



---

Fachausschuss

Jahrgang 30 Heft 1

**Management der  
Anwendungsentwicklung  
und -wartung (WI MAW)**

ISSN 1610 5753

im FB Wirtschaftsinformatik

Mai 2024

---

## Inhalt

Fachbeiträge.....	3
Ankündigungen .....	49
Berichte .....	61
Organisation .....	67

**48. WI-MAW-Rundbrief**

## Inhaltsverzeichnis

### Fachbeiträge

<i>What Got You Here Won't Get You There - A multi-case study on the challenges in the transition from traditional towards continuous data practices in the embedded systems domain</i> Helena H. Olsson, Jan Bosch .....	3
<i>Evaluation eines LLM-basierten Text2SQL-Prototyps</i> Gero Pocha, Alexander Rachmann .....	15
<i>Generative künstliche Intelligenz und Kreativität: Erkenntnisse aus grauer Literatur</i> Alexander Rachmann, Alexander Hiller .....	25
<i>How does the role of a Product Owner relate to the role of a Software Product Manager?</i> Timo Toikkanen, Sami Hyrynsalmi, Maria Paasivaara .....	37

### Ankündigungen

Call for Participation: Digital Product Management Week, Göteborg, 10. - 14. Juni 2024 ....	49
Call for Participation: KI-basierte Analyse rollenbasierter Gesprächsverläufe, SOWI Gebäude Uni Innsbruck, 10. Juni 2024 .....	51
Call for Papers: PVM 2024 – Neues Arbeiten in Projekten – Teamarbeit neu interpretiert, TH Mittelhessen, Campus Friedberg, 26 + 27. September 2024 – Einreichfrist 15. Juni 2024 .....	53
Buchhinweis: Veränderungen im Software Engineering durch den Einsatz von „low code“ orientierten Softwareentwicklungsansätzen, ISBN 978-3-8325-5733, erscheint im 2. Quartal 2024.....	57

### Berichte

Tagungsbericht zur 1. Internationalen Konferenz für Software Produktmanagement (ICSPM2023) (Dimitri Petrik) .....	61
Bericht zum ESAPI 2023 Workshop „KI-Szenarien im Zeitalter von ChatGPT & Co“ (An- dreas Schmietendorf).....	63

### Organisation

Der Fachausschuss „Management der Anwendungsentwicklung und –wartung“ WI-MAW und die Fachgruppen Vorgehensmodelle für die betriebliche Anwendungsentwicklung WI-VM Projektmanagement WI-PM Software Produktmanagement WI-PrdM stellen sich vor .....	67
---	----



# What Got You Here Won't Get You There

## A multi-case study on the challenges in the transition from traditional towards continuous data practices in the embedded systems domain

Helena H. Olsson, Jan Bosch

Malmö University, Department of Computer Science and Media Technology, 211 19 Malmö,  
helena.holmstrom.olsson@mau.se, <https://orcid.org/0000-0002-7700-1816>

Chalmers University of Technology, Department of Computer Science and Engineering,  
41296 Gothenburg, Sweden,

jan.bosch@chalmers.se, <https://orcid.org/0000-0003-2854-722X>

**Abstract:** For decades, product data has been collected and used for quality assurance, for post-deployment defect detection and for informing the next generation of products. Across industry domains, and with the online domain leading the way, companies have adopted experimentation and data driven practices such as A/B testing to evaluate product performance, customer behaviors and for determining what adds value to customers. However, with the rapid changes that new digital technologies bring, companies are moving towards continuous value delivery and monetization models in which they offer their products as-a-service or offer services to complement and extend their existing products. In this transition, the traditional way of post-deployment data collection and use is no longer sufficient. While companies realize this, they experience difficulties in making the changes they need to transition towards continuous practices and new ways of working with data. As a result, companies risk wasting development efforts on functionality that have little or no customer value and they lose out on the competitive advantages that come with insights derived from continuous collection and use of data. In this paper, we explore the challenges companies experience in the transition from traditional towards continuous practices and the implications this shift has on their ways of working with data.

**Keywords:** Digitalization, digital transformation, data practices, continuous practices, continuous customer value delivery.

## 1 Introduction

Due to digitalization and technologies such as software, data, and artificial intelligence (AI), companies across domains are experiencing a fundamental shift in how to develop, deliver and monetize customer value. As recognized by e.g., [BO21], [PH14], [HE18], [OS22], digital technologies and digitization of products allow companies to expand and improve value creation in their existing products at the same time as they can provide customers with entirely new value in the form of e.g., data-driven service offerings and digital products. With the many opportunities to collect and leverage data generated by products in the field, companies are focusing their efforts on exploiting this data for competitive advantage [PF13]. By collecting, processing and actively using data generated by connected products, companies can advance not only their software engineering practices and their products, but also the understanding of their customers and what adds value to them. For companies in the embedded systems domain, software is rapidly becoming the central differentiator, whereas the traditional technologies such as mechanics and electronics are becoming commodities [Bo15], [BO21], [OB20]. With value creation being shifted from hardware to software, these companies experience a situation in which they can update and continuously improve their products after manufacturing and deployment at the customer. While this significantly extends the lifetime of a product and the experience of the customer, it also allows for new digital business models and recurring revenue

streams [BO21], [Ge20], [LP15]. For decision-making, the increasing availability and access to data allows for entirely new ways-of-working characterized by data-driven approaches to e.g., feature prioritization, customer validation, and product and service innovation. As examples, companies that used to be heavy on waterfall approaches to development are adopting iterative and customer-centric methods such as design thinking and lean start-up to understand and empathize with customers [BO21], [CF18], [DP19]. Moreover, techniques such as A/B testing and feature experimentation are being explored to continuously evaluate and improve customer value [Fa17], [Fa17], [Is21], [Li21].

However, with the rapid transition towards continuous value delivery and monetization models, the traditional way of post-deployment data collection and use is no longer sufficient. Although companies realize this, they struggle in making the changes they need to transition towards continuous practices and new ways of working with data. In this paper, and to address this problem, we explore data practices in embedded systems companies. Based on multi-case study research, we explore the challenges companies experience in the transition from traditional towards continuous practices and the implications this shift has on their ways of working with data. The contribution of this paper is three-fold. First, we identify the key challenges that companies in the embedded systems domain experience in the transition towards continuous data practices. Second, we derive four organizational anti-patterns that we see reduce the benefits of data practices in large software-intensive embedded systems companies. Third, we provide a set of recommendations to help companies evolve beyond their current state.

The remainder of the paper is structured as follows. In section 2, we review literature and related work on digitalization and data practices. In section 3, we describe the research method that was adopted in the study. In section 4, we present our empirical findings, and we identify the key challenges the case companies experience. In section 5, we derive four organizational anti-patterns that we see reduce the benefits of data practices in software-intensive embedded systems companies and we provide a set of recommendations to help companies advance their current data practices. In section 6, we discuss threats to validity. Finally, in section 7, we conclude the paper, and we outline opportunities for future research.

## **2 Background and related work**

### **2.1 Digitalization and digital transformation**

At the core of digitalization and digital transformation is the opportunity for continuous value delivery to customers. Technologies such as software, data and AI transform the ways in which business entities operate, how they create value and how value is delivered and monetized with customers. According to [MHB15], digital transformation brings the opportunity for increases in sales and productivity, innovations in value creation, as well as novel forms of interaction with customers. In previous research, data is recognized as a key component to innovations and opportunities for new value creation and monetization [BO21], [BZN15], [IV19], [Kr22], [MHB15]. In [BZN15], data is referred to as the “new oil” and capitalizing on data is described as increasingly important for a business to remain competitive. As recognized in this research, businesses are developing new business models specifically designed to create additional business value by extracting, refining, and capitalizing on data. Similarly, in [KB19], the authors refer to data-driven business models as service-oriented business models which use data as key for new value propositions to customers. In our own previous work [BO21], we study how companies are transforming towards digital companies and how data is critical in this evolution. Also, we outline how data is becoming an asset as the basis for data driven and digital services allowing for recurrent revenue streams [OB22]. As examples of such services, data is used to provide insights, recommendations, and actions to customers. As a next step, data from customers can be aggregated and used to provide comparative analysis and insights. However,

in the transition towards continuous practices, the traditional and often ad-hoc way of data collection is no longer sufficient [OB13]. In the following sections, we describe data practices and how these practices are shifting in character. As recognized in previous research, this shift brings numerous opportunities, but also challenges as companies need to adopt new ways in which they work with data.

## **2.2 Data practices**

The collection and use of data in embedded systems companies is not a new phenomenon. Previous research has described the many benefits with using data as the basis for e.g., feature prioritization, for understanding feature usage, for innovation purposes, and for decision-making in organizations [BE12], [OB14], [Ro20]. If looking at the online domain, data practices such as A/B testing and feature experimentation are used on a continuous basis to learn how the introduction of new features affect user experience, satisfaction, and system performance [Fa22], [Fa17], [KT17]. In [Au21], the authors describe how decisions regarding what features to build are difficult for software development organizations and that the effect of an idea and its return-on-investment might not be clear before its launch. Moreover, the evaluation of an idea might be expensive. In most organizations, this leads to a situation in which decisions are taken based on opinions and experience and on assumptions on what adds value to customers. With A/B testing, organizations can evaluate different versions of a software feature and collect data on which version performs the best [De17], [KTX20]. Based on this, data driven decisions can be taken regarding future development, improvement, and optimization of features. In [RBR22], A/B testing is described as an experiment-driven software engineering approach where assumptions about product features and requirements are continuously tested with users with the intent to reduce the risk of wasting development resources on requirements of little or no value to users. Similarly, [Da21], [Fa17], [Is21], and [OB14], recognize how use of data for experimentation purposes helps companies in the embedded systems domain minimize the risk of developing software that does not deliver value to customers. However, despite well-known benefits, the adoption of data driven practices is relatively slow. While the opportunities are many, the transition towards fully automated data practices requires not only new techniques and tools but also architectures and infrastructures as well as new competences and skills [Li21].

## **2.3 Towards continuous data practices**

To realize the opportunities that come with continuous value delivery, companies are increasingly adopting continuous practices. For years, companies have been adopting DevOps practices. In [Mu20], DevOps is defined as a set of practices that helps to build a collaboration between software development and operations which reduces the software development lifecycle and helps in continuous and fast delivery of high-quality software. With DevOps, customer value is created in short cycles and deployed on a frequent, or even continuous, basis [Lw16]. Similarly, DataOps is a practice which aims at bridging the gap between data and operations teams and is viewed as an application of DevOps but for data analytics [Mu20]. DataOps practices are viewed as effective means to help companies make meaningful use of data and for keeping track of data and the purpose for which it was collected [BZ12], [Fi21]. In a report from Gartner, DataOps is described as a collaborative data management practice with the goal of delivering value faster by creating predictable delivery and change management of data, data models and related artifacts [DO23]. For companies in the embedded systems domain, DataOps is one of several continuous practices that support continuous value delivery to customers but that requires a fundamental shift in how these companies work with data. In what follows, we present a study in which we explore the challenges embedded systems companies experience in the transition from traditional towards continuous practices and the implications this shift has on their ways of working with data.

### 3 Research Method

The research presented in this paper is part of a long-term collaboration between 17 companies in the embedded systems domain and five Swedish universities ([www.software-center.se](http://www.software-center.se)). For more than a decade, we have had the privilege to engage with these companies in case study research [Ea08], [Ma12] within the field of software engineering. The companies represent different domains, and they share the similar experiences of digital transformation and the challenges and opportunities that come with new technologies such as software, data, and AI. Currently, all companies are exploring data practices and how to benefit from these. In this paper, we report on on-going research that was initiated in January 2022 in which we explore data practices in a selected set of the companies. In alignment with our research interests, we adopted a qualitative research approach with case studies as our method. Case study research is well-suited for research concerned with identifying patterns of action and for studying organizational contexts in which emphasis is put on stakeholder's perceptions and experiences [Ea08].

#### 3.1 Case companies

As our empirical basis, we selected a set of primary and secondary case companies. As described in [Ge09] and [SG08], case study selection is critical as case study research is about something larger than the cases themselves. Typically, the chosen cases are asked to represent a population of cases that is larger than the cases themselves, and therefore, background cases play an important role for analysis. In accordance with this, the empirical findings we present build on research in three companies that were selected as primary cases. As secondary cases, we selected three companies that experience opportunities and challenges very similar to the primary case companies. The three *primary case companies* are briefly described below:

**Case company A:** A company providing product development, marketing, engineering, sales and support for crew planning and optimization. For this paper, we engaged with a team involved in development of new service offerings and roles representing software, architecture, and portfolio.

**Case company B:** A company manufacturing trucks, buses, and construction equipment as well as a supplier of marine systems. For this paper, we engaged with a team responsible for new service innovation and development.

**Case company C:** A company developing autonomous driving and advanced driver-assistance systems. For this paper, we engaged with product owners in areas such as fleet insight, data driven data management and data governance, as well as roles involved in business development and strategy.

In addition to the three primary case companies, we studied three *secondary case companies*:

**Case company D:** A company developing pump units, circulator pumps, submersible pumps, and centrifugal pumps. For this paper, we engaged with roles responsible for product management, sales, and architecture.

**Case company E:** A company manufacturing network cameras, access control, and network audio devices for the security and surveillance industries. For this paper, we engaged with roles responsible for software development, architecture, platform development, engineering, and management.

**Company F:** A company providing networking and telecommunications solutions and services. For this paper, we engaged with roles in management and software engineering.

### 3.2 Data collection and analysis

For data collection, we engaged in workshop sessions with both the primary case companies and the secondary case companies involved in our study. These workshops were organized either online or at the different company sites and gave us the opportunity to meet with key stakeholders in teams involved in, and responsible for, data collection, data analytics and data usage. Although our research collaboration with these companies covers more than a 10-year period, the study we report on in this paper was initiated in January 2022 and is on-going. During 2022, we organized on-site workshops at all primary case companies (company A, B and C) and at the three secondary case companies (company D, E and F). In addition, we met with all case companies in online workshops. The on-site workshop sessions lasted for 2 – 4 hours and involved 4 – 8 people. The online workshop sessions were typically shorter (30 minutes – 1-hour sessions) and a way to follow-up, share ideas and results and for monitoring progress in between the on-site meetings. In total, the research we report on in this paper is based on 12 on-site company workshops and 18 online workshops. In addition to these workshops, we met with both the primary and the secondary case companies at *two larger cross-company full-day events* that we organized to report our preliminary findings and get company feedback. As this research is on-going, we have company workshops with all case companies scheduled during spring 2023 and our intention is to scale the interactions with the secondary case companies. For data analysis, we adopted an interpretive approach [Ea08], [Ma12], [Wa95]. As suggested by [Wa95] the generalizations that are made based on case study research are valuable for other organizations that experience similar challenges in similar contexts to the case companies.

## 4 Empirical findings

For decades, the case companies involved in our study have harvested huge amounts of data from e.g., development and test fleets, from internal systems, and from products in the field. They have sophisticated infrastructures and systems in place for adding events, enabling queries and questions, and for introducing and monitoring metrics. So far, the data has been used primarily for quality assurance of products, for post deployment defect detection, and for informing the next generation of products. However, due to digitalization and the many opportunities that come with new digital technologies, the companies are experiencing a rapid shift from traditional and product-oriented business models towards continuous and service-oriented business models. In this transition, the traditional, and often ad-hoc, way of post-deployment data collection and use is no longer sufficient. Instead, all the case companies seek to adopt new and more continuous ways of working with data. This implies a shift towards periodic and, in the end, continuous and automated collection, processing, and use of data.

Below, we identify the *key challenges* the case companies experience in the shift from traditional towards continuous data practices. We structure our findings according to generic challenges that are prevalent both in traditional and continuous practices and challenges that we identify as unique for continuous data practices.

### 4.1 Generic challenges that are prevalent in both traditional and continuous data practices

**Combining qualitative and quantitative data:** Both the primary and the secondary case companies involved in our study experience challenges with how to effectively combine different data sources. This involves how to generate insights that build on both qualitative data generated within the company, and quantitative data generated by products in the field. In traditional data practices, organizations rely heavily on individuals to collect, process, and



analyze data based on requests from e.g., product management or from development teams. Often, such requests are ad-hoc requests concerning a specific feature and its behavior and requires manual efforts in identification and analysis of relevant data. In the shift towards continuous practices, companies need automated solutions that help convert qualitative data into quantitative data to enable frequent monitoring and control of the data analytics pipeline. As a common challenge in our case companies, people report on difficulties in e.g., combining feedback from customers with data from internal build systems (continuous integration and continuous deployment systems) and data generated by products in the field. While customer feedback typically reflects qualitative experiences of the overall system and individuals' perceptions on usability, build system and product data reflects quantitative measures of internal efficiency and performance.

**Incorporating external data with internal data:** Throughout our study, we learnt that the case companies experience difficulties when trying to incorporate external data produced by third parties, and other relevant data sources generated outside company boundaries, into their own data streams and as a complement to their own data sources. For example, while sources such as e.g., market trend data and social media data are viewed as sources that provide potentially valuable insights about customer behaviors, these are also perceived as more challenging to keep accurate to avoid invalid or inaccurate data. As companies are moving towards continuous data practices, one of the main challenges is to ensure that the time and period of the internal data is aligned with the time and period of the external data.

**Understanding the surrounding context of a metric:** Although the case companies have a large set of metrics in place in their systems, they lack effective mechanisms that help them understand the surrounding context of a specific metric. For example, a metric can capture a certain action taken by a user but very often information such as when the activity took place, who initiated and performed the action, what the purpose or use case of the action was etc., is missing. As a result, the analysis of what a metric reflects becomes difficult as the context of the data that is collected is lacking. For example, one of the primary case companies describes a situation in which the same metrics are used to capture two user groups with very different behaviors. While one customer group interacts with the system on a very frequent basis for solving highly complex optimization tasks, the other customer group uses the system to perform basic tasks and with as little interaction as possible. Metrics are the same however, and during one of our workshops one of the product managers described the situation as “...*comparing apples and pears*” referring to the same metrics being used to capture two very different use cases and user scenarios.

## 4.2 Challenges unique for continuous data practices

**Limited scope of metrics:** In the transition towards continuous data practices, the scope of metrics becomes increasingly important. While metrics in traditional data practices tend to focus primarily on performance and quality aspects of the system, metrics in continuous practices need to capture not only a certain aspect at a certain point in time, but also how aspects of the system change over time. In our study, the case companies report on challenges with regards to the scope of metrics and how to ensure that existing metrics capture accurate data on e.g., frequency, rate, duration, or interval of an event. Especially, the case companies experienced challenges with regards to how to continuously integrate changes in data while maintaining and ensuring high quality of data.

**Metrics are “static” and viewed as “cast in stone”:** In the case companies we studied, metrics tend to show feature usage in terms of activation of the feature rather than providing insights that capture usage patterns, complexity of a user task, purpose of use or how feature usage changes over time etc. Most often, metrics represent the behavior of a feature or of a system as it was once specified in a requirement (static), rather than providing an understanding of what

the feature or system should do and what could be expected from it in its evolution (dynamic). In addition, and as experienced in most of the case companies, people at different levels and in different functions often view metrics as “cast in stone”, leading to a situation in which the introduction of new metrics, experimentation with metrics, and removal of existing metrics is regarded very difficult even if this is key for continuous data practices.

**Monitoring overall improvement:** All case companies have DevOps practices in place and periodic deployment of software allow them to continuously measure and monitor basic key performance indicators (KPIs). However, we noticed that despite this, the companies often lack mechanisms to determine if things are overall improving or not. In several workshops, teams reported how they continuously improve specific features and how they can monitor these improvements over time. Still, they are unable to understand to what extent, or if at all, the feature improvements they do contribute to an overall improvement of the system. In our experience, this is due to an unclear desired state meaning that teams and organizations don’t know, or don’t align, on what they are optimizing for. Also, slow customer activation of frequently deployed software functionality makes it difficult for continuous monitoring of overall system improvement. As one example, one case company experiences a situation in which the data coming back from their products is so complex that most developers have difficulties in understanding it. In one of our onsite workshops, one of the key stakeholders reflected on this and described it as *“You either have hardware people or you have software people but to combine these skills are hard. In addition, we have a problem with granularity as there are so many variables and factors that interact.”* For the company, this results in a situation where teams track a sub-set of metrics but where improvement of overall system performance is more difficult to monitor. This situation is valid also for the other case companies as they all report on difficulties in fully benefitting from the data they have available. Often, there is little agreement on what KPIs are the critical ones and hence, how team and feature level metrics correspond to high-level business metrics. Also, one problem is that initial data collection is often concerned with a specific use case while continuous improvements of this use case might make the case for analytics a different one and therefore, contextual data is required to fully benefit from data and analytics as mechanisms to monitor overall system improvement.

## 5 Discussion

### 5.1 Organizational anti-patterns

As reported in this paper, the use of data practices is challenging for companies with systems involving not only digital technologies but also mechanics and electronics. As a generalization of what we see happen in the case companies, we derive four anti-patterns that reduce the benefits of data driven practices in large-software-intensive embedded systems companies. An anti-pattern in software engineering, project management, and business processes is a common response to a recurring problem that is ineffective and that risks being highly counterproductive [Ga95]. In both the primary and the secondary case companies, we see the following anti-patterns (Table 1):

Anti-pattern:	Description:
<b>The “worthwhile many versus the vital few” anti-pattern<sup>1</sup></b>	The lack of a shared understanding of the desired state makes prioritization and resource allocation

<sup>1</sup> The ‘vital few’ is a term derived from the 80-20 rule (the Pareto principle) which asserts that about 80% of all outcomes are the direct result of only 20% of all inputs.

	difficult. This results in individuals and teams spending time and efforts on activities that don't directly contribute to business value and success.
<b>The “homonym” anti-pattern</b>	Companies fail to realize that they use the same metrics to capture two (or more) different customer groups with different preferences. This results in difficulties in analyzing and benefitting from any data that is collected, as well as an inadequate metrics system reflecting only a subset of what it potentially could.
<b>The “what got you here won't get you there” anti-pattern</b>	KPIs reflect requirements that was once accurate and that describe feature and system behavior but that have stopped being true. This results in an insufficient metrics system reflecting current state but that fail in capturing desired state, i.e., what the feature and system should do and what we expect from it over time.
<b>The “Alice in Wonderland – if you don't know where you're going, any road will take you there” anti-pattern</b>	The lack of a desired outcome and agreed upon key value factors make individuals and teams optimize for their own best but without a holistic understanding of the business. This results in conflicting KPIs, suboptimization of team efforts, and misalignment in business outcomes.

Table 1: Anti-patterns that reduce the benefits of data practices.

## 5.2 Recommendations

In this section, we provide a set of recommendations with the intention to help companies advance their data practices and evolve beyond their current state. The recommendations are inductively derived based on the insights we gained during this study, and during our long-term collaboration with the case companies. To further advance data practices in software-embedded systems companies, we recommend the following:

**Experiment with proxy metrics that are indicative of customer value:** Typically, the things companies wish to measure are difficult to measure. For example, most companies struggle with how to clearly describe what constitutes customer value and therefore, measuring customer value becomes a very difficult task. The consequence is that companies stop trying and fall back on traditional metrics focusing on internal efficiency, product performance and quality as indirect measures of value. Instead, we recommend that companies hypothesize proxy metrics that are indicators of customer value, measure these over multiple releases and do correlation analysis between confirmed customer value delivery and the proxy metric with the intent of identifying metrics that have a strong correlation with customer value.

