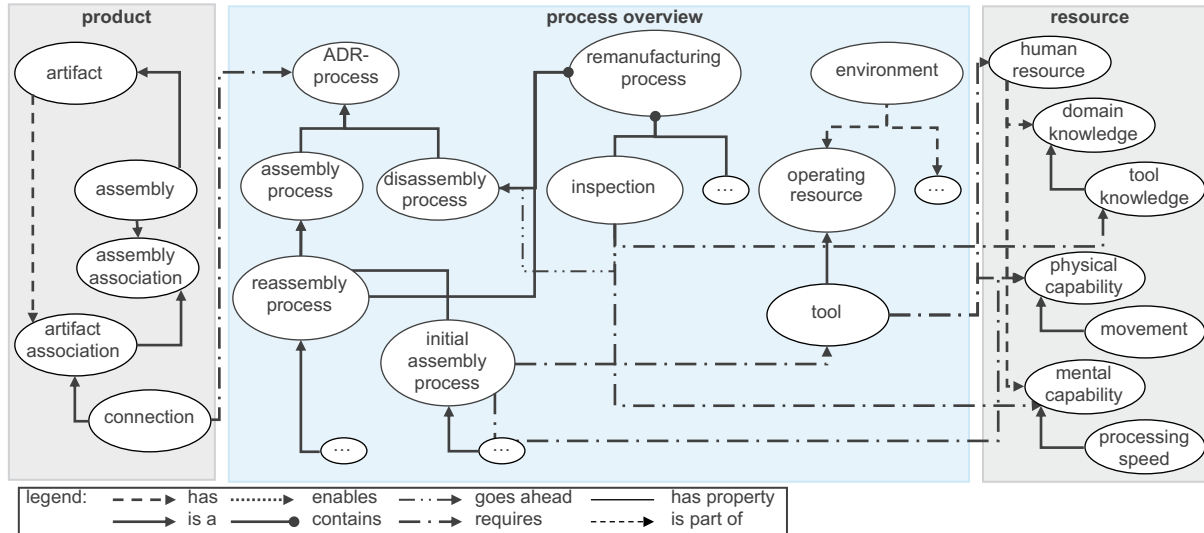


Figure 1: Overview of the process MDM

regarding mental capabilities.

In Figure 2, a focused view on ADR process and the interdependencies between the domains is presented. Based on the connection from the product model, which deploys the information of the used joining technology, the ADR process are determined on the product side via the “requires” relation to the connection. This provides information such as the type of connection and basic parameters. The ADR process of the MDM is subdivided into the assembly and disassembly processes. These are subclasses of the ADR process and are associated with the subclasses “separation operations” in case of the disassembly process. The assembly process needs to be differentiated between the initial assembly and the reassembly, because of varying inspection requirements. The initial assembly process is based on new components which implies less needs for additional quality control processes. The distinction is based on the initial state of the components to be assembled. The reassembly process instead has a different set of linked quality control processes to guarantee a good quality of the products. That means

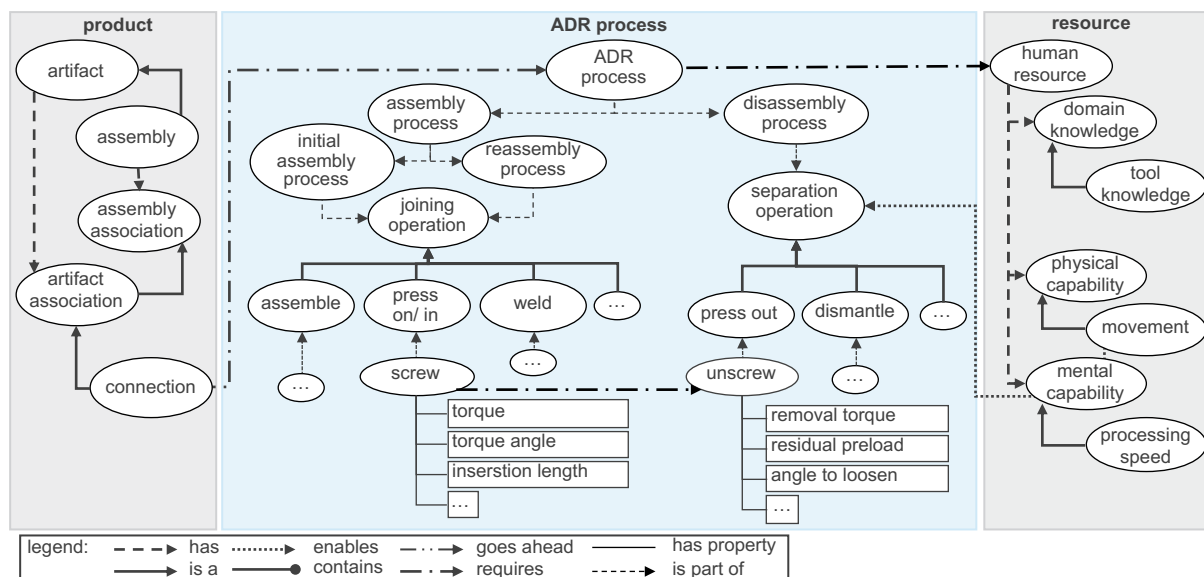


Figure 2: Detailed representation of the ADR-processes of the process MDM

in case of the reassembly processes the identical joining operations must be processed in combination with additional quality control processes.

The assembly process and the disassembly process are divided into joining and separation operations which are individually further subdivided. The individuals of the operations are linked to the required capabilities. The differentiation between manufacturing processes is based on DIN 8580 [16,19], the further differentiation of the assembly processes and the used methods are based on DIN 8593 [16] and the disassembly processes are described by DIN 8588 [17] and DIN 8589 [20]. To identify necessary properties of the joining and separation operations, basic literature from the production domain is analyzed. The screw operation is described by attributes like torque, torque angle and insertion length. Components that have already been used may require additional processes to be carried out to obtain a product of the same initial quality as in the initial assembly.

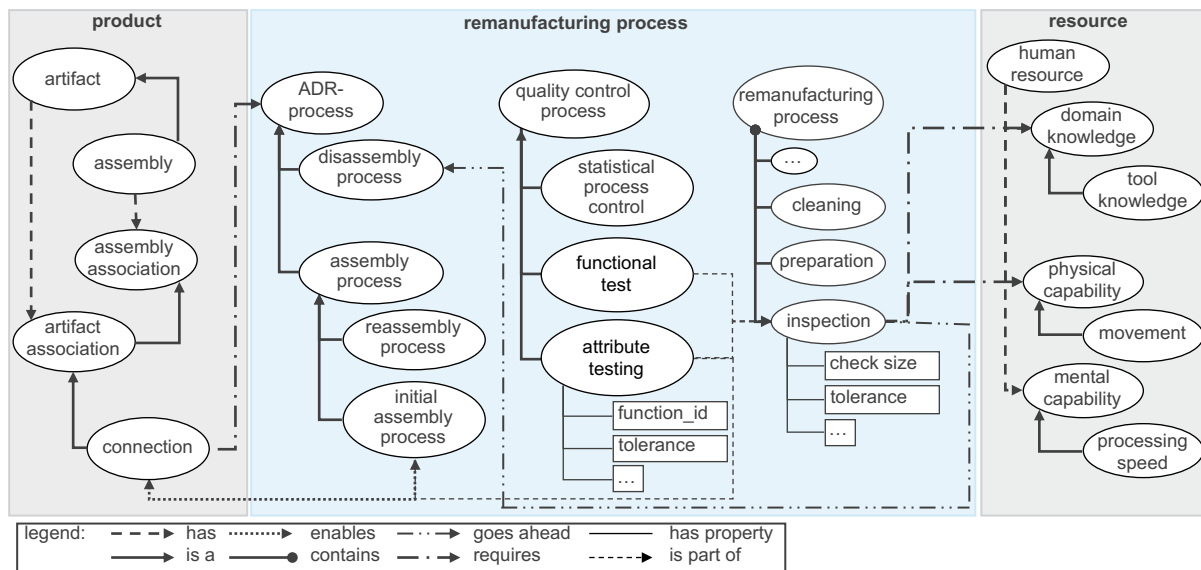


Figure 3: Detailed representation of the remanufacturing processes of the process MDM

The interdependence between the remanufacturing processes to the quality control processes and the ADR process are represented in figure 3. The remanufacturing processes are defined based on CHATTI et al. [18] and are limited to the processes which can be allocated to processes which were defined by the author and reach from the collection of products to the re-commercialization. It consists of ten steps including, for example, sorting, testing, cleaning and inspection. The developed MDM only contains seven of the ten processes, which include the manufacturing processes and excludes logistical and business processes. Each process has its specific requirements to the capabilities of the resources. Besides that, quality control processes are described which subsets are linked to specific ADR-processes. The processes which are linked by the “contains”-relation to the remanufacturing process are a subset of the higher-level classes “quality control processes” and “ADR processes”. Contained processes require in this case a differentiation of capabilities than during the manufacturing of regular products. This implies that the employees need to have more developed mental capabilities, because of the high variety of the initial state during the manufacturing process. The identification of the initial state is the reason that the inspection is needed before disassembling the product. The definition of the inspection for this approach is based on the definition of CHATTI et al. [18].

The inspection is the process to ensure that the product or the process meets the requirements and detect the faults. In the inspection, different kinds of quality control procedures are applied, such as visual analysis or a measurement of the check size. The inspection is central part of the remanufacturing processes and requires specific capabilities of the employees. In addition the subsets of the quality control processes are defined and related to the inspection by the “is part of”-relation. The different

processes are also linked to specific ADR processes with the same relation due to the fact that the employees have to control different properties during the process. In figure 4, the environment is illustrated. In addition to the actual processes that require certain employee capabilities, the objects in the process environment also place demands on employee capabilities. The objects in the environment include the tools to be used, but also the actual assembly workplace. In the case of the tools used, the properties of the tool place demand the physical skills of the employees, for example, due to weight. In addition, domain knowledge is required from which the employee recognizes and applies the appropriate tools, but also has the authority to use the tools [21]. The selection and application of tools in disassembly also represent a special requirement. This is because, for example, a connection cannot be loosened due to rust, and the employee must recognize which tool needs to be used instead. As shown in Figure 4, a “requires” relation is used to connect the product structure to the required tool. The specification of the connection defines the properties of the tool. This also specifies central parameters of the tool, such as the tightening torque or the wrench size in the case of a screw connection. Therefore, the class “tool” also is connected by the “requires” relation to the domain knowledge, more specific the tool knowledge. The tool knowledge in this case describes the knowledge about the safety instruction for using the tool as well as the key properties that might have to be adjusted by the employee, for example the tightening torque of the torque wrench. Also, the employee must be capable of using the tool regarding physical capabilities, for example the movement of the tool which depends on the weight.

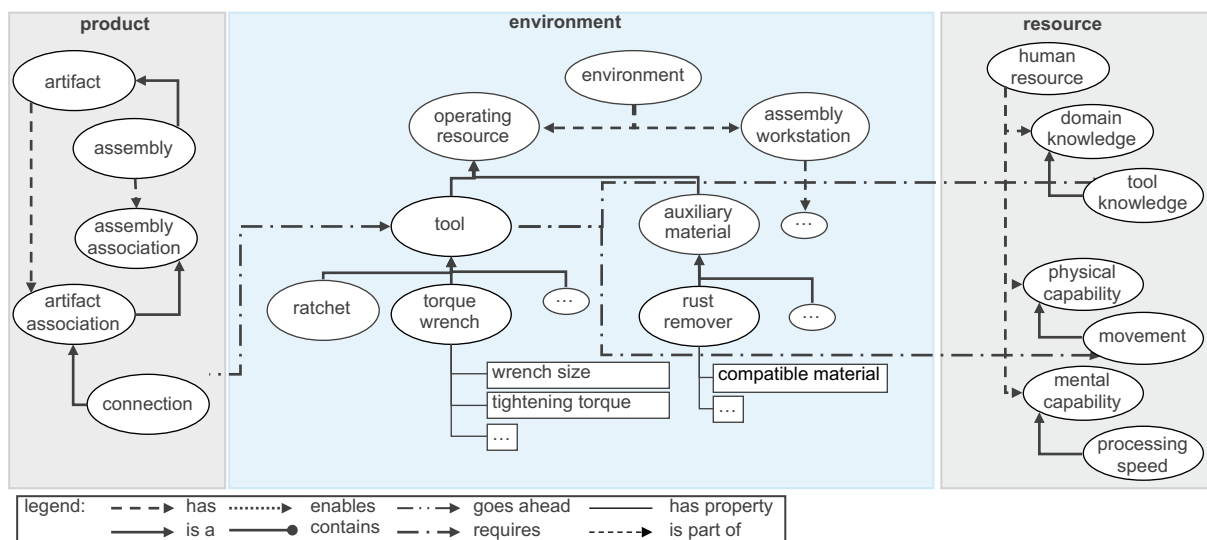


Figure 4: Detailed representation of the assembly system of the process MDM

5 Verification and Validation of the MDM

Based on the requirements, the developed MDM is verified. As shown in Figure 2, the MDM represents the separation processes, their properties and the interdependencies which fulfill the requirements R3 and R5. Also, the quality assurance processes and their properties are represented as shown in Figure 3. The MDM is validated according to the ontology engineering methodology [6] using the competence questions. By answering the competence questions, it becomes clear whether the MDM contains all the necessary information for the use case.

Which joining processes are required for the joint connection? The competence question is answered by the “require” relation between the class “connection” and the class “ADR processes” and their subclasses. The class “connection” contains the necessary information within their properties for example the instance of the connection has the property “connection type” with the value “screw” then the class “screw” is connected by the required relation.

Which separation process is required for the joint connection? The competence question is answered by two different queries. The first possible query uses the relation between the joining operation and the associated separation operation using the relation “require” between the classes. The second possible query is the direct path between the class “connection” and the subclasses of “separation operation”.

What process parameters must be observed for the joining process? The necessary information is defined on the product domain and are associated with the class “connection”. Due to the connection between the instance of class “connection” to the instance of the subclass “joining operation” the necessary information is accessible.

What process parameters must be observed for the separation process? The necessary information is accessible by two different queries. The first query uses the relation of the class “separation operation” with the class “connection”. The second query is possible using the relation between the instance of the subclass of the “joining operation” and “separation operation”

What tools are needed for ADR processes? The competence gets answered by the association between the instance of the subclass of “joining operation“ or “separation operation“ and the instance of the subclass “tool“. The subclasses of ADR-processes are described by the property „tool name“ which defines the class of the tool. Further information which defines for example the size of the wrench is associated to the class “connection” and the “ADR process“.

What quality control processes are needed? The competence question is answered by the relation of the instances of the subclasses from the “ADR processes”. Every Instance of this subclass has specific relation to instances of the class “quality control processes”. In addition, in case of the disassembly processes is the inspection associated with the relation “goes ahead”.

In summary, the mapping MDM answers the derived competence questions. The MDM contains all information that are necessary to represent the joining and separation operations and their parameters. The combination of the representation of required tools and quality control processes enables queries to identify required resources. Therefore, the mapping MDM is validated based on competence questions.

6 Conclusion

Evaluating needed capabilities to enable ADR processes by analyzing product features requires a connection between the domains. Therefore, approaches are suitable which use the process domain. The development of the mapping MDM is based on requirements, which were validated within an expert workshop. The workshop is also used to define the competence questions which are used to validate the MDM. The systematic literature analysis shows that existing approaches do not cover all processes and interdependencies which are necessary to derive the required capabilities. The identified approaches were expended to fulfill the requirements and answer the competence questions. Regarding RQ2 the needed expansion includes the representation of quality assurance processes, remanufacturing processes and the interdependence to the capabilities of the human resource. The developed MDM represents the ADR process, remanufacturing process and the environment. The modeled granularity in accordance to RQ1 represents the single instance of the required processes to, for example, screw the product. Therefore, the MDM enables a mapping of the product feature with the required process which then maps the capabilities. The resulting mapping MDM builds a logical structure using the underlying rules which was utilized in a created ontology. The links are used to perform queries that allow the required skills to be inferred for specific product features in form of connections. This relation makes it possible to check whether the required employees’ capabilities are available within the company. The paper thus lays the foundation for evaluating employees’ capabilities using knowledge-based AI methods such as knowledge graphs.

7 Acknowledgement

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – project no. 543074930 – as part of the SPP 2443 (Hybrid Decision Support in Product Creation, project no. 521148212, www.spp2443.de).

