Towards Structured Knowledge Reuse to Support Design for Manufacturability: A Case-Based Reasoning Approach

Kokulan Thanabalan[1,3,*], Lukas Malburg[1,3], Leonhard Kunz[2], Ralph Bergmann[1,3]

[1]University Trier, Business Information Systems II – Artificial Intelligence and Intelligent Information Systems, 54296 Trier
[2]University of Kaiserslautern-Landau (RPTU), 67663 Kaiserslautern
[3]German 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

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

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:

$$\text{Case} = \langle \text{Meta Data, F, B, S, Solution, Outcome} \rangle$$

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.

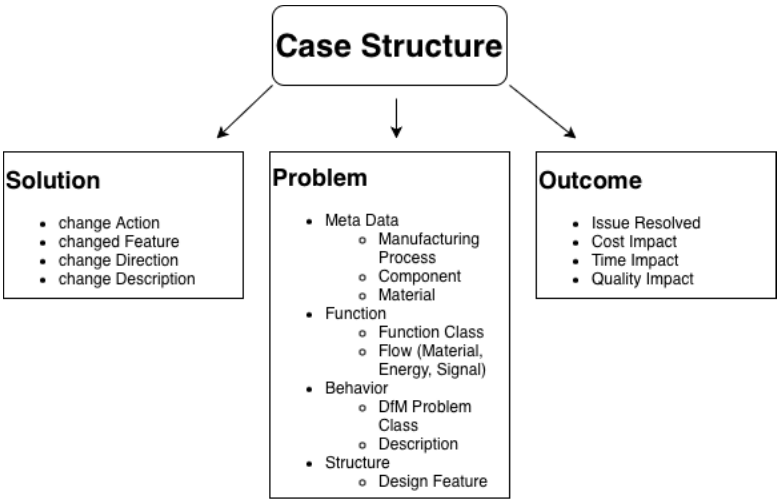

Figure 1: Case Representation Structure

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.

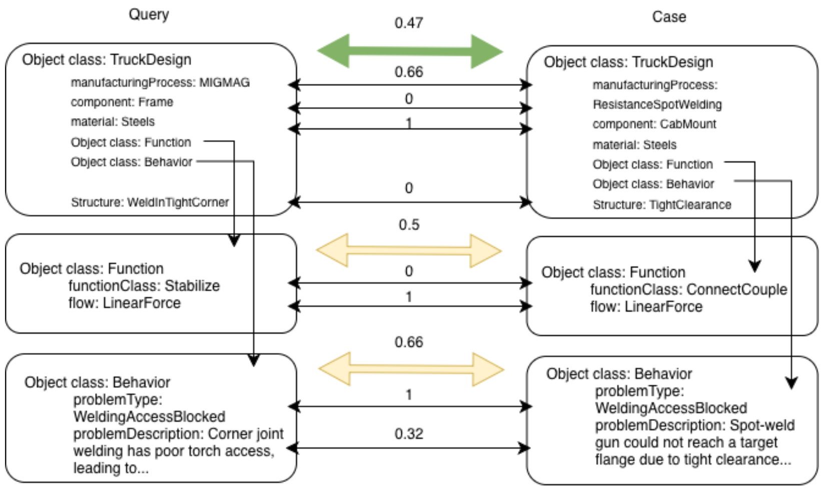

Figure 2: Example Retrieval with Similarity Results

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 appeares 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 DFMPro." 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. Wartzack, 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.