**Seek to continuously shorten feedback cycles with customers:** The ability to continuously evaluate improvements relies on accurate and frequent feedback from customers. In addition to being critical for validation of improvements, customer feedback is important as companies seek to minimize development efforts and investments in between proof points. The shorter feedback cycle, the faster teams learn, and the smaller the investment is if a feature turns out to not prove valuable. One of the key enablers to this is DevOps practices. We recommend companies to adopt DevOps capabilities for at least parts of their systems as it allows for the opportunity to start running A/B experiments to measure the impact and value of new features on a frequent basis.

**Develop an experimentation infrastructure:** The ability to run experiments provides companies with the foundation to learn about feature and system usage and about what adds value to customers. A/B testing is a powerful mechanism for identifying and evaluating value and it forms the basis for a hierarchical value model in which metrics at different levels contribute to an overall understanding of customer and business value. We recommend companies to develop an experimentation infrastructure and capabilities involving e.g., randomization algorithms, assignment methods, and the data processing mechanisms. To scale these practices, we recommend companies to have dedicated groups that build, manage, and improve the experimentation infrastructure so that it can be effectively employed by many teams.

**Develop a hierarchical value model:** Establishing a quantitative understanding of the use of the system is critical. Companies can either use historical data to understand usage of a certain feature or instrument the software to start collecting data revealing feature usage. To evaluate feature improvements, and to understand if the overall system is improving as desired, companies need a hierarchical value model. In such a model, there needs to be a single, or at least very few, high-level metrics that translates into lower-level metrics and to which the lower-level metrics contribute. A hierarchical value model details the relationships between metrics and incorporates the tradeoff among lower-level metrics. With a hierarchical value model in place, companies can continuously monitor and evaluate value to ensure that development efforts and investments are allocated to functionality with proven customer value.

**Continuously validate the hierarchical value model and evolve it over time:** To avoid misalignment between teams and sub-optimization of efforts, the value model needs to be maintained over time. This involves having teams responsible for parts of the system and its related metrics, it involves having teams responsible for the overall alignment of the value model as it evolves, and it involves continuous introduction and evaluation of new metrics. Over time, evaluating and adjusting the high-level metric(s) and understanding causes and effects becomes easier. By running experiments and interpreting the results companies will not only learn what metrics work best for certain types of experiments but also learn how to develop and introduce new metrics into the value model.

## 6 Threats to validity

As the basis for our understanding of digital transformation and the role of data in the shift towards continuous practices, we reviewed contemporary literature on this topic. Based on this, we conducted multi-case study research in selected primary and secondary case companies in the embedded systems domain. To mitigate validity threats, and to address construct validity [Ma12], we started each workshop with sharing our view on digitalization and the impact digital transformation has on the ways in which companies work with data. By doing this, we established a common understanding, and we could focus the discussions using terminology that was familiar for everyone involved. With regards to external validity, our research contributions are related to what Walsham [Wa95] defines as “drawing of specific implications” and as a contribution of “rich insights”.

## 7 Conclusions and future research

At the core of digitalization and digital transformation is the shift towards continuous value delivery to customers. For companies in the embedded systems domain, this shift implies that product sales and transactional business models are increasingly being complemented with service sales and recurring revenue streams. For companies in the embedded systems domain, this includes traditional service approaches where the physical product is offered as a service

or where other aspects, such as maintenance of the product, are provided as a service [BO21]. In this transition, data practices are critical as they lay the basis for new data driven and digital offerings. In this paper, we explore the challenges companies experience in the transition from traditional towards continuous data practices and the implications this shift has on their ways of working with data. In the paper, we identify the key challenges that companies in the embedded systems domain experience in the transition towards continuous data practices. Second, we derive four organizational anti-patterns that we see reduce the benefits of data practices in large software-intensive embedded systems companies. Third, we provide a set of recommendations to help companies evolve beyond their current state.

In future research, we aim to study how to better combine and make effective use of different data sources that companies have available. Our goal is to provide a holistic understanding of the opportunities that continuous data practices bring and how different organizational roles can benefit from these for decision-making purposes.

## References

- [Au21] Auer, F.; Ros, R.; Kaltenbrunner, L.; Runeson, P.; and Felderer, M: Controlled experimentation in continuous experimentation: Knowledge and challenges. *Information and Software Technology*, 134, p.106551, 2021.
- [Bo15] Bosch, J: Speed, data, and ecosystems: the future of software engineering. *IEEE Software*, 33(1), pp.82-88, 2015.
- [BE12] Bosch, J.; and Eklund, U: Eternal embedded software: Towards innovation experiment systems. In *Leveraging Applications of Formal Methods, Verification and Validation. Technologies for Mastering Change*, Springer Berlin Heidelberg, pp. 19-31, 2012.
- [BO21] Bosch, J.; and Olsson, H.H: Digital for real: A multicase study on the digital transformation of companies in the embedded systems domain. *Journal of Software: Evolution and Process*, 33(5), p.e2333, 2021.
- [BZN15] Brownlow, J.; Zaki, M.; Neely, A.; and Urmetzer, F: Data and analytics-data-driven business models: A Blueprint for Innovation. *Cambridge Service Alliance*, 7, February, pp.1-17, 2015.
- [BZ12] Buse, R.P.; and Zimmermann, T.; Information needs for software development analytics. In *2012 34th International Conference on Software Engineering (ICSE)*, pp. 987-996, IEEE, June 2012.
- [CF18] Corral, L.; and Fronza, I: Design thinking and agile practices for software engineering: an opportunity for innovation. In *Proceedings of the 19th Annual SIG Conference on Information Technology Education*, pp. 26-31, September 2018.
- [Da21] Dakkak, A.; Zhang, H.; Mattos, D.I.; Bosch, J.; and Olsson, H.H: Towards continuous data collection from in-service products: Exploring the relation between data dimensions and collection challenges. In *Proceedings of the 28th Asia-Pacific Software Engineering Conference (APSEC)*, pp. 243-252, IEEE, December 2021.
- [DO23] DataOps, <https://www.gsrtnr.com/en/information-technology/glossary/dataops>, accessed January 26th, 2023.
- [De17] Deng, A.; Dmitriev, P.; Gupta, S.; Kohavi, R.; Raff, P.; and Vermeer, L: A/B Testing at Scale: Accelerating Software Innovation. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '17)*. Association for Computing Machinery, New York, NY, USA, pp. 1395–1397, 2017.
- [DP19] Dobrigkeit, F.; and de Paula, D: Design thinking in practice: understanding manifestations of design thinking in software engineering. In *Proceedings of the 2019 27th ACM joint meeting on European software engineering conference and symposium on the foundations of software engineering*, pp. 1059-1069, 2019

- [Ea08] Easterbrook, S.; Singer, J.; Storey, M-A.; and Damian, D: Selecting empirical methods for software engineering research. *Guide to advanced empirical software engineering*, pp. 285–311, Springer, 2008.
- [Fa22] Fabijan, A.; Dmitriev, P.; Bosch, J.; Olsson, H.H: The Evolution of Continuous Experimentation in Software Product Development: From Data to a Data-Driven Organization at Scale. In: Bosch, J.; Carlson, J.; Olsson, H.H; Sandahl, K.; and Staron, M. (eds) *Accelerating Digital Transformation*. Springer, Cham, 2022.
- [Fa17] Fabijan, A.; Dmitriev, P.; Olsson, H.H.; and Bosch, J: The Benefits of Controlled Experimentation at Scale, In *Proceedings of the 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, pp. 18–26, 2017.
- [Fa17] Fagerholm, F.; Guinea, A.S.; Mäenpää, H.; and Münch, J: The RIGHT model for continuous experimentation. *Journal of Systems and Software*, 123, pp.292-305, 2017.
- [Fi21] Figalist, I.; Elsner, C.; Bosch, J.; Olsson, H.H: Fast and curious: A model for building efficient monitoring- and decision-making frameworks based on quantitative data. *Information and Software Technology*, Volume 132, 106458, ISSN 0950-5849, 2021.
- [Ga95] Gamma, E.; Helm, R.; Johnson, R.; Johnson, R.E.; and Vlissides, J: *Design patterns: elements of reusable object-oriented software*. Pearson Deutschland GmbH, 1995.
- [Ge20] Gebauer, H.; Arzt, A.; Kohtamaki, M.; Lamprecht, C.; Parida, V.; Witell, L.; and Wortmann, F: How to convert digital offerings into revenue enhancement–conceptualizing business model dynamics through explorative case studies, *Industrial Marketing Management*, vol. 91, pp. 429– 441, 2020.
- [Ge09] Gerring J: What is a case study and what is it good for? *Am Polit Sci Rev*. 98(2), pp. 341-354, 2009.
- [HE18] Hunke, F.; and Engel, C: Utilizing data and analytics to advance service: Towards enabling organizations to successfully ride the next wave of servitization, Springer International Publishing, pp. 219-231, 2018.
- [Is21] Issa Mattos, D.; Dakkak, A.; Bosch, J.; and Olsson, H.H: The HURRIER process for experimentation in business-to-business mission-critical systems. *Journal of Software: Evolution and Process*, p.e2390, 2021.
- [IV19] Ivančić, L.; Vukšić, V.B.; and Spremić, M: Mastering the digital transformation process: Business practices and lessons learned. *Technology Innovation Management Review*, 9(2), 2019.
- [KT17] Kohavi, R.; and Thomke, S: The Surprising Power of Online Experiments, *Harvard Business Review*, October 2017.
- [KTX20] Kohavi, R.; Tang, D.; and Xu, Y: *Trustworthy online controlled experiments: A practical guide to A/B testing*. Cambridge University Press, 2020.
- [Kr22] Kraus, S.; Durst, S.; Ferreira, J.J.; Veiga, P.; Kailer, N.; and Weinmann, A: Digital transformation in business and management research: An overview of the current status quo. *International Journal of Information Management*, 63, 102466, 2022.
- [KB19] Kühne, B.; and Böhmman, T: Data-Driven Business Models: Building the Bridge between Data and Value. In *Proceedings of European Conference on Information Systems*, June 2019.
- [Li21] Liu, Y.; Bosch, J.; Olsson, H.H.; and Lantz, J: An architecture for enabling A/B experiments in automotive embedded software. In *Proceedings of the 45th Annual Computers, Software, and Applications Conference (COMPSAC)*, pp. 992-997, IEEE, 2021.
- [LP15] Loebbecke, C; and Picot, A: Reflections on societal and business model transformation arising from digitization and big data analytics: A research agenda, *The Journal of Strategic Information Systems*, vol. 24, no. 3, pp. 149–157, 2015.
- [Lw16] Lwakatare, L.E.; Karvonen, T.; Sauvola, T.; Kuvaja, P.; Olsson, H.H.; Bosch, J.; and Oivo, M.: Towards DevOps in the embedded systems domain: Why is it so hard? In *Proceedings*

- of the 49th Hawaii International Conference on System Sciences, pp. 5437-5446, IEEE, 2016.
- [MHB15] Matt, C.; Hess, T.; and Benlian, A: Digital transformation strategies. *Business & Information Systems Engineering*, 57, pp.339-343, 2015.
- [Ma12] Maxwell, J.A: *Qualitative research design: An interactive approach*. Sage publications, 2012.
- [Mu20] Munappy, A.R.; Mattos, D.I.; Bosch, J.; Olsson, H.H.; and Dakkak, A: From ad-hoc data analytics to Dataops. In *Proceedings of the International Conference on Software and System Processes*, pp. 165-174, 2020.
- [OB13] Olsson, H.H.; and Bosch, J: Post-deployment data collection in software-intensive embedded products. In *Software Business. From Physical Products to Software Services and Solutions: In Proceedings of the 4th International Conference on Software Business (ICSOB)*, Springer Berlin Heidelberg, pp. 79-89, 2013.
- [OB14] Olsson, H.H.; and Bosch, J: The HYPEX model: from opinions to data-driven software development. In *Continuous software engineering*, Cham: Springer International Publishing, pp. 155-164, 2014.
- [OB20] Olsson, H.H.; and Bosch, J.; *Going Digital: Disruption and transformation in software-intensive embedded systems ecosystems*. *Journal of Software: Evolution and Process*, 32(6), p.e2249, 2020.
- [OB22] Olsson, H.H.; and Bosch, J: Living in a Pink Cloud or Fighting a Whack-a-Mole? On the Creation of Recurring Revenue Streams in the Embedded Systems Domain. In *Proceedings of the 48th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, pp. 161-168, IEEE, 2022.
- [OS22] Ostmeier, E.; and Strobel, M: Building skills in the context of digital transformation: How industry digital maturity drives proactive skill development. *Journal of Business Research*, 139, pp.718-730s, 2022.
- [PH14] Porter, M.E.; and Heppelmann, J.E: How smart, connected products are transforming competition. *Harvard business review*, 92(11), pp.64-88, 2014.
- [PF13] Provost, F.; and Fawcett, T: Data science and its relationship to big data and data-driven decision making. *Big data*, 1(1), pp.51-59, 2013.
- [Ro20] Ros, R: Continuous experimentation with product-led business models: A comparative case study. In *Proceedings of the 11th International Conference on Software Business (ICSOB)*, Springer International Publishing, pp. 143-158, 2020.
- [RBR22] Ros, R.; Bjarnason, E.; and Runeson, P: A Theory of Factors Affecting Continuous Experimentation (FACE). *arXiv preprint arXiv:2210.05192*, 2022.
- [SG08] Seawright, J.; and Gerring J: Case selection techniques in case study research: a menu of qualitative and quantitative options. *Polit Res Q*. 2008;61(2): 294-308, 2008.
- [Wa95] Walsham, G: Interpretive case studies in is research: nature and method, *European Journal of information systems*, vol. 4, no. 2, pp. 74– 81, 1995.

# Evaluation eines LLM-basierten Text2SQL-Prototyps

Gero Pocha, Alexander Rachmann

CBS International Business School

gero.pocha@cbs-mail.de, a.rachmann@cbs.de

**Abstract:** In diesem Beitrag wird ein Prototyp zur Generierung von SQL-Code aus Text vorgestellt. Der Prototyp wird evaluiert. Es zeigt sich, dass vor allem vermeintliche einfache Prompts die besten Ergebnisse zeigen.

## 1 Einleitung

Large Language Models (LLMs) sind fortschrittliche Systeme der künstlichen Intelligenz, die darauf trainiert sind, menschenähnliche Texte zu generieren und zu verstehen. Eine möglicher Anwendungsfall ist die Generierung von Programmcode durch die Eingabe von natürlicher Sprache; noch genauer: die Generierung von Datenbankabfragen („Text2SQL“). Der betriebliche Einsatzzweck dahinter ist, dass eine Datenanalyse durch Nicht-Techniker noch niedrigschwelliger und noch intuitiver als bisher stattfinden könnte, wenn syntaktisch und semantisch korrekter Code generiert werden kann.

Im Folgenden wird ein Prototyp für Text2SQL vorgestellt. Dazu wird zuerst die Interaktion mit einem LLM vorgestellt (Kapitel 2 „Prompting“), danach werden die technischen Rahmenbedingungen für Text2SQL erläutert. In Kapitel 4 wird ein Prototyp vorgestellt, der in Kapitel 5 evaluiert wird. Kapitel 6 schließt den Beitrag.

## 2 Large Language Models und Prompting

Ein zentrales Merkmal von LLMs ist ihre Verhaltensweise, die auf Menschen als nicht deterministisch wirkt. Im Gegensatz zu deterministischen Systemen, die bei gleicher Eingabe immer dasselbe Ergebnis liefern, können LLMs bei derselben Anfrage unterschiedliche Antworten generieren. Dies liegt an ihrer Fähigkeit, aus einer Vielzahl potenzieller Antworten zu „wählen“, basierend auf Wahrscheinlichkeiten, die während des Trainings gelernt wurden. Diese Eigenschaft fördert einen natürlichen und menschenähnlichen Dialog (Holtzman et al. 2020, S. 1 ff.; Ouyang et al. 2023, S. 1 ff.) und damit, so die Annahme, so einer verbesserten Mensch-Computer-Interaktion. Auf der anderen Seite: Bei der Abfrage einer Datenbank gibt es immer nur eine begrenzte Anzahl an Abfragen, die eine bestimmte Bedeutung abfragen. Die Varianz in der Formulierung eines LLM kann hier also zum Nachteil werden, wenn eine SQL-Abfrage syntaktisch oder semantisch falsch formuliert wird.

Die Interaktion mit einem LLM passiert i.d.R. mit einem Prompt. Im Folgenden werden vier Arten des Prompting unterschieden:

- Zero-Shot Prompting (ZS): Mit ZS reagiert ein LLM auf eine Anfrage, ohne zuvor spezifische Beispiele für diese Art von Aufgabe gesehen oder darauf trainiert worden zu sein. Effektives Zero-Shot Prompting erfordert daher die Entwicklung von Prompts, die



dieses latente Wissen aktivieren, um neuartige Aufgabenstellungen zu bewältigen (Heston / Khun 2023, S. 199; Tingiris 2021, S. 7.).

- One-Shot Prompting (OS): Beim OS erhält das Modell ein einzelnes Beispiel der Aufgabenstellung, bevor es die eigentliche Anfrage bearbeitet. Dieses Beispiel dient als Leitfaden, um das Modell auf die gewünschte Art der Antwort zu konditionieren. Das Beispiel soll sowohl repräsentativ für die Anfrage sein als auch genügend Kontext bieten, damit das Modell die Anfrage im gewünschten Rahmen beantworten kann (Kovriguina et al. 2023, S. 4.).
- Chain-of-Thought Prompting (CT) ist eine Technik, bei der Prompts so gestaltet werden, dass das LLM seinen Lösungsweg Schritt für Schritt darlegt. Dieser Ansatz ist besonders nützlich bei komplexen Problemstellungen, bei denen eine direkte Antwort nicht offensichtlich ist (Wei et al. 2022, S. 1 ff.; Zhang et al. 2022, S. 2 f.).
- Beim Role Prompting (RP) wird das LLM durch eine geschickte Formulierung der Eingabeaufforderung in die Rolle eines Experten versetzt.

Im Allgemeinen ist akzeptiert, dass ein Prompt eine innere Struktur aufweisen sollte (White et al. 2023). Zum Beispiel differenziert OpenAI zwischen dem System-Prompt und User-Prompt (OpenAI o.J.). Ein User-Prompt ist der von einem Benutzer eingegebene Text, der als Eingabe für das Modell dient. System-Prompts hingegen sind vordefinierte Eingaben, die das Verhalten des Modells steuern oder spezifische Reaktionen auslösen sollen. Sie werden verwendet, um die Modellantworten zu formen oder zu leiten (Kim et al. 2023, o. S.; Wu et al. 2024, S. 3.). Eine weitere mögliche Struktur ist z.B. die Angabe einer Aufgabe, eines dazugehörigen Kontextes, Eingabedaten und dem Ausgabeformat (Vogelsang 2024).

### 3 Text2SQL

Text2SQL bezeichnet den Transfer von natürlicher Sprache zu SQL-basierten Datenbankabfragen. Diese Umwandlung ermöglicht es Nutzern, mit Datenbanken zu interagieren, ohne spezifisches Wissen über SQL-Syntax oder Datenbankstrukturen besitzen zu müssen. Die grundlegende Idee besteht darin, die Zugänglichkeit und Benutzerfreundlichkeit von Datenbankabfragen zu erhöhen, indem intuitive, natürlichsprachliche Eingaben verwendet werden, um komplexe Datenextraktionen und -analysen durchzuführen (Zhong et al. 2017, S. 7 ff.).

Grundlegend für Text-to-SQL-Systeme ist die Anwendung von Natural Language Processing und Machine Learning. Diese Systeme nutzen oft Deep Learning Modelle, um die Semantik der natürlichsprachlichen Anfragen zu verstehen und in die entsprechenden SQL-Befehle zu übersetzen (Zhong et al., 2017). Ein Schlüsselkonzept dabei ist das Verständnis des Kontexts und der Absicht hinter der Nutzeranfrage, was die Generierung präziser und funktionaler SQL-Abfragen ermöglicht (Zhang 2019, S. 1 ff; Zhong et al. 2017, S. 1 ff.).

Ein wesentlicher Aspekt, der für die Funktionalität von Text-to-SQL-Systemen von Bedeutung ist, ist das Verständnis der Struktur der zugrundeliegenden Datenbank durch das LLM. Damit die Konvertierung von natürlicher Sprache in SQL-Abfragen erfolgreich ist, muss das Modell nicht nur die Anfrage des Benutzers verstehen, sondern auch wissen, wie die Daten organisiert sind, einschließlich der Tabellenstrukturen, Spaltennamen und Beziehungen zwischen den Tabellen. Dieses Wissen ermöglicht es dem Modell, die Anfrage korrekt in eine SQL-Abfrage zu übersetzen, die gültige und erwartete Ergebnisse liefert (Zhang 2019, S. 3 ff; Zhong et al. 2017, S. 7 ff.).

Die Notwendigkeit, dass das Modell die Datenbankstruktur kennt, bedeutet jedoch nicht zwangsläufig, dass diese Informationen direkt in das Benutzer-Prompt aufgenommen werden müssen. Stattdessen können diese Details als Teil des System-Prompts oder der Modellkonfiguration hinterlegt werden. Moderne Ansätze in der Text-to-SQL-Konvertierung

beinhalten daher oft eine Phase des Trainings oder der Feinabstimmung, in der das Modell mit der spezifischen Datenbankstruktur vertraut gemacht wird. Diese Phase kann durch die Verwendung von Metadaten über die Datenbank, wie Schemadefinitionen und Beziehungskarten, unterstützt werden (Bogin et al. 2019, S. 1 ff.).

## **4 Demonstration anhand eines Prototyps mit der OpenAI API**

### **4.1 Designüberlegungen zum Prototyp**

Die OpenAI API ist eine Programmierschnittstelle zur Nutzung von ChatGPT (Gallardo et al. 2023, S. 26 f.; Kublik / Saboo 2022, S. 13 ff.). Die API bietet den Parameter Temperature an; mit diesem kann die „Zufälligkeit“ der Antworten des LLMS gesteuert werden: So führt ein niedriger Temperaturwert zu konservativeren und vorhersehbareren Antworten, während ein höherer Wert die Wahrscheinlichkeit erhöht, dass die generierten Antworten vielfältiger und weniger vorhersehbar sind (Kublik / Saboo 2022, S. 24 ff.).

Für die Demonstration wird einerseits eine Datenbank benötigt; andererseits ein Programm, welches anhand der OpenAI API Prompts ins SQL umwandelt und an die Datenbank schickt.

Für die Realisierung der Datenbank wurde auf eine MySQL Datenbank auf Microsoft Azure zurückgegriffen. Um den Prototypen eine praktisch relevante Aufgabe zustellen, wurde auf das in Abbildung 1 dargestellte Datenbankschema zurückgegriffen. Das Kernstück ist die Tabelle Lieferanten. Ein Lieferant dieser Tabelle hat bestimmte Verträge und liefert bestimmte Produkte zum Kunden. Pro Lieferanten und Produkt wird der Umsatz auf Tagesebene erfasst. Die Tabellen wurden für die Demonstration mit beispielhaften Angaben befüllt.

Das Programm soll die vier Prompting-Ansätze ZS, OS, CT und RP nacheinander abarbeiten.

Der Prototyp soll anhand von drei Kriterien bewertet werden:

1. Die Genauigkeit, d.h. ob die der korrekte SQL-Code generiert wurde. Korrekt ist ein Code, der sowohl syntaktisch wie auch semantisch richtig ist.
2. Ein Token, eine Maßeinheit für generierten Text, ist Maß für Kosten, da die Nutzung der OpenAI API über die Anzahl der generierten Token abgerechnet wird. Laut OpenAI entspricht ein Token ungefähr vier Zeichen in der englischen Sprache oder  $\frac{3}{4}$  Wörtern oder 100 Token entsprechen ungefähr 75 Wörtern.
3. Die Laufzeit gibt an, wie lange das Programm gelaufen ist.