References

- [1] I. Gräßler, J. Pottebaum, P. Nyhuis, M. Schmidt, B. Grewe, F. Vollenkemper, T. Hesse and T. Meinecke, “Evolving Design for Assembly, Disassembly and Reassembly into a new paradigm: Design-for-Capabilities with Hybrid Decision Support as an enabler for circular products,” in *Stuttgarter Symposium für Produktentwicklung SSP 2025*: Stuttgart: Fraunhofer IAO, 2025, pp. 509–518.
- [2] J. Ast and P. Nyhuis, “Approach for determining functional flexibility of the workforce based on training losses and employee specific risks,” *Procedia CIRP*, vol. 107, pp. 839–844, 2022. DOI: <https://doi.org/10.1016/j.procir.2022.05.072>.
- [3] I. Gräßler, D. Roesmann and J. Pottebaum, “Human Factors in der integrierten Produktentwicklung,” *Zeitschrift für wirtschaftlichen Fabrikbetrieb*, vol. 118, no. 3, pp. 149–152, 2023. DOI: <https://doi.org/10.1515/zwf-2023-1029>.
- [4] I. Gräßler, J. Pottebaum, P. Nyhuis, R. Stark, K.-D. Thoben and P. Wiederkehr, “Hybride Entscheidungsunterstützung in der Produktentstehung,” *Industry 4.0 Science*, vol. 41, no. 1, pp. 18–25, 2025. DOI: <https://doi.org/10.30844/I4SD.25.1.18>.
- [5] M. M. Baysal, U. Roy, R. Sudarsan, R. D. Sriram and K. W. Lyons, “The Open Assembly Model for the Exchange of Assembly and Tolerance Information: Overview and Example,” in *Volume 4: 24th Computers and Information in Engineering Conference*: ASME/EDC, 2004, pp. 759–770.
- [6] Y. Sure, S. Staab and R. Studer, “Ontology Engineering Methodology,” in *Handbook on Ontologies*, S. Staab and R. Studer, eds.: Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 135–152.
- [7] R. K. Lee, H. Zheng and Y. Q. Lu, “Human-Robot Shared Assembly Taxonomy: A step toward seamless human-robot knowledge transfer,” *ROBOTICS AND COMPUTER-INTEGRATED MANUFACTURING*, vol. 86, 2024. DOI: <https://doi.org/10.1016/j.rcim.2023.102686>.
- [8] B. M. Hasan, “An Ontology based Approach for Knowledge Sharing between Product Design and Assembly Process Planning (APP),” in , 2024, pp. 326–331.
- [9] S. K. Das and A. K. Swain, “An ontology-based modelling and reasoning framework for assembly process selection,” *INTERNATIONAL JOURNAL OF ADVANCED MANUFACTURING TECHNOLOGY*, vol. 120, 7-8, pp. 4863–4887, 2022. DOI: <https://doi.org/10.1007/s00170-022-09002-9>.
- [10] A. Claeys, S. Hoedt, M. Schamp, H. van Landeghem and J. Cottyn, “Ontological Model for Managing Context-aware Assembly Instructions,” in , Vol. 51, 2018, pp. 176–181.
- [11] Q. W. Bao, G. Zhao, Y. Yu, S. Dai and W. Wang, “The ontology-based modeling and evolution of digital twin for assembly workshop,” *INTERNATIONAL JOURNAL OF ADVANCED MANUFACTURING TECHNOLOGY*, vol. 117, 1-2, pp. 395–411, 2021. DOI: <https://doi.org/10.1007/s00170-021-07773-1>.
- [12] Y. Ghanjaoui, J. H. Bussemaker, J. Biedermann and B. Nagel, eds., *An Ontology-Based approach for the co-development and optimization of aircraft cabin design and assembly architectures* (2024).
- [13] E. Järvenpää, P. Luostarinen, M. Lanz and R. Tuokko, “Presenting capabilities of resources and resource combinations to support production system adaptation,” in *2011 IEEE International Symposium on Assembly and Manufacturing (ISAM)*, 2011, pp. 1–6.
- [14] B. R. Ferrer, B. Ahmad, D. Vera, A. Lobov, R. Harrison and J. L. Lastra, “Product, process and resource model coupling for knowledge-driven assembly automation,” *AT-AUTOMATISIERUNGSTECHNIK*, vol. 64, no. 3, pp. 231–243, 2016. DOI: <https://doi.org/10.1515/auto-2015-0073>.
- [15] K. Y. Kim, D. G. Manley and H. Yang, “Ontology-based assembly design and information sharing for collaborative product development,” *COMPUTER-AIDED DESIGN*, vol. 38, no. 12, pp. 1233–1250, 2006. DOI: <https://doi.org/10.1016/j.cad.2006.08.004>.
- [16] DIN Deutsches Institut für Normung, DIN 8593-8:2003. *Fertigungsverfahren Fügen (8593)* (Berlin: DIN Media GmbH, 2003).
- [17] DIN Deutsches Institut für Normung, DIN 8588. *Fertigungsverfahren Zerteilen (8588)* (Berlin: DIN Media GmbH, 2013).
- [18] S. Chatti, L. Laperrière, G. Reinhart and T. Tolio, *CIRP Encyclopedia of Production Engineering* Berlin, Heidelberg: Springer Berlin Heidelberg, 2019.
- [19] DIN Deutsches Institut für Normung, DIN 8580. *Fertigungsverfahren (8580)* (Berlin: DIN Media GmbH, 2022).
- [20] DIN Deutsches Institut für Normung, DIN 8589. *Fertigungsverfahren Spanen (8589)* (Berlin: DIN Media GmbH, 2003).
- [21] I. Gräßler, D. Roesmann, C. Cappello and E. Steffen, “Skill-based worker assignment in a manual assembly line,” in *Procedia CIRP*, 2021, pp. 433–438.

Harnessing Tacit Knowledge from Sustainable Product Engineering through Artificial Intelligence

Iris Graessler¹, Johanna Mansheim^{1*}, Jan Pfeifer¹

¹Paderborn University, Heinz Nixdorf Institute, Chair for Product Creation, 33102 Paderborn

*Corresponding author. E-Mail address: johanna.mansheim@hni.uni-paderborn.de; Tel.: +49 5251/60-6252

Abstract: Tacit knowledge is particularly valuable in product engineering. Experiences reside in minds of employees and are not systematically documented but could have a significant influence on sustainable product engineering. Existing approaches do not offer sufficient support for harnessing technical tacit knowledge regarding sustainable product engineering. In this paper a three-step method is presented: The method contains knowledge acquisition, requirements extraction und requirements validation for quality assurance. Acquisition is performed by support artifacts like interview guidelines. Extraction is supported by Artificial Intelligence, converting statements into requirements. These requirements are validated using a specific questionnaire. The developed approach is applied in a funded project of the European Union (EU) with automotive industry partners. The method supports product engineers to collect tacit knowledge from experienced employees, transform unsystematic knowledge into standardized technical requirements and validate them. Application of this method enables creation of validated requirements that can be applied throughout the company and are not exclusively dependent on a single employee.

Keywords:

Tacit Knowledge, Sustainability, Product Engineering, Artificial Intelligence

1 Introduction

Leading industrial companies are placing rising emphasis on the sustainability of their products to address future developments. A significant proportion of a product's environmental impact is determined during engineering [1]. Formalized guidelines for engineering sustainable products are provided by ISO 14000 series such as Life Cycle Assessment in ISO 14040, allowing industrial companies to evaluate their products' sustainability [2]. Companies build their specific implementation frameworks based on such general standards and guidelines. Experiential knowledge is held by individual employees in product engineering. Experiential knowledge is undocumented and unsystematic [3]. Experiential knowledge is considered Tacit Knowledge (TK), and will always remain only partially usable for engineering sustainable products as it cannot be externalized completely [3]. Extracting TK provides support for product engineering for recurring challenges such as increasing complexity of products and changes due to new technologies, especially regarding sustainability [4]. To systematically counteract loss of TK, extraction of TK needs to be conducted in a structured and scalable manner. To be able to use extracted TK in engineering, its quality must be ensured.

To address these engineering challenges, a comprehensive approach is needed, assuming great potential in the utilization of Artificial Intelligence (AI) through Large Language Models (LLM). Currently, various individual methods for extracting TK exist in research and are already established in practice. These methods are formulated in general terms, not adapted to a specific technical topic and do not include quality assurance [5]. At the same time, LLM-based AI tools evolve technologically with great

potential to process unstructured information like natural language and even hand-written notes. The paper at hand is based on the assumption that this potential can help to overcome the high efforts of current practices in analyzing TK documentations. Based on the identified challenges and potentials, three research questions (RQ) are formulated:

RQ1: *What approaches are most beneficial for acquiring tacit knowledge in product engineering?*

RQ2: *Which methods and extracts from existing approaches need to be adapted to incorporate sustainability in engineering context?*

RQ3: *Which constituents are required in a) a questionnaire for acquisition and b) a prompt for extraction to generate automatized input to requirements specifications from technical tacit knowledge?*

To answer these RQs, the paper provides a brief overview of background in terms of state of the art, before introducing the applied scientific approach. The method for harnessing TK from sustainable product engineering and utilizing it in a requirements engineering project is proposed as a combination of acquiring, extracting and validating TK including results of a systematic literature review, method description and presentation of artifacts. An application with sample data and contributions by 20 experts is used for initial validation of the method, leading to conclusion and outlook.

2 State of the Art

Sustainable product engineering considers the entire product life cycle [2]. Products have various environmental impacts, which accumulate over their product life cycle [2]. Environmental impacts can be divided into different impact categories such as climate change, acidification and human toxicity [1]. Environmental impacts must be already considered during the engineering phase [1]. To conduct sustainability assessments and take its results into account in engineering, a high amount of technical expertise is required [1]. This expertise often refers to the unconscious part of a person's knowledge and is considered Tacit Knowledge (TK). TK is not documented and is therefore intangible [6]. TK rises from experience and is expressed through intuitive actions [7]. These experiences allow future actions to be improved [7]. Verbalization of TK is difficult, partially even impossible [8]. Relevant examples in product engineering are the understanding of an issue and the development of a problem-solving strategy based on engineering know-how through personal knowledge and experience [4]. These problem-solving strategies are often required to make appropriate decisions in engineering processes [4]. In comparison to TK, explicit knowledge is communicated and documented. Explicit knowledge can be shared and is detachable from the person. For companies, both types of knowledge are crucial to success and of strategic importance in product engineering, for example for defining requirements [7]. The processing of TK can be very time-consuming. To utilize support from Artificial Intelligence (AI) regarding TK, machine-readable and, thus, explicit artifacts are needed. AI enables technical systems based on algorithms to solve tasks that require intelligence [9]. Generative AI is a subfield of AI that involves creation of new data. LLMs provide one framework for generative AI in the field of human and technical language processing. The answers of an LLM are generated based on statistical methods and training data to provide logical answers in terms of syntax and semantics [10].

3 Scientific Approach

To address these RQs, a five-step research approach is formulated. Through a systematic literature review (SLR), related work on harnessing TK in the context of product engineering is identified (1). Relevant approaches for the acquisition of TK are reflected in generic knowledge management literature and evaluated based on defined criteria in the context of sustainability (2). Based on recurring elements in related work, a method for comprehensive extraction and validation of TK is developed (3). The

developed method is applied in an EU funded project with twenty experts from the automotive sector (4). Finally, the method is validated against the defined criteria (5).

4 Method for acquiring, extracting and validating Tacit Knowledge

This section delineates outcomes of the first three steps of the scientific approach. Results are divided into different subsections. Results of the SLR are presented and evaluated. The developed method and different artifacts are introduced and applied with twenty industry experts.

4.1 Systematic literature review

The SLR is used to identify relevant approaches for acquiring and extracting TK in product engineering. The SLR approach is based on the PRISMA schema [11]. The search string combines key words based on standard literature with Boolean operators “AND” and “OR”. This search string consists of four areas: method, TK, knowledge extraction and engineering. Scopus and Web of Science databases were selected for their engineering-related content. Google Scholar is added to expand scope of results and ensure completeness. To supplement findings of other databases, number of documents from Google Scholar is limited to 100. The findings and search string of SLR are shown in Figure 1.

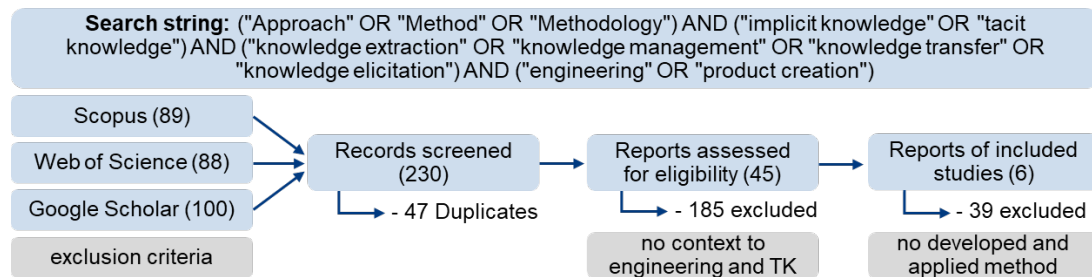


Figure 1: Results of the SLR

A total of 277 findings were retrieved. 47 duplicates were removed from results and 230 records were screened based on their titles and abstracts. 45 reports were assessed for eligibility in detail. The publications fulfil criteria such as context to TK, context to engineering and paper in English. Furthermore, an eligible publication needs to describe a method or approach that is developed, applied or validated. These criteria result in a total of six publications, whose approaches are examined in detail.

4.2 Evaluation of approaches and methods

The following evaluation considers the various knowledge acquisition and extraction methods identified in the six approaches. Overall, approaches have different orientations like development of a collaborative framework and application of a specific method. CURRAN et al. [12] develop a method for systematically reusing knowledge in a knowledge management system. SHANKAR et al. [13] define a framework based on the SECI model (socialization, externalization, combination, internalization) for transferring knowledge within companies. Both approaches refer to different methods such as expert interviews and observations for knowledge extraction, but do not provide detailed process models. Linking and application of methods remain unclear. KWONG and LEE [14] use narrative interviews to gather knowledge from experts in airline industry. WHYTE and CLASSEN [15] utilize storytelling, which is similar in application to narrative interviews. LIU et al. [16] use interviews that contain defined scenarios to obtain problem-solving suggestions from experts. This method is similar to the established critical incident technique. GARCIA-PEREZ and AYRES [17] develop a collaborative approach to knowledge representation. The first step in the preparation process is to conduct interviews to identify relevant use cases. In the next step, these use cases with unclear knowledge elements are addressed in group discussions.

In addition to methods identified through SLR, references in related work are reflected and complemented by further methods. While SLR results are related to the overall concept proposed in this paper, these methods for knowledge acquisition are derived from standard literature and are well established in qualitative research. The selected methods are shown in Table 1.

Table 1: Selected methods for acquiring tacit knowledge

Method	Description
Structured interview	stringent answering of questions from the prepared interview guideline, which cannot be deviated from [18]
Semi-structured interview	prepared interview guideline, which can be deviated from, with a mixture of structure and openness and a flexible focus of the questions [18]
Unstructured interview	open interview without interview guidelines [18]
Narrative interview	interview with open questions about personal experiences and experiential knowledge in detail; interviewee takes up most of the talking (narrative flow) [5,6]
In-depth interview	interview with comparatively more interventions by the researcher to identify rules, actions, types of actions and typologies; meaningful content must be determined in advance [18]
Critical incident technique (CIT)	interview in which the interviewee is asked specific questions about situations they have experienced to obtain a detailed description of the situation, behavior and resulting outcome; particularly positive and negative experiences [19]
Thinking aloud	thoughts are expressed parallel with a primary task to provide insight into problem-solving behavior of a test subject; task is focus and verbalization occurs alongside task [5]
Group discussion	discourse with six to ten participants to reflect different perspectives and to gather collective knowledge; researcher takes role of moderator [5]
Questionnaire	predefined number of questions sent to respondents; different types of questions, e.g. open questions with free text and questions answered by rating on a scale [20]

The list of methods provides a broad overview of methods for TK acquisition. The list of methods is evaluated with members of the EU funded project based on six criteria. Criteria include repeatable results (C1), text-based results (C2), anonymized results (C3), inherent process model (C4), low-effort adaptability (C5) and quality assurance of knowledge (C6). Degree of fulfilment of criteria is divided into completely fulfilled, partially fulfilled and not fulfilled. The results of the method shall be reproducible. C1 is completely fulfilled if results achieve the same structure and subject areas after repeated application. The experts' knowledge and the resulting outcomes of the method shall be represented in text form. C2 is completely fulfilled if results are available in text form immediately after application as an export. The criterion is partially fulfilled if results can be converted into text form (with an additional step). Sharing TK shall be possible anonymously to prevent participants from being influenced and potentially holding back. Results shall also be archived anonymously and do not allow any conclusions to be drawn about individuals. The criterion can be considered completely fulfilled if the results can be collected and archived anonymously to fulfill C3. The collection and processing of TK shall follow a methodical and inherent process model. C4 is completely fulfilled if the method specifies a systematic procedure for collection and processing of TK. Artifacts of the method shall be adapted with minimal effort. Additional questions and topics can be added or deleted. C5 is completely fulfilled if the method can be supplemented without changing the current version and with minimal resources. The knowledge gained shall be checked for correctness. Feedback shall be provided to pass on high-quality knowledge. C6 is fulfilled if the method provides a way of knowledge validation.

The results of evaluation of methods based on the criteria are shown in Table 3. The evaluation shows that all methods carry opportunities with regard to individual criteria. Some, like structured interviews, can be interpreted as more frequently applicable, others are adequate for very specific purposes like analysis of nonverbal actions. To cover all these characteristics, methods are combined in the developed

approach. The method shall be adaptable to different setups in companies, with specific measures on knowledge management, business intelligence or quality management. None of the methods consider quality assurance of knowledge in their individual application. Quality and accuracy of knowledge is highly important for incorporating TK into technical processes in product engineering. By combining methods, control of TK should be considered and made possible. The methods that completely and partially fulfil most of the criteria are questionnaires, semi-structured interviews, narrative interviews and CIT. Based on the evaluation, these methods are selected as most beneficial for further use and considered suitable as a core of the TK acquisition approach (RQ1).

Table 3: Evaluation of selected methods

Methods/Criteria	C1	C2	C3	C4	C5	C6
Structured interview	●	◐	●	◐	○	○
Semi-structured interview	◐	◐	●	◐	●	○
Unstructured interview	○	◐	●	○	●	○
Narrative interview	◐	◐	●	◐	●	○
In-depth interview	◐	◐	●	◐	◐	○
CIT	◐	◐	●	◐	●	○
Thinking aloud	◐	◐	●	◐	◐	○
Group discussion	○	○	○	◐	●	○
Questionnaire	●	●	●	◐	●	○
Key: C = Criteria ● = completely fulfilled ◐ = partially fulfilled ○ = unfulfilled						

4.3 Method for harnessing Tacit Knowledge in Sustainable Product Engineering

With reference to RQ2, a three-step approach is developed based on results of the SLR (see Figure 2). The method contains knowledge acquisition, requirements extraction und requirements validation. The three phases are performed sequentially. In **knowledge acquisition**, experts are interviewed using semi-structured interviews according to KATENKAMP [18]. Openly formulated questions are asked to collect knowledge about sustainable product engineering. To extract relevant data and information within product engineering, various topics like materials and assembly are addressed in the interview guideline. In addition to technical categories, narrative questions and CIT questions are used at the beginning of interview. These two methods were selected as being suitable for use alongside the semi-structured interview. Narrative questions and CIT intend to encourage interviewees to talk freely about their experiences and prepare them for specific questions. Result of the first phase is documented machine-readable TK. TK can be recorded as transcripts in individual documents and a table. TK takes on an initial explicit character through documentation but remains unstructured. Statements may be duplicated and contradictory, which means that recorded knowledge should not be archived and used in this form.

In **requirements extraction**, a LLM converts acquired TK into requirements. Requirements are structured according to the SOPHIST requirements templates [21]. The detailed prompt (cf. parallel work in [22]) is used to provide context to the AI and to explain the task and response format in detail. For example: “In sustainable product engineering, screw connections should be used instead of riveted connections.”. Attributes such as “ID”, “domain of expertise” and “lifecycle phase” are assigned to the requirements. Use of AI ensures rapid processing and sorting of unstructured data. The level of expertise can be used as a basis for initial prioritization of requirements: Requirements obtained from statements of the interviewees with expert knowledge and extensive knowledge are more likely to be correct; requirements obtained from statements made by interviewees with basic knowledge, little knowledge and no knowledge require more detailed validation. In **requirements validation**, requirements are

checked for validity using a questionnaire as an evaluation form. Through a five step Likert scale [20] identified requirements are validated for plausibility and cross-employee knowledge is utilized. Application of the developed approach facilitates conversion of TK into validated requirements. Validated requirements can be transferred to a database in a defined format.

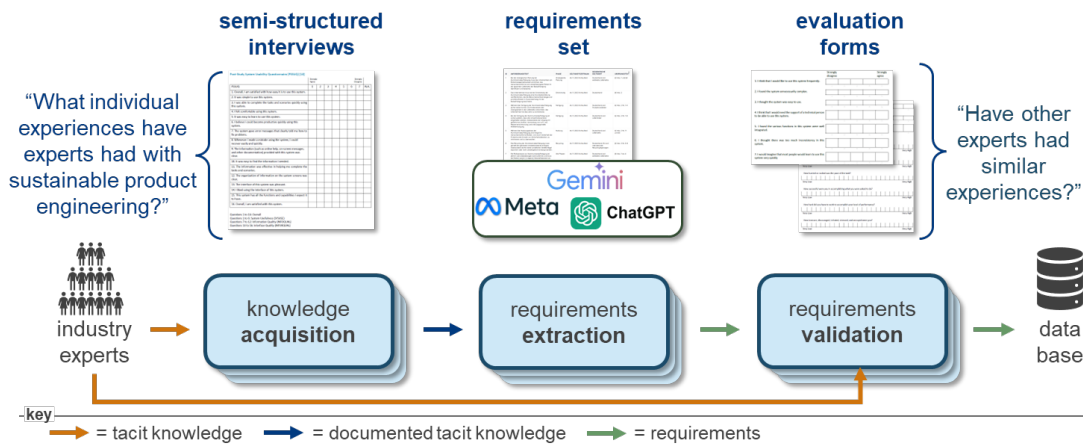


Figure 2: Developed method for harnessing tacit knowledge

4.3.1 Interview guideline for TK acquisition

An interview guideline is developed as part of the first step “knowledge acquisition”. The interview guideline enables systematic querying of TK in expert interviews. Its first part consists of a brief introduction of interviewer and context. In the second part, CIT [19] and narrative questioning technique [5] are applied. These techniques get experts into narrative flow and into thinking intensively about their actions. In the third part, the interview guideline is created taking technical guidelines into account. Important areas of product engineering are considered in relation to sustainability. The main-feature list of VDI/VDE 2206 [23] contains various relevant criteria for product engineering, such as interfaces, manufacturing and recycling. The interview guideline is divided into areas according to the main features. In the interview, for example, experiences with disassembly-friendly joining methods can be requested under topic of ‘assembly’. Within these areas, explicit and targeted experiences are queried in comparison to the first part. Areas from the main-feature list are supplemented by VDI 4800 [24] and VDI 2243 [25]. In areas of materials, manufacturing and assembly, for example, questions are asked on topics of lightweight construction, reparability and modularization. These topics are taken from VDI 4800 [24], which covers resource efficiency and resource conservation. The area of recycling can be enriched with further questions from VDI 2243 guideline [25] on recycling-oriented product engineering. Questions in each area are preceded by a five-point Likert scale [20]. Interviewees use this scale to assess their level of knowledge. This scale includes the levels “no knowledge”, “little knowledge”, “basic knowledge”, “extensive knowledge” and “expert knowledge”. This assessment enables classification and evaluation of knowledge elements during requirements extraction. To better illustrate technical questions, they are posed using a product example from industry partner. An excerpt from the interview guideline is shown in Figure 3.

Interview guidelines can be converted into a questionnaire to make TK extraction less resource intensive. Such questionnaire can therefore be conducted without another person acting as an interviewer. Questions from the interview guideline can be transferred verbatim into a questionnaire. A free text field is provided as a response option. This allows experts to answer questions without restriction. The Likert scale for assessing expertise can also be included in a questionnaire.

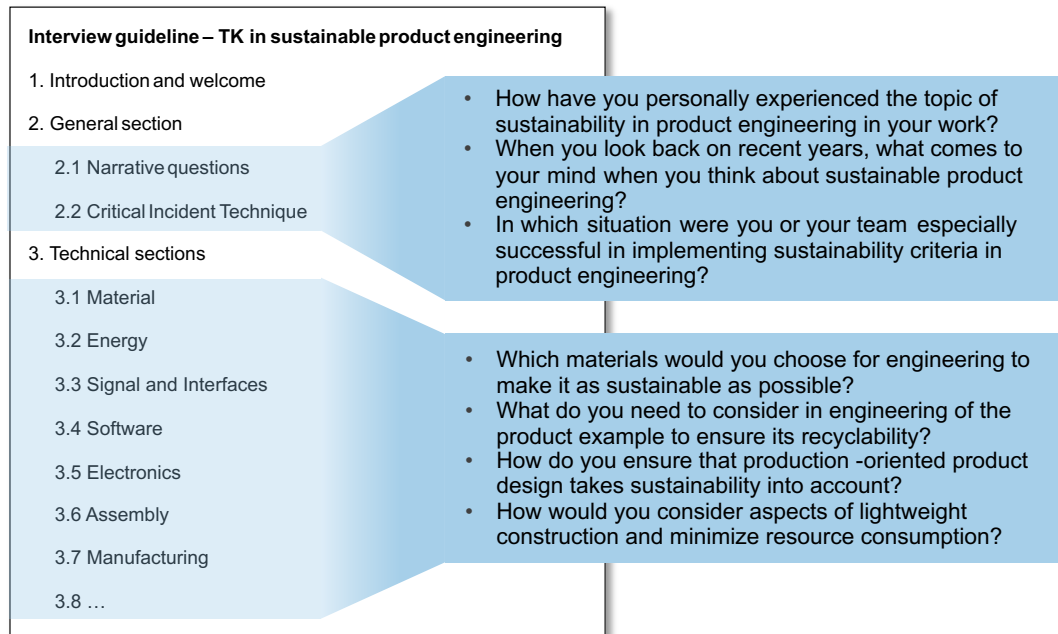


Figure 3: Exemplary questions from the interview guideline

4.3.2 Prompt for requirements extraction

In the next step, requirements are extracted from documented TK (cf. RQ3). An Excel spreadsheet serves as the source document. Often even this means a realistic type of data source; in other cases, requirements are exported from requirements management tools like DOORS or Polarion to Excel. Extraction is performed using an LLM. Due to large volume of textual data, use of an LLM saves time during text analysis. ChatGPT 5.0 is selected as LLM based on previous research with technologies like support vector machines and transformer models [26], resulting selection criteria, literature research and technical tests. The query to ChatGPT is based on a prompt. This prompt is created in advance and refined through tests with ChatGPT. Revisions like precise columns of the table to be analyzed and sample formulations for requirements are made. The prompt is divided into sections that are intended to provide ChatGPT with adequate previous knowledge to solve the task. The CO-STAR framework according to PENG et al. is used to structure the prompt and is extended to include role [27]. To ensure that individual parts are clear for AI despite long text, hashtags are used [28]. The structure of the formulated prompt is shown in Figure 4.

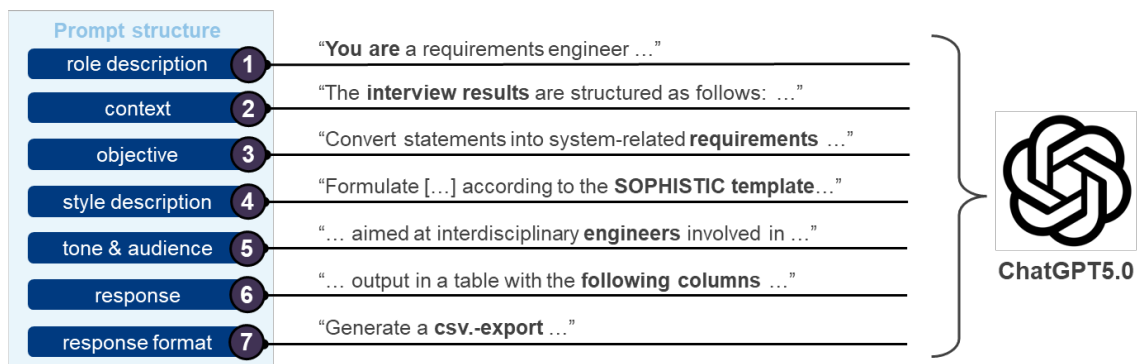


Figure 4: Structure of the formulated prompt

First, a "role description" is added to the framework. It describes the person represented by AI. "Context" provides AI with background information about the scenario. "Objective" precisely defines the task for AI model so that AI can best meet expectations. Next is the "style description" to define a

template. “Tone“ determines type of wording, e.g. formal. The description of an “audience” explains who the answer given is aimed at to meet expectations. “Response” defines exactly what the output should look like. Response can be, for example a table and a specific file format. ChatGPT is utilized as an agent performing requirements engineering tasks. To provide context for AI, the prompt states where statements in the resulting table (see “response format” csv in Figure 4) come from and how the table is structured. Further refinement of this table in the final step can be achieved by including “task”, “level of expertise”, “life cycle phase” and “style”. Task describes which columns should be analyzed and provides an example of how to derive requirements. In addition to requirements, level of expertise of interviewee and life cycle phase to which the requirement relates should be specified to influence solutions of AI in a targeted manner. The individual life cycle phases are used in accordance with GRAESSLER and POTTEBAUM [29]. Style defines the appearance of requirements using the SOPHIST template [21] and specifies criteria that requirements should meet. Tone and audience help to tailor the response to a specific use case. The audience ‘engineers’ generates greater attention to technical terms and a technical perspective. Last section of the prompt explains format of desired responses. Requirements should be displayed in a table. The individual columns with desired content are listed. This allows AI to output requirements directly in the expected form to be used for evaluation.

4.3.3 Questionnaire for requirements validation

A questionnaire is used to check requirements in terms of correctness. The extracted requirements are listed in this questionnaire and grouped by topics like assembly. Experts can evaluate requirements using a five-point Likert scale [20]. Objective correctness of requirements is classified using the levels “I completely agree”, “I agree”, “I disagree”, “I completely disagree” and “I don't know”. A text box below allows additional information about requirements to be added and corrections to be made. These corrections are collected after the questionnaire has been completed and used to revise initial requirements. Requirements can be weighted based on evaluations. Requirements that are rated as incorrect by majority of experts should be removed from collection. Requirements about which majority of experts are uncertain should be reviewed through research and consolidation with additional experts. Requirements that are rated as correct by majority of experts can be included in a database.

5 Initial validation of the developed approach

The method described in section 4.3 is applied in an EU funded project. Semi-structured interviews based on the developed interview guideline¹ are conducted with twenty selected experts. During application, diversity of interview results highlighted importance of flexibility in the interview guideline. Results are then checked by an employee of the industry partner and revised if necessary to ensure that no sensitive data is disseminated. Results are then compiled in form of an Excel spreadsheet and can be evaluated. Data is anonymized to prevent any conclusions about specific individuals being drawn from an evaluation of the table. Extraction of requirements using AI worked after revisions to the prompt. ChatGPT 5.0 evaluates the table based on the final prompt and formulates requirements.

Table 4 shows five examples of requirements that are derived from the statements made in interviews. They are all related to sustainability and refer to different life cycle phases. Requirements provide general constraints for engineering. The requirements do not contain any specific technical details such as length of a component and specific materials. This is due to imprecise information provided by interviewees. In the interviews, participants only provide general context on sustainable engineering. If interviewees express themselves unclearly, AI can hardly draw logical conclusions from their statements. Accordingly, a standardized response format is necessary. This provides AI with uniformly

¹ <https://zenodo.org/records/18360632>

structured responses, which AI can analyze much more precisely. Validation of requirements using questionnaire was also successful. Questionnaire is implemented in a digital version using Microsoft Forms. Results allowed requirements to be grouped. Incorrect requirements are excluded immediately.

Table 4: Derived requirements from the interview statements

ID	Requirement	Life cycle phase	Level of expertise
1	The system should quantify energy usage during operation and output energy usage as a parameter.	Use	High
2	The system should only use materials whose sustainability characteristics have been verified in the material database.	Engineering	Expert
3	The system must be designed in a way that allows for the segregated separation of the main components by type for material recycling.	Decommissioning	High
4	The system must consist mainly of recyclable and corrosion-resistant materials whose surface is suitable for low-pollutant disposal and recycling.	Engineering	Basic
5	The system must avoid permanent connections such as adhesive bonds.	Engineering	High

The developed method meets the defined criteria from 4.2. The prescribed process involving various artifacts such as the prompt ensures that results are presented and evaluated in the same way. The interview guideline ensures that all experts are asked same set of questions, thereby supporting consistency across interviews (cf. C1). Extracted TK can subsequently be presented in text-based formats, such as tables or transcripts, enabling further processing by AI (cf. C2). To promote unbiased feedback, the questionnaire is conducted in a fully anonymous manner, and only a neutral interviewer is present during interviews (cf. C3). Overall, the approach follows a structured three-step process (cf. C4), while remaining flexible, as questions and additional topics can easily be incorporated into both interview guideline and questionnaire (cf. C5). Finally, a questionnaire is used to validate identified requirements and to ensure accuracy (cf. C6). The approach was successfully applied. TK extraction can be conducted either by semi-structured interviews or questionnaires. This approach allowed the concept to be tested through its application. Improvements like a standardized response format and more precise technical questions in interviews are elaborated to address problems encountered during application.