In dem Programm soll daher jeweils eine Messung der Laufzeit, Prüfung der Akkuratheit und Messung der Kosten eingebaut werden, um diese Evaluierung des Programms zu ermöglichen.

### **4.2 Erstellung des Aufgabenkatalogs**

Da die Tabellen extra so entworfen wurden sind, dass viele mögliche Zusammenhänge erkennbar sind, können die Aufgaben recht herausfordernd gestaltet werden. Die 15 Aufgaben werden jeweils in fünf einfache, fortgeschrittene und herausfordernde Aufgaben aufgeteilt:

- Für die einfachen Aufgaben wurden nur Situationen gewählt, in denen nur eine Tabelle relevant ist. Zwei Aufgaben lauten „Nenn mir alle Lieferanten mit einer Bewertung von mindestens vier.“ und „Wie viele unserer Lieferanten sitzen in Deutschland?“.
- Für die fortgeschrittenen Aufgaben wurden Sachverhalte gewählt, in denen zwei Tabellen angesprochen werden. Z. B. „Wie viele Verträge zwischen uns und der koffeinCola GmbH laufen dieses Jahr aus?“ oder „Was ist die Adresse von dem Lieferanten, bei dem wir das Produkt mit dem höchsten Standardpreis beziehen?“.
- Für die herausfordernden Aufgaben wurde solche gewählt, deren Beantwortung mindestens 3 Tabellen benötigt, z. B. „Wie heißt das Produkt mit dem niedrigsten

Standardpreis, von dem Lieferanten, der die meisten Verträge mit uns hat, die am 31.12.2024 enden?“.

Der vollständige Aufgabenkatalog befindet sich im Repository <https://github.com/GeroPocha/TextToSqlRepository>.

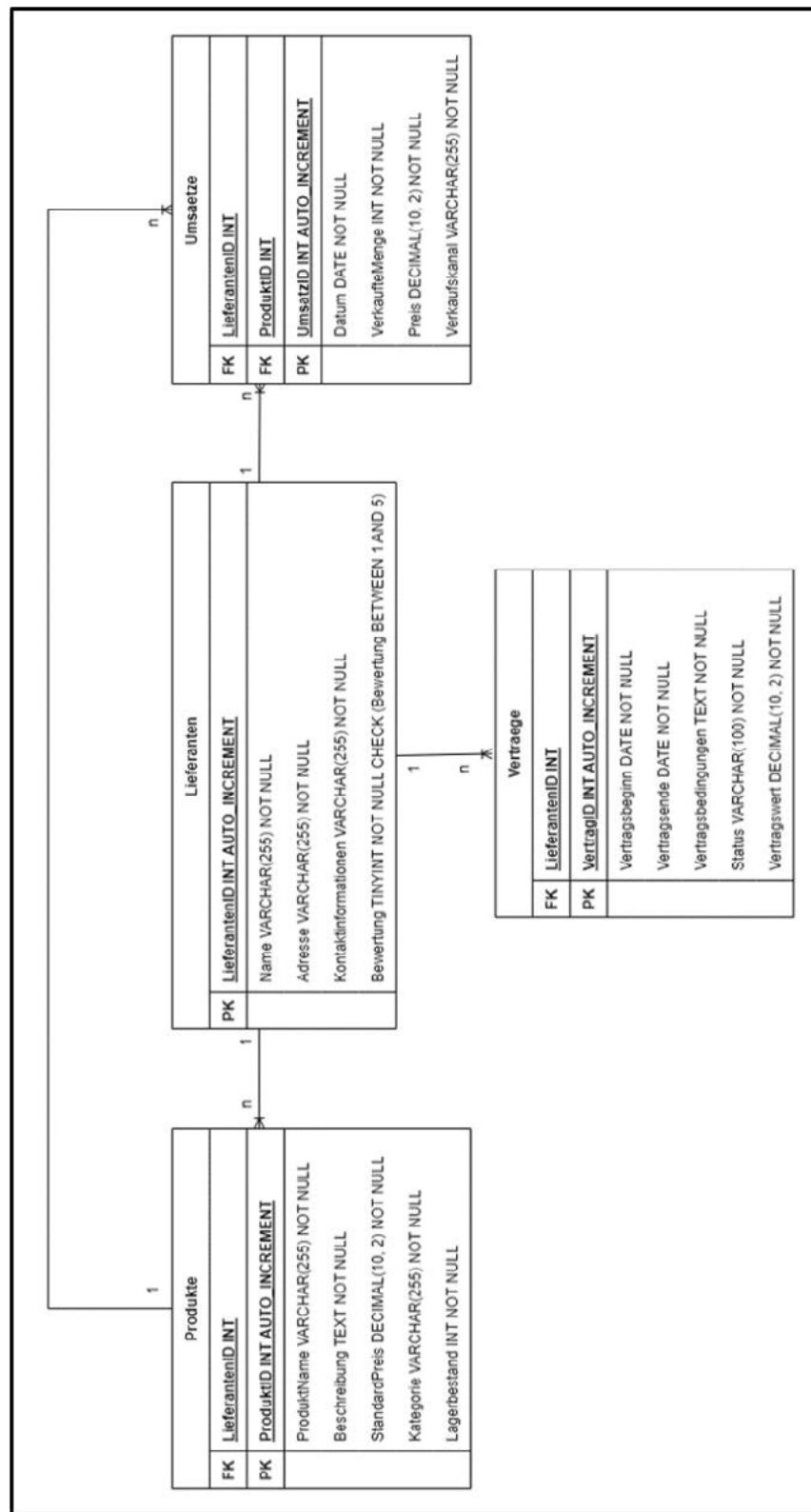


Abbildung 1: Datenbankschema

### 4.3 Erstellung der Prompts

Die OpenAI API unterscheidet zwischen dem System-Prompt und dem User-Prompt. Im System-Prompt wird dem LLM der Kontext zum eigentlichen User-Prompt gegeben; der System-Prompt ist über die bekannte ChatGPT-Webbrowser-Schnittstelle nicht nutzbar. Das System-Prompt ist für alle vier Prompting-Strategien des Prototyps unterschiedlich. Eine Gemeinsamkeit hat das Systemprompt jedoch bei allen vier der Strategien, nämlich das Nennen der Tabellen und Felder der Datenbank, die benutzt wird. Ohne wäre es für die OpenAI API schlichtweg nicht möglich eine Query zu erstellen, wenn nicht klar ist worauf sich bezogen wird. Tabelle 1 zeigt, wie sich die Prompts in der Demonstration zusammensetzen. Beispiele: Ein ZS besteht nur aus dem Kontext „Datenmodell“ im System-Prompt und dem User-Prompt. Ein CT besteht aus dem Kontext „Datenmodell“, dem Kontext „Vorgehen“ und dem User-Prompt.

Promptbestandteil	Prompt	ZS	OS	RP	CT
System Prompt: Kontext „Datenmodell“	Benutze folgende Datenbank als Grundlage: 'UnternehmenDB' mit den Tabellen 'Lieferanten', 'Produkte', 'Umsaetze' und 'Vertraege'. - 'Lieferanten' enthält 'LieferantenID', 'Name', 'Adresse', 'Ansprechpartner', 'Bewertung'. - 'Produkte' enthält 'ProduktID', 'ProduktName', 'Beschreibung', 'StandardPreis', 'Kategorie', 'Lagerbestand', 'LieferantenID'. - 'Umsaetze' enthält 'UmsatzID', 'ProduktID', 'LieferantenID', 'Datum', 'VerkaufteMenge', 'Preis', 'Verkaufskanal'. - 'Vertraege' enthält 'VertragID', 'LieferantenID', 'Vertragsbeginn', 'Vertragsende', 'Vertragsbedingungen', 'Status', 'Vertragswert'. Dir wird nun in natürlicher Sprache eine Aufgabe geben, du musst basierend auf der UnternehmenDB aus der Eingabe eine Query mit gültiger SQL Syntax machen. Gib nichts außer der Query aus.	X	X	X	X
System Prompt: Ausgabeformat	Beispiel: Wenn der Nutzer fragt: "Zeige mir alle Produkte in der Kategorie 'Elektronik', die einen Lagerbestand unter 50 haben.", dann lautet die SQL-Abfrage: "SELECT * FROM Produkte WHERE Kategorie = 'Elektronik' AND Lagerbestand < 50;".		X		
System Prompt: Kontext „Rolle“	Als Datenbankexperte für 'UnternehmenDB', spezialisiert auf die Analyse und Verwaltung der Tabellen 'Lieferanten', 'Produkte', 'Umsaetze', und 'Vertraege', ist es deine Aufgabe, datengetriebene Lösungen zu finden. Dein tiefgehendes Verständnis der Datenbankstruktur ermöglicht es dir, komplexe Abfragen effizient zu konstruieren.			X	
System Prompt: Kontext „Vorgehen“	Um eine SQL-Abfrage basierend auf einer natürlichsprachlichen Anfrage zu generieren, gehe wie folgt vor: 1. Identifiziere, welche Informationen gesucht sind. Bestimme die Felder, die aus der Datenbank selektiert werden sollen. 2. Ermittle, aus welchen Tabellen diese Felder bezogen werden können. Berücksichtige dabei die Struktur der 'UnternehmenDB' und die Beziehungen zwischen den Tabellen 'Lieferanten', 'Produkte', 'Umsaetze' und 'Vertraege'. 3. Prüfe, unter welchen Bedingungen die Informationen selektiert werden sollen. Identifiziere die Kriterien für die Filterung der Daten und wie diese Kriterien auf die Felder und Tabellen angewendet werden. 4. Überlege, ob und wie die Daten aus verschiedenen Tabellen miteinander verknüpft werden müssen, um die gesuchten Informationen zu erhalten. Dies kann den Einsatz von JOIN-Operationen oder die Kombination von mehreren Abfragen beinhalten.				X

5. Formuliere die SQL-Abfrage, indem du die identifizierten Felder selektierst, die entsprechenden Tabellen ansprichst, die festgelegten Bedingungen anwendest und die notwendigen Verknüpfungen zwischen den Tabellen herstellst.					
User Prompt:	Wie viel Lieferanten haben wir mit einer Bewertung von mindestens 3,				
Aufgabe und Eingabedaten (Beispiel)	mit deren Produkte wir schonmal an einem Tag über 5000€ im Online Verkauf verdient haben?	X	X	X	X

Tabelle 1: Prompts in der Demonstration. ZS kürzt Zero-Shot ab; OS One-Shot, RP Role Prompt, CT Chain of Thought.

#### 4.4 Codeausschnitte des Prototyps

Die Demonstration wurde mit Python programmiert; es werden nur inhaltlich interessante Codestellen hier vorgestellt. Der vollständige Code ist unter <https://github.com/GeroPocha/TextToSqlRepository> abrufbar.

In dem Array System-Prompts liegen alle System-Prompts (Zeile 40 im Code; siehe auch Tabelle 1, Zeilen 1 bis 4). Für jeden dieser vier Ansätze wird eine Schleife über drei definierte „Temperature“-Werte (0, 0.5, 1) durchgeführt (Zeile 42). Innerhalb dieser Schleife erfolgt eine weitere Iteration über den User-Prompts-Katalog (Zeile 46 im Code; siehe auch Tabelle 1, Zeile 5).

```

40 for prompt_name, prompt_details in System-Prompts.items():
41     prompt_text = prompt_details['prompt']
42     for temperature in [0, 0.5, 1]:
43         start_time = time.time()
44         korrekte_ergebnisse = 0
45         verbrauchte_tokens = 0
46         for index, task in enumerate(User-Prompts):
47             response = client.chat.completions.create(
48                 model="gpt-3.5-turbo",
49                 messages=[
50                     {"role": "system", "content": prompt_text},
51                     {"role": "user", "content": task}
52                 ],
53                 temperature=temperature
54             )
55             verbrauchte_tokens += response.usage.total_tokens
56             generated_query = response.choices[0].message.content
57             result = query_database(cursor, generated_query)
58             if result and clean_result(result) == loesungen[index]:
59                 korrekte_ergebnisse += 1
60         end_time = time.time()

```

In jeder Iteration wird das aktuelle System-Prompt zusammen mit der jeweiligen Aufgabe an die OpenAI API gesendet, um eine SQL-Query zu generieren (Zeile 47 bis 53). Pro Systemprompt und Temperature Wert wird außerdem die Laufzeit der Anwendung in Millisekunden festgehalten.

Der Prototyp loggt die gesamte Genauigkeit, die Genauigkeit pro Temperature, die aufgabenspezifische Genauigkeit, die gesamte Laufzeit, die Laufzeit pro Temperature sowie die gesamten Kosten. Pro Systemprompt und Temperature Wert wird außerdem die Laufzeit der Anwendung in Millisekunden festgehalten. Der Prototyp loggt die gesamte Genauigkeit, die Genauigkeit pro Temperature, die aufgabenspezifische Genauigkeit, die gesamte Laufzeit, die Laufzeit pro Temperature sowie die gesamten Kosten. Genauigkeit meint hier, ob eine Aufgabe korrekt beantwortet wurde.

## 4.5 Ergebnisauszug

Um die Ergebnisse des Prototyps verständlich zu machen, werden im Folgenden zwei Ergebnisse des Prototyps dargestellt. Beide Ergebnisse basieren auf dem gleichen User Prompt („Wie viel Lieferanten haben wir mit einer Bewertung von mindestens 3, mit deren Produkte wir schonmal an einem Tag über 5000€ im Online Verkauf verdient haben?“). CT0 gibt hier aus (Zeilennummer wurden hinzugefügt):

```

01 SELECT COUNT(DISTINCT L.LieferantenID)
02 FROM Lieferanten L
03 JOIN Produkte P ON L.LieferantenID = P.LieferantenID
04 JOIN Umsaetze U ON P.ProduktID = U.ProduktID
05 WHERE L.Bewertung >= 3
06 AND U.Datum = (
07     SELECT MAXIMUM(Datum)
08     FROM Umsaetze
09     WHERE Preis * VerkaufteMenge > 5000
10     AND Verkaufskanal = 'Online'
11 )
12 AND U.Preis * U.VerkaufteMenge > 5000
13 AND U.Verkaufskanal = 'Online';

```

Hierin tauchen zwei Fehler auf: erstens ein Syntaxfehler (Zeile 07, „MAXIMUM“ anstatt „MAX“); zweitens ein semantischer Fehler, da gar nicht nach dem maximalen Datum gefragt wurde.

Die Strategie ZS0,5 findet den richtigen SQL-Code; die Ausführung des untenstehenden Codes ergibt die korrekte Lösung „3“ (basierend auf der beispielhaften Datenbasis des Experiments).

```

01 SELECT COUNT(DISTINCT L.LieferantenID)
02 FROM Lieferanten L
03 JOIN Produkte P ON L.LieferantenID = P.LieferantenID
04 JOIN Umsaetze U ON P.ProduktID = U.ProduktID
05 WHERE L.Bewertung >= 3
06 AND U.Preis * U.VerkaufteMenge > 5000
07 AND U.Verkaufskanal = 'Online';

```

## 5 Evaluation

Nachdem das Programm 10-mal ausgeführt wurde, wurden die Kennzahlen erfasst und ein Durchschnitt berechnet. Die Tabelle 2 zeigt diese Werte.

Ø-Wert	ZS0	ZS0,5	ZS1	OS0	OS0,5	OS1	CT0	CT0,5	CT1	RP0	RP0,5	RP1
Genauigkeit	12/ 15	12/ 15	11/ 15	12/ 15	11,6/ 15	11,4/ 15	1,5/ 15	1,9/ 15	3,3/ 15	10,9/ 15	10,7/ 15	9,6/ 15
Tokens	5330	5306	5304	6330	6307	6314	10920	10922	10920	6158	6173	6184
Laufzeit (ms)	25286	26227	24642	24103	22414	21722	23263	23369	24446	23068	23408	21260

Tabelle 2: Durchschnittswerte der vier Prompt Engineering Strategien. ZS kürzt Zero-Shot ab, OS One-Shot, CT Chain-Of-Thought und RP Role Prompting, die 0, 0,5 und 1 Werte stehen jeweils für die Temperature Werte.

Bei der Betrachtung der Genauigkeit fallen fünf bemerkenswerte Punkte auf:

1. Zunächst ist festzustellen, dass das CT im Vergleich zu den anderen Ansätzen die niedrigsten Genauigkeitswerte aufweist. Dies steht im Widerspruch zur gängigen Meinung, die CT in vielen Fällen als überlegen gegenüber ZS und OS betrachtet (Badhan o. J.; Wei et al. 2022, S. 1 ff.).
2. Weiterhin zeigt sich, dass – abgesehen vom CT – niedrigere Temperatureinstellungen zu höherer Genauigkeit führen. Insbesondere liefert eine Temperatureinstellung von 0 bessere Ergebnisse als eine Einstellung von 0,5, und eine Einstellung von 0,5 wiederum bessere Ergebnisse als eine von 1. Dies deutet darauf hin, dass präzisere und weniger variable Antworten generiert werden, wenn die Unsicherheit im Modell durch niedrigere Temperaturen reduziert wird.
3. Die Ansätze, die am genauesten waren, sind ZS0, ZS0,5 und OS0.
4. Bei der Analyse der Kosten, die sich in der Anzahl der verbrauchten Tokens ausdrückt, ist entscheidend zu berücksichtigen, dass diese primär von der Länge des Systemprompts abhängen. Die Daten zeigen zwar Unterschiede in den Token-Zahlen zwischen den Methoden – ZS bewegt sich von 5330 bis 5304 Tokens, OS variiert minimal um 6300 Tokens, CT verzeichnet durchweg hohe Werte um 10920 Tokens, und RP weist 6158 bis 6184 Tokens auf –, jedoch spiegeln diese Zahlen hauptsächlich die jeweilige Komplexität und Länge der verwendeten System Prompts wider. Diese Beobachtung unterstreicht, dass die reine Betrachtung der Token-Anzahl ohne Kontext der Prompt-Länge wenig aussagekräftig bezüglich der tatsächlichen Effizienz oder Kostenwirksamkeit der verschiedenen Ansätze ist. Daher sollte dieser Aspekt bei der Bewertung der Kosten-Effektivität der Methoden berücksichtigt werden, um eine angemessene Einschätzung zu ermöglichen.
5. Die Laufzeitanalyse der Prompting-Strategien offenbart, dass OS durchweg die schnellsten Antworten liefert, was auf eine effiziente Informationsverarbeitung hindeutet. CT hingegen benötigt aufgrund seiner schrittweisen Lösungsansätze mehr Zeit. ZS und RP liegen zeitlich dazwischen, wobei ZS leicht längere Laufzeiten aufweist, da es ohne konkrete Beispiele arbeiten muss. RP profitiert von klar definierten Rollen, die eine schnellere Antwortfindung ermöglichen.

Basierend auf den drei analysierten Kriterien erscheint OS0 der ausgewogenste Ansatz für diesen Anwendungsfall. Der Ansatz bietet eine hohe Genauigkeit, die vergleichbar mit den besten Ergebnissen des ZS ist, und hält gleichzeitig die Kosten und die Laufzeit in einem moderaten Rahmen.

## 6 Fazit

Das Ziel des Beitrags bestand in der Demonstration einer Datenbankabfrage mit einer LLM. Dabei stand die Untersuchung im Fokus, wie verschiedene Prompt Engineering Strategien die Effizienz und Genauigkeit dieses Vorhabens beeinflussen. Nach der Durchführung des Programmes war erkennbar, dass ZS0, ZS0,5 sowie OS0 mit 12 richtig beantworteten Aufgaben von 15 durchschnittlich die höchste Genauigkeit der 12 Konfigurationen vorzuweisen hatten.

## Literaturverzeichnis

### Badhan o. J.

Badhan, Matihilli: Comprehensive Guide to Chain-of-Thought Prompting. <https://www.mercity.ai/blog-post/guide-to-chain-of-thought-prompting>, o. J., Abruf am 24. März 2024.

### Bogin et al. 2019

Bogin, Ben; Gardner, Matt; Berant, Jonathan: Representing Schema Structure with Graph

- Neural Networks for Text-to-SQL Parsing. <https://aclanthology.org/P19-1448.pdf>, 2019, Abruf am 17. März, 2024.
- Devin et al. 2018**  
Devlin, Jacob; Chang, Ming-Wei; Lee, Kenton; Toutanova, Kristina: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Google AI Language.
- Gallardo et al. 2023**  
Gallardo, Christian; Machuca, Cristian; Semblantes, Yadira: ChatGPT API. A brief overview and integration in software development. International Journal of Engineering Insights (2023) 1, S. 25 – 29.
- Heston / Khun 2023**  
Heston, Thomas; Khun, Charya: Prompt Engineering in Medical Education. In: International Medical Education (2023) 2, S. 198 – 205.
- Holtzman et al. 2020**  
Holtzman, Ari; Buys, Jan; Du, Li; Forbes, Maxwell; Choi, Yejin: The curious case of neural text degeneration. <https://arxiv.org/pdf/1904.09751.pdf>, 2020, Abruf am 22. März, 2024.
- Jujumilk3 2024**  
Jujumilk3: leaked-system-prompts. <https://github.com/jujumilk3/leaked-system-prompts>, 2024, Abruf am 06. März 2024.
- Kim et al. 2023**  
Kim, Tae Soo; Lee, Yoonjoo; Shin, Jamin; Kim, Young-ho; Kim, Juho: EvalLM: Interactive Evaluation of Large Language Model Prompts on User-Defined Criteria. <https://ar5iv.labs.arxiv.org/html/2309.13633>, 2023, Abruf am 10. März 2024.
- Kong et al. 2023**  
Kong, Aobo; Zhao, Shiwan; Chen, Hao; Li, Qicheng; Qin, Yong; Sun, Ruiqi; Zhou, Xin: Better Zero-Shot Reasoning with Role-Play Prompting. <https://arxiv.org/pdf/2308.07702.pdf>, 2023, Abruf am 10. März 2024.
- Kovriguna et al. 2023**  
Kovriguna, Liubov; Teucher, Roman; Radyush, Damiil; Mouromtsev, Dmitry: SPARQLGEN: One-Shot Prompt-based Approach for SPARQL Query Generation. <https://ceur-ws.org/Vol-3526/paper-08.pdf>, 2024, Abruf am 10. März 2024.
- Kublik / Saboo 2022**  
Kublik, Sandra; Saboo, Shubham: GPT-3. The ultimate Guide to building NLP Products with OpenAI API. New York 2022.
- OpenAI o.J.**  
OpenAI: Prompt Engineering. <https://platform.openai.com/docs/guides/prompt-engineering>, Abruf am 16. April, 2024.
- Ouyang et al. 2023**  
Ouyang, Shuyin; Zhang, Jie M.; Harman, Mark; Wang, Meng: LLM is Like a Box of Chocolates: the Non-determinism of ChatGPT in Code Generation. <https://arxiv.org/pdf/2308.02828.pdf>, 2023, Abruf am 22. März, 2024.
- Radford et al. 2018**  
Radford, Alec; Narasimhan, Karthik; Salimans, Tim; Sutskever, Ilya: Improving Language Understanding by Generative Pre-Training. [https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language\\_understanding\\_paper.pdf](https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf), 2018, Abruf am 22. März, 2024.
- Schmidt et al. 2023**  
Schmidt, Douglas; Spencer-Smith, Jesse; Fu, Quchen; White, Jules: Cataloging Prompt Patterns to Enhance the Discipline of Prompt Engineering. [https://www.dre.vanderbilt.edu/~schmidt/PDF/ADA\\_Europe\\_Position\\_Paper.pdf](https://www.dre.vanderbilt.edu/~schmidt/PDF/ADA_Europe_Position_Paper.pdf), 2024, Abruf am 06. März 2024.