6 Conclusion and Outlook

In this paper, a method is described to facilitate conversion of tacit knowledge into validated requirements. Methods for acquiring and extracting TK are analyzed and evaluated. Most suitable methods for acquiring and extracting TK are semi-structured interviews advanced with narrative questions, CIT and questionnaires. An approach is developed from evaluation results using these methods. In the first step, TK is collected using an interview format. Technical guidelines are consulted to focus the approach specifically on sustainable product engineering. To ensure the accuracy of requirements in the final step, a feedback mechanism in form of a questionnaire is proposed. To enable an initial classification of requirements, interviews include an individual assessment of experts' knowledge on each topic area. Extracting requirements with AI offers a low-effort alternative to manually reviewing large amounts of data. AI requires a detailed prompt including sufficient context and a precise description of the expected answer to derive requirements. Overall, LLMs offer an efficient way to extract requirements. However, these requirements should not be accepted without validation. Future research will explore expanded use of LLMs and GPTs, for example through chatbot-conducted interviews, where this further automation enables scalability in companies with many employees and reduces effort required to extract TK. The developed approach supports engineers in sustainable product engineering by providing systematic technical knowledge that has been acquired through TK.

7 Acknowledgements

The research leading to these results has received funding from the European Union's Horizon Europe Program under the Digi4Circular Project, grant agreement n° 101177586. Results are reflected by the scope of DFG SPP2443 (coordination, PN521148212).

References

- [1] T. C. McAloone and D. C. A. Pigosso, "Ökodesign," in Pahl/Beitz Konstruktionslehre, B. Bender and K. Gericke, eds.: Berlin, Heidelberg: Springer, 2021, pp. 975–1021.
- [2] ISO 14040, "Environmental management - Life cycle assessment. Principles and framework", Geneva, 2006.
- [3] F. Lehner, "Wissensmanagement. Grundlagen, Methoden & technische Unterstützung", 8. Aufl., München: 2025.
- [4] H. Wang and G. Peng, "Collaborative Knowledge Management Through Product Lifecycle. A Computational Perspective", 1st ed. 2023, Singapore: Springer Nature Singapore; Imprint Springer, 2023.
- [5] G. Mey and K. Mruck, eds., "Handbuch Qualitative Forschung in der Psychologie. Band 2: Designs und Verfahren", 2. Auflage, Wiesbaden: Springer Fachmedien, 2020.
- [6] U. Reisach, C. B. Nakhosteen, C. Erlach and W. Orians, "Wissenstransfer bei Fach- und Führungskräftewechsel. Erfahrungswissen erfassen und weitergeben", 2. Auflage, München: Hanser, 2025.
- [7] C. Erlach, A. Mittelmann, C. B. Nakhosteen, M. Della Schiava and G. Terhoeven, "Praxis Wissenstransfer. Erfolgsstrategien und bewährte Lösungswege", München: Hanser, 2025.
- [8] S. Porschen-Hueck, "Austausch impliziten Erfahrungswissens. Neue Perspektiven für das Wissensmanagement", Wiesbaden: VS Verl. für Sozialwissenschaften, 2008.
- [9] S. Matzka, "Künstliche Intelligenz in den Ingenieurwissenschaften. Maschinelles Lernen verstehen und bewerten", Wiesbaden: Springer Vieweg, 2021.
- [10] J. Fuchs, C. Helmerich and S. Holland, "Transforming System Modeling with Declarative Methods and Generative AI", in AIAA SCITECH 2024 Forum: Reston, Virginia: American Institute of Aeronautics and Astronautics, 2024.
- [11] Page, M. J. et al., "Updating guidance for reporting systematic reviews: development of the PRISMA 2020 statement", *Journal of clinical epidemiology*, vol. 134, pp. 103–112, 2021.
- [12] R. Curran, W. J. Verhagen and M. J. van Tooren, "A multidisciplinary implementation methodology for knowledge based engineering: KNOMAD", in *Expert Systems with Application*, 2010.
- [13] R. Shankar, S. Acharya and A. Baveja, "Soft-system knowledge management framework for new product development", in *Journal of KM*, 2009.
- [14] E. Kwong and W. B. Lee, "Knowledge elicitation in reliability management in the airline industry", in *Journal of KM*, vol. 13, no. 2, pp. 35–48, 2009.
- [15] G. Whyte and S. Classen, "Using storytelling to elicit tacit knowledge from SMEs", in *Journal of KM*, 2012.
- [16] L. Liu, Z. Jiang and B. Song, "A novel two-stage method for acquiring engineering-oriented empirical tacit knowledge", *International Journal of Production*, 2014.
- [17] A. Garcia-Perez and R. Ayres, "Collaborative development of knowledge representations - A novel approach to knowledge elicitation and transfer", in 9th European Conference on KM, 2008, pp. 217–224.
- [18] O. Katenkamp, "Implizites Wissen in Organisationen. Konzepte, Methoden und Ansätze im Wissensmanagement", Wiesbaden: VS Verl. für Sozialwiss, 2011.
- [19] H. Hughes, "Critical Incident Technique", in *Exploring methods in information literacy research*, S. Lipu, K. Williamson and A. Lloyd, eds.: Wagga Wagga, N.S.W: Centre for Information Studies Charles Sturt University, 2007.
- [20] N. Baur and J. Blasius, eds., "Handbuch Methoden der empirischen Sozialforschung", 3. Auflage, Wiesbaden: Springer VS, 2022.
- [21] R. Kluge, "Die SOPHISTen Schablonen für alle Fälle", 1.th ed. Nürnberg: SOPHIST GmbH, 2013.
- [22] I. Graessler and J. Pfeifer, "Performance evaluation of Large Language Models in extracting requirements sets from technical documents", *International Symposium on Hybrid Intelligence in Product and Production Engineering*, 2026.
- [23] VDI/VDE 2206, "Development of mechatronic and cyber-physical systems", Berlin: Beuth Verlag, 2021.
- [24] VDI 4800, "Ressourceneffizienz und Ressourcenschonung. Methodische Grundlagen, Prinzipien und Strategien", Düsseldorf: Beuth Verlag, 2023.
- [25] VDI 2243, "Recycling-oriented product development", Düsseldorf: Beuth Verlag, 2002.
- [26] I. Gräßler, D. Preuß, L. Brandt and M. Mohr, "Efficient Extraction of Technical Requirements Applying Data Augmentation", in 2022 IEEE International Symposium on Systems Engineering (ISSE): IEEE, 2022, pp. 1–8.
- [27] J. Peng, H. Zhongyuan, Z. Huan, Y. Jingyan and L. Chang, "A multilingual text detoxification method based on few-shot learning and co-star framework", *CLEF 2024*, 2024.
- [28] S. Lemeš, "Prompt Engineering", in *Artificial Intelligence in Industry 4.0: The future that comes true: Academy of Sciences and Arts of Bosnia and Herzegovina*, 2024, pp. 159–170.
- [29] I. Graessler and J. Pottebaum, "Generic Product Lifecycle Model: A Holistic and Adaptable Approach for Multi-Disciplinary Product–Service Systems", *Applied Sciences*, vol. 11, no. 10, p. 4516, 2021.

Systematic Review of Engineering Simulation Models for Life Cycle Assessment

Iris Graessler¹, Simon Aydin^{1*}, Sven Rarbach¹

¹Paderborn University, Heinz Nixdorf Institute, Chair for Product Creation, 33102 Paderborn

*Corresponding author. E-Mail address: simon.aydin@hni.uni-paderborn.de

Abstract: Manufacturing companies increasingly integrate the strive towards sustainability into product engineering to reduce resource consumption and environmental impacts. A substantial share of a product's environmental performance is determined during engineering. Early assessments rely on generic data, which is gradually replaced by more concrete simulation and primary data as product maturity increases. The effective use of engineering simulation models for Life Cycle Assessment (LCA) therefore remains a central challenge under heterogeneous, distributed and rapidly evolving data. This article presents a systematic review of product engineering approaches for integrating simulation models into LCA. A seven-step research approach is applied to identify, categorize and evaluate existing approaches across CAx tools. Relevant simulation parameters for sustainability assessment, including material properties, process descriptors, consumables and waste streams, are identified and analyzed. Based on identified data characteristics, the potential of Data Science and Artificial Intelligence methods for LCA data preparation is assessed. A conceptual knowledge-graph-based decision support approach is derived to enable structured integration, traceability and reuse of simulation data for sustainability assessment. The approach is evaluated using criteria from literature. Structured integration of simulation data improves early-stage data quality and supports decision-making in sustainable product engineering.

Keywords:

Product Engineering, Simulation Model, Sustainability, Life Cycle Assessment

1 Motivation

Leading companies are progressively engineering their products to reduce resource consumption and environmental impact. A significant amount of a product's environmental impact is determined during engineering. Maximizing the scope for design in engineering necessitates making valid estimations about the product's sustainability throughout the entire life cycle [1]. This assessment requires consolidating a wide variety of information and data from different IT-systems. In Life Cycle Assessment (LCA), generic database values are used to make initial estimates within the product life. These values are gradually replaced by more concrete simulation and primary data as the product matures. Primary data consists of real data from production processes and other sources. Due to high volume, diversity and distribution across different IT-systems, applicability of the data is limited. Using simulation data carries the potential to improve the quality of sustainability assessments at an early stage. Simulations currently used for sustainability assessments primarily describe the environmental impacts of processes involved in product life. Integrating existing engineering simulation models into tools such as Computer-Aided Design (CAD), Computer-Aided Engineering (CAE), Computer-Aided Manufacturing (CAM) and Finite Element Analysis (FEA) improves the quality of LCA and utilization of simulation for hybrid decision making in sustainable product creation (see Figure 1).

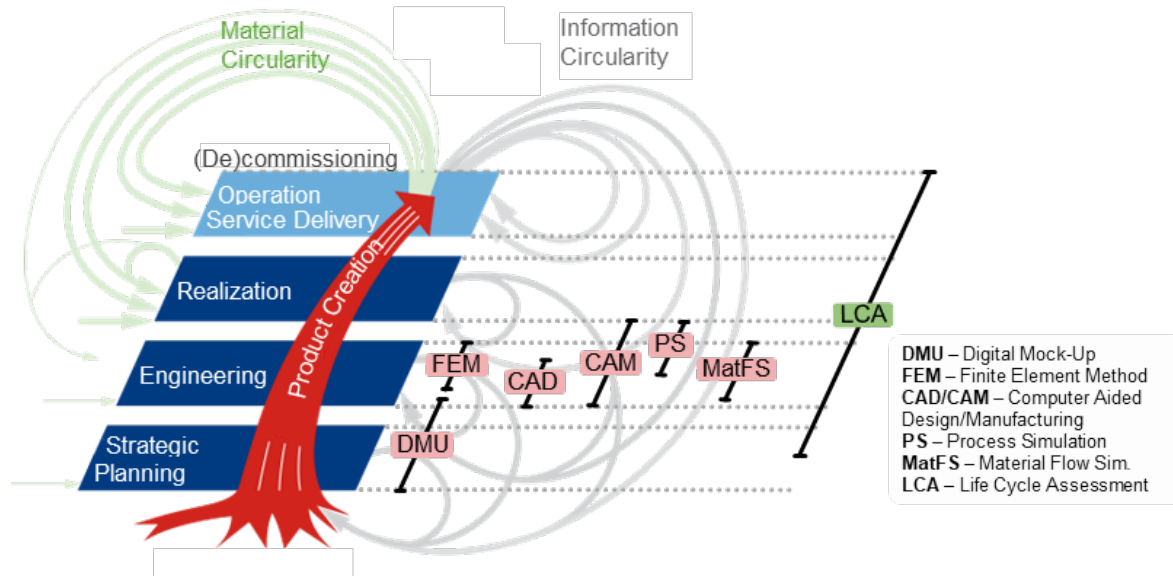


Figure 1: Potentials of simulation models in product engineering for dynamic Life Cycle Assessment in accordance with [2]

Current considerations of dynamic LCA in product engineering only marginally consider established simulation models, but focus on individual models and consider exemplary model elements. Established reviews prioritize process simulations over analysis of engineering simulation models. Current approaches have not yet been sufficiently adapted to recent technologies, such as Data Science and Artificial Intelligence (DS/AI), which offer enhanced potential to handle data for sustainability assessments. Considering these limitations, a review of approaches for integrating simulation models into LCA is conducted, analyzing data regarding sustainability and the potential of established DS/AI approaches. Two research questions (RQ) are formulated:

RQ1: Which product engineering approaches exist for integrating simulation models into LCA?

RQ2: Which existing DS/AI methods can potentially be utilized for data preparation for LCA?

The aim of this paper is to systematically compare existing approaches for integrating simulation models into LCA. Therefore, a conceptual approach for their implementation is derived and initially validated in an experimental laboratory environment. Section 2 presents the applied research approach. This is followed in section 3 by a state-of-the-art review. Section 4 describes the systematic analysis, evaluation and synthesis of identified approaches. Based on this analysis, parameters from existing methods are identified and a new conceptual approach for integrating simulation data into LCA is developed.

2 Scientific approach

The scientific approach is based on the Design Research Methodology (DRM) and focuses on descriptive research [3]. A seven-step process, derived from the DRM's four phases, is used to systematically analyze the integration of simulation data into LCA. As part of the research project Decide4ECO funded by the German Federal Ministry of Economy and Energy (BMWE) relevant decision points and success factors are identified (1). Based on the PRISMA scheme [4], a systematic literature review is conducted to identify approaches in which simulation model results are used for LCA (2). The identified approaches are analyzed regarding integration of engineering simulation models into LCA (3) and concerned parameters (4). Based on characterizations of data and information, potentials for using DS/AI methods are examined (5). A conceptual implementation of a knowledge graph for structuring data is carried out in a production laboratory environment (6). Based on this preparation, the resulting new conceptual approach is validated (7).

3 State of the art

This section introduces key concepts and principles of sustainable product engineering, with focus on simulation models across the product life cycle, data science and data space. Manufacturing companies increasingly account for ecological sustainability alongside functional and economic requirements in product engineering [5]. Early engineering decisions define key product characteristics. These decisions largely determine life-cycle environmental performance [5,6]. **Sustainability integration into product engineering** is supported by methods such as LCA. These methods enable the evaluation of environmental performance and circularity despite limited data availability [7]. As early design decisions largely determine a product's environmental footprint, systematic and quantitative assessment methods are required to evaluate sustainability impacts [6]. According to the ISO 14044, **LCA** is a standardized method for environmental sustainability assessment across the product life cycle [8,9]. This structured procedure ensures a transparent, consistent and reproducible assessment of environmental impacts. Potential environmental impacts are calculated based on life cycle inventory (LCI) data describing emissions and resource consumption. This enables engineers to make informed decisions regarding sustainable products [10]. Reliable LCA results depend on trustworthy, high-quality **LCI data**. Such data requires preparation involving complex transformations and quality checks. Availability of such data is currently limited by confidentiality and heterogeneous formats [11]. Within the LCA domain, data encompasses heterogeneous forms ranging from primary, site-specific measurements to aggregated secondary datasets, processed data products and non-standard outputs such as analysis protocols or software tools. **Data spaces** provide a viable approach to address challenges regarding inventory data. The technology is a key enabler of secure, sovereign and interoperable data exchange across manufacturing and supply-chain networks, helping to overcome fragmented and siloed data structures [12]. Integrating secure, sovereign data transfer enables collaborative LCA calculations across value creation networks [13]. **Simulation models** in tools like CAD, CAE, CAM, CAPP (Computer-Aided Process Planning) extend modeling and simulation to support sustainable product engineering. Customization and sustainability variables are integrated into design and process planning within these tools, enabling assessment of manufacturing energy use, emissions, waste generation and production flexibility. Knowledge-based software extensions increase interoperability among CAX tools to optimize engineering times and enable manufacturing decisions that consider LCA results [14]. **DS/AI approaches** offer practical means to improve accuracy and completeness of LCA. Method selection depends on study objective, data type and available volume [15]. DS/AI improve product engineering decisions while maintaining resource use. DS forms the basis for machine learning (ML) and ontology-based reasoning. AI systems display intelligent behavior, analyze environments, take actions with autonomy and achieve specific goals [16].

4 Systematic review of engineering simulation models

In this section, a systematic analysis is presented of the relevant approaches for implementing simulation models in sustainable product engineering. A conceptual model is developed based on an analysis of the approaches regarding the relevant aspects and parameters used.