**Tingiris 2021**

Tingiris, Steve: Exploring GPT-3. An unofficial first look at the general-purpose language processing API from OpenAI. Birmingham 2021.

**Vaswani et al. 2023**

Vaswani, Ashish; Shazeer, Noam; Parmar, Niki; Uszkoreit, Jakob; Jones, Llion; Gomez, Aidan N.; Kaiser, Łukasz; Polosukhin, Illia: Attention Is All You Need. <https://arxiv.org/pdf/1706.03762.pdf>, 2023, Abruf am 22. März, 2024.

**Vogelsang 2024**

Vogelsang, Andreas: Prompting the Future: Integrating Generative LLMs and Requirements Engineering. In: D. Mendez, A. Moreira, J. Horkoff, T. Weyer, M. Daneva, M. Unterkalmsteiner, S. Bühne, J. Hehn, B. Penzenstadler, N. Condori-Fernández, O. Dieste, R. Guizzardi, K. M. Habibullah, A. Perini, A. Susi, S. Abualhaija, C. Arora, D. Dell'Anna, A. Ferrari, S. Ghanavati, F. Dalpiaz, J. Steghöfer, A. Rachmann, J. Gulden, A. Müller, M. Beck, D. Birkmeier, A. Herrmann, P. Mennig, K. Schneider. Joint Proceedings of REFSQ-2024 Workshops, Doctoral Symposium

**Wei et al. 2022**

Wei, Jason; Wang, Xuezi; Schuurmans, Dale; Bosma, Maarten; Ichter, Brian; Xia, Fei; Chi, Ed; Le, Quoc; Zhou, Denny. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. <https://arxiv.org/pdf/2201.11903.pdf>, 2022, Abruf am 10. März 2024.

**White et al. 2023a**

White, Jules, Quchen Fu, Sam Hays, Michael Sandborn, Carlos Olea, Henry Gilbert, Ashraf Elnashar, Jesse Spencer-Smith, and Douglas C. Schmidt. "A prompt pattern catalog to enhance prompt engineering with chatgpt." arXiv preprint arXiv:2302.11382 (2023).

**White et al. 2023b**

White, Jules, Sam Hays, Quchen Fu, Jesse Spencer-Smith, and Douglas C. Schmidt. "Chatgpt prompt patterns for improving code quality, refactoring, requirements elicitation, and software design." arXiv preprint arXiv:2303.07839 (2023)

**Wu et al. 2024**

Wu, Yuanwei; Li, Xiang; Liu, Yixin; Zhou, Pan; Sun, Lichao: Jailbreaking GPT-4V via Self-Adversarial Attacks with System Prompts. <https://arxiv.org/pdf/2311.09127.pdf>, 2024, Abruf am 10. März 2024.

**Zhang et al. 2019**

Zhang, Rui; Yu, Tao; Yang, Kai; Yasunaga, Michihiro; Wang, Dongxu; Li, Zifan; Ma, James; Li, Irene; Yao, Qingning; Roman, Shanell; Zhang, Zilin; Radev, Dragomir R.: Spider: A Large-Scale Human-Labeled Dataset for Complex and Cross-Domain Semantic Parsing and Text-to-SQL Task. <https://arxiv.org/pdf/1809.08887.pdf>, 2019, Abruf am 17. März 2024.

**Zhang et al. 2022**

Zhang, Zhuosheng; Zhang, Aston; Li, Mu; Smola, Alex: Automatic Chain of Thought Prompting in Large Language Models. <https://arxiv.org/abs/2210.03493> 2022, Abruf am 10. März 2024.

**Zhong et al. 2017**

Zhong, Victor; Xiong, Caiming; Socher, Richard: SEQ2SQL: Generating structured queries from natural language using reinforcement learning. <https://arxiv.org/pdf/1709.00103.pdf>, 2017, Abruf am 17. März 2024.

# Generative künstliche Intelligenz und Kreativität: Erkenntnisse aus grauer Literatur

Alexander Rachman, Alexander Hiller

CBS International Business School

a.rachmann@cbs.de, a.hiller@cbs.de

**Abstract:** Fortschritte in generativer künstlicher Intelligenz verändern die Arbeitswelt von Wissensarbeitern derzeit stark; auch kreative Arbeit scheint davon betroffen zu sein. Im Beitrag wird anhand von grauer Literatur erhoben, welche Erfahrungen zum Einsatz von Kreativitätstechniken mit generativer KI vorliegen. Es wird deutlich, dass einerseits viele verschiedene Kreativitätstechniken in der grauen Literatur benannt werden. Andererseits dominiert die Technik des Promptings gegenwärtig die Wahrnehmung. Das breite Feld von Prompting-Techniken wird dargestellt.

## 1 Einleitung

Die Veröffentlichung von ChatGPT im Herbst 2022 löste einen Hype um generative künstliche Intelligenz (GKI) aus. Unter GKI werden solche KI-Ansätze verstanden, die auf Basis von unbeaufsichtigtem oder teilweise überwachtem maschinellen Lernen künstliche Artefakte durch die Anwendung von Statistiken und Wahrscheinlichkeiten generieren (Baidoo-Anu & Ansah, 2023; Hu, 2022; Jovanovic & Campbell, 2022). GKI spielt beispielsweise in den verschiedenen Phasen der Softwareentwicklung eine Rolle; darunter in der Programmierunterstützung oder der Formulierung von Software-Anforderungen (Fan et al., 2023; Hou et al., 2023). Die zunehmende Verfügbarkeit von GKI weckt Erwartungen, kreative Aufgaben von Menschen zu unterstützen bzw. vollständig zu übernehmen (Brynjolfsson et al., 2023; Dwivedi et al., 2023; Jo, 2023; McKinsey, 2023). Trotz dieser Erwartungen ist die aktuelle Diskussion über die spezifische Nutzung von Kreativitätstechniken im Kontext von GKI begrenzt.<sup>1</sup> In diesem Beitrag wird im Rahmen einer Analyse von grauer Literatur explorativ untersucht, welche Erkenntnisse zum Einsatz von Kreativitätstechniken mit GKI gegenwärtig vorliegen. Der Wertbeitrag dieser Untersuchung ist die Dokumentation von Erfahrungen zum Einsatz von generativer KI und Kreativitätstechniken bis Herbst 2023.

## 2 Kreativität in der Softwareentwicklung

Kreativität bezeichnet den Prozess, in dem originelle oder modifizierte Ideen, Lösungen oder Erkenntnisse entstehen. Dieser Prozess wird maßgeblich von dem Wissen und der Motivation der kreativ tätigen Person beeinflusst und verändert gleichzeitig deren Wissen und Motivation. Im Verlauf des kreativen Prozesses beeinflussen sich Wissen und Motivation gegenseitig (Runco & Chand, 1995). Kreativität entfaltet sich im Rahmen des gestalterischen Prozesses des Designs (Abbildung 1). Die gezielte Anwendung von Kreativitätstechniken wird in der Softwareentwicklung als essenziell angesehen (Groeneveld et al., 2021). Zu den bekanntesten Kreativitätstechniken zählen beispielsweise Storyboarding, morphologische Analyse, laterales

---

<sup>1</sup> Beispielsweise wird in dem viel beachteten Aufsatz von Dell'Acqua et al. (2023), der die positive Wirksamkeit von GKI in der Wissensarbeit experimentell darlegt, sehr allgemein geschrieben: „Approximately half of the participants (385 consultants) tackled a series of tasks where they were prompted to conceptualize and develop new product ideas, focusing on aspects such as creativity, analytical skills, persuasiveness and writing skills.“

Denken, Six Thinking Hats und Brainstorming (Leopoldino et al., 2016). Künstliche Intelligenz wird als relevantes Werkzeug der Kreativität betrachtet (Esling & Devis, 2020; Runco, 2023).

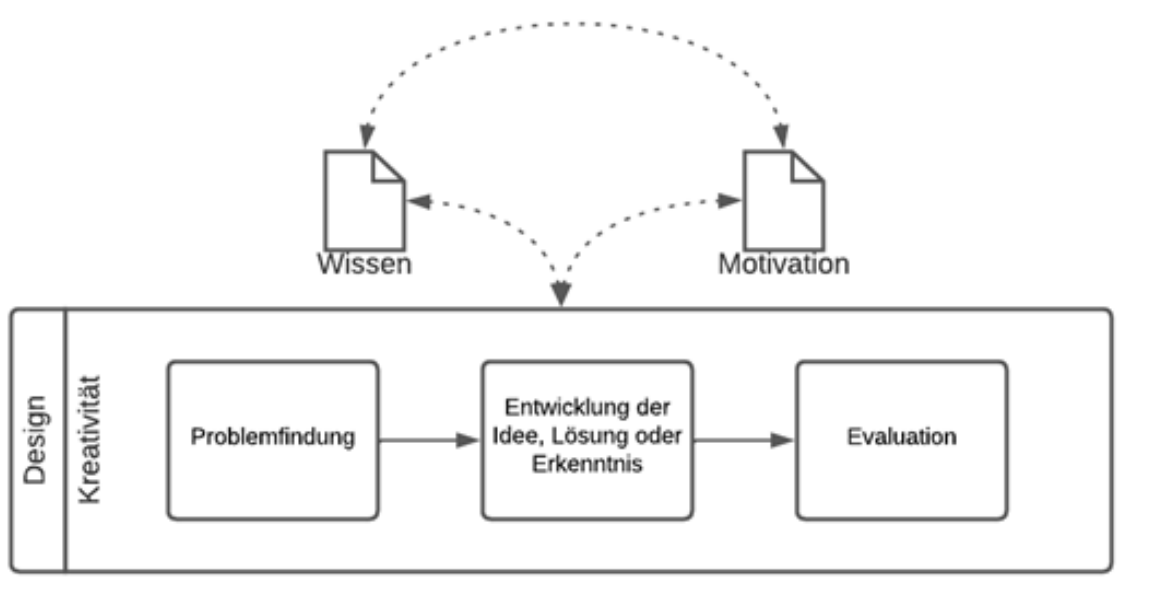


Abbildung 1: Zweistufiges Modell des kreativen Denkens, vereinfacht nach Runco und Chand (1995) und in BPMN formuliert

### 3 Forschungsmethodik

Auf Basis einer systematischen Literaturliteraturanalyse in grauer Literatur wird die Frage beantwortet, welche Erfahrungen über die Nutzung von Kreativitätstechniken beim Einsatz von GKI gegenwärtig vorliegen. Weiterhin wird die meistdiskutierte Kreativitätstechnik anhand von Beispielen vorgestellt. Die Forschungsmethodik basiert auf der Annahme, dass im nicht-wissenschaftlichen Bereich schneller Erkenntnisse generiert und publiziert werden können, da weniger oder keine Qualitätsvorgaben existieren. Daraus folgt eine, in der Tendenz, geringere Qualität bzw. Verlässlichkeit der Erkenntnis (Adams et al., 2017). Es ist also davon auszugehen, dass sich der hier vorgestellte Erkenntnisstand durch zukünftige Arbeiten deutlich verändern wird.

Es wurde eine zweistufige Suche im Rahmen der Literaturliteraturanalyse durchgeführt:

1. Zunächst erfolgte die Identifikation relevanter grauer Literatur mithilfe der Suchmaschine Google. Allgemein wird der Suchmaschine Google eine hohe Suchergebnisqualität zugestanden (Lewandowski, 2015). Unter Verwendung des zusammengesetzten Suchbegriffs "generative ai creativity techniques" erfolgte eine Untersuchung der ersten 20 Suchergebnisse. Aus diesen 20 Suchergebnissen konnten 18 Treffer für die weitere Analyse genutzt werden.<sup>2</sup> Zwei kostenpflichtige Treffer fanden keine Berücksichtigung. Die Google-Suche wurde am 01.08.2023 durchgeführt.
2. In der zweiten Stufe wurde die meistbenannte Technik, das Prompting, genauer analysiert. Dazu wurden fünf ausgewählte Expertenquellen genutzt.

<sup>2</sup> Alphabetisch sortiert: 123rf (2023); Brownell (2023); Daza (2023); Cepelak (2023); Duarte (2023); Eapen et al. (2023); Forrester (2023); Frey (2023); IBM (o.J.); Lawton (2023); Moon (2023); Mukherjee & Chang (2023a); Mukherjee & Chang (2023b); NewMutation Technology (2023); Remote Symphony (2023); Stackpole (2023); Stahl (2023); Wolfson (2023)

## 4 Erkenntnisse

### 4.1 Übersicht der Techniken

Innerhalb der untersuchten Stichprobe wurden 24 Kreativitätstechniken identifiziert, wobei die Mehrzahl dieser Techniken in fünf verschiedenen Quellen nachgewiesen werden konnte (Brownell, 2023; Eapen et al., 2023; Frey, 2023; Mukherjee & Chang, 2023b; Wolfson, 2023). In Tabelle 1 sind die identifizierten Kreativitätstechniken aufgelistet. Die zweiten Spalte von Tabelle 1 enthält den Namen der Kreativitätstechnik, wie er in der jeweiligen Veröffentlichung benannt wurde. Identische Kreativitätstechniken aus unterschiedlichen Quellen wurden zusammengeführt. In der dritten Spalte sind die jeweiligen Kreativtechniken einer Gruppe zugeordnet, welche im Folgenden näher erläutert werden. Die vierte Spalte führt auf, in welchen Quellen die Kreativtechniken gefunden wurden.

#	Name der Kreativitätstechnik	Gruppe	Quelle
01	Analyzing and learning from data	T	Frey (2023)
02	Balance of divergent and convergent thinking	D	Mukherjee, A. & Chang, H. (2023b)
03	Balance of novelty and usefulness	D	Mukherjee, A. & Chang, H. (2023b)
04	BLEU metric	T	Mukherjee, A. & Chang, H. (2023b)
05	Co-creation / collaboration	T	Eapen et al. (2023); Frey (2023)30.04.2024 10:26:00
06	Creating multimedia mood boards	En	Brownell (2023)
07	Criteria-oriented idea evaluation	Ev	Eapen et al. (2023)
08	Divergent thinking	D	Eapen et al. (2023)
09	Encourage lateral thinking	D	Brownell (2023)
10	Enhancing skills	Ev	Frey (2023)
11	Expanding artistic horizons	En	Frey (2023)
12	Filling in the details	En	Brownell (2023)
13	Focus on the flaws	En	Brownell (2023)
14	Giving you confidence in the ideas you already have	Ev	Brownell (2023)
15	Idea generation	En	Frey (2023)
16	Idea refinement	En	Eapen et al. (2023)
17	Personalization	T	Frey (2023)
18	Prompting	T	Brownell (2023; 2023; 2023; 2023; 2023; 2023; 2023; 2023; 2023)
19	Rapid prototyping	En	Frey (2023)
20	Rounding out your ideas	En	Brownell (2023)
21	Rubber ducking	En	Brownell (2023)
22	Solving the blank page problem / Overcoming creative blocks	En	Brownell (2023); Frey (2023)
23	Thinking outside the box	D	Brownell (2023)
24	Trisociation	D	Eapen et al. (2023)

Anmerkung. (Legende: T = Allgemeine Technik, D = Denkweise, En = Technik zur Entwicklung von Ideen etc., Ev = Technik zur Evaluation).

Tabelle 1: Überblick über Kreativitätstechniken in der Stichprobe

## 4.2 Gruppierung der Techniken

Zur Strukturierung der 24 Kreativitätstechniken wurden fünf Gruppen gebildet, die sich aus den fünf Bereichen der Kreativität (Abbildung 1) ableiten. Wie es aus grauer Literatur zu erwarten ist, sind die Kreativitätstechniken nicht vollständig definiert. Es fehlen teilweise genaue Verfahrensbeschreibungen, erforderliche Eingaben und resultierende Ausgaben. Daher ist die Zuordnung einer Kreativitätstechnik zu einer Gruppe nicht immer eindeutig. Die Autoren haben eine Kreativitätstechnik jener Gruppe zugeordnet, die aus ihrer Sicht am schlüssigsten erscheint.

- Gruppe T beschreibt allgemeine Kreativitätstechniken. Diese können sowohl in der Problemfindung, der Entwicklung von Ideen, Lösungen oder Erkenntnissen, der Evaluation, dem Bereich Wissen oder dem Bereich Motivation eingesetzt werden (siehe Abbildung 1). Fünf Kreativitätstechniken sind dieser Gruppe zugeordnet.
- Gruppe D beschreibt Denkweisen, d.h. eine Kreativitätstechnik ohne konkrete Vorgehensweise, sondern einen Rahmen, in dem kreatives Denken stattfinden kann. Kreativitätstechniken der Gruppe D sind dem Bereich des Wissens und / oder der Motivation zugeordnet. Sechs Kreativitätstechniken sind dieser Gruppe zugeordnet.
- Gruppe P beschreibt Kreativitätstechniken zur Problemfindung. Keine Kreativitätstechnik ist dieser Gruppe zugeordnet.
- Gruppe En beschreibt Kreativitätstechniken zur Entwicklung von Ideen, Lösungen oder Erkenntnissen. Zehn Kreativitätstechniken sind dieser Gruppe zugeordnet.
- Gruppe Ev beschreibt Kreativitätstechniken zur Evaluation von Ideen, Lösungen oder Erkenntnissen. Drei Kreativitätstechniken sind dieser Gruppe zugeordnet.

## 4.3 Prompting

In der Stichprobe ist das Prompting die am häufigsten erwähnte Kreativitätstechnik (Tabelle 1). Hierbei handelt es sich um eine textbasierte Interaktionsform mit GKI. Um die gegenwärtige Anwendung des Promptings zu beschreiben, wurden fünf Quellen der grauen Literatur ausgewertet und dabei 45 Varianten des Promptings identifiziert.

1. OpenAI, der Entwickler von ChatGPT, stellt mit Shieh (2023) acht Rule of Thumbs für den Umgang mit Prompts zur Verfügung (Tabelle 2).
2. Github ist der Produzent von GitHub Co-Pilot, einer führenden GKI in der Softwareentwicklung. Mit Ziegler und Berryman (2023) stellt GitHub einen Entwicklerleitfaden mit sechs Schritten für die Codeproduktion mit GKI zur Verfügung (Tabelle 3).
3. Alphabet ist ein führendes Unternehmen im Bereich des maschinellen Lernens, welches die technische Grundlage für die Entwicklung von GKI bildet. Als Tochterunternehmen von Alphabet, stellt Google (o.J.) sechs Best Practices zum Prompting zur Verfügung (Tabelle 4). Darüber hinaus beschreibt Google (o.J.) fünf Arten von Prompts (Tabelle 5).
4. Der „Prompt Engineering Guide“ von Saravia (2023) ist ein häufig zitiertes Repository auf GitHub, in dem 14 Prompting-Techniken beschrieben werden (Tabelle 6).
5. Ende 2023 veröffentlichte OpenAI (2023) einen Guide, der Strategien und Taktiken zum Prompting beinhaltet (Tabelle 7).

#	Rule of Thumb
25	1. Es soll die aktuelle Version des zugrundeliegenden Modells genutzt werden.
26	2. Der Prompt soll unterteilt werden in Kontext und Anweisung. Zuerst soll die Anweisung gegeben werden, danach der Kontext.
27	3. Der Prompt soll spezifisch, beschreibend und so detailliert wie möglich sein. Dies bezieht sich auf den Kontext, das erwartete Ergebnisse und dessen Länge, Format, Stil etc.
28	4. Das erwünschte Ausgabeformat soll durch Beispiele verdeutlicht werden.

29	5. Der Prompt-Verlauf soll mit der Zero-shot-Technik starten, dann mit der Few-shot-Technik fortgeführt werden. Falls das nicht zum Erfolg führt, sollte im Detail nachgearbeitet werden („fine-tune“).
30	6. Es sollen unpräzise Formulierungen reduziert werden.
31	7. Negativ-Formulierungen sollen vermieden werden, stattdessen sollen Positivformulierungen verwendet werden.
32	8. Spezifisch für die Produktion von Softwarecode: Es sollen “Leitwörter” verwendet werden um bestimmte Entwurfsmuster zu nutzen.

Tabelle 2: Rule of Thumbs nach Shieh (2023)

#	Schritt	Beschreibung
33	1. Gathering context	Der Programmierer sollte Informationen bereitstellen über die verwendete Programmiersprache und den technischen Kontext des zu erstellenden Codes.
34	2. Snippeting	Co-Pilot arbeitet mit Snippets, d.h. Codeabschnitten von ca. 60 Zeilen. Der Programmierer kann dafür sorgen, dass der Co-Pilot diese klug auswählen kann.
35	3. Dressing them up	Dem Co-Pilot sollen die wichtige Kontextinformationen zur Verfügung gestellt werden, i.d.R. mit einem Verweis zum Pfad der Datei.
36	4. Prioritization	Die Kontextinformationen sollen vom Programmierer mit Prioritäten versehen werden. Die Priorisierungen sorgen für eine bevorzugte Verarbeitung durch Co-Pilot.
37	5. The AI does its thing	Das Modell schlägt Code vor.
38	6. Now, over to you	Der Programmierer muss festlegen, wie viel Code von Co-Pilot produziert werden soll. Der Programmierer übernimmt das Ergebnis des Co-Pilots.

Tabelle 3: Sechs Schritte der Codeproduktion durch Prompting nach Ziegler und Berryman (2023)

#	Best Practice
39	1. Es soll deutlich gemacht werden, welche Inhalte oder Informationen am wichtigsten sind.
40	2. Der Prompt soll strukturiert werden in eine Rolle, dann den Kontext, dann Eingabedaten und die Anweisung.
41	3. Es sollen spezifische, vielfältige Beispiele verwendet werden, um das Modell einzugrenzen und genauere Ergebnisse zu erzielen.
42	4. Es sollen Einschränkungen verwendet werden, um den Umfang der Modellausgabe einzuschränken. So wird vermieden, dass sich die Anweisungen auf sachliche Ungenauigkeiten beziehen.
43	5. Komplexe Prompts sollen in einfachere Prompts zerlegt werden.
44	6. Das Modell soll aufgefordert werden, seine eigenen Antworten zu bewerten oder zu prüfen, bevor sie erstellt werden. Beispiele: „Beschränken Sie Ihre Antwort auf drei Sätze.“, „Berücksichtigen Sie auf einer Skala von 1 bis 10, wie kurz und prägnant“ Ihre Aussage ist.“ „Glauben Sie, dass das richtig ist?“.

Tabelle 4: Sechs Best Practices des Promptings nach Google (o.J.)

Art	Beschreibung
45 Direct prompting (Zero-shot)	Ein direkter Prompt ist die einfachste Form eines Prompts. Es werden keine Beispiele zur Anweisung hinzugefügt.
46 Prompting with examples (One-, few-, and multi-shot)	One-shot-Prompts beinhalten ein Beispiel. Few-shots-Prompts einige Beispiele, Multi-shot-Prompts entsprechend mehrere Beispiele.

47	Chain-of-thought prompting	Mit Chain-of-Thought (CoT) liefert die KI eine Begründung für das Ergebnis.
--	Zero-shot CoT	Kombination von #45 und #47.
48	Strategien für Iterationen	<ol style="list-style-type: none"> <li>1. Es sollen wichtige Wörter, Wortgruppen oder Ideen wiederholt werden.</li> <li>2. Es soll das gewünschte Ausgabeformat angegeben werden (CSV, JSON usw.).</li> <li>3. Es sollen Großbuchstaben / Übertreibungen / hyperbolische Formulierungen verwendet werden, um wichtige Punkte oder Anweisungen zu unterstreichen.</li> <li>4. Es sollen Synonyme oder alternative Formulierungen verwendet werden.</li> <li>5. Es sollen Sandwichverfahren mit langen Aufforderungen ausprobiert werden: Fügen Sie dieselbe Anweisung an verschiedenen Stellen hinzu.</li> <li>6. Es sollen Aufforderungsbibliotheken genutzt werden<sup>3</sup>. Prompt Hero und diese Prompt Gallery sind zwei gute Ausgangspunkte.</li> </ol>

Tabelle 5: Arten von Prompts nach Google (o.J.)

Technik		Beschreibung
--	Zero-Shot prompting	Siehe #45.
--	Few-Shot prompting	Siehe #46.
--	Chain-of-thought prompting	Siehe #47.
49	Self-consistency	Es werden mehrere Prompts mit few-shot CoT geschrieben. Die KI wird aufgefordert, aus den Prompts ein Ergebnis zu generieren. Durch die few-shots CoT wird eine Konsistenz erzwungen.
50	Generated knowledge prompting	Der KI werden Fakten mitgeteilt („Knowledge“). Diese Fakten helfen der KI die richtigen Antworten zu generieren.
51	Tree of thought	Der Tree of Thought (ToT) ist ähnlich zur CoT, baut diese aber in einer Baumstruktur auf.
52	Retrieval augmented generation	Ähnlich wie in #50 wird der KI-Wissen als Grundlage mitgegeben. Dies wird durch das Modell der KI explizit unterstützt.
53	Automatic reasoning and tool-use	Das Large Language Model einer KI wird in einen „frozen state“ versetzt; aus einer „Task library“ werden vordefinierte Aufgaben ausgewählt.
54	Automatic prompt engineer	Die KI wird als Schlussfolgerungs-, Bewertungs-, Variationsmodell genutzt. Diese drei Nutzungsformen werden nacheinander gepromptet.
55	Active prompt	Die KI wird genutzt um die menschegebenen Beispiele (#46) zu verbessern. Die KI wird mehrmals mit #47 gepromptet. Die KI generiert mehrere Antworten, die vom Menschen bewertet werden. Aus diesen Bewertungen ergeben sich Schlussfolgerungen auf die Ursprungsfrage.
56	Directional stimulus prompting	Das Ergebnis der KI wird durch spezifische „Stimulus“ beeinflusst.
57	ReAct	Es wird unterschieden zwischen „reasoning traces“ und „task-specific actions“. Diese beiden Strukturen erweitern eine CoT (#47).

<sup>3</sup> Referenzierte Beispiele in Google (o.J.): <https://prompthero.com/> und <https://developers.generativeai.google/prompt-gallery>.

58	Multimodal chain of thought	In MCoT werden nicht nur Texteingaben, sondern auch Bilder verwendet.
59	Graph prompting	Es wird eine Graphenstruktur verwendet.

Tabelle 6: Prompting-Techniken nach Saravia (2023)

Strategie	
60	Es sollen klare Anweisungen geschrieben werden. Dies beinhaltet die Nutzung von Details, die Auswahl einer Persona, die Strukturierung des Prompts, die Beschreibung der notwendigen Schritte zur erfolgreichen Beantwortung des Prompts, die Angabe von Beispielen und sowie die Angabe der Länge der Antwort.
61	Es soll ein Referenztext angegeben werden. Dabei soll angegeben werden, wo der Referenztext zu finden ist. Zusätzlich soll das LLM angewiesen werden, diese Referenz entsprechend zu zitieren.
62	Die Aufgabe im Prompt soll in kleinere Teilaufgaben untergliedert werden. Das LLM soll die Ausgabe klassifizieren; bei langen und vielen Posts sollen Techniken zur Komplexitätsreduktion verwendet werden.
63	Dem LLM soll Zeit zum „Denken“ eingeräumt werden. Dazu kann das LLM aufgefordert werden die eigene Lösung zu erklären oder einen inneren Monolog zu führen. Weiterhin kann das LLM gefragt werden, ob in einem vorherigen Versuch etwas vergessen oder falsch gemacht wurde.
64	Sollte das LLM eine Schwäche besitzen, die nicht durch eigenes Verhalten zu verringern ist, sollte der Output von anderen Tools als Input für das LLM genutzt werden.
65	Veränderungen in den Ergebnissen des LLM durch unterschiedliche Prompts sollten systematisch geprüft werden.

Tabelle 7: Sechs Strategien und Taktiken nach OpenAI (2023)

### 4.3.1 Erkenntnisse zum Prompting

Folgenden Erkenntnisse lassen sich aus der Stichprobe zum Prompting ziehen:

- Die Techniken #45 bis #47 werden von mehreren Quellen benannt und es scheint Konsens über deren Gebrauch zu bestehen.
- Es besteht eine Unterscheidung zwischen prozessualen und nicht-prozessualen Techniken. Beispiele für prozessuale Techniken sind #45 bis #47 und #49 bis #59, während #25 bis #28 zu den nicht-prozessualen Techniken gehören.
- Die Techniken sind in ihrer Nutzungsweise unterschiedlich:
  - Indem Co-Pilot den Code selbst schreibt, der im weiteren als Prompt verwendet wird, ist Co-Pilot selbstreferenziell (#33-#38). ChatGPT bzw. dessen textbasierte Dialogform ist nicht selbstreferenziell, da ChatGPT den Prompt nicht selbst verändern kann.
  - Manche Techniken erfordern die direkte Unterstützung durch die KI, wie #52 bis #59, während andere Techniken nicht explizit von der KI unterstützt werden müssen.
- Die beschriebenen Techniken scheinen noch nicht eindeutig definiert zu sein; einige basieren auf arXiv-Dokumenten, andere werden oberflächlich beschrieben aber nicht wissenschaftlich ausdefiniert (z.B. Saravia (2023)).

### 4.3.2 Ausgewählte Prompting-Beispiele

In Abbildung 2 ist ein Beispiel für einen direkten Prompt (#45) dargestellt<sup>4</sup>. #26 bis #28 sowie

<sup>4</sup> #25 wird nicht angewendet, da zum Zeitpunkt des Schreibens die aktuelle Version für den Autor nicht verfügbar war.



#39 bis #42 werden eingehalten.

```
Prompt:
Create a four-column spreadsheet of 3 highest-rated science fiction
movies, year of release, average audience rating, and top 3 keywords
from audience reviews. Use Markdown syntax. Make sure to cite the
source of the audience rating.

Response:
Certainly, here's a four-column spreadsheet of the 3 highest-rated
science fiction movies in Markdown syntax, including their year of
release, average audience rating, and the top 3 keywords from
audience reviews:
| Movie Title           | Year of Release | Average Audience Rating |
Top 3 Keywords from Audience Reviews |
|-----|-----|-----|
| Blade Runner 2049    | 2017           | 87% (Rotten Tomatoes) |
Visually stunning, Thought-provoking, Atmosphere |
| Inception            | 2010           | 91% (Rotten Tomatoes) |
Mind-bending, Dream, Heist |
| Interstellar         | 2014           | 86% (Rotten Tomatoes) |
Epic, Emotional, Mind-bending |
Source of the audience rating: Rotten Tomatoes
(www.rottentomatoes.com). Please note that audience ratings may
change over time, and this data is accurate as of my knowledge cutoff
```

Abbildung 2: Beispiel für One-Shot-Prompting mit Kontextinformationen zur Datenauswahl und -format. Inhaltlich gekürzt nach Google (o.J.), ausgeführt unter ChatGPT.

In Abbildung 3 ist ein Chain-of-thought Prompt dargestellt, d.h. die KI stellt eine Erläuterung bereit, die das Ergebnis erklärt.

```
Prompt:

The odd numbers in this group add up to an even number: 4, 8, 9, 15,
12, 2, 1.
A: Adding all the odd numbers (9, 15, 1) gives 25. The answer is
False.
The odd numbers in this group add up to an even number: 15, 32, 5,
13, 82, 7, 1.
A:

Response:

In this group of numbers (15, 32, 5, 13, 82, 7, 1), the odd numbers
are 15, 5, 13, 7, and 1. Let's add them up:
15 + 5 + 13 + 7 + 1 = 41
So, the sum of the odd numbers in this group is 41, which is an odd
number. Therefore, the statement "The odd numbers in this group add
up to an even number" is False.
```

Abbildung 3: Beispiel für Chain-of-Thought prompting nach Google (o.J.), ausgeführt unter ChatGPT.

In Abbildung 4 ist ein Generated knowledge prompt dargestellt. Es ist erwähnenswert, dass die KI die Frage „Part of golf is trying to get a higher point total than others. Yes or No?“ ohne erweitertes Wissensstatement tendenziell falsch beantwortet. Erst durch das Hinzufügen des Wissens erkennt die KI die korrekte Antwort.

**Prompt:**

Question: Part of golf is trying to get a higher point total than others. Yes or No?

Knowledge: The objective of golf is to play a set of holes in the least number of strokes. A round of golf typically consists of 18 holes. Each hole is played once in the round on a standard golf course. Each stroke is counted as one point, and the total number of strokes is used to determine the winner of the game.

Explain and Answer:

**Response:**

No, the objective of golf is not to get a higher point total than others. In golf, the goal is to complete a round of 18 holes in as few strokes as possible. Each stroke is counted individually, and the player with the lowest total number of strokes at the end of the round is the winner. In other words, the aim is to have the lowest score or point total, not a higher one, to be successful in golf.

Abbildung 4: Beispiel für ein Generated Knowledge Prompt nach Saravia (2023), ausgeführt unter ChatGPT.

## 5 Fazit

Der vorliegenden Artikel präsentiert eine systematische Literaturanalyse in der grauen Literatur zur Nutzung von Kreativitätstechniken mit GKI bis Herbst 2023. Die Analyse zeigt die Verwendung einer Vielzahl unterschiedlicher Kreativitätstechniken im Zusammenhang mit GKI. Insgesamt wurden 24 Kreativitätstechniken identifiziert. Die Technik des Prompting dominiert dabei die Aufmerksamkeit. Es wurden 45 Varianten des Promptings in einer vertiefenden Suche gefunden. Prompting ist dabei nicht als einzelne Technik zu verstehen, sondern als ein System unterschiedlicher Techniken. Es ist anzunehmen, dass sich diese Techniken zukünftig stark verändern werden, da sich GKI-Technologien dynamisch weiterentwickeln und weitere Erfahrungen zum effektiven Einsatz des Promptings, mit und ohne Verbindung zu klassischen Kreativitätstechniken, gesammelt werden.

## Literatur

123ref. (2023, Mai 26). *Unleashing Creativity: The Power of Generative AI in Transforming the Content Landscape*. <https://www.blog.123rf.com/unleashing-creativity-the-power-of-generative-ai-in-transforming-the-content-landscape>

Adams, R. J., Smart, P., & Huff, A. S. (2017). Shades of Grey: Guidelines for Working with the Grey Literature in Systematic Reviews for Management and Organizational Studies. *International Journal of Management Reviews*, 19(4), 432–454. <https://doi.org/10.1111/ijmr.12102>

Baidoo-Anu, D., & Ansah, L. O. (2023). *Education in the Era of Generative Artificial Intelligence (AI): Understanding the Potential Benefits of ChatGPT in Promoting Teaching and Learning* (SSRN). [https://papers.ssrn.com/sol3/papers.cfm?abstract\\_id=4337484](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4337484)

Brownell, B. (2023, Juni 19). *10 ways generative AI can make you more creative*. <https://www.descript.com/blog/article/10-ways-generative-ai-can-make-you-more-creative>

Brynjolfsson, E., LI, D., & Raymond, L. R. (2023). *Generative AI at Work*. <https://www.nber.org/papers/w31161>

Cepelak, C. (2023). *Using Generative AI to Boost Your Creativity: Explore art, music, and literature*

with the help of generative AI models! <https://www.datacamp.com/blog/using-generative-ai-to-boost-your-creativity>

Daza, M. C. (2023). How Generative AI Revolutionizes the Creative Process? *Encora Insight*. <https://www.encora.com/insights/from-inspiration-to-creation-how-generative-ai-revolutionizes-the-creative-process>

Dell'Acqua, F., McFowland, E., Mollick, E., Lifshitz-Assaf, H., Kellogg, K., Rajendran, S., Kraymer, L., Candelon, F., & Lakhani, K. R. (2023). Navigating the Jagged Technological Frontier: Field Experimental Evidence of the Effects of AI on Knowledge Worker Productivity and Quality. *Harvard Business School Technology & Operations Mgt. Unit Working Paper*, 24(013). [https://papers.ssrn.com/sol3/papers.cfm?abstract\\_id=4573321](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4573321)

Duarte, D. (2023). *Creative or Generative AI: Unleashing the Power of Artificial Intelligence in the Creative Process*. <https://medium.com/@davidtduarte/creative-or-generative-ai-unleashing-the-power-of-artificial-intelligence-in-the-creative-process-2b98809c39e9>

Dwivedi, Y. K., Kshetri, N., Hughes, L., Slade, E. L., Jeyaraj, A., Kar, A. K., Baabdullah, A. M., Koochang, A., Raghavan, V., Ahuja, M., Albanna, H., Albashrawi, M. A., Al-Busaidi, A. S., Balakrishnan, J., Barlette, Y., Basu, S., Bose, I., Brooks, L., Buhalis, D., ... Wright, R. (2023). Opinion Paper: "So what if ChatGPT wrote it?" Multidisciplinary perspectives on opportunities, challenges and implications of generative conversational AI for research, practice and policy. *International Journal of Information Management*, 71, 102642. <https://doi.org/10.1016/j.ijinfomgt.2023.102642>

Eapen, T. T., Finkenstadt, D. J., Folk, J., & Venkataswamy, L. (2023). *How Generative AI Can Augment Human Creativity*. <https://hbr.org/2023/07/how-generative-ai-can-augment-human-creativity>

Esling, P., & Devis, N. (2020). Creativity in the era of artificial intelligence. *CoRR*, abs/2008.05959. <https://arxiv.org/abs/2008.05959>

Fan, A., Gokkaya, B., Harman, M., Lyubarskiy, M., Sengupta, S., Yoo, S., & Zhang, J. M. (2023). *Large Language Models for Software Engineering: Survey and Open Problems*. <https://doi.org/10.48550/ARXIV.2310.03533>

Forrester. (2023). *How Marketing Creativity Is Revolutionized By Generative AI*. <https://www.forbes.com/sites/forrester/2023/06/08/how-marketing-creativity-is-revolutionized-by-generative-ai/?sh=44e4bdd936e4>

Frey, T. (2023). *The Difference Between Human Creativity and Generative AI Creativity*. <https://futuristspeaker.com/artificial-intelligence/the-difference-between-human-creativity-and-generative-ai-creativity>

Google. (o.J.). Prompt Engineering für die generative KI. *Get Started with Machine Learning. E-Learning Kurs von Google*. <https://developers.google.com/machine-learning/resources/prompt-eng?hl=de>

Groeneveld, W., Luyten, L., Vennekens, J., & Aerts, K. (2021). Exploring the Role of Creativity in Software Engineering. *Proceedings of the 43rd International Conference on Software Engineering: Software Engineering in Society*, 1–9. <https://doi.org/10.1109/ICSE-SEIS52602.2021.00009>

Hou, X., Zhao, Y., Liu, Y., Yang, Z., Wang, K., Li, L., Luo, X., Lo, D., Grundy, J., & Wang, H. (2023). *Large Language Models for Software Engineering: A Systematic Literature Review* (arXiv:2308.10620). arXiv. <http://arxiv.org/abs/2308.10620>

Hu, L. (2022). *Generative AI and Future: GAN, GPT-3, DALL·E 2, and what's next*. <https://pub.towardsai.net/generative-ai-and-future-c3b1695876f2>

IBM. (o.J.). *The quest for AI creativity*. <https://www.ibm.com/watson/advantage-reports/future-of-artificial-intelligence/ai-creativity.html>

Jo, A. (2023). The promise and peril of generative AI. *Nature*, 614(1), 214–216.

Jovanovic, M., & Campbell, M. (2022). Generative artificial intelligence: Trends and prospects.

*Computer*, 55(10), 107–112.

Lawton, G. (2023). *What is generative AI? Everything you need to know*.  
<https://www.techtarget.com/searchenterpriseai/definition/generative-AI>

Leopoldino, K. D. M., González, M. O. A., de Oliveira Ferreira, P., Pereira, J. R., & Souto, M. E. C. (2016). Creativity techniques: A systematic literature review. *Product: Management and Development*, 14(2), 95–100.

Lewandowski, D. (2015). Evaluating the retrieval effectiveness of web search engines using a representative query sample. *Journal of the Association for Information Science and Technology*, 66(9), 1763–1775. <https://doi.org/10.1002/asi.23304>

McKinsey. (2023). *The state of AI in 2023: Generative AI's breakout year*.  
<https://www.mckinsey.com/capabilities/quantumblack/our-insights/the-state-of-ai-in-2023-generative-ais-breakout-year#/>

Moon, S. (2023). *Generative AI vs. Human creativity?* <https://www.thedatahunt.com/en-insight/generative-ai-vs-human-creativity>

Mukherjee, A., & Chang, H. (2023a). *The Creative Frontier of Generative AI: Managing the Novelty-Usefulness Tradeoff* (arXiv:2306.03601). arXiv. <http://arxiv.org/abs/2306.03601>

Mukherjee, A., & Chang, H. H. (2023b). *Managing the Creative Frontier of Generative AI: The Novelty-Usefulness Tradeoff*. <https://cmr.berkeley.edu/2023/07/managing-the-creative-frontier-of-generative-ai-the-novelty-usefulness-tradeoff>

NewMutation Technology. (2023). *The Rise of Generative AI: Unleashing Creativity and Innovation*.  
<https://www.linkedin.com/pulse/rise-generative-ai-unleashing-creativity-innovation-technology>

OpenAI. (2023). *Prompt engineering*. <https://platform.openai.com/docs/guides/prompt-engineering>

Remote Symphony. (2023). *Exploring the Frontier of Creativity: Generative AI Unleashed*.  
<https://remotesymphony.com/insights/exploring-the-frontier-of-creativity-generative-ai-unleashed/>

Runco, M. A. (2023). Updating the Standard Definition of Creativity to Account for the Artificial Creativity of AI. *Creativity Research Journal*, 0(0), 1–5.  
<https://doi.org/10.1080/10400419.2023.2257977>

Runco, M. A., & Chand, I. (1995). Cognition and creativity. *Educational Psychology Review*, 7(3), 243–267. <https://doi.org/10.1007/BF02213373>

Saravia, E. (2023). *Prompt-Engineering-Guide*. <https://github.com/dair-ai/Prompt-Engineering-Guide>

Shieh, J. (2023). *Best practices for prompt engineering with OpenAI API*.  
<https://help.openai.com/en/articles/6654000-best-practices-for-prompt-engineering-with-openai-api>

Stackpole, B. (2023). *Why generative AI needs a creative human touch*. <https://mitsloan.mit.edu/ideas-made-to-matter/why-generative-ai-needs-a-creative-human-touch>

Stahl, A. (2023). *The Rise of Generative AI: A New Era of Creativity and Innovation*.  
<https://www.linkedin.com/pulse/rise-generative-ai-new-era-creativity-innovation-alexander-stahl>

Wolfson, S. (2023). *This Is Not a Bicycle: Human Creativity and Generative AI*.  
<https://creativecommons.org/2023/02/21/this-is-not-a-bicycle-human-creativity-and-generative-ai/>

Ziegler, A., & Berryman, J. (2023). *A developer's guide to prompt engineering and LLMs*.  
<https://github.blog/2023-07-17-prompt-engineering-guide-generative-ai-llms/>



# How does the role of a Product Owner relate to the role of a Software Product Manager?

Timo Toikkanen, Sami Hyrynsalmi, Maria Paasivaara

LUT University, Department of Software Engineering, Lahti, Finland

sami.hyrynsalmi@lut.fi, maria.paasivaara@lut.fi

**Abstract:** In the Scrum Guide, the Product Owner (PO) is defined as being accountable for maximizing the value of the product they are responsible for. Thus, a Product Owner shares many responsibilities with a Software Product Manager (SPM), who is defined as a role governing the creation of the highest possible value to the business from the product. Despite the vast popularity of Scrum and other software development methods based on it, the role of a Product Owner has not received much academic research yet. This study contributes to the literature by assessing the similarities and differences between Product Owners and Software Product Managers using exploratory semi-structured interviews with 16 Agile software professionals. The study shows that the concept of product value is not always evident to Product Owners responsible for maximizing it. In addition, we identify five Product Owner Scenarios. Depending on the Product Owner Scenario, Product Owners' responsibilities overlap to a varying degree with Software Product Manager's responsibilities defined in the ISPMA SPM framework. Overall, further work is required to clarify the role and responsibilities of a Product Owner in various types of real-life organisations.

**Keywords:** Product Owner, Software product Management, Product Owner Scenario

## 1 Motivation

Customers invest in software for a reason. There needs to be a benefit that the customer will receive from the capabilities provided by the software. There are countless opportunities to leverage software for value creation. Assuming a typical, modern software development setting, the responsibility for maximizing the value falls on the shoulders of a nominated *Product Owner* (PO) [SS20]. Considering the ubiquity of software, the decisions made by POs have an impact on modern life in many ways. The PO is also a probable key contributor to the success of many contemporary businesses.

The role of a PO originates from Scrum [Ke19]. The Scrum Guide [SS20] describes the PO as a single person, not a committee, responsible for maximizing the product's value resulting from the Scrum team's work. A blog post by ScrumAlliance explains that the PO is responsible for tactical and strategic product decisions and is typically closely involved with the business side of the organization. At the same time, POs are given specific responsibilities to the Scrum team [SS20], which would typically be considered a part of the development organization. The blog post portrays the PO as a connector between product strategy and the development team. The attributes of great POs include being empowered, knowledgeable, empathetic, available, and decisive. The role is unique and challenging.

However, despite the substantial popularity of agile software development methods and Scrum in the scientific community, there is surprisingly little research on POs and product ownership. For instance, as of the end of January 2023, Scopus has indexed only 41 and 58 articles where "*product owner\**" is a part of the title or the keywords (either defined by the authors or the

indexing database), respectively. For comparison, a Scopus search with the term "*scrum*" returned 1,220 and 1,978 articles with respective limitations. The number of studies explicitly addressing POs is remarkably low. For example, in a recent bibliographical study conducted in 2021, the authors found merely 142 studies addressing software or digital product management [HSS21].

Nevertheless, the PO is given the strategic responsibility for maximizing the product's value [SS20]. In addition, the PO has specific operational responsibilities to the Scrum team, mainly related to managing the product backlog [SS20]. Combining the two might seem like a lot for one person. While POs are accountable for maximizing the value of the product, Ebert [Eb07] defines software product management (SPM) as '*the discipline and role that governs a product from its inception to market/customer delivery to generate the biggest possible value to the business.*' The ISPMA SPM framework [Ki22] provides a structured view of the elements of software product management. The above suggests a potential overlap and conflict between the roles of a PO and a software product manager.

Thus, this study focuses on exploring the potential overlap between the roles and responsibilities of a PO and the elements of software product management, as defined in [Ki22], via two research questions:

**RQ1** How do Product Owners understand value in the context of their products?

**RQ2** How does the role of a software product manager, as defined by the ISPMA SPM framework, relate to that of a Product Owner in Scrum?