4.1 Derivation of relevant engineering decision points

The analysis of relevant approaches is based on the collection and evaluation of user perspectives. Key engineering decisions are recorded for transfer to specific engineering processes [17]. These decision points are characterized by specific features such as relevant roles, key figures and questions and serve as a reference point for the integration of simulation models [18]. The decision points were identified in workshops with application partners from the automotive and consumer goods sectors. Based on an analysis of existing engineering processes, relevant decision points regarding sustainability of products

are identified. The identified characteristics of the decision points include: requirements change, solution principle change, material change, geometry change, production process change and supplier change. These points describe core artifacts and characteristics of sustainable product engineering and are further subdivided according to the respective decision-making situation.

4.2 Identification of relevant approaches using simulation models for LCA

To identify relevant approaches for the use of simulation models in LCA, a systematic literature review is conducted in accordance with the PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) guidelines. The objective of the present study is to identify existing approaches for integrating simulation models into LCA-related applications. The basis for this study is a structured screening of the scientific databases Scopus and Web of Science. These databases provide quality-controlled literature sources. In addition, Google Scholar offers a way to check for completeness. Topic-specific search terms are derived from the research objective and combined into search strings using Boolean operators. The search strings are applied to the databases and bibliometric data of all retrieved publications are extracted for further analysis. The classification of existing tools providing engineering simulation models from ROMERO et al. [14] forms the basis for derivation of search terms. Each search string comprises three conceptual components: keywords related to product engineering, Life Cycle Assessment and the respective simulation approach (see Figure 2).

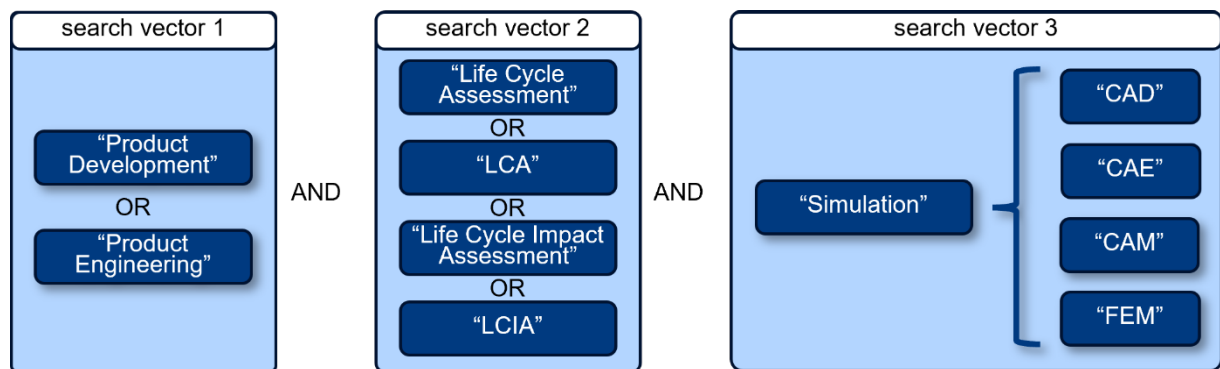


Figure 2: Search string formulation on three term vectors for identifying simulation approaches for LCA in product engineering

Results of the study are presented in Table 1. In database search, 738 publications are identified, of which 681 remain after removing duplicates. Following a screening process involving the evaluation of titles and abstracts, publications are selected based on their relevance to enhance the quality of LCA data through the integration of LCA with simulation tools, as well as the identification of simulation-based parameters relevant to LCA. This process resulted in the selection of 110 publications for a full-text assessment. Following the full-text analysis according to the same criteria, 22 publications are selected. Due to thematic overlap among the search strings, individual publications are retrieved multiple times and therefore identified as duplicates and included only once.

Table 1: Results of the systematic literature review

PRISMA Phase → tool category	CAD	CAE	CAM	FEM	Total
Total records	385	97	85	171	738
Total records (without duplicates)	370	81	79	151	681
Literature assessed for eligibility	36	13	13	48	110
Studies included in review	7	1	5	9	22

4.3 Categorization of relevant approaches

Identified approaches are analyzed regarding implementation of simulation models into sustainable product engineering. Analysis of the articles reveals three recurring characteristics of implementation approaches. The approaches are classified into engineering-level, process-level and tool-embedded model integration, as shown in Table 2 (RQ1).

Table 2: Categorization of included studies according to level of integration mentioned

(E) ngineering-level	(P) rocess-level	(T) ool-embedded
[8–10,14,19–31]	[9,19,21,25,27,32,33]	[8,10,14,23,28,33–36]

Engineering-level integration involves mapping product and process properties. Relevant data is extracted, harmonized and imported from simulation software into an LCI database. Therefore, LCI values are available for use in LCA software for sustainability calculations. CHEN, TAO and YU [20] propose a feature-based paradigm in which product features are systematically mapped to operational features. This allows CAD geometry and associated attributes to be translated into part- and process-level LCI entries. In ASNAFI et al. approach [19] translation of data provided by simulation tools is performed by a "Lifecycle data integrator" for LCI. The software tool collects relevant product and production attributes from heterogeneous sources, aligns the data in a unified data model and transfers it to LCA software. The approach further includes a "Life Cycle Analyzer" that enables feedback of sustainability data from impact assessment into simulation software. Complementary enterprise-level patterns include Product Lifecycle Management (PLM)- and mechanical Bill of Material (mBOM)-driven workflows. These workflows transition mBOM to Sphera LCA for Experts, which are used for systematic mapping of materials and processes [22,23]. Life-cycle engineering tools for additive manufacturing are enhanced to automate LCI generation from CAD/slicer outputs when mass/volume and material data are provided [21]. **Process level** integration links real or simulated manufacturing operations such as CAM and Computerized Numerical Control (CNC) outputs to environmental analysis. ANDRIANKAJA et al. [32] describe STEP-NC and ANGEL approaches in which toolpaths, cutting parameters and execution traces are used to compute operation level energy and material flows. These approaches are combined with discrete event simulation or Petri net models to generate temporally resolved machine state and power demand series for use in LCA. GAHA et al. [24] build on data extraction from simulation software and present a CAD/CAPP integration providing real time scenarios. LCI entries are assigned to product features, provide scenarios from CAD and CAPP and translate LCA results into the knowledge context of engineers. **Tool-embedded implementations** involve native integration into simulation tools. Native CAD plugins and cloud LCA services are used to provide engineers with environmental feedback inside modeling environments. MATHIEUX et al. [23] and SCHILLING et al. [34] publish examples such as SolidWorks Sustainability and Siemens NX Sustainability, in which environmental indicators are imported directly into CAD tools. BLIBECH [8] presents a One Click LCA integration with Building Information Modeling (BIM).

4.4 Evaluation of relevant approaches regarding decision making in engineering

For systematic review, product engineering approaches for integration of simulation models into LCA are evaluated with respect to their capacity to support decision making across engineering activities by enhancing data quality at initial stages. Evaluation criteria are established to assess these approaches into a structured manner. The criteria include support for common simulation tools within engineering workflows, interoperable export of simulation outputs for LCI ingestion, dynamic updating of LCI databases during engineering decisions, interpretable feedback for engineers and mapping of metadata and features to ensure consistency and traceability of simulation data. Table 3 summarizes how integration approaches identified in literature meet decision-relevant criteria. The categorization of these

approaches remains consistent with the classification established in Table 2. A black circle indicates fulfilled, a half-black/half-white circle partially fulfilled and a white circle not fulfilled.

Table 3: Results of evaluation relevant approaches regarding decision making

Category	Author	Approach	common simulation tools	export to LCI	dynamic LCI update	actionable feedback	metadata / feature mapping
E	MATHIEUX et al. [23]	CAD-based LCI extraction	●	●	◐	◐	●
E	BLIBECH [8]	Feature-based BIM–LCA	◐	●	◐	◐	◐
E	KO et al. [22]	PLM-driven mapping	●	●	◐	◐	●
E / P	FAVI et al. [21]	Additive manufacturing automated LCI	●	●	◐	●	●
E / T	CHEN et al. [20]	Feature-based CAD integration	●	●	◐	◐	●
P	ANDRIANKAJA et al. [32]	CAM/CNC integration with dynamic LCA	●	●	●	●	◐
P	ANDRIANKAJA et al. [32]	Machine simulation LCA	●	●	●	●	◐
T	MATHIEUX et al. [23]	Native CAD plugin	●	●	○	◐	◐
T	SCHILLING et al. [34]	Native CAD plugin	●	●	○	◐	◐
T	SCHILLING et al. [34]	Cloud-based LCA service	○	●	◐	◐	◐

● = fulfilled ◐ = partially fulfilled ○ = not fulfilled

The following examples illustrate typical approaches and link to the limitations identified for the three integration categories. **Engineering-level** integration remains limited by incomplete lifecycle information in early CAD models, as discussed by MATHIEUX et al. and BLIBECH [8,23]. Critical attributes such as use-phase energy, service life and detailed material inventories are often unavailable and are approximated using proxies, which reduces result robustness [8,23]. Feature-based CAD–LCA approaches proposed by CHEN et al. and GAHA et al. mitigate this limitation by explicitly mapping geometric and semantic features to process-relevant parameters and LCI flows [20,24]. These mappings improve traceability and enable systematic population of foreground inventories. Additional modelling effort is required for novel materials when database entries are missing, as shown in multiscale and material-focused studies by MIO et al. and BIANCHI et al. [9,31]. **Process-level** integration enables temporally resolved and operation-specific assessment but requires detailed CAM/CNC machine or simulation data and standardized translation mechanisms [30,32]. The ANGEL project mentioned by ANDRIANKAJA et al. exemplifies a strong operational approach by converting machine-state and power time series into process LCI entries, which supports iterative process planning with environmental metrics [32]. High data requirements and limited industrial adoption of enabling standards such as STEP-NC remain main constraints on scalability [30,32]. **Tool-embedded** model-level integration yields rapid model-level feedback inside CAD, BIM or PLM environments but often reports simplified indicators rather than full cradle-to-grave inventories [23,36]. Absence of continuous synchronization

with background LCA databases limits consistency in iterative design [8,36]. Best practice combines native in-tool indicators with PLM-driven mBOM exports and strict metadata governance to improve interpretability and reduce manual reconciliation across iterations [22,34].

4.5 Identification of relevant parameters

Based on the analyzed approaches, relevant parameters of simulation models are identified to enable a comparison across studies. Table 4 represents a summary of these parameters, including typical units and data sources, to ensure consistent and reproducible characterization.

Table 4: Identified relevant parameter

Parameter	Suggested unit	Typical source	References
geometry and mass	kg; mm ³ or m ³	CAD, mBOM, AM slicer, Parametric CAE	[21–23,26,27]
material properties	material ID; kg·m ⁻³	CAD metadata, PLM, multiscale / molecular dynamics, material database	[9,20,22,31]
mechanical material parameters	GPa; dimensionless	atomistic and multiscale simulations	[26,31]
process descriptors	G-code/vector; m·min ⁻¹ ; mm·rev ⁻¹ ; mm	CAM output, STEP-NC, NC simulation	[20,32]
temporal and energy profiles	s or h; kW	machine simulation, DES logs, machine control traces	[9,21,32]
consumables and waste streams	L; g or kg; cycles/parts	process simulation, shopfloor data, CAM / AM slicer, tool-life models	[22,32]
lifetime and use-phase variables	h; cycles; %; timestamped logs	durability simulation, monitoring, digital twin, in-service sensors	[9,21,33]
end-of-life and circularity parameters	%; categorical	PLM, EoL modelling	[9,27]

Geometry and mass metrics include part mass, nominal volume, additive manufacturing support volume and topology variants. They represent primary drivers of material flows and are extracted from CAD systems, mBOM structures and slicer or topology-optimization outputs. Material properties are required to match components to background inventories. Material properties derived from multiscale and molecular dynamics simulations are an alternative when suitable database records are unavailable. Mechanical material parameters, such as elastic modulus and Poisson's ratio, are necessary to support robust material substitution decisions. Process descriptors, including toolpath geometry, cutting speed, feed rate and depth of cut, combined with temporal and energy profiles, enable conversion of machine behavior into kWh and other LCI entries. Consumables and waste streams, such as coolant, lubricant, chips, support structures and tool wear, are captured for waste accounting and end-of-life modelling. Lifetime and use-phase variables, including service life, duty cycles and remaining useful life, are required to allocate use-phase impacts. End-of-life and circularity parameters further support modelling of recycling rates, recovery scenarios and circularity pathways. Support for handling data is essential for managing LCI data, particularly when using simulation and primary data.

4.6 Potentials of Data Science and Artificial Intelligence

Product engineering faces multiple challenges for LCA. These challenges can be grouped into three categories: data availability and quality (missing data, low-quality measurements, costly LCI data collection), data heterogeneity and scale (large volumes, redundant simulation outputs, inconsistent formats and units, fragmented sources) and integration and mapping issues (poor interoperability, domain-specific taxonomy and unit mismatches, mapping simulation outputs to LCI properties) [15,16]. DS/AI methods address distinct stages of LCI data preparation. Data architecture tools, blockchain, data

mining and data engineering support data collection, transfer and storage. Theoretical models and conceptual frameworks support regionalization of impacts, dynamic LCA and handling of big data. ML techniques can be applied to predict LCA results, decision support and benchmarking [15]. Knowledge graphs are positioned within both data architecture and conceptual frameworks due to incorporate structural data modelling and formalized ontologies [18]. ML is the most studied DS approach for LCA and is most applied to predict LCA results [15]. Typical targets include characterization factors, LCI entries and unit process inventories. The method commonly appears under the label “surrogate LCA” and is applied when new products or systems lack measured inventory data. Studies employ models that range from linear regressions to advanced neural networks. Model choice depends on available training data, feature complexity and required interpretability [15]. DS/AI methods increase availability, consistency and usability of LCI data, enable rapid scenario analysis and prediction of LCA results into early engineering stages and support automated decision-making and reducing uncertainty (RQ2).

4.7 Conceptual approach for simulation models in sustainable product engineering

A conceptual model for integrating simulation data into sustainability assessments is presented, which is based on analysis of existing approaches and necessary parameters derived from them. Building on preliminary work relating to impact analysis like model-Based Effect-Chain Analysis (MECA) for Complex Systems [37] and associated data models, this approach involves transferring identified simulation models to a knowledge graph [18]. The knowledge graph serves as a central representation that integrates all information relevant to assess the environmental impact of technical products. Based on the semantic model in form of an ontology, heterogeneous and extensive datasets can be integrated, analyzed and interpreted into the knowledge graph. The existing published model is supplemented by parameters listed in Table 4. These parameters enable assessments of LCAs based on realistic simulation data integrated in early stages of product engineering.

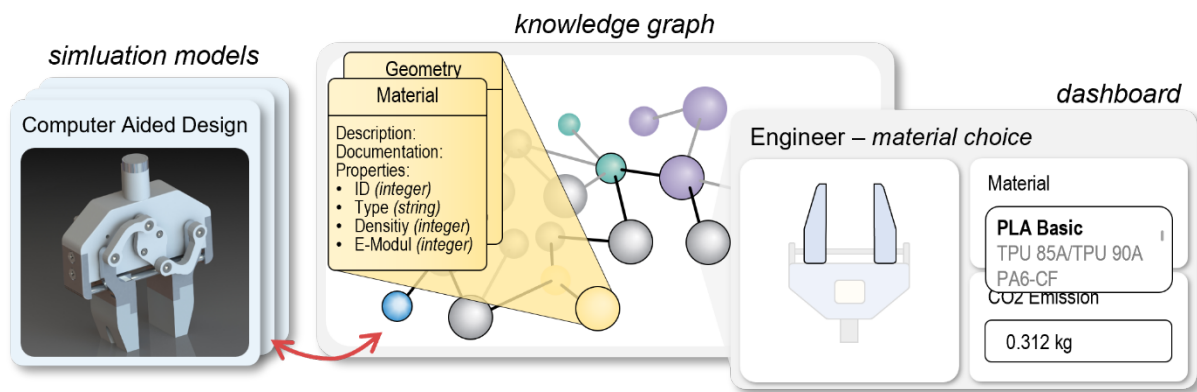


Figure 3: Knowledge Graph for implementation of simulation models for sustainable product engineering

Technically, the knowledge graph is implemented in Neo4J. A defined data model allows parameters from different source systems to be integrated, contributing to the harmonization of diverse data formats. Importing data via generic CSV tables allows for efficient integration of large, diverse datasets. The implementation is demonstrated using the example of a decision point for a material change. Based on a geometry created in CAD, relevant information on the component geometry and composition can be derived. Combining this information with other data stored in the knowledge graph, such as emission factors or energy consumption from additional simulation models, allows for initial conclusions to be drawn about the resulting environmental impacts. Role-specific dashboards provide corresponding calculations, such as CO₂ equivalents, using queries formulated in Cypher. These dashboards enable engineers to evaluate engineering decisions, such as material selection, based on concrete simulation data and immediately determine their environmental impact (see Figure 3). Integrating all relevant life cycle parameters (see Table 4) enables conclusions to be drawn regarding the entire product life cycle.