To answer the presented questions, this study adopts a qualitative research approach with semistructured interviews. 16 experienced POs and Agile practitioners were interviewed, yielding 901 minutes of recorded interviews. For the data analysis, the Gioia method [GCH13] was applied. The remainder of the study is structured as follows. Section 2 covers the previous work on the roles of a PO and a software product manager. Section 3 discusses the research approach, and Section 4 the results. The implications of this work, as well as limitations and conclusions, are discussed in Section 5.

## 2 Background

Ken Schwaber and Jeff Sutherland introduced Scrum at an ACM research conference in 1995, implying that its history predates the Agile manifesto. Scrum has evolved significantly between 1995 and the 2020 Scrum Guide [Ve20], which as of 2022, can be considered the authoritative definition by the two co-founders. The Scrum Guide [SS20] introduces Scrum as a '*lightweight framework that helps people, teams and organizations generate value through adaptive solutions for complex problems*' and states that it is founded on empiricism and lean thinking. Scrum intends to be applicable also outside of software development.

According to the Scrum Guide [SS20], a Scrum team comprises a Scrum Master, developers, and a PO. The Scrum Guide [SS20] describes Scrum teams as cross-functional and selforganizing groups of individuals collaborating to deliver a product. According to the Scrum Guide [SS20], the size of a Scrum team should be less than ten persons, while Verheyen [Ve21] suggests that teams have the highest cohesion when the number of people is between five and seven. Each Scrum team has one and only one PO accountable for maximizing the product's value [SS20]. However, the provided framework does not give any definition or measure for the value of the product. The PO may represent the needs of several internal and external stakeholders. The PO defines a product goal that serves as a target for the team. Verheyen [Ve20] explains that the product goal should be derived from a longer-term product vision, although the Scrum Guide [SS20] does not mention the product vision. Product goal-related business expectations and ideas, in other words, requirements expressed by the PO, are

continuously captured as items in a product backlog. The PO is responsible for creating backlog items, ordering them, and communicating the product backlog to the team. The PO has authority and responsibility over the product backlog. The PO is responsible for ensuring that attendees are prepared to discuss the highest priority backlog items and their relationship with the product goal. The PO also proposes how the sprint could increase the product's value. Developers discuss with the PO to define a sprint goal and select the backlog items to be implemented in the sprint. The selected backlog items may be broken down into tasks [Su10].

Kittlaus [Ki12] discusses the potential conflicts between the roles of a PO and a Software Product Manager. The Scrum PO is a member of the development team, whereas the ISPMA SPM framework<sup>1</sup> [KF17, Ki22] represents development as one of the seven functional areas of Software Product Management. According to Kittlaus [Ki22], assigning the two roles to the same person is problematic because the operational responsibilities of a PO need to leave more time for the strategic responsibilities of a Software Product Manager. Kittlaus [Ki12] proposes that, in larger organizations, the two roles should be separate but dependent. In small organizations, the two roles may be assigned to the same person, taking care of the responsibilities of a PO as well as all the applicable parts of the ISPMA SPM framework [Ki12].

In academic research, Sverrisdottir et al. [SIJ14] interviewed five POs and found that the roles and responsibilities varied significantly between organizations. Based on a somewhat limited sample of five POs, Sverrisdottir et al. [SIJ14] concluded that the role and the responsibilities are seldom in perfect conformance with the Scrum Guide [SS20]. Bass et al. [Ba18] noted that few studies report how POs perform their role and what the related activities are. Bass et al. [Ba18] interviewed 55 POs and provided a grouping of the activities identified. However, detailed descriptions of the activities were left for further work. According to Bass et al. [Ba18], their research shows that POs perform a wide range of challenging activities requiring authority to influence.

Unger-Windeler et al. [UWKS19] conducted a mapping study to identify, analyze, and categorize existing research literature on the role of a PO. They found a need for additional, profound insights into the relationship between the roles of a PO and a Product Manager. One more takeaway from Unger-Windeler et al. [UWKS19] is that '*No PO role is like the other*'. Pursuing a similar line of thought, a LinkedIn post by Rafael Calovi<sup>2</sup> challenges the reader to find three persons who agree on the best definition of the role of a PO.

### 3 Research Method

This study is exploratory and uses a qualitative approach. We chose a semi-structured interview method to collect data and uncover unexpected perspectives. However, it is worth noting that open-ended questions may produce data that can be challenging to code and analyse [KP02]. The interview instrument consisted of eight parts, specifically formulated to address the research questions of the study. The first three parts introduced the study, interviewee, and the organization represented by the interviewee. The following section assessed conformance to the Scrum Guide [SS20], while the subsequent parts examined the meaning of product value and the potential overlap between the roles of a PO and a Product Manager. Lastly, the interview concluded with two straightforward questions that asked the interviewee to identify the essential skills required for the role and any challenges they may face.

---

<sup>1</sup> International Software Product Management Association (ISPMA) is an organization fostering software product management excellence. ISPMA has published a reference framework for software product management, see <https://ispma.org/bok/>

<sup>2</sup> Calovi, R., 2021. Safe is too prescriptive. [Online]. Available: <https://www.linkedin.com/pulse/safe-tooprescriptive-rafael-calovi/>



The interviews were conducted during the summer and autumn of 2022. All interviews were conducted one-on-one using Microsoft Teams video conferencing. Participants could choose to have the interview in English or Finnish. The objective of the interviews was to yield a sample that would bring out new insights into the role of a PO. The selection criteria for the participants were defined accordingly. Initially, the premise was that the participants of the study would be currently or previously working as POs. Candidates with at least five years of experience working in the software industry were preferred in the selection process. The intention was to select candidates representing different types of software development organizations, each with its own development processes. Interviewees from organizations that did not use Scrum were seen to add variety to the sample.

#	Org.	Len.	Summary
1	A	56'	Nearly 30 years of experience in the software industry, two years in the role of a PO
2	A	61'	The youngest professional among the participants with one year of experience in the software industry, recently taken over the role of a PO
3	A	80'	Approximately 20 years of professional experience in the software industry, four years in the role of a PO
4	A	47'	Approximately 20 years of professional experience, three to four years in the role of a PO
5	B	60'	More than 10 years of experience in the software industry, two years in the role of a PO
6	C	65'	Assumed the role of a PO during a period of five and half years
7	D	56'	Experienced Agile practitioner, never worked as a PO. Nevertheless, capable of explaining how the role and the related responsibilities were defined in the organization
8	B	52'	Various roles in the software industry, three years in the role of a PO
9	E	54'	Background in both industry and academia, three and half years of experience in the role of a PO
10	F	52'	Several years of experience working in the PO and Product Manager type of positions
11	G	43'	PO on a mission to develop business analytics for the organization
12	H	51'	Background working as an Agile Consultant and a PO, currently working as a consultant specializing in due diligence
13	I	44'	Long background in the software development industry, currently working as a PO
14	J	75'	Various positions in information technology including PO responsibilities
15	D	45'	PO based in Sweden
16	K	59'	Various PO and Product Management positions

Table 1: The interviewees with their respective organizations and the length of the interview records in minutes

The interviewed POs were initially recruited from personal networks. The sample was expanded throughout the interview process by asking interviewees to refer to other POs whom they knew. This kind of sampling technique is known as snowball sampling [Go61] or chain referral sampling. However, because people are likely to know and provide referrals to other people with similar traits, the reliance on personal networks may introduce sampling bias. The

resulting sample is non-statistical and not necessarily representative of the whole population of professionals identifying themselves as POs. Conceptually, with a non-statistical sample, findings cannot be generalized back to the population. Therefore, any of the results must be considered exploratory and not conclusive, in line with its objectives.

The plan was to continue the interview and participant recruitment process until patterns or repetitions arose. A total of 16 experts were interviewed for the study (see Table 1). All except for one of the interviewees were currently or previously working as POs. Nevertheless, this person was capable of explaining how the role and its related responsibilities were defined in the organization. All but one of the interviewees met the criteria for having at least five years of experience in the software industry. The sample includes participants representing smaller companies as well as some of the largest technology companies in the world. While most of the participants worked in the software product business, some worked in the software service business. It's worth noting that the sample includes participants from organizations using the job title 'Product Manager' instead of 'Product Owner'.

The organizations represented by the interviewees are briefly described in Table 2, along with a brief description of the industry in which they operate. Some of the organizations are part of large, multinational companies. The roles and responsibilities of a PO may vary from one part of the organization to another, implying that the experiences of the respondents are not necessarily generalizable to the entire organization. To maintain anonymity, no further information on the respondents or the organizations is not disclosed.

#	Company Sector	Identified Product Owner Scenario
A	Communication industry	SAFe-like Organisation
B	Software service and product company	Compact Organisation
C	Medical technology	Separate Management Product
D	Insurance sector, respondents from Finland and Sweden	Internal Customer
E	Software products for the medical sector	SAFe-like Organisation
F	Online advertisement	Internet Company
G	Forest industry	Internal Customer
H	Management consultancy	-
I	Medical technology	Separate Management Product
J	Online retail	Internet Company
K	Prominent technology company	Internet Company

Table 2: Summary of the Organisations

The recorded audio was transcribed by a professional transcription company using naturalized transcription, also referred to as 'intelligent verbatim' transcription, which aims to follow the conventions of written language [Bu00], ignoring the characteristics of spoken languages, such as repetition, filled pauses, and grammatical errors. As a result, 901 minutes of recorded audio were transcribed to about 100,000 words and 137 pages of text. The transcribed text was anonymized to ensure that the respondents or their employers could not be identified. QSR International's NVivo software was used as the tool for conducting qualitative data analysis. The coding of the interview data was guided by the Gioia method [GCH13], which is a widely accepted approach for qualitative data analysis. The method allows for inductive concept creation while maintaining '*qualitative rigour*' [GCH13]. The analysis process was iterative in nature. Along the process, concepts emerge from the data, and the process results in a three-

layer abstraction hierarchy. While reading the interview transcriptions, codes for first-order categories were developed to mark parts of the text that were interpreted to convey a common message. First-order categories were grouped into more abstract second-order themes. Second-order themes were mapped to even more abstract codes referred to as overarching dimensions.

## 4 Results

### Product Owner Scenarios

The roles and responsibilities of Product Owners can vary depending on the organizational structure to which they belong. Unger-Windeler et al. [UWKS19] concluded their mapping study as follows: ‘We hypothesize that the description of the Product Owners environment – especially in terms of organizational structure and the collaboration with traditional management roles – will make a difference in the description of this role.’ The results of this study support their hypothesis, as five distinct Product Owner Scenarios emerged from the interview data and analysis. These scenarios are primarily characterized by organizational structure and business model. Table 2<sup>3</sup>

shows the mapping between organizations and their respective Product Owner Scenarios. The five identified Product Owner Scenarios are:

**Internal Customer.** Product Owners are typically members of software development teams responsible for creating software, such as business analytics, for internal use within an organization. In the Internal Customer scenario, the PO’s role is primarily focused on meeting the internal customer’s needs and requirements for the software. This scenario generally limits the commercial aspects of the role. However, the PO may also be involved in tasks such as internal invoicing, product marketing within the organization, and potentially even sourcing.

**Compact Organisation.** In the Compact Organisation scenario, Product Owners are often perceived as versatile ‘jack-of-all-trades’ individuals who assume many of the responsibilities traditionally associated with software product management. While some aspects of product strategy may be handled by higher-level personnel within the company, several respondents representing Compact Organisations noted that Product Owners still manage a heavy workload.

**Separate Product Management.** In the Separate Product Management scenario, the Product Owner is a member of a development team who interfaces with a separate product management function, which is typically located outside of the development organization. In this scenario, the PO is responsible for product planning related to the software component of the product.

**SAFe-like Organisation.** SAFe-like Organisations can be considered a special case of the Separate Product Management scenario. In the interviews, the POs representing SAFelike Organisations were generally technically oriented, with a focus on requirements engineering. One distinguishing characteristic of these organizations is that the longer-term product roadmap is controlled by the product management function rather than the PO. While the respondents affiliated with this scenario noted that their organizations did not claim full compliance with the SAFe framework, they were clearly influenced by it. As such, the name ‘SAFe-like’ was used to describe this scenario.

**Internet Company.** In the Internet Company scenario, the organizations conduct most of their business and interact with customers online, and some may use the SaaS business model.

---

<sup>3</sup> Respondent 12 shared experiences from various organizations instead of focusing on Organisation H. Thus, Organisation H is not explicitly dealt with in this study and is not assigned to any Product Owner Scenario.

In this scenario, the interviewees representing the organization were referred to as product managers rather than Product Owners. Unlike in the Separate Product Management and AFe-like Organisation scenarios, the product managers were members of development teams responsible for specific product functionalities. The development teams worked relatively independently of each other, but coordination between product managers from different teams was still required. Each team and product manager had specific contributions to the organization's business objectives, and the development teams had autonomy in defining their own ways of working.

### **RQ1: Value in the context of PO's products**

The study aimed to challenge the respondents' understanding of the meaning of value. Respondent 7, representing the Internal Customer scenario, noted that since the purpose of the product was to automate a process, the product would not provide user value to any individual. Instead, the product would provide business value to the organization and be measured in terms of cost savings rather than revenue generation. In contrast, the POs in the Compact Organisation scenario focused on the economic success of the product from the vendor's perspective. They tracked sales revenue and product development costs, indicating that they took responsibility for the profitability of the product.

*'Since I come from a sales background, the revenue brought in is what matters to me. Are customers willing to pay for the product, how is it priced, and is the business around the product profitable... And, of course, I have been contemplating the value for the customers.'*

— Respondent 8 representing a Compact Organisation

Respondents 6 and 13, representing the Separate Product Management scenario, associated user value with the economic success of the product. Respondent 6 explained that the product needs to be user-friendly to build lasting relationships with customers, highlighting the importance of user value. Overall, the interviewed POs primarily approached product value from the perspective of the user. The cross-tabulation of the data revealed that user-centeredness was particularly prevalent in SAFe-like organizations. Notably, the POs in these organizations did not refer to any monetary indicators of the product value.

Two of the interviewees associated technical debt with product value. Technical debt can manifest itself as quality issues, delayed deliveries, and increased costs, indirectly affecting customers and users. Interviewees suggested that when ordering the product backlog, the POs should consider the need for refactoring. The organizations in the Internet Company scenario stood out from the rest in quantifying product value. The decision-making of the POs in these organizations was guided by product metrics and objectives. Through product analytics, they appeared capable of establishing a strong link between user value and the financial performance of the product.

To conclude, the results highlight the context-dependent nature of value. All interviewees presented user-centered viewpoints, reflecting the Scrum Guide [SS20] and the principles of Agile that emphasize customer satisfaction. In the Internal Customer scenario, the business case of the product was based on cost saving rather than revenue generation. The POs in the SAFe-like Organisation scenario did not refer to any financial indicators of the product value. On the other end of the spectrum, the most business-oriented POs were observed in the Compact Organisation scenario and in the Internet Company scenario. The organizations in the Internet Company scenario were advanced in using quantifiable data to measure product value and POs' success in maximizing it.

## **RQ2: How does the Product Owner role relate to the role of a Product Manager?**

RQ2 outlines the intersection between the role of a PO and the role of a Product Manager as defined by the ISPMA SPM framework. The interview questions and the coding of the data for RQ2 draw inspiration from the ISPMA SPM framework. According to Kittlaus [Ki22], the framework provides a holistic view of software product management activities.

When comparing the accountabilities of a PO defined in the Scrum Guide [SS20] to the range of activities included in the ISPMA SPM framework, the latter is remarkably wider. The PO responsibilities defined in the Scrum Guide [SS20] are mainly related to development, which is only one of the seven areas covered by the ISPMA SPM framework. In contrast, the ISPMA SPM framework indicates that Software Product Managers are directly accountable for product strategy and planning, and they either participate in or coordinate the other activities included in the framework. However, the Scrum Guide [SS20] also assigns the PO with the all-encompassing responsibility of maximizing the product's value.

Cross-tabulation showed differences between PO scenarios and their relationship to the surrounding organization. In the Internal Customer scenario, it is assumed that the product is developed for internal use within the organization. The responsibilities related to the business leadership of the product are limited. For example, marketing may be limited to the internal promotion of the product, whereas sales and fulfillment may be limited to the definition of internal Service Level Agreements (SLAs). However, as in any other scenario, the Internal Customer scenario requires the PO to understand user needs, steer development, and communicate the value of the product. The interviewed POs in the Internal Customer scenario were involved in some of these activities, with respondents 11 and 15 being not only involved in but responsible for product planning.

The interviewees in the Compact Organisation scenario are Product Managers as defined by the ISPMA SPM framework, who additionally take on the role of a PO in development teams. As Product Managers, they are responsible for or participate in, the wide range of activities defined by the ISPMA SPM framework. Respondent 5 explained the practice of organizing 'roadmap meetings' for product strategy and product planning-related decision-making. The roadmap meetings also ensured that the resulting decisions were adequately communicated within the organization. In general, Respondents 5 and 8 indicated active involvement in product strategy, product planning, marketing, sales, and fulfillment, as well as delivery services and support, typically in collaboration with other stakeholders of the organization.

The Separate Product Management scenario and the SAFe-like Organisation scenario have many similarities. In both scenarios, POs are part of development teams and interface with a separate Product Management function. POs in these scenarios are primarily responsible for development activities, with a focus on requirements engineering. Meanwhile, product managers are responsible for product strategy and planning, and they take on the business leadership of the product. According to the ISPMA SPM framework, product management 'orchestrates' product marketing, sales, and fulfillment, as well as delivery services and support. In the Separate Product Management scenario and in the SAFe-like Organisation scenario, the POs participate in related activities as technical experts, but they do not orchestrate or coordinate them. However, the Separate Product Management and SAFe-like Organisation scenarios are not the same. In SAFe, POs take input from the program backlog defined by the product management. In comparison, Respondent 13, representing the Separate Product Management scenario, had a greater degree of autonomy in defining the product roadmap, practically excluding the hardware components of the product.

In the Internet Company scenario, the respondents see themselves as product managers, but they also have a close relationship with the development teams. Respondent 10 is part of a development team, whereas Respondents 14 and 16 hold senior product manager roles and lead initiatives that involve multiple development teams. These teams are organized around business

areas, and it's worth noting that the product created by a development team may only be a part of the overall product offered to the market. Consequently, the product created by a development team may differ from what the market perceives as a specific product.

Product managers in the Internet Company scenario are empowered within their teams and business areas. They have clear business objectives to meet. Nevertheless, Respondent 10 explained that many of the aspects of product strategy are defined higher up in the organization. These aspects of product strategy, such as pricing, are broader than the business area of the team, implying that the representatives of this Product Owner Scenario only need to address some of the activities defined in the ISPMA SPM framework. The Product Manager is only responsible for some of the things that would be required for an individually branded, stand-alone product.

To summarise the results of RQ2, there were significant differences in how POs relate to the surrounding organization. In addition to being POs in development teams, the interviewees representing the Compact Organisation scenario generally fulfill the role of a Software Product Manager as defined by the ISPMA SPM framework. In the SAFe-like Organisation scenario, the PO takes a sharp focus on development. In the Internet Company scenario, development teams have been organized around business areas, allowing each of the development teams to have a Product Manager with meaningful business objectives. Nevertheless, different teams contribute to the overall offering of the company.

It is possible that Internet Companies chose to position Product Managers within development teams in order to emphasize the importance of their role in driving product development and ensuring that the final product meets the needs of the business. By naming the role 'Product Manager', these companies may have sought to highlight the business leadership aspects of the role, indicating that the Product Manager is responsible not only for overseeing the technical aspects of product development but also for ensuring that the product aligns with the overall business strategy and goals.

Based on the study, the Product Owner is primarily a technical role within software development organizations. However, in the Compact Organisations scenario, the interviewed Product Managers also assumed the role of PO in development teams, similar to the Internet Companies scenario. These scenarios were the only ones where the interviewees assumed ISPMA SPM framework-like responsibilities. Table 3 summarises the results of RQ2.

<b>Product Owner Scenario</b>	<b>Relation of Product Owner role to SPM framework</b>
Internal Customer	The ISPMA SPM framework is only partially applicable in this scenario.
Compact Organisation	The POs assumed the role of a Product Manager as defined in the ISPMA SPM framework.
Separate Product Management SAFe-like Organisation	The POs focused on the development area of the ISPMA SPM framework.
Internet Company	The respondents assumed the role of a Product Manager as defined in the ISPMA SPM framework, although there may be several development teams and Product Managers contributing to the market offering.

Table 3: Summary of RQ2 results

## 5 Summary

The key observations of this research are summarised as follows:

**Firstly**, if the parties involved in a discussion do not agree on the definition of product value, it is nearly meaningless to talk about maximising it. This paper argues that careless use of the term product value might create a false sense of professionalism that does not exist. Every now and then, the PO should take a step back and think strategically about where the value of the product is and whether there is a way to measure it.

Value is co-created with customers instead of being produced by companies. The roles of the producer and consumer become intertwined in value creation. The authors suggest that a service lens on value creation [BLV14] can help POs to understand the value of the product and support innovation in the software business. Given that the value is highly individual and ultimately judged by the customer, POs can only anticipate value. Nevertheless, the anticipated value should be captured in a value proposition conveying the benefit provided to the customer.

**Secondly**, appropriately defined metrics can help POs to make informed product decisions, and when linked to an organisation's business objectives, product and business metrics can also add a great deal of depth to the role of a PO. The study found that the POs representing the Internet Company scenario were far ahead of the rest in quantifying product value. However, outside of this scenario, most of the interviewees approached product value by anticipating the product's usefulness or usability, and only a few relied on user-research in their decision-making. The paper questions who would drive the implementation of product analytics if not POs themselves and proposes that POs should consider measuring product value more effectively and making it visible to their teams.

**Thirdly**, POs representing the SAFe-like Organisation scenario reported a disconnect from the users of the product. In this scenario, the POs mainly interfaced with Product Managers rather than directly with users. Assuming that the objective of the PO is to maximize the usefulness or usability of the product, the disconnect from the users is concerning. Whether this is a common problem in organizations applying the market-leading framework for Large-scale Agile could be dealt with in future research.

**Fourthly**, according to the Scrum Guide [SS20], the PO is held accountable for maximizing the value of the product. The guide defines the internal responsibilities within the development team, but it does not aim to explain what POs should do to maximize value. Nevertheless, it is a product leadership role that shares many similarities with the role of a Product Manager. The ISPMA SPM framework outlines the activities typically carried out by Product Managers. The organizations represented by the interviewed POs are widely different from each other. In some organizations, POs fulfill the role of a Product Manager as defined by the ISPMA SPM framework. In other organizations, POs focus purely on the development activities of the framework.

### Limitations and future work

The selection of participants may pose a threat to internal validity, even though exploratory research does not aim to confirm any causal relationships or provide conclusive results. As discussed in the context of interview planning, the sample is non-statistical, and the representativeness of the sample cannot be guaranteed in terms of internal validity. Given the heterogeneity of the Product Owner population and the broad scope of the research questions, a larger sample size may have provided new perspectives, but it could also have resulted in many repetitions.

It's worth noting that multiple respondents represented Organisations A, B, and D in our study. While each respondent provided their own unique perspective, there were no significant

discrepancies between those representing the same organization. By using semi-structured interviews, we were able to ask follow-up questions as necessary, which could have increased the internal validity of our results. However, the limitations discussed earlier may reduce the generalisability, or external validity, of our findings. Further research would be required to draw conclusive results.

## Conclusions

This paper presents a study on the role of a Product Owner and how it relates to the role of a Software Product Manager. The PO is responsible for maximizing the value of the product. Through empirical inquiry with 16 software development professionals, this study shows that POs have varying perceptions of what constitutes value. Additionally, while the role of a PO overlaps with that of a SPM, the specific responsibilities of a Product Owner vary between different companies.

## References

- [Au21] Auer, F.; Ros, R.; Kaltenbrunner, L.; Runeson, P.; and Felderer, M: Controlled experimentation in continuous experimentation: Knowledge and challenges. *Information and Software Technology*, 134, p.106551, 2021.
- [Ba18] Bass, J.M.; Beecham, S.; Razzak, M.A.; Canna, C.N.; Noll, J.: An empirical study of the product owner role in scrum. In: *Proceedings of the 40th International Conference on Software Engineering: Companion Proceedings*. pp. 123–124, May 2018.
- [BLV14] Bettencourt, L.A.; Lusch, R.F.; Vargo, S.L.: A service lens on value creation: marketing’s role in achieving strategic advantage. *California management review*, 57(1):44–66, 2014.
- [Bu00] Bucholtz, M.: The politics of transcription. *Journal of pragmatics*, 32(10):1439–1465, 2000.
- [Eb07] Ebert, Christof: The impacts of software product management. *Journal of Systems and Software*, 80(6):850–861, 2007.
- [GCH13] Gioia, Dennis A.; Corley, Kevin G.; Hamilton, Aimee L.: Seeking Qualitative Rigor in Inductive Research: Notes on the Gioia Methodology. *Organizational Research Methods*, 16(1):15–31, 2013.
- [Go61] Goodman, L.A.: Snowball sampling. *The annals of mathematical statistics*, pp. 148–170, 1961.
- [HSS21] Hyrynsalmi, Sami; Suominen, Arho; Seppänen, Marko: A Bibliographical Study of Software Product Management Research. In: *2021 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC)*. pp. 1–8, 2021.
- [Ke19] Kelly, A.: *The Art of Agile Product Ownership*. Apress, 2019.
- [KF17] Kittlaus, H.-B; Fricker, Samuel A.: *Software Product Management — The ISPMA-Compliant Study Guide and Handbook*. Springer, 2017.
- [Ki12] Kittlaus, H.-B.: Software product management and agile software development: conflicts and solutions. In: *Software for People*. Springer Berlin, Heidelberg, pp. 83–96, 2012.
- [Ki22] Kittlaus, H.-B: *Software Product Management: The ISPMA®-Compliant Study Guide and Handbook*. Springer Berlin, Heidelberg, Germany, 2022.
- [KP02] Kitchenham, B.A.; Pfleeger, S.L.: Principles of survey research: part 3: constructing a survey instrument. *ACM SIGSOFT Software Engineering Notes*, 27(2):20–24, 2002.
- [SIJ14] Sverrisdottir, H.S.; Ingason, H.T.; Jonasson, H.I.: The role of the product owner in scrum-comparison between theory and practices. *Procedia-Social and Behavioral Sciences*, 119:257–267, 2014.



- [SS20] Schwaber, K.; Sutherland, J.: The Scrum Guide — The definitive guide to Scrum: The rules of the Game, November 2020.
- [Su10] Sutherland, J.: Jeff Sutherland's Scrum Handbook. Scrum Training Institute, Boston, 2010.
- [UWKS19] Unger-Windeler, C.; Klünder, J.; Schneider, K.: A mapping study on product owners in industry: identifying future research directions. In: 2019 IEEE/ACM International Conference on Software and System Processes (ICSSP). IEEE, pp. 135–144, May 2019.
- [Ve20] Verheyen, G.: Scrum – A Brief History of a Long-Lived Hype. White Paper, December 2020.
- [Ve21] Verheyen, G.: Scrum – A Pocket Guide – 3rd edition: A Smart Travel Companion. Van Haren, Hertogenbosch, the Netherlands, 2021.

# Digital Product Management Week in Göteborg im Juni

## Calls for Participation und Registrierung

<https://digitalproductmanagementweek.org/>

**Registrierung:** <https://digitalproductmanagementweek.org/registration>

10. – 14. Juni 2024, Göteborg, Schweden

Werte Mitglieder der Fachgruppe Software Produktmanagement und des Fachausschusses Management der Anwendungsentwicklung und -Wartung,

Vom 10. bis 14. Juni findet in Göteborg im Lindholmen Science Park die „Digital Product Management Week“ statt, an deren Organisation die Fachgruppe beteiligt ist. Die „Digital Product Management Week“ fungiert als Dach für unterschiedliche Veranstaltungen, die alle für Sie von Interesse sein könnten. Die Teilnahme als solche ist kostenlos, Sie müssen lediglich Reisekosten bzw. Übernachtung budgetieren. Folgende Veranstaltungen erwarten Sie:

- Die „1st International Conference on Digital Product Management“ (ICDPM) als wissenschaftliche Konferenz – hier waren Einreichungen in Form von wissenschaftlichen Beiträgen (full paper von bis zu 12 Seiten + Literaturverzeichnis oder short paper von bis zu 6 Seiten plus Literaturverzeichnis) bis 3.4. möglich. Die ICDPM findet am Mittwoch, 12. Juni statt, der Tagungsband erscheint bei Springer als Teil der Reihe LNBP Proceedings, Details finden Sie hier: <https://digitalproductmanagementweek.org/icdpm>
- Die „Summer School on Digital Product Management“, die sich explizit an Nachwuchswissenschaftler:innen richtet und vom 10.-13. Juni stattfindet. Hier sind Bewerbungen noch bis 30.04.2024 möglich: <https://digitalproductmanagementweek.org/summerschool>
- Der 6. „Software Product Management Summit Europe“ als stärker praxisorientierte Tagung, die am Dienstag, 11. und Mittwoch, 12. Juni stattfindet. Das Programm finden Sie hier: <https://digitalproductmanagementweek.org/spm-summit-call-for-contributions>
- Einen „Startup Roundtable“, der am Donnerstag, 13. Juni stattfindet und zu dem Sie hier mehr Informationen finden: <https://digitalproductmanagementweek.org/startup>

Wir freuen uns, dass einige Einreichungen aus den Kreisen der Fachgruppe und des Fachausschusses eingegangen sind und würden uns freuen, auch einige Teilnehmerinnen und Teilnehmer bei der wissenschaftlichen Konferenz sowie den weiteren Veranstaltungen, die im Rahmen der „Digital Product Management Week“ in Gothenburg stattfinden werden, begrüßen zu dürfen!

Beste Grüße

Andreas Helferich & Dimitri Petrik



## **KI-basierte Analyse rollenbasierter Gesprächsverläufe**

(Ziele und Möglichkeiten der angewandten KI in der Mediationsforschung)

10. Juni 2024 (12:00 bis 18:00 Uhr)

SOWI Gebäude, Uni Innsbruck – SR 14 im 4. OG, Universitätsstraße 15

Im Mittelpunkt des Workshops stehen die Ergebnisse, mit Hilfe von Large Language Models – (kurz LLMs) durchgeführter KI-Analysen transkribierter Mediationssitzungen. Ergebnisse beziehen sich u.a. auf die prozentuale Verteilung von Gesprächsanteilen, gesprächsverlaufsorientierte Sentimentanalysen oder auch kontextbasierte Analysen anonymisierter Transskripte. Darüber hinaus entstanden Ergebnisse im Zusammenhang mit einer notwendigen Anonymisierung der verarbeiteten Transskripte. Im Workshops sollen die Ergebnisse einem breiten Fachpublikation zur Diskussion gestellt werden.

*Andreas Schmietendorf (HWR Berlin/OvG-Universität Magdeburg):*

Eröffnung/Impuls: KI-basierte Analyse rollenbasierter Gesprächsverläufe

*Hans-Dieter Will (BAFM e.V.):*

Bedürfnisse der Mediationsforschung

*Peter Münte (Uni Innsbruck):*

Forschung im Diskurs durchgeführter Mediationssitzungen

*Walter Letzel (TU Berlin):*

Ziele der angewandten KI in der Mediationsforschung

*Sandro Hartenstein (HWR Berlin/OvG-Universität Magdeburg):*

KI-basierten Analyse von Transskripten durchgeführter Mediationssitzungen

Moderierte Diskussionsrunde:

- Datenproblem der Mediationsforschung (Umgang mit Primär- und Sekundärdaten)
- Ergebnisse einer KI-basierten Analyse von transkribierten Mediationssitzungen
- Ergebnisinterpretation und Herausforderungen für die Mediationsforschung

Zur Deckung der Unkosten ist die Einnahme einer kleinen Tagungspauschale (max. 50 Euro für Imbiss und Raummiete) geplant. Aktuell laufen diesbezüglich noch Abstimmungen mit potentiellen Sponsoren.

Anmeldung via Walter Letzel:

Email: [wl@letzel-consult.de](mailto:wl@letzel-consult.de)

Infos auch unter:







## **Call for Papers / Aufruf zur Einreichung von Beiträgen**

### **Neues Arbeiten in Projekten – Teamarbeit neu interpretiert**

TH Mittelhessen, Campus Friedberg, 26.+27.09.2024

Zehnte gemeinsame Tagung der Fachgruppen Vorgehensmodelle und  
Projektmanagement im Fachbereich Wirtschaftsinformatik der Gesellschaft für  
Informatik e.V. (GI), in Kooperation mit der Fachgruppe IT-Projektmanagement der  
GPM e.V., sowie dem PMI Germany Chapter e.V.

### **Ausgangslage**

Das Arbeiten in verschiedensten Projektformaten hat sich in nahezu allen Feldern der Wissensarbeit etabliert, um der zunehmenden Komplexität bei der Erarbeitung von Lösungen gerecht zu werden. Agile und hybride Vorgehensmodelle bzw. Organisationsansätze haben sich in verschiedensten Ausprägungen in der Projektarbeit etabliert und zum Mindset-Shift von der Ergebnis- zur Lösungsorientierung beigetragen. Agile Muster werden dabei von der Projektebene auf die Unternehmensebenen skaliert und führen zu neuen Governance-Strukturen und -prozessen sowie Führungsverständnissen (u.a. Agile Leadership). Die Weiterentwicklung der Projektarbeit selbst und vor allem neue Wege des Zusammenarbeitens in Projektteams scheinen heute aus dem Blickfeld der Forschung zu geraten. Daher ist es an der Zeit, die Strukturen der agierenden Teams wieder ins Blickfeld zu rücken.

Die Arbeit in Projekten hat sich in den letzten Jahren deutlich verändert. Die Flexibilisierung von Wissensarbeit hinsichtlich Raum, Zeit und Arbeitsform wurde während der Pandemie gewollt forciert und hat den agierenden Individuen immer neue Freiheiten gewährt, die zu erweiterten und neuen Möglichkeiten des mobilen Arbeitens (z.B. Home Office, Coworking, Arbeitsnomadentum, Workation etc.) geführt haben. In der Folge sind damit auch die Wünsche und Ansprüche des Individuums immer mehr in den Mittelpunkt gerückt und die Kultur des Ermöglichens wird zum Führungsziel. Der War-for-Talents im IT-Sektor hat sein Übriges dazu beigetragen, dass Individualwünsche akzeptiert werden. Das Projektteam selbst und die Kooperation in und zwischen Teams haben sich mit der Digitalisierung stark verändert. Selbstregulierende Prozesse in den Teams gewinnen an Bedeutung. Virtualisierung führt zu neuen Formaten und Ritualen in Projekten. KI-basierte Tools werden zum neuen Hilfsmittel bis hin zum Co-Piloten bei Entwicklungsarbeiten. Der Faktor Mensch scheint ausgehebelt zu werden und die Gefahr der weiteren Isolierung der Teammitglieder ist real.

Wie wird sich das Arbeiten in Projektteams in einer veränderten und zunehmend von neuen KI-Tools geprägten Arbeitswelt weiterentwickeln? Wie kann Teamarbeit neu interpretiert und vor allem gestaltet werden? Diese Fragen aus Sicht von Wissenschaft und Praxis zu vertiefen und Lösungen zu diskutieren bildet den Themenschwerpunkt der PVM 2024 mit dem Leitthema *Neues Arbeiten in Projekten – Teamarbeit neu interpretiert*.

## **Ziele der Fachtagung**

Ziel der Veranstaltung ist es, einerseits einem Fachpublikum fundierte Ansätze aus der Wissenschaft mit Erfahrungen zu deren Anwendung in der Praxis vorzustellen und andererseits für die Praxis relevante Themen der Wissenschaft vorzustellen, um Raum für die fachübergreifende Diskussion und den Erfahrungsaustausch zu geben.

## **Themenschwerpunkte**

Um diese Fragen im Spannungsfeld zwischen Academia und Praxis zu diskutieren, laden wir in diesem Jahr insbesondere – aber nicht ausschließlich – zu Beiträgen zu folgenden Themenkomplexen ein:

- Projektarbeit und Führung in verteilt und zunehmend virtualisiert arbeitenden Teams
  - Attraktive Arbeitsmodelle für das IT-Projektmanagement in Zeiten des War-for-Talents
  - Teamstrukturen neu definiert - Erfahrungen agiler Governancestrukturen (u.a. SAFe, LeSS, Holocracy, Spotify-Modell) und Implikationen für künftige Teamkonzepte
  - Neue Wege der Teamführung und des Leadership in IT-Projekten
  - Wege zum Empowerment von Projektmitarbeitenden für eine höhere Team-Performance
- Moderne IT-Lösungen und deren Einfluss auf individuelle Arbeitssituation und die Zusammenarbeit in und von Projektteams
  - KI als Arbeitswerkzeug und Co-Pilot in der Projektarbeit.
  - KI als Unterstützung im Projektmanagement (Planung, Steuerung, Reporting)
  - Chancen und Grenzen der Virtualisierung von Projektarbeit
- Moderne Teamkonzepte im Einklang mit (neuen) Vorgehensmodellen in IT-Projekten
  - Anpassung von Vorgehensmodellen an die neuen Arbeitsformen von Individuen & Teams
  - Ideen und systemische Ansätze für post-agile Vorgehensmodelle
  - Transparenz zu Projektfortschritten in fremd- und selbstregulierten Teams
  - Toolunterstützung für moderne Teams (u.a. Zusammenarbeit, Management)

## **Beitragskategorien**

### **Wissenschaftliche Beiträge / Fachbeiträge**

Hierunter fallen Ergebnisse abgeschlossener Forschungsvorhaben und fundierte wissenschaftliche Ergebnisse, um diese mit den Teilnehmern aus Wissenschaft, Studium und Praxis zu diskutieren.

### **Denkanstöße und Impulse**

Eine wichtige Aufgabe der GI-Fachgruppen ist es, sich mit der Zukunft des Fachgebiets zu beschäftigen. Reifende Ideen oder kontroverse bzw. provokative Ansichten können in

Impulsvorträgen vorgestellt werden, welche anschließend im Auditorium diskutiert werden sollen. Durch den Austausch sollen Denkanstöße und Impulse für die Teilnehmenden und für die künftige Fachgruppenarbeit entstehen. Die Einreichung kann als Kurzbeitrag erfolgen.

### **Berichte aus der Praxis**

In Praxisberichten sollen sowohl positive als auch negative Projekterfahrungen berichtet werden. Die Einreichung kann als Foliensatz oder Kurzbeitrag erfolgen.

Im Review-Prozess wird jede dieser Inhaltskategorien nach passenden Kriterien bewertet.

### **Diskussion bereits publizierter Arbeiten**

Ähnlich dem „Journal-first“-Modell anderer internationaler Konferenzen, können Beiträge eingereicht werden, welche bereits in renommierten Journalen und Konferenzen publiziert oder zur Publikation/Präsentation angenommen wurden.

## **Hinweise für Studierende und NachwuchswissenschaftlerInnen**

Insbesondere fordert das Programmkomitee auch Studierende und NachwuchswissenschaftlerInnen auf, mit eigenen Beiträgen in allen oben aufgeführten Kategorien aktiv an der Tagung teilzunehmen, um auf diesem Weg die Community der Fachgruppen kennenzulernen. Studierende können einen Zuschuss zur Teilnahme an der Tagung beantragen.

## **Formate, Einreichung und Fristen**

Als Formate sind vorgesehen:

- Langbeiträge mit maximal 10 Seiten
- Kurzbeiträge mit maximal 5 Seiten
- Foliensätze (für Praxisberichte)

Weitere Details zur Einreichung (Vorlagen, Seitenzahlbegrenzungen, Einreichungssystem) findet man auf der Website <https://pvm-tagung.de/beitrag-einreichen>

Ein herausragender Beitrag wird mit dem Best Paper Award prämiert.

Bitte beachten Sie folgende Fristen:

- **15.06.2024:** Einreichung des Beitrags durch die Autoren
- 15.07.2024: Benachrichtigung der Autoren
- 30.07.2024: Einreichung finaler Version
- 31.08.2024: Einreichung der finalen Vortragsfolien

## **Tagungsband und Indizierung**

Die Textbeiträge werden im gedruckten Tagungsband in den GI Lecture Notes in Informatics publiziert (<https://www.gi.de/service/publikationen/lni.html>) und von dblp (<http://dblp.uni-trier.de/>) indiziert. Foliensätze werden nicht im Tagungsband publiziert.



## **Tagungsort**

Die Tagung findet am 26. und 27. September 2024 in Friedberg statt:

TH Mittelhessen

Campus Friedberg

Wilhelm-Leuschner-Straße 13

61169 Friedberg

<https://www.thm.de>

## **Kontakt**

Auf der Webseite (<https://www.pvm-tagung.de>) und auf LinkedIn (<https://www.linkedin.com/company/pvm-tagung>) werden laufend aktualisierte Informationen zur Tagung bereitgestellt.

Für Rückfragen wenden Sie sich bitte an [info@pvm-tagung.de](mailto:info@pvm-tagung.de).

Für das Programmkomitee der Tagung:

Dr. Martin Bertram (Vorstandsmitglied PMI Germany Chapter)

Prof. Dr. Martin Engstler (Hochschule der Medien, Stuttgart)

Prof. Dr. Eckhart Hanser (Sprecher der Fachgruppe Vorgehensmodelle)

Prof. Dr. Axel Kalenborn (Stv. Sprecher der Fachgruppe Projektmanagement)

Prof. Dr. Oliver Linssen (Stv. Sprecher FG Vorgehensmodelle &  
Sprecher der FG IT-Projektmanagement der GPM)

Alexander Volland (Sprecher der Fachgruppe Projektmanagement)

Dr. Enes Yigitbas (Stv. Sprecher der Fachgruppe Vorgehensmodelle)

# **Buchhinweis: Veränderungen im Software Engineering durch den Einsatz von „low code“ orientierten Softwareentwicklungsansätzen**

Autoren: Andreas Schmietendorf, Michael Knuth  
Verlag: <https://www.logos-verlag.de>  
Kontakt via: <https://blog.hwr-berlin.de/schmietendorf>  
Erscheint: 2. Quartal 2024  
ISBN: 978-3-8325-5733-1

## **Abstrakt zum Buch**

Die No-Code bzw. Low-Code Softwareentwicklung versucht auf eine programmiersprachenorientierte Codierung eingesetzter Algorithmen im Diskurs benötigter Softwarelösungen weitgehend zu verzichten. Argumentiert wird dieser Ansatz zumeist durch immer schneller und preiswerter benötigte Softwarelösungen, aber auch mit unzureichend zur Verfügung stehenden Experten für die Softwareentwicklung.

Darüber hinaus soll mit Low-Code die Zusammenarbeit mit fachlichen Auftraggebern im Sinne einer agilen und domänenspezifischen Vorgehensweise unterstützt werden. Ggf. wird sogar eine ausschließlich fachlich orientierte Entwicklung verfolgt, so dass Fachabteilungen benötigte Lösungen mit Hilfe so genannter „Citizen Developers“ selbst bereitstellen können. Gerade dieser Ansatz wird häufig mit einem Kulturwandel hinsichtlich der Bereitstellung von Software in Verbindung gebracht, da fachspezifische Kenntnisse der einhergehenden Prozesse und Daten einen unmittelbaren Einfluss auf die unternehmerische Digitalisierung ausüben können.

Die Begriffe No-Code und Low-Code werden durch [Bitkom 2020] in folgender Weise definiert:

*„Als No Code (NC) bezeichnet man Plattformen, mit denen ganz ohne Programmieraufwand Applikationen erstellt werden können. Ein »No-Coder« fügt seine Anwendung per Mausklick aus vorkonfektionierten Bestandteilen zusammen, ohne dass er den Code manuell ergänzen oder anpassen kann.“*

*„Bei Low Code (LC) ist das Grundprinzip, dass in einer grafischen Entwicklungsumgebung Applikationen per Drag-and-Drop aus vorgefertigten Bausteinen zusammengesetzt werden. Doch der Clou bei Low Code ist, dass bestehende Bausteine mittels klassischen Codes von der Entwicklungsabteilung ergänzt und individuell angepasst werden können.“*

Im Low-Code-Manifest finden sich 14 Kernprinzipien, wobei das folgende Prinzip auf die Beziehung zwischen No-Code und Low-Code eingeht (Quelle: [LCMani 2024]):

*„No-Code verstehen wir als eine besondere Ausprägung von Low-Code, gänzlich ohne Programmierung.“*

Im Vortrag soll auf potentielle Einsatzszenarien, Auswahlkriterien für Low-Code Entwicklungsplattformen, Veränderungen im Software Engineering sowie Risiken und Grenzen dieses Ansatzes eingegangen werden. Ein Schwerpunkt bezieht sich auf den Einsatz generativer KI-Modelle (u.a. Large Language Models). Entsprechend trainierte Modelle bieten die Möglichkeit, Texte, Bilder, Videos oder eben auch Quellcode und Diagramme generativ bereitzustellen. Konkrete Ansätze finden sich als integraler Bestandteil von Low-Code-Plattformen und dienen der Entwicklungsunterstützung und Qualitätssicherung, wie z.B.:

- Pega *GenAI* - <https://www.pega.com>
- OutSystems *AI Mentor* - <https://www.outsystems.com>
- Power Apps *Copilot* - <https://www.microsoft.com>

Ein weiterer Aspekt bezieht sich auf die Integration von vortrainierten bzw. ggf. trainierbaren KI-Modellen in die eigene Applikation. Hier finden sich vielfältige Angebote für diverse KI-gestützte Funktionen, die z.B. im Diskurs von Sentiment-Analysen unter Rapid-API abrufbar sind (vgl. <https://rapidapi.com/collection/sentiment-analysis>).

Letztendlich finden sich auch Low-Code-Plattformen, die zur Erstellung von KI-Lösungen (ggf. auch wissenschaftlichen Experimenten) eingesetzt werden können. Dabei existieren sowohl cloudbasierte Lösungen, wie z.B. das Machine Learning Studio<sup>1</sup> (<https://studio.azureml.net>), bzw. lokal (On Premise) einsetzbare Low-Code-Plattformen, wie z.B. KNIME (<https://www.knime.com>).

## **Bucherstellung im Kontext generativer KI**

Im Zusammenhang mit der Verfügbarkeit einer generativen KI (u.a. Large Language Models kurz LLMs), die auf der Basis durch Menschen erstellter Inhalte (allgemein Trainingsdaten) u.a. Texte, Bilder, Videos oder auch Quellcode generativ bereitstellen kann, stellt sich grundsätzlich die Frage nach der Sinnfälligkeit ein neues Buch bereitzustellen. Das gilt umso mehr im Diskurs der Informationswissenschaften bzw. korrespondierender Ingenieurdisziplinen.