5 Discussion

The seven-step approach enables systematic and comprehensible analysis of existing methods and development of a conceptual model for incorporating simulation data into sustainability assessments. While the combination of structured literature analysis, parameter-based evaluation and conceptual implementation provides a methodologically consistent basis. In relation to the previously defined assessment criteria, the knowledge graph-based approach offers advantages in terms of interoperability integrating heterogeneous simulation data and consistently mapping metadata to ensure traceability. While dynamic updating of LCI-relevant information during technical decision-making processes is conceptually possible, practical implementation depends on additional interfaces and degrees of automation. Support for common simulation tools is not native; existing engineering workflows can only be integrated to a limited extent via standardized exports. The semantic structure of the knowledge graph simultaneously allows transparent and interpretable feedback on the results for engineers. The approach meets key criteria in conceptual terms but has limitations in terms of practical scalability.

6 Conclusion

The article presents a systematic analysis of existing engineering simulation models for sustainable product design. Using a seven-step research approach, the relevant approaches are identified, categorized and analyzed based on simulation parameters relevant to sustainability assessment, such as mechanical material parameters, process descriptors, consumables and waste streams. Based on this categorization, the potential use of DS/AI is identified and a conceptual model for a knowledge graph based decision support is developed. Relevant parameters are implemented into a knowledge graph to serve as a central representation integrating information for assessing environmental impact of technical products. The approach is validated using previously identified criteria. The approach enables early improvement of data quality using simulation data. Identified data and information can be systematically transferred to sustainability assessment, providing engineers with a solid foundation for making hybrid decisions regarding engineering sustainable products.

7 Acknowledgements

The authors gratefully acknowledge funding of the project Decide4ECO (13MX002G) by the German Federal Ministry of Economy and Energy (BMWE) from the European Union – NextGenerationEU. The results are solely those of the authors and do not necessarily reflect the views of the European Union or the European Commission. Neither the European Union nor the European Commission can be held responsible for them. Results are reflected by the scope of DFG SPP2443 (coordination, PN521148212).

References

- [1] T. C. McAloone and D. C. A. Pigosso, "Ökodesign," in *Pahl/Beitz Konstruktionslehre*, B. Bender and K. Gericke, eds.: Berlin, Heidelberg: Springer Berlin Heidelberg, 2021, pp. 975–1021.
- [2] I. Gräßler and J. Pottebaum, "Generic Product Lifecycle Model: A Holistic and Adaptable Approach for Multi-Disciplinary Product–Service Systems," *Applied Sciences*, vol. 11, no. 10, p. 4516, 2021.
- [3] L. T. Blessing and A. Chakrabarti, *DRM, a Design Research Methodology* London: Springer London, 2009.
- [4] M. J. Page, J. E. McKenzie, et al. "The PRISMA 2020 statement: an updated guideline for reporting systematic reviews," *Systematic reviews*, vol. 10, no. 1, p. 89, 2021.
- [5] M. Inoue, K. Lindow, R. Stark, K. Tanaka, Y.-E. Nahm and H. Ishikawa, "Decision-making support for sustainable product creation," *ADVANCED ENGINEERING INFORMATICS*, vol. 26, no. 4, pp. 782-792, 2012.
- [6] B. Rotondo, C. Bakker, R. Balkenende and V. Arquilla, "Integrating Circular Economy Principles in the New Product Development Process: A Systematic Literature Review and Classification of Available Circular Design Tools,".
- [7] C. Favi and M. Marconi, "Product Eco-Design in the Era of the Circular Economy," *Sustainability*, vol. 17.
- [8] O. Blibech, "LCA Software Tools, CAD Tools and Engineering Design Processes: Challenges with Process Modelling for Sustainable Productions," in : Institute of Electrical and Electronics Engineers Inc, 2024, pp. 79–86.

- [9] I. Bianchi, A. Forcellese, C. Mignanelli, M. Simoncini and T. Verdini, "Development and Life Cycle Analyses of Carbon Fiber Reinforced Polymer Tubular Parts for Metal Replacement in Aerospace Applications", 2025.
- [10] G. Alexandridis, D. Tzetzis and ..., "Biomimicry in product design through materials selection and computer aided engineering," ... Science and Engineering, 2016.
- [11] A. Ghose, "Can LCA be FAIR? Assessing the status quo and opportunities for FAIR data sharing," *The International Journal of Life Cycle Assessment*, vol. 29, no. 4, pp. 733-744, 2024.
- [12] M. Gabellini, L. Civolani, M. Ronchi, L. D. Naldi and A. Regattieri, "Data Spaces in Manufacturing and Supply Chains: A Review and Insights from European Initiatives," *Applied Sciences*, vol. 15, no. 11, p. 5802, 2025.
- [13] H. Perera, U. Atmojo and V. Vyatkin, "1.02.P-Tu005 Life Cycle Assessment in Paper and Pulp Industry : Addressing the Data Exchange Challenges,".
- [14] D. Romero, J. Osorio, M. C. Bentacur, G. Estrada and A. Molina-Gutierrez, eds., *Next generation computer-aided tools: Supporting integrated Sustainable mass-customized product developments* (2011).
- [15] S. Bahmutsky, I. Turner, V. Arulnathan and N. Pelletier, "Advancing life cycle assessment through data science: A critical review of algorithms, tools, and data challenges," *Sustainable Production and Consumption*, vol. 61.
- [16] Iris Gräßler, Jens Pottebaum, Peter Nyhuis, Rainer Stark, Klaus-Dieter Thoben and Petra Wiederkehr, "Hybrid Decision Support in Product Creation," *Improving performance with data science and artificial intelligence*, vol. 41.
- [17] Iris Gräßler, Sven Rarbach and Jens Pottebaum, "Datenqualität in der Entwicklung kreislauffähiger Produkte," *Entscheidungsunterstützung für die zirkuläre Wertschöpfung durch Datenökosysteme*, vol. 41, no. 2, pp. 12-19, 2025.
- [18] I. Graessler and S. Rarbach, "Model-based Impact Analysis for Engineering Sustainable Products in Value Creation Networks," *Procedia CIRP*, vol. 136, pp. 207-212, 2025.
- [19] N. Asnafi, Z. Zhou and J. Tao, eds., *CAX/PLM/ERP-LCA integration to support product life cycle engineering: A conceptual framework* (IOP Publishing Ltd, 2021).
- [20] Z. R. Chen, J. Tao and Yu, SR, "A Feature-based CAD-LCA Software Integration Approach for Eco-design," vol 109.
- [21] C. Favi, L. Murgese, S. Gallozzi, C. Chiacchietta, M. Marconi and M. Mandolini, "Definition of a life cycle engineering tool to support the ecodesign in additive manufacturing: application to FDM technology," *International Journal on Interactive Design and Manufacturing*, 2025.
- [22] N. Ko, R. Graf, T. Buchert, M. Kim and D. Wehner, "Resource Optimized Product Design - Assessment of a Product's Life Cycle Resource Efficiency by Combining LCA and PLM in the Product Development," in *Procedia CIRP* 2016, Vol. 70.
- [23] F. Mathieux, L. Roucoules, L. Lescuyer and D. Brissaud, eds., *Connecting CAD and PLM systems with ecodesign software: Current experiences and future opportunities* (2007).
- [24] R. Gaha, B. Yannou and A. BenAmara, "A new eco-design approach on CAD systems," *International Journal of Precision Engineering and Manufacturing*, vol. 15, no. 7, pp. 1443-1451, 2014.
- [25] J. Tao, Yu, SR and ASME, "DEVELOPMENT OF LCA MODULE INTEGRATED WITH CAD FOR ECO-ASSESSMENT OF MACHINED PRODUCTS," in , 2018.
- [26] E. A. Calado, M. Leite and A. Silva, "Integrating life cycle assessment (LCA) and life cycle costing (LCC) in the early phases of aircraft structural design: an elevator case study," ... *Journal of Life Cycle Assessment*, 2019.
- [27] P. Cicconi, D. Landi and M. Germani, "An Ecodesign approach for the lightweight engineering of cast iron parts," *International Journal of Advanced Manufacturing Technology*, vol. 99, 9-12, pp. 2365-2388, 2018.
- [28] C. Favi, D. Landi, R. Garziera and M. Rossi, "Fostering Design for Sustainability through the Adoption of Computer-Aided Engineering Tools in the Development of Energy-Related Products," *Sustainability*, 2024.
- [29] Y. He, T. Huang, Y. Wang, Y. Nie, Y. F. Li and ..., "A new lightweight design method integrating shape optimization with life cycle assessment for extrusion dies," *Journal of Cleaner ...*, 2017.
- [30] A. Kurdi, N. Alhazmi, H. Alhazmi and T. Tabbakh, "Practice of simulation and life cycle assessment in tribology—A review," *MATERIALS*, 2020.
- [31] A. Mio and M. Fermeglia, "Multiscale modelling techniques in Life Cycle Assessment: application to product design," *Procedia CIRP*, 2022.
- [32] H. Andriankaja, J. Le Duigou, C. Danjou and B. Eynard, "Sustainable machining approach for CAD/CAM/CNC systems based on a dynamic environmental assessment,"
- [33] G. Mengesha, "Shaping tomorrow's Skylines: A Review of Emerging Methods and Technologies in Sustainable Structural Engineering," Available at SSRN 5298042, 2025.
- [34] L. Schilling, S. Spitzer, R. Kupfer, C. Klaus and M. Gude, "Integrating environmental assessment into aerospace product development using the VDI 2221 methodology," *Discover Sustainability*, vol. 6, no. 1, 2025
- [35] H. Huang, G. Ameta and ASME, "Towards a Computational Framework for Energy Estimation - Needs, Requirements, and Its Generic Shell," in , 2010, 891-+.
- [36] A. Jakobsen and T. Tambo, "Comparative Evaluation of Product Lifecycle Management Systems: Current State of Sustainability Representation and Future Perspectives," *Torben, Comparative Evaluation of ...*, 2024.
- [37] I. Gräßler, D. Wiechel, A.-S. Koch, D. Preuß and C. Oleff, "Model-Based Effect-Chain Analysis for Complex Systems," *Proceedings of the Design Society*, vol. 2, pp. 1885-1894, 2022.

Corporate resilience through generic AI-based workflows in strategic product planning

Iris Graessler¹, Deniz Oezcan^{1*}

¹Paderborn University, Heinz Nixdorf Institute, 33102 Paderborn

*Corresponding author. E-Mail address: deniz.oezcan@hni.upb.de

Abstract: In increasingly volatile and uncertain markets, corporate resilience has become a critical capability in strategic product planning. Companies face significant challenges in systematically monitoring and interpreting heterogeneous environmental data originating from diverse sources, formats, and temporal contexts. While predefined workflows and decision trees can support strategic analysis, they often lack the flexibility required to cope with dynamic market conditions and foresight-related data from extreme dispersed and heterogeneous sources. This paper proposes a method to enhance corporate resilience through the application of generic, reusable AI-based workflows in strategic product planning. The approach integrates Data Science and Artificial Intelligence methods into modular, visually modelled workflows that enable hybrid human-AI decision-making. Based on a systematic literature review and an analysis of industrial challenges, key success factors and resilience criteria are identified. These insights are used to develop a method that supports internal and external analyses, scenario-based strategy development, and adaptive implementation monitoring within a generic workflow structure. The method leverages techniques such as machine learning and generative AI to process structured and unstructured data, identify patterns, and support real-time strategic assessments. Validation with decision-makers from medium-sized companies demonstrates improved transparency, repeatability, and cross-functional collaboration compared to predefined workflows.

Keywords:

Strategic Product Planning, Artificial Intelligence, workflow, resilience, decision-making

1 Motivation

In an increasingly volatile and uncertain business environment, corporate resilience has emerged as a key strategic capability in Strategic Product Planning (SPP). Good resilience enables companies to anticipate decisions in advance and create scope for decision-making in areas where scope for decision-making is very limited due to uncertainty. A major challenge is monitoring environmental influences and their impacts on the company [1]. Although data such as public statistics and trend descriptions is available, it is extremely diverse in terms of data formats, access channels, geographical and temporal contexts. Additionally, time-series data can be decisive, but only available in diverse temporal resolutions and with unclear semantic. Such extreme data is therefore only usable with enormous effort. Predefined workflows and decision trees enable to manage these complex influences but are inflexible and rigid [2]. Dynamic markets in particular require flexible and adaptive analysis mechanisms to support informed real-time decision-making. Innovative approaches in the field of Data Science (DS) and Artificial Intelligence (AI) supported systems, have potential to automate processing of heterogeneous environmental data and provide access to new insights using information in real-time assessments [3]. Developing such methods for hybrid decision support are essential for ensuring the future competitiveness and sustainable product planning and engineering in companies [4]. Therefore,

a method is needed for providing guidance for internal and external analysis, scenario-based strategy development, and adaptive implementation monitoring in a modular workflow structure in SPP.

The structure is as follows: Motivation is followed by state of the art in the relevant areas. Section three describes research questions and approach. The fourth section covers the development of the method. First the industrial challenges are analyzed and followed by a systematic literature review. Success factors are derived from review, which serve as basis for developing the method. The penultimate section presents the validation and discussion. This is followed by short summary and outlook.

2 State of the Art

The state of the art covers the three relevant sub-areas of this paper: SPP, corporate resilience in strategic decision-making and graphical workflow-based decision support in this context.

Strategic Product Planning (SPP) refers to the continuous, methodically guided business process in which companies align their product portfolios with long-term market, technology, and customer requirements, identify future product ideas, and make systematic decisions about product development [5]. SPP goes beyond pure operational project planning and encompasses a holistic analysis of opportunities, risks, and competitive advantages in order to achieve sustainable competitive advantages and set concrete engineering tasks [6]. SPP uses many methods to support decision-makers in important decision-making processes. One key method here is the scenario technique [7]. Comprehensive data models and integration of processes provide the basis for holistic insights in SPP [8, 9].

Corporate resilience is a central concept in management and strategy research discourse that describes the ability of companies to not only survive disruptions in an uncertain, complex, and volatile environment, but also to adapt, learn, and remain competitive. Resilience is strategically embedded in decision-making and management processes to enable continuous adaptation, risk absorption, and restoration of performance [10]. In an organizational context, resilience is understood as a dynamic capability that encompasses multiple levels—individuals, teams, structures, and external networks—and requires active adaptation to environmental uncertainties. This strategic perspective makes resilience an important design goal in SPP, not only as a response to crises, but as a proactive component of long-term corporate planning [11].

Graphical workflow modelling serves to interact with complex decision-making logic, data flows, and process steps comprehensible to human actors, thereby increasing both transparency and user acceptance [2]. Visual decision support system approaches typically combine structured representation techniques such as flowcharts, influence diagrams, or low-code platforms with AI methods to make data analysis and decision logic accessible [12]. Graphical AI workflow modelling must address not only technical integration problems but also organizational knowledge and domain expertise to bridge the gap between expert knowledge and algorithmic processing. Visual models not only support the configuration of AI components, but also promote human-in-the-loop interactions by enabling decision-makers to directly interpret and control assumptions, data paths, and algorithms. [13,14]

3 Research Questions and approach

Key objective of this research is a method for improving corporate resilience through the application of generic, reusable workflows that integrate AI into SPP. The method combines conceptual modelling with digital tools for visual workflow design, enabling decision-makers to structure, simulate and adapt planning processes across organizational contexts. It helps to combine the benefits of established methods and DS/AI contributions in terms of hybrid intelligence. This paper contributes answers to three research questions (RQ):

- RQ1: Which criteria are proposed in literature to assess corporate resilience in the SPP process using generic, AI-based workflows?
- RQ2: What success factors are relevant for the effective integration of generic AI-based workflows into strategic product planning to cope with extreme data?
- RQ3: What advantages does the method offer compared to predefined workflows?

In order to answer these three research questions, motivation and industrial challenges (1) are discussed, and a systematic literature review based on the PRISMA approach [15] is conducted (2). This review focuses on generic, reusable workflow design in SPP and its added value. The success factors from relevant approaches are recorded and used to define a method (3) that provides guidance for internal and external analysis, scenario-based strategy development and adaptive implementation monitoring in an AI-based workflow structure using DS/AI methods. The method is developed in the EU data science project CREXDATA. It is validated and discussed with decision-makers in medium-sized companies (4) by identifying strategic weaknesses, increasing adaptability and ensuring long-term profitability. The integration of graphical workflow tools into strategic decision-making improves transparency and repeatability and enables cross-functional collaboration and digital hybrid decision support in SPP, shown in Figure 1.