Aus Sicht der Autoren führt an der Verwendung generativer KI-Ansätze zukünftig kein Weg vorbei. Obwohl uns die vielfältigen Nachteile (u.a. ungeprüfte Inhalte, unzureichende Zitierfähigkeit, unklare Urheberrechte) durchaus bewusst sind, halten wir die sich daraus ergebenden Möglichkeiten dennoch für gravierend. Mit Hilfe der KI-basierten Auseinandersetzung konnte das Themengebiet des Buches frühzeitig strukturiert bzw. bezüglich der betrachteten Sachverhalte abgegrenzt werden. Im Zusammenhang mit der Erstellung des vorliegenden Buchs wurde das ChatGPT-System zu den folgenden für das Buch zentralen Themenkomplexen befragt:

- Wie entstand die „low code“ Idee in den letzten 30 Jahren, welche Meilensteine gab es?
- Welche Möglichkeiten und Grenzen bieten sich mit einer „low code“ basierten Softwareentwicklung?
- Für welche Anwendungsszenarien eignet sich eine „low code“ basierte Softwareentwicklung und für welche eher nicht?
- Welche Anforderungen bzw. Kriterien sollten „low code“ Entwicklungsplattformen erfüllen?
- Wie könnte eine Klassifizierung bestehender Low-Code-Plattformen aussehen?

Die bereitgestellten Antworten (liegen als Anlagen dem Buch bei) konnten für eine erste eher oberflächliche Reflektion der Themenstellung durchaus überzeugen. Dem entsprechend stellt sich die Frage nach darüber hinaus gehenden Inhalten, die dem Buch eine Berechtigung geben bzw. entsprechende Alleinstellungsmerkmale verleihen. Aus Sicht der Autoren finden sich diese mit den folgenden Sachverhalten:

- Empirisch gesicherte Bewertung der durch die KI bereitgestellten Aussagen, durch die Reflektion zitierfähiger Fachbeiträge bzw. verfügbarer Expertenaussagen. → Verifikation KI-Aussagen
- Bereitstellung von im Umgang mit „low code“ Lösungen gewonnen Erfahrungen, welche durch dokumentierte Fallstudien nachvollzogen und vor allem reproduziert werden können. → Praxis-Erfahrungen

---

<sup>1</sup> soll durch Azure Machine Learning in 2024 abgelöst werden

- Bereitstellung personalisierter Aussagen die so einen Beitrag zum Meinungsdiskurs in der einschlägigen Community leisten und so einer kritischen Diskussion unterzogen werden können. → Praxis-Erfahrungen
- Reflektion praktischer Erfahrungen aus Industrie und Forschung, so dass z.B. auch Beziehungen hinsichtlich der KI unbekannter Einflussfaktoren wiedergegeben werden können. → Praxis-Erfahrungen
- Vermeidung allgemeiner Aussagen (aktuell typisch für KI-ChatBots) durch die Belegung bzw. Referenzierung konkret eingesetzter Modelle, Methoden oder auch Architekturansätze. → Vermeidung oberflächlicher Allgemeinaussagen
- Kreative Reflektion der „low code“ Themenstellung, so dass sich einhergehende Möglichkeiten aber auch Risiken leichter auf potentiell eigene Anwendungsszenarien abbilden bzw. bewerten lassen. → Anwendungsfälle LowCode

Die hier gewählte offensive Auseinandersetzung mit KI-basierten Ergebnissen soll im Sinne von Erfolgskriterien für die Bewertung der Buchinhalte herangezogen werden. Darüber hinaus erhoffen wir uns die Etablierung eines kritischen Diskurses (webbasierte Diskussionsplattform), so dass die Buchinhalte eine sukzessive Verbesserung in nachfolgenden Auflagen erfahren können. Für die inhaltliche Bearbeitung der Kapitel wurde das ChatCPT-System durchaus zu Rate gezogen, eine unreflektierte Übernahme der generierten Inhalte erfolgte allerdings an keiner Stelle!

## Übersicht zu den Inhalten der Buchkapitel

Im ersten Kapitel erfolgt ein kurzer geschichtlicher Abriss zur Entwicklung des Low-Code Ansatzes, ebenso werden die aktuellen Treiber dieses Entwicklungstrends verdeutlicht bzw. auf potentielle Interessengemeinschaften (Communities) eingegangen. Es gilt, für die Komplexität des Low-Code-Einsatzes im Kontext des Lebenszyklus derart erstellter Softwareanwendungen zu sensibilisieren. Trotz der massiven Reduktion benötigter Quelltextfragmente und der stark visuell betriebenen Entwicklung bleiben die klassischen Aufgaben des Software Engineerings (Analyse, Entwurf, Implementierung, Test) dennoch in angepasster Form erhalten. Ob der durch die jeweils verwendete Low-Code-Plattform vordefinierten Softwarearchitekturen verändern sich insbesondere die Aufgaben des Entwurfs. Neben der Auswahl einer konkreten Plattform geht es hier jetzt um die Komposition der benötigten Komponenten und ggf. einzusetzenden Schnittstellen zu externen Funktions- und Datenservices. Diese abstrakte bzw. modellorientierte Auseinandersetzung mit den verfügbaren Möglichkeiten und technologischen Details der einzusetzenden Softwarearchitektur bildet die Grundlage sich stärker auf die fachlichen Anforderungen konzentrieren zu können.

Kapitel 2 setzt sich mit existierenden Analysen und Einschätzungen zum Thema Low-Code auseinander. Berücksichtigt werden dabei sowohl wissenschaftliche Arbeiten als auch praxisorientierte Bewertungen von unabhängigen Analysten. Darüber hinaus finden sich fundierte Arbeiten (z.B. fachlich orientierte Fallstudien), welche durch Anbieter korrespondierender Plattformen zur Verfügung gestellt werden.

Im dritten Kapitel werden Funktionen und Architekturen von Low-Code-Plattformen aus einer produktunabhängigen Perspektive erläutert. Unter anderem soll auf benötigte Basiselemente der Low-Code-Entwicklung, die Art der Zusammenarbeit zwischen Fach- und Softwareentwicklern, die Performance/Effizienz der Softwareentwicklung sowie Aspekte des Software-Engineering und Services-Managements eingegangen werden.

Das vierte Kapitel fokussiert auf den Auswahlprozess aber auch die Auswahlkriterien für eine konkrete Low-Code-Plattform. Auf der Grundlage einer Auseinandersetzung mit existierenden Arbeiten zu dieser Problemstellung, aber auch den durch die Autoren eingebrachten Erfahrungen wird die Idee einer kriterien- und risikoorientierten Nutzwertanalyse vermittelt.

Die aktuell durch den Einsatz von Low-Code-Plattformen postulierten Vorteile werden sich aus Sicht des Autors erst im Diskurs der Softwarewartung bestätigen. Bereits bei der Auswahl einer Low-Code-Plattform sollte diesem Aspekt Rechnung getragen werden. Daher gilt es das Potential des Anbieters einzuschätzen und potentielle Abhängigkeiten von ausschließlich einem Anbieter (vendor lock in) zu vermeiden.

Obwohl wir mit dem Buch keine produktspezifische Einführung in eine konkrete Low-Code-Plattform verfolgen, sollen im Kapitel 5 dennoch typische Lösungen hinsichtlich ihrer Grundzüge vorgestellt werden. Im Detail wird insbesondere auf durch die Anbieter fokussierte Anwendungstypen, Aspekte der Entwicklungs- und Laufzeitumgebung, die einhergehende Community und mögliche Architekturansätze eingegangen.

- „Joget“ als webbasierte Open-Source-Software zur Entwicklung von Workflow- und Geschäftsprozessmanagementanwendungen.
- „Budibase“ als Open-Source-Software zum Erstellen von Apps und Workflows auf der Basis vielfältiger Datenquellen.
- „OutSystems“ zur Entwicklung von Apps bzw. Portallösungen, Geschäftsprozessautomation und Komponenten zur Systemintegration.
- „Node-RED“ versteht sich als Werkzeug für „event-driven applications“ und zielt insbesondere auf den IoT-Bereich ab.
- „PowerApps“ als Datenplattform mit vielfältigen Schnittstellen und vordefinierter Vorlagen zur Erstellung von Apps.
- „Taipy“ als Open-Source-Bibliothek zur Entwicklung von Front-Ende und Back-End-Lösungen im Bereich datenintensiver Lösungen.
- „KNIME“ Integration so genannter Nodes zur Abbildung von Ansätzen des maschinellen Lernens und des Data-Minings.

Probleme und Entwicklungstendenzen sind dann Gegenstand des 6. Kapitels. Hier sollen die Möglichkeiten und Grenzen des Low-Code-Ansatzes noch einmal deutlich herausgestellt werden. Dafür soll zunächst auf KI-orientierte Aspekte eingegangen werden. Mit Hilfe der KI (insbesondere so genannte generative KI – LLMs) lassen sich vielfältige Aufgaben der Softwareentwicklung unterstützen, ebenso können zu implementierende KI-Lösungen (Prognosen, Klassifikationen, Bild- und Mustererkennung, Data Science, ...) selbst zum Gegenstand einer Low-Code-Entwicklung werden. Eingegangen wird auch auf Erfahrungen, die im Zusammenhang mit einer gescheiterten Low-Code-Einführung gewonnen wurden. Abschließend soll noch einmal auf Herausforderungen im Diskurs der Zusammenarbeit zwischen Citizen-Development und klassischen IT-Experten (Entwicklung und Betrieb) eingegangen werden.

Die vielfältig zur Erstellung des Buches verwendeten Literatur- und Internetquellen werden im abschließenden Quellenverzeichnis zusammengefasst, so dass interessierte Leser die Darstellungen bzw. Ausführungen des Buches nachvollziehen können bzw. eine ggf. benötigte Vertiefung in spezielle Sachverhalte unterstützt wird.

## **Bibliografische Angabe zum Buch**

[Bitkom 2021] ERP Trend-Check 2021 - Bitkom Whitepaper, [https://www.bitkom.org/sites/main/files/2021-04/bitkom\\_erp\\_trend-check\\_2021.pdf](https://www.bitkom.org/sites/main/files/2021-04/bitkom_erp_trend-check_2021.pdf)

[LCMani 2024] DAS LOW-CODE MANIFEST, Low-Code Association e.V., <https://www.low-codeassociation.org/wp-content/uploads/2023/02/Low-Code-Manifest.pdf>

## **Tagungsbericht zur 1. Internationalen Konferenz für Software Produktmanagement (ICSPM2023)**

Software Product Management in the Era of Data-Driven Products, Services and  
Ecosystem

3. und 4. Mai 2023, Frankfurt

von Dimitri Petrik

Im Mai 2023 fand die 1. Internationale Konferenz für Software Produktmanagement (ICSPM) in Frankfurt statt. Die Konferenz wurde von den Sprechern der GI-Fachgruppe Software-Produktmanagement (WI-PrdM) im Rahmen des Fachausschusses für Management der Anwendungsentwicklung und -wartung (WI-MAW) organisiert und zusammen mit dem International Software Product Management Association (ISPMA) Austräger. Die Konferenz verfolgte das Ziel die Lücke zwischen Forschung und Praxis, um das Software-Produktmanagement im Zeitalter datengesteuerter Produkte, Dienstleistungen und Ökosysteme zu schließen. Somit wurde Technologietransfer als charakteristisches Merkmal der Konferenz verstanden. Die ISPMA ist ein offener, gemeinnütziger Verband von Experten, Unternehmen und Forschungsinstituten, der branchenübergreifende Exzellenz im Software-Produktmanagement fördert.

Thematisch passend zur Konferenz wurden zwei Keynote-Sprecher eingeladen:

Prof. Dr. Jan Bosch (Chalmers Universität für Technologie) und Patrizia Sickinger (Mercedes-Benz Group AG). Jan Bosch ist Professor an der Chalmers Universität Technologie in Göteborg, Schweden, und Direktor des Software Center ([www.software-center.se](http://www.software-center.se)), einer strategischen, partnerfinanzierten Zusammenarbeit, die sich auf das Thema Digitalisierung zwischen 17 großen europäischen Unternehmen (einschließlich Ericsson, Volvo Cars, Volvo Trucks, Saab Defense, Scania, Siemens und Bosch) und fünf Universitäten konzentriert. Zuvor war er als Vice President of Engineering Process bei Intuit Inc tätig, wo er auch die offenen Innovationsbemühungen leitete und das zentrale mobile Technologieteam anführte. Seine Keynote behandelte die Verschiebung im Produktmanagement von "was zu bauen ist" zu "welche Ergebnisse sind zu erreichen".

Patrizia Sickinger ist Teamleiterin in der IT-Abteilung für die Implementierung von Produktportfoliomanagement, Marketing und Qualitätsmanagement für unternehmensweite Backend-Infrastrukturprodukte sowie Dienstleistungen innerhalb der Organisation "Dynamic Platforms & CDO". Sie ist seit ihrem Studium der Betriebswirtschaftslehre von 1980 bis 1983, als eine der ersten Studentinnen an der Hochschule für Wirtschaft und Gesellschaft, bei der Mercedes-Benz AG beschäftigt. Während dieser Zeit hatte sie verschiedene Positionen in den Bereichen Personalwesen, Forschung und Entwicklung, Finanzen und Controlling sowie IT inne und ihr Team war am Aufbau eines Datenökosystems für Mercedes-Benz zum Zeitpunkt

der Konferenz betraut. In ihrer Keynote wurden die Erfahrungen bei der Implementierung eines umfassenden Portfoliomanagements für klassische IT-Produkte, Dienstleistungen und Lösungen, die nicht nur den Betrieb, sondern auch das Marketing für diese Produkte überwacht sowie die Qualität der damit verbundenen Prozesse verbessert, vorgestellt. Die sechs akzeptierten Beiträge decken eine breite Varianz unterschiedlicher Aspekte des Software-Produktmanagements ab.

Von insgesamt 12 eingereichten Beiträgen wurden sechs in einem Peer-Review-Verfahren akzeptiert. Die angenommenen wissenschaftlichen Artikeln adressierten folgende Themen. Der Beitrag von Moroz und Kollegen beschreibt das Themenfeld der Produktoperationen (kurz: Produkt-OPS) im Kontext der Analyse von Grauliteratur, also den Publikationen aus der Praxis. Der Beitrag von Katharina Wagner, Felix Schönhofen und Georg Herzwurm befasst sich mit der Entwicklung und Bewertung eines Kompetenzmodells für das Produktmanagement in softwaregetriebenen Branchen und Geschäftsmodellen. Helena H. Olsson und Jan Bosch thematisieren die Herausforderungen, mit denen Unternehmen konfrontiert sind, wenn sie neue, datengetriebene Geschäftsmodelle oder Dienstleistungen einführen, bestehende Umsatzmodelle auf "as a Service"-Modelle umstellen und etablierte Praktiken im Bereich der Datennutzung überdenken. Yevgen Bogodistov, Antonia Schwaiger und Daniel Beimborn nutzten ein diskretes Auswahl-Experiment, um die Unterschiede in den Erwartungen des Top-Managements, des mittleren Managements und der operativen Ebene bezüglich Digitalisierungsinitiativen zu analysieren und leiteten daraus Schlüsselfaktoren für eine erfolgreiche Implementierung ab. Basierend auf 16 Experteninterviews mit Produktverantwortlichen untersuchen Timo Toikkanen und Kollegen die Frage, welche Ähnlichkeiten und Unterschiede zwischen der Rolle des Produktverantwortlichen im agilen Scrum-Ansatz und der Rolle des Software-Produktmanagers bestehen. Im letzten Artikel präsentieren Marcus Pietzker und seine Mitautoren ein Rahmenwerk zur Entwicklung von Geschäftsmodellen für Produktplattformen für Unternehmenssoftware.

Im Anschluss an die Konferenz konnten mit Hilfe der GI die eingereichten Beiträge in einem wissenschaftlichen Tagungsband publiziert werden. Der Tagungsband der 1st International Conference on Software Product Management (ICSPM) ist bei der digitalen Bibliothek der Gesellschaft für Informatik e.V. erschienen und ist unter folgendem Link erreichbar:

<https://dl.gi.de/handle/20.500.12116/41773>.

Der Beitrag von Helena H. Olsson und Jan Bosch sowie der Beitrag von Timo Toikkanen et. al aus dem Tagungsband wurden zusätzlich als herausragende Beiträge auch als Bestandteil dieses Rundbriefs ausgewählt.

# Bericht zum ESAPI 2023 Workshop KI-Szenarien im Zeitalter von ChatGPT & Co (ESAPI 2023)

Öffentlicher Expertenworkshop, inspiriert von den Themen des IFAF-Forschungsprojekts TAHAI (TrustAdHocAI – Details siehe QR-Code)



21. November 2023 Fraunhofer IESE Kaiserslautern



Eröffnung des hybrid gestalteten Workshops durch:

Dr. Andreas Jedlitschka  
(Fraunhofer IESE bzw. Sprecher GI FG Data Science)  
Prof. Dr. Andreas Schmietendorf  
(HWR Berlin/Uni Magdeburg bzw. ProjL TAHAI)

## Ziele des Workshops

Im Mittelpunkt des Workshops stand die Idee, ein Szenario der künstlichen Intelligenz aus verschiedenen Perspektiven zu diskutieren. Konkret galt das Interesse der Frage, inwieweit als Daten zur Verfügung stehende Gesprächsverläufe (Transskripte) mittels generativer Sprachmodelle untersucht werden können.



## Diskussionsbeiträge

Prof. Dr. Claudia Nass  
(*HS Mainz/Fraunhofer IESE*)

Möglichkeiten des Design Thinking im Diskurs von KI-Szenarien

- Bedürfnisse der Menschen beim KI-Design in den Mittelpunkt stellen.
- Sinnvoller Einsatz von Design Frameworks (Design Thinking, User Centered-, Human Centered-Design)
- Ethische Aspekte berücksichtigen, was sollten wir mit dieser Technologie machen, Werte und Akzeptanz

Prof. Dr. Erik Rodner  
(*HTW Berlin*)

KI-Robustheit evaluieren sowie Stärken und Schwächen aufzeigen (Online-Beitrag)

- Vertrauen in KI-Lösungen (Sicherheit, Robustheit, Reputation, ...) kann mit Hilfe der Evaluierung bzw. Benchmarking eingesetzter KI-Modelle erreicht werden.
- Sprachmodelle implizieren typischerweise einen systematischen BIAS (z.B. Verzerrungen durch geschlechterspezifische oder auch ortsspezifische Besonderheiten).

Dr. Peter Münte  
(*Universität Innsbruck*)

Fachliche Daten transkribierter Gesprächsverläufe im Mediationskontext

- Reale Mediationsverfahren mit ggf. mehreren Sitzungen implizieren sehr umfangreiche Transkripte (typ. 1000 Seiten) – eine manuelle Auswertung ist eher für Mikroanalysen geeignet.
- KI könnte einen Beitrag leisten, um Emotionen, Kompromissfähigkeiten oder auch mediatorische Haltungen direkt oder indirekt zu erfassen.

Sandro Hartenstein  
(*HWR Berlin*)

Prototypische Analyse von KI-APIs für LLMs (Cloud vs. OnPremise)

- Überblick zu möglichen NLP-Ansätzen und Einordnung von LLMs
- Einsatz verschiedener LLMs (GPT 3.5, GPT 4 und LLAMA 2) zur Analyse exemplarischer Transskripte.
- Die Interpretation der Analyseergebnisse muss in Zusammenarbeit mit den fachlichen Experten erfolgen (es gilt Hypothesen zu bestätigen bzw. zu widerlegen).

Walter Letzel  
(*TU Berlin*)

Konzeptioneller Ansatz zur Analyse und Bewertung von Mediationssitzungen (Online-Beitrag)

- Erfassen von quantitativen (Häufigkeiten – z.B. Gesprächsanteile) und qualitativen Merkmalen (Mediationsphasen, ...)
- Einsatz von strukturprüfenden (z.B. Intervention durch Doppeln) und strukturendeckenden Verfahren (z.B. Stimmungen über Sentimentanalysen erfassen)

Prof. Dr. Ralf Schnieders  
(*HTW Berlin*)

Compliance-Aspekte im Diskurs der Datenüberlassung - Datenschutz, Urheberrecht und Haftung

- Der Umgang mit personenbezogenen Daten im Forschungsdiskurs stellt eine ressourcenmäßig zumeist nicht abbildbare Herausforderung dar.
- Entsprechende Daten sollten für die Verarbeitung dringend einer Anonymisierung unterzogen werden.

Prof. Dr. Jens Heidrich  
(HS Mainz/Fraunhofer IESE)

Auswertung und Abschlussdiskussion

- Möglichkeiten und Grenzen von KI-Sprachmodellen (speziell "Large Language Models").
- Gefahr des Übergehens zum Teil unbekannter Stakeholder.
- Bedarfs interdisziplinär zusammengesetzter KI-Teams.
- Konsequenzen des LLM-Einsatzes lassen aktuell schwer abschätzen.

## Fazit und Ausblick

Auf der Grundlage der vorgenannten Impulsvorträge entstand eine lebhafte und vor allem interdisziplinär geprägte Diskussion zu den Möglichkeiten und Grenzen von KI-Sprachmodellen (speziell "Large Language Models"). Zur Geltung kamen dabei insbesondere die Sichtweisen der Informatik, der Soziologie, der Psychologie aber auch der Rechtswissenschaften.

Aus fachlicher Sicht (Fokus im Projekt TAHAI) galt das Interesse insbesondere der KI-basierten Verarbeitung anonymisierter Transskripte, alternativ wäre auch die direkte und abermals KI-basierte Verarbeitung mitgezeichneter Audiosequenzen denkbar. Die manuell erstellten Transskripte wurden in Anlehnung an den GAT 2 Standard (Gesprächsanalytisches Transkriptionssystem) erstellt.

Als besondere Herausforderung stellte sich dabei die Aufgabe der Anonymisierung heraus. Insbesondere bei umfangreichen Transskripten erscheint eine manuell durchgeführte Anonymisierung, ob des damit einhergehenden Aufwands nur noch bedingt zielführend. Dem entsprechend wäre hier ebenfalls eine mit Hilfe von Algorithmen und Programmen bzw. KI gesteuerte Anonymisierung sinnvoll. Dabei sollten sowohl die rollenbasierte Identifikation der Gesprächspartner als auch ggf. innerhalb des Textes referenzierte Personen berücksichtigt werden. Aus Sicht des Autors bleibt dabei allerdings ein Restrisiko, d.h. eine Algorithmen gesteuerte Anonymisierung kann ggf. nur mit einer Wahrscheinlichkeit erfüllt werden. Im Zusammenhang mit den Herausforderungen der Compliance (z.B. EU DSGVO bzw. EU AI Act) entstehen für den wissenschaftlichen Bereich dabei immense und ggf. auch unkalkulierbare Herausforderungen.

Konkrete Tests bezogen sich auf Quantifizierungs- und Klassifizierungsaspekte aber auch Sentimentanalysen von Transskripten. Dafür wurden die KI-Modelle ChatGPT 3.5, ChatGPT 4 sowie LLAMA2 prototypisch zum Einsatz gebracht. Im Zusammenhang mit der Bewertung erzielter Ergebnisse zeigten sich massive Herausforderungen hinsichtlich der fachspezifischen Ergebnisinterpretation bzw. -validierung aber auch der Erklärbarkeit der eingesetzten Large Language Models. Gerade Fragen der Nachvollziehbarkeit eingesetzter KI-Modelle sind für einen vertrauenswürdigen Einsatz essentiell. Weitere Untersuchungen sollen sich auf ein mediationsspezifisches Training der LLM-Ansätze, aber auch die Erklärbarkeit (häufig auch als Explainable Artificial Intelligence – kurz XAI) entsprechender Lösungen fokussieren.

## Dank

Ein solcher Workshop wäre ohne professionelle Referenten nicht möglich, denen gilt ein herzliches Dankeschön! Ein besonderer Dank gilt den Gastgebern des Workshops vom Fraunhofer

IESE (Herrn Dr. Andreas Jedlitschka und Herrn Prof. Dr