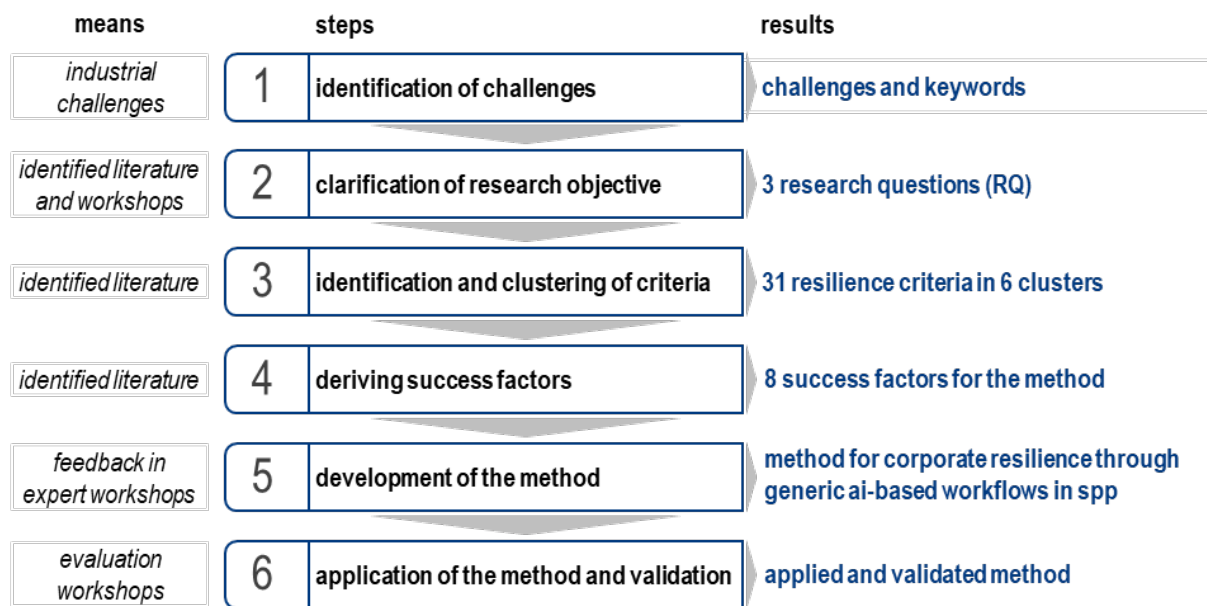


Figure 1: Scientific Approach

4 Development of the method

The aforementioned approach systematically links rigorous scientific research with industrial practice. First, the industrial challenges in SPP are analyzed to identify the practical problem context (4.1). Based on these findings, a systematic literature review is conducted to identify relevant concepts and criteria related to corporate resilience and AI-supported decision-making (4.2). The findings are then clustered and summarized into a set of relevant success criteria that address the identified challenges (4.3). Finally, these success factors are operationalized through the development of a modular, AI-based workflow method for strategic product planning (4.4).

4.1 Industrial challenges

Companies operate in an increasingly uncertain environment, which under current conditions is no longer characterized by isolated crises but by continuous disruption. Strategic planning is no longer as

promising as it was in predictable situations due to the new conditions. According to a recent study, only 8% of the companies surveyed are able to maintain their plans over a period of more than one year in a volatile environment [16]. Cross-industry studies emphasize that strategic decisions in SPP must be actively reconsidered in order to remain competitive in the long term and to be able to place products on the market. These developments highlight the industrial challenge: SPP today must not only anticipate customer requirements and future scenarios, but also have the ability to respond continuously to external shocks and highly volatile data environments [17]. Classic sequential planning models and rigidly predefined workflows reach their limits here because they are not sufficiently flexible, adaptive or data-centric. Companies must design their SPP in such a way that it is not only efficient, but also robust against uncertainties, adaptive to change and transparent in decision support [18,19]. AI-based, generic workflows represent an innovative approach to addressing these requirements by integrating techniques such as machine learning, predictive analytics and interactive modelling to navigate complex data conditions and specifically strengthen corporate resilience.

4.2 Systematic literature review and criteria identification

Based on the industrial challenges, a systematic literature analysis is conducted to answer the RQs defined in section 2. In a first step, criteria to assess organizational resilience in the strategic SPP using generic, AI-based workflows are identified. The research approach is based on the PRISMA guidelines [15]. For research, the databases of Scopus, Web of Science, Science Direct and IEEE Xplore are used to identify relevant literature. Four thematic clusters feed the search string for the databases:

1. organizational resilience, corporate resilience
2. strategic product planning, development
3. artificial intelligence, data science
4. workflow, decision support, modelling

The search results in a total of 353 hits. The literature is narrowed down to 87 results by focusing on strategic decision support and relevant work from the last 10 years. From these 87 results, 16 articles are relevant by skimming and scanning. The analysis and recording of success criteria answers RQ1 by an overview of criteria for assessing corporate resilience in the SPP process using generic, AI-based workflows is shown in Table 1.

Table 1: Criteria for assessing corporate resilience in SPP

<i>Nr.</i>	<i>Criteria</i>	<i>Source</i>	<i>Nr.</i>	<i>Criteria</i>	<i>Source</i>
1	customer focus[18]	[21]	17	generation of capital	[22]
2	external collaboration	[21,22]	18	location & marketing	[22]
3	digitization	[21,22]	19	proactivity	[19,23]
4	crisis management	[21]	20	environmental uncertainty	[19,22]
5	innovation	[21,24]	21	change management	[19]
6	corporate culture	[20,21]	22	product excellence	[19,24]
7	leadership	[21]	23	adaptability	[19]
8	digitalization of service	[24]	24	redundancy	[23]
9	strategic collaboration	[22,24]	25	diversity	[23]
10	customer intimacy	[24]	26	prudence	[23]
11	agile use of resources	[24]	27	embeddedness	[23]
12	improved revenue model	[24]	28	customer loyalty	[20,24]
13	vertical integration	[22]	29	continuous experimentation	[20]
14	quality management	[22]	30	risk taking	[20]
15	business certifications	[22]	31	financial reserves	[20,22]
16	globalization	[22,23]			

The identified 31 resilience-related criteria from the literature are systematically categorized into six higher-level dimensions representing distinct but complementary resilience capabilities in SPP. The categorization follows a functional logic, assigning each criterion to the dimension in which it primarily contributes to resilience, thereby avoiding conceptual redundancy while preserving analytical clarity. Six resilience capabilities are visualized in Figure 2, which enables corporate resilience, action planning, and decision making. This takes place in the SPP of the generic product life cycle. The six resilience capabilities are classified in the image, starting from the right to left, using graphical workflow modelling.

C1 – Anticipatory Capability

Anticipatory capability describes the ability in SPP to identify relevant changes in the market, technology and environmental context at an early stage and to systematically integrate them into strategic decision-making processes within the company. The focus here is on the forward-looking analysis of customer needs, market trends and external uncertainties, as well as the continuous alignment of product and corporate strategy with future requirements.

C2 – Adaptive Decision-Making Capability

Adaptive decision-making capability describes the ability of organizations to flexibly adapt strategic product decisions when conditions, assumptions or targets change. In the context of early stages of product development, this adaptability is crucial, as strategic decisions are often made under conditions of high uncertainty and subsequent corrections are only possible to a limited extent.

C3 – Robustness under Extreme Data Condition

Robustness under extreme data condition describes the ability in SPP to remain capable of acting even under conditions of incomplete, contradictory or highly volatile data. This explicitly includes the

capability to process such extreme data, instead of neglecting it. This robustness is particularly relevant in dynamic markets where valid decision-making bases are often only available to a limited extent.

C4 – Transparency and Explainability

The cluster of transparency and explainability focuses on the traceability, documentability and comprehensibility of strategic decision-making processes. Transparency is a key prerequisite for acceptance, trust and organizational learning ability, particularly when using AI-based analysis and decision support systems.

C5 – Learning and Feedback Capability

The learning and feedback capability describes the ability of organizations to systematically learn from past decisions, experiments and their results, and to integrate this knowledge into future strategic product decisions. Resilience is understood here not as a static characteristic, but as a dynamic, learning-based process.

C6 – Cross-Functional Integrability

Cluster of cross-functional integrability describes the ability in SPP to integrate different functional, organizational and external perspectives into consistent decision-making processes. This ability to integrate is particularly crucial in the early stages of product development, as market, technology, cost and organizational aspects must be considered in parallel.

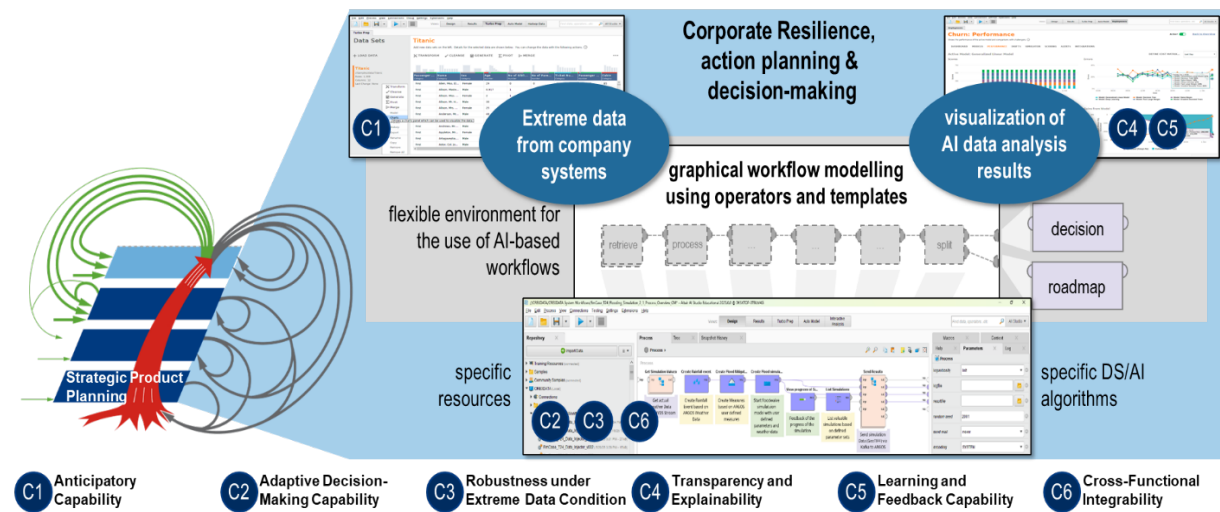


Figure 2: Resilience cluster for corporate resilience through AI-based workflows in the generic product life cycle

The 31 criteria identified in the literature are systematically assigned to six capabilities in Table 2. The six capabilities provide the conceptual basis for analyzing corporate resilience in the context of SPP, as these criteria describe what characterizes resilience in SPP (RQ1).

Table 2: Clusters of organizational resilience assessment criteria

Nr.	Resilience Capabilities	Categorized Resilience Criteria
C1	Anticipatory Capability	1, 10, 19, 20, 22, 28
C2	Adaptive Decision-Making Capability	5, 11, 21, 23, 26, 30
C3	Robustness under Extreme Data Conditions	4, 24, 27, 31
C4	Transparency and Explainability	3, 8, 14, 15
C5	Learning and Feedback Capability	5, 29, 30
C6	Cross-Functional Integrability	2, 6, 7, 9, 13, 16, 17, 18, 25

4.3 Relevant success criteria

The six capabilities of corporate resilience identified in the SPP are examined to derive success factors of the proposed method. Investigating success factors for integrating generic, AI-based workflows aims to establish the framework conditions, mechanisms and modelling principles necessary for the resilience clusters derived in RQ1 to be effective in real-world scenarios involving dynamic and highly volatile data. These success factors are clear and address technological and organizational aspects. They describe how generic workflows must be designed, implemented, and utilized to enhance an organization's adaptability, robustness, transparency, and learning capacity. Cross-cutting success factors can be derived from the six resilience capabilities.

Table 3: Success Criteria

Nr.	Success Factors	Description	Resilience Capability						
SF1	Modular and Generic Workflow Design	AI-based workflows must be modular in structure and generic in design so that individual analysis, evaluation or decision-making modules can be flexibly combined, replaced or expanded.	<table><tr><td></td><td>C2</td><td>C3</td><td></td><td></td><td>C6</td></tr></table>		C2	C3			C6
	C2	C3			C6				
SF2	Human-in-the-Loop Integration	The effective integration of AI workflows requires the explicit involvement of human decision-makers in analysis, interpretation, and decision-making processes.	<table><tr><td></td><td>C2</td><td></td><td>C4</td><td>C5</td><td></td></tr></table>		C2		C4	C5	
	C2		C4	C5					
SF3	Visual Workflow Modelling and Explainability	Visually modeled workflows increase transparency, traceability, and organization-wide comprehensibility of AI-supported decision-making processes.	<table><tr><td></td><td></td><td></td><td>C4</td><td></td><td>C6</td></tr></table>				C4		C6
			C4		C6				
SF4	Integration of Heterogeneous Data Sources	Ability to systematically integrate structured and unstructured internal and external data, such as extreme data.	<table><tr><td>C1</td><td></td><td>C3</td><td></td><td></td><td></td></tr></table>	C1		C3			
C1		C3							
SF5	Scenario-Based and Iterative Decision Logic	Successful AI workflows do not support singular “optimal” decisions, but enable scenario-based, iterative evaluations.	<table><tr><td>C1</td><td>C2</td><td></td><td></td><td>C5</td><td></td></tr></table>	C1	C2			C5	
C1	C2			C5					
SF6	Continuous Learning and Feedback Mechanisms	AI-based workflows must incorporate feedback loops to continuously improve foresights, models, and decision-making logic.	<table><tr><td></td><td>C2</td><td></td><td></td><td>C5</td><td></td></tr></table>		C2			C5	
	C2			C5					
SF7	Organizational Data and AI Governance	Clear rules on data quality, model use, responsibilities, and decision-making rights are essential for sustainable use.	<table><tr><td></td><td></td><td>C3</td><td>C4</td><td></td><td></td></tr></table>			C3	C4		
		C3	C4						
SF8	Cross-Functional and Cross-Domain Collaboration Enablement	AI-based workflows must actively support cross-functional collaboration, for example through shared data models and collaborative interfaces.	<table><tr><td>C1</td><td></td><td></td><td></td><td></td><td>C6</td></tr></table>	C1					C6
C1					C6				

RQ2 is addressed by identifying eight success factors that enable the effective integration of generic, AI-based workflows into SPP. These factors include modular workflow design, human-in-the-loop integration, visual and explainable workflow modelling, heterogeneous data integration, scenario-based decision logic, continuous learning mechanisms, organizational AI governance, and cross-functional collaboration support. Together, these success factors operationalize corporate resilience by embedding adaptability, robustness, and transparency directly into strategic planning workflows.

4.4 Development of the method based on the success factors

The synthesized success factors are used to create a guide for internal and external analyses, scenario-based strategy development, and adaptive monitoring within a modular workflow structure. Internal analyses include, for example, SWOT analysis (strengths, weaknesses, opportunities, threats) and analysis of production KPIs, while external analyses include market-specific KPIs and key figures. A

closely linked to specific use cases and difficult to adapt without extensive reconfiguration. Another important positive aspect was the support for cross-functional collaboration. The modular structure enabled different organizational units (product management, production, marketing) to make domain-specific contributions via specialized operators while maintaining a common analytical framework. This facilitated coordination between functions and reduced the coordination effort, which is particularly important in medium-sized companies with limited analytical resources. Decision-makers also appreciated the concept of digital hybrid decision support, which combines AI-based analysis with human judgement (human-in-the-loop). Nonetheless, several limitations were identified during validation. The method currently relies on a qualitative assessment involving a limited number of decision-makers, which limits the generalizability of the results. The effectiveness of the approach depends on the availability and quality of the data. Although the method is designed to handle heterogeneous data sources, incomplete, inconsistent or poorly managed data can compromise the reliability of AI-based analyses and thus the certainty of decisions. The validation demonstrates that the proposed method effectively enhances transparency, repeatability, and cross-functional collaboration in SPP, thereby addressing RQ3. At the same time, the discussion highlights important limitations related to data dependency, organizational maturity, and scalability. These findings underline both the practical relevance and research potential of generic AI-based workflows as a foundation for resilient strategic decision-making.

6 Summary and Conclusion

Based on extensive literature research and a review of generic, reusable workflow designs in SPP, aspects of increasing corporate resilience are identified. This is supplemented by the inclusion of challenges from industrial practice. These challenges include the early identification and assessment of opportunities and risks, as well as the use of heterogeneous data sources. DS/AI enables users to solve data-related problems and make quick and transparent decisions. DS/AI approaches such as generative AI in Large Language Models enable automated extraction of information and trends from unstructured external data sources like news articles or market reports. Machine learning algorithms such as supervised and unsupervised learning methods are used to identify patterns and anomalies within KPI analyses that are relevant for risk assessment and strategy monitoring. Six resilience capabilities are identified based on 31 resilience criteria, answering RQ1 based on systematic literature analysis. RQ2 is answered by deriving success factors from these clusters and their correlation with tasks in SPP. Targeting these success factors, the paper presents a method that provides guidance on internal and external analyses, scenario-based strategy development, and adaptive monitoring within a modular workflow structure. Internal analyses include SWOT-analysis and production KPI analysis, for example, whereas external analyses include market-specific KPIs and key figures. This modular workflow structure enables different KPIs to be monitored and independent operators for market analysis, risk assessment and strategy monitoring to be operated internally for SPP. The method has been validated with decision-makers in medium-sized companies. The decision-making process in the SPP process was improved through integration of graphical workflow tools. The decision-makers highlighted transparency and repeatability, based on the success factors, which answer RQ3. They also emphasized cross-functional collaboration and digital hybrid decision support.

7 Acknowledgements

The research leading to these results has received funding from the European Union's Horizon Europe Programme under the CREXDATA Project, grant agreement n° 101092749. We extend our gratitude to all collaborators, partners, and contributors involved in CREXDATA for their dedication. Results are reflected by the scope of DFG SPP2443 (coordination, PN521148212).

References

- [1] R. Stark, G. Seliger and J. Bonvoisin, eds., *Sustainable manufacturing. Challenges, solutions and implementation perspectives*, Cham, Switzerland, 2017).
- [2] J. Deuse, R. Klinkenberg and N. West, *Industrielle Datenanalyse*, Wiesbaden: Springer Fachmedien Wiesbaden, 2024.
- [3] I. Gräßler, J. Pottebaum, R. Stark, P. Nyhuis, K.-D. Thoben and P. Wiederkehr, "Hybrid Decision Support in Product Creation. Improving performance with data science and artificial intelligence," *Industry 4.0 Science*, vol. 41, no. 1, pp. 18-25, 2025. DOI: <https://doi.org/10.30844/I4SD.25.1.18>.
- [4] F. Nyffenegger, J. Ríos, L. Rivest and A. Bouras, eds., *Product Lifecycle Management Enabling Smart X*, Springer International Publishing, 2020.
- [5] I. Grässler, S. Rarbach and B. Grewe, "Modell der Strategischen Produktplanung. Digitale Zwillinge für kreislauffähige Produkte und Produktionsprozesse," *Industry 4.0 Science*, vol. 2025, 2025.
- [6] P. Brandtner and M. Mates, "Artificial Intelligence in Strategic Foresight – Current Practices and Future Application Potentials," in *The 2021 12th International Conference on E-business, Management and Economics*: New York, NY.
- [7] I. Graessler and D. Oezcan, "Quality Key Figures for Developing Future Scenarios," *Human Systems Engineering and Design (IHSED2024): Future Trends and Applications*, vol. -2, 2024. DOI: <https://doi.org/10.54941/ahfe1005553>.
- [8] I. Graessler, J. Pottebaum and P. Scholle, "Integrated Process and Data Model for Agile Strategic Planning," 11th International Workshop on Integrated Design Engineering IDE Workshop, 2017.
- [9] I. Graessler, P. Scholle and J. Pottebaum "Integrated process and data model for applying scenario-technique in requirements engineering," *Proceedings of the 21st International Conference on Engineering Design ICED 17*, 2017.
- [10] W. Su and S. Junge, "Unlocking the recipe for organizational resilience: A review and future research directions," *European Management Journal*, vol. 41, no. 6, pp. 1086-1105, 2023. DOI: <https://doi.org/10.1016/j.emj.2023.03.002>.
- [11] A. Annarelli and F. Nonino, "Strategic and operational management of organizational resilience: Current state of research and future directions," *Omega*, vol. 62, pp. 1-18, 2016. DOI: <https://doi.org/10.1016/j.omega.2015.08.004>.
- [12] A. Kovari, "AI for Decision Support: Balancing Accuracy, Transparency, and Trust Across Sectors," *Information*, vol. 15, no. 11, p. 725, 2024. DOI: <https://doi.org/10.3390/info15110725>.
- [13] R. W. Belk, D. Belanche and C. Flavián, "Key concepts in artificial intelligence and technologies 4.0 in services," *Service Business*, vol. 17, no. 1, pp. 1-9, 2023. DOI: <https://doi.org/10.1007/s11628-023-00528-w>.
- [14] I. Graessler and D. Oezcan, "Graphical AI workflow modelling: Identifying relevant competencies in AI-based automation of business processes," in *Human Systems Engineering and Design (IHSED2025): Future Trends and Applications*, AHFE International: AHFE International, 2025, <http://doi.org/10.54941/ahfe1006785>.
- [15] M. J. Page, J. E. McKenzie, P. M. Bossuyt, I. Boutron, T. C. Hoffmann, C. D. Mulrow, L. Shamseer, J. M. Tetzlaff, E. A. Akl, S. E. Brennan, R. Chou, J. Glanville, J. M. Grimshaw, A. Hróbjartsson, M. M. Lalu, T. Li, E. W. Loder, E. Mayo-Wilson, S. McDonald, L. A. McGuinness, L. A. Stewart, J. Thomas, A. C. Tricco, V. A. Welch, P. Whiting and D. Moher, "The PRISMA 2020 statement: an updated guideline for reporting systematic reviews," *Systematic reviews*, vol. 10, no. 1, p. 89, 2021. DOI: <https://doi.org/10.1186/s13643-021-01626-4>.
- [16] C. Fuchs and R. Tischler, "Resilient Planning in Volatile Markets. Building flexibility and speed into forecasting and planning to navigate uncertainty," *BARC Research Study*, vol. 2025.
- [17] J. Tüllner and J. He, "Global Crisis and Resilience Survey 2023. Die Resilienz-Revolution hat begonnen," PwC, 2023.
- [18] I. Graessler and D. Oezcan, "Identifying future competence demands in production companies for strategic decision-making," in *19th CIRP Conference on Intelligent Computation in Manufacturing Engineering*, 2025.
- [19] K. Dindarian, *Unvorhergesehenes als Chance sehen – Black Swan*Cham: Springer International Publishing, 2024.
- [20] R. Swaminathan, "How Can Resilience Create and Build Market Value?," *Journal of Creating Value*, vol. 8, no. 2, pp. 204-218, 2022. DOI: <https://doi.org/10.1177/23949643221125287>.
- [21] M. Radic, P. Herrmann, P. Haberland and C. R. Riese, "Development of a Business Model Resilience Framework for Managers and Strategic Decision-makers," *Schmalenbach Journal of Business Research*, vol. 74, no. 4, pp. 575-601, 2022. DOI: <https://doi.org/10.1007/s41471-022-00135-x>.
- [22] S. Syamsul Hadi, "New Perspective on the Resilience of SMEs Proactive, Adaptive, Reactive from Business Turbulence: A Systematic Review," *JOURNAL OF XI'AN UNIVERSITY OF ARCHITECTURE & TECHNOLOGY*, XII, V, 2020. DOI: <https://doi.org/10.37896/JXAT12.05/1524>.
- [23] M. Reeves and K. Whitaker, "Innovating Management Innovation," *Strategic Management Review*, vol. 3, no. 1, pp. 157-167, 2022. DOI: <https://doi.org/10.1561/111.00000040>.
- [24] T. Eriksson, M. Heikkilä and N. Nummela, "Business model innovation for resilient international growth," *Small Enterprise Research*, vol. 29, no. 3, pp. 205-226, 2022. DOI: <https://doi.org/10.1080/13215906.2022.2092890>.
- [25] I. Brüggemann, S. Buse and N. Villarreal, *How AI can increase resilience in small and medium-sized companies*, 2025.

generation at inference time. Therefore, PP is theoretically compatible with all other approaches investigated and should be usable in tandem. The plain LLM condition is retained as a baseline and as a fallback should fine-tuned LLMs for SysML v2 become available, but the validation demonstrates that it is not suitable as a default strategy for SysML v2 model generation in the considered setting. With a detailed prompt, the plain LLM achieves approximately 72 percent content coverage but only 12 percent representability, and with an abstract prompt its representability drops to 2 percent. Overall, the electric bike instantiation supports the selector assumptions introduced in Chapter 3. The implications for method selection are consolidated in Chapter 5.

5 Conclusion and Outlook

This work proposes a selector-based approach to automated SysML v2 generation, using the techniques of RAG, GCD, and PP. The results show that prompt completeness strongly influences both architectural fidelity and syntactical correctness, with detailed prompts resulting in significantly higher performance across all approaches. GCD improved the syntax under both prompt versions, indicating its usability as a safeguard that increases the proportion of outputs directly usable in engineering tools, but limits LLM token generation and can result in content coverage failures. RAG achieved the highest content and syntax scores when using detailed prompts, but shows sensitivity to abstract prompts. PP delivered moderate gains by converting almost correct outputs into syntactically correct SysML v2 artifacts, but did not substantially increase architectural completeness when the model output strongly deviated from the reference.

These findings highlight the need to dynamically adapt LLM output in engineering workflows to address both semantic correctness and content improvements. RAG support enhances the reuse of domain knowledge when the user prompt is sufficiently detailed. PP functions as a fallback that allows repair stages to increase syntactic correctness without modifying the content of the output. Plain LLMs alone fail to meet the dual requirements of representability and fidelity in most underspecified scenarios, but should not be generally dismissed, as future LLMs will most likely show improved performance, once a large enough dataset of SysML v2 syntax is publicly available for the LLMs to be trained on.

The results and the fact that PP can in theory be used in tandem with other approaches indicate that a hybrid approach of RAG and PP most likely delivers the best performance when sufficient knowledge from previous developments is available, and needs further research. Especially considering the iterative nature of smart product development, where previous knowledge is typically available, this should be the most used method for engineering assistance. Further research should also be done using graph-based RAG approaches as well as more complex PP tools that include multiple LLMs working on different specialized aspects of a system model, such as requirements, use-case diagrams, or completeness and mapping between various layers of a SysML v2 model.

References

- [1] T. Tomiyama, E. Lutters, R. Stark and M. Abramovici, "Development capabilities for smart products," *CIRP Annals*, vol. 68, no. 2, pp. 727-750, 2019. DOI: <https://doi.org/10.1016/j.cirp.2019.05.010>.
- [2] A. Albers, N. Bursac and E. Wintergerst, "Produktentgenerationsentwicklung - Bedeutung und Herausforderungen aus einer entwicklungsmethodischen Perspektive," in *Stuttgarter Symposium für Produktentwicklung 2015*.
- [3] M. Kretzschmar, M. P. Dammann, S. Schwoch, E. Berger, B. Saske and K. Paetzold-Byhain, "Key Concepts, Potentials and Obstacles for the Implementation of Large Language Models in Product Development," in *Entwerfen Entwickeln Erleben 2024 : Menschen, Technik und Methoden in Produktentwicklung und Design*: Technische Universität Dresden, 2024, pp. 381-393.
- [4] D. Stenholm and D. Bergsjö, "Barriers to reuse of codified engineering knowledge in product development - a literature review," *International Journal of Product Lifecycle Management*, vol. 12, no. 4, p. 307, 2020. DOI: <https://doi.org/10.1504/IJPLM.2020.112773>.

- [5] L. Jokinen and S.-P. Leino, "Hidden product knowledge: problems and potential solutions," *Procedia Manufacturing*, vol. 38, pp. 735-744, 2019. DOI: <https://doi.org/10.1016/j.promfg.2020.01.099>.
- [6] W. J. Verhagen, P. Bermell-Garcia, R. E. van Dijk and R. Curran, "A critical review of Knowledge-Based Engineering: An identification of research challenges," *Advanced Engineering Informatics*, vol. 26, no. 1, pp. 5-15, 2012. DOI: <https://doi.org/10.1016/j.aei.2011.06.004>.
- [7] S. Schleifer, A. Lungu, B. Kruse, S. van Putten, S. Goetz and S. Wartzack, "Minimal Data, Maximal Impact: Language Model-based Pipelines for the Automatic Generation of Use Case Diagrams from Requirements," in *DS 133: Proceedings*, pp. 240-249.
- [8] L. T. Blessing and A. Chakrabarti, *DRM, a Design Research Methodology* London: Springer London, 2009.
- [9] M. Abramovici, J. C. Göbel and H. B. Dang, "Semantic data management for the development and continuous reconfiguration of smart products and systems," *CIRP Annals*, vol. 65, no. 1, pp. 185-188, 2016. DOI: <https://doi.org/10.1016/j.cirp.2016.04.051>.
- [10] C. M. Gardiner, "Knowledge sharing success and resistance in an engineering department: A case study," *Journal of Management & Organization*, vol. 22, no. 2, pp. 254-271, 2016. DOI: <https://doi.org/10.1017/jmo.2015.30>.
- [11] P. Krenz, S. Basmer, S. Buxbaum-Conradi, T. Redlich and J.-P. Wulfsberg, "Knowledge Management in Value Creation Networks: Establishing a New Business Model through the Role of a Knowledge-Intermediary," *Procedia CIRP*, vol. 16, pp. 38-43, 2014. DOI: <https://doi.org/10.1016/j.procir.2014.01.006>.
- [12] D. Birch, H. Liang, P. H. J. Kelly, G. Mullineux, T. Field, J. Ko and A. Simondetti, "Multidisciplinary Engineering Models: Methodology and Case Study in Spreadsheet Analytics," 2014. DOI: <https://doi.org/10.48550/arXiv.1401.4582>.
- [13] M. Chami and J.-M. Bruel, eds., *A Survey on MBSE Adoption Challenges* (2018).
- [14] A. Akundi, W. Ankobiah, O. Mondragon and S. Luna, "Perceptions and the extent of Model-Based Systems Engineering (MBSE) use – An industry survey," in *2022 IEEE International Systems Conference (SysCon)*: IEEE, 2022, pp. 1-7.
- [15] K. Henderson and A. Salado, "The Effects of Organizational Structure on MBSE Adoption in Industry: Insights from Practitioners," *Engineering Management Journal*, vol. 36, no. 1, pp. 117-143, 2024. DOI: <https://doi.org/10.1080/10429247.2023.2210494>.
- [16] A. M. Madni and M. Sievers, "Model-Based Systems Engineering: Motivation, Current Status, and Needed Advances," in *Disciplinary Convergence in Systems Engineering Research*, A. M. Madni, B. Boehm, R. G. Ghanem, D. Erwin and M. J. Wheaton, eds.: Cham: Springer International Publishing, 2018, pp. 311-325.
- [17] A. Mahboob and S. Husung, "A Modelling Method for Describing and Facilitating the Reuse of SysML Models during Design Process," *Proceedings of the Design Society*, vol. 2, pp. 1925-1934, 2022. DOI: <https://doi.org/10.1017/pds.2022.195>.
- [18] R. Mendieta, J. L. de La Vara, J. Llorens and J. M. Álvarez-Rodríguez, "Towards Effective SysML Model Reuse," in *Proceedings of the 5th International Conference on Model-Driven Engineering and Software Development*: SCITEPRESS - Science and Technology Publications, 2017, pp. 536-541.
- [19] M. S. d. Biase, S. Marrone and A. Palladino, *Towards Automatic Model Completion: from Requirements to SysML State Machines*, 2022.
- [20] M. Ariffud, "LLM statistics 2025: Comprehensive insights into market trends and integration," [Online]. Available: <https://www.hostinger.com/tutorials/llm-statistics>. [Accessed 6 November 2025].
- [21] A. Akundi, J. Ontiveros and S. Luna, "Text-to-Model Transformation: Natural Language-Based Model Generation Framework," *Systems*, vol. 12, no. 9, p. 369, 2024. DOI: <https://doi.org/10.3390/systems12090369>.
- [22] H. Kourani, A. Berti, D. Schuster and W. M. P. van der Aalst, "Process Modeling with Large Language Models," in *Enterprise, Business-Process and Information Systems Modeling*, Vol. 511, H. van der Aa, D. Bork, R. Schmidt and A. Sturm, eds.: Cham: Springer Nature Switzerland, 2024, pp. 229-244.
- [23] J. K. DeHart, "Leveraging Large Language Models for Direct Interaction with SysML v2," *INCOSE International Symposium*, vol. 34, no. 1, pp. 2168-2185, 2024. DOI: <https://doi.org/10.1002/iis2.13262>.
- [24] D. Mollahassani, M. Becker, T. Eickhoff, J. Pickel, S. Goetz, S. Wartzack and J. C. Göbel, "Framework for circular AI-driven model-based systems engineering," *Proceedings of the Design Society*, vol. 5, pp. 1883-1892, 2025. DOI: <https://doi.org/10.1017/pds.2025.10202>.
- [25] B. Johns, K. Carroll, C. Medina, R. Lewark and J. Walliser, "AI Systems Modeling Enhancer (AI-SME): Initial Investigations into a ChatGPT-enabled MBSE Modeling Assistant," *INCOSE International Symposium*, vol. 34, no. 1, pp. 1149-1168, 2024. DOI: <https://doi.org/10.1002/iis2.13201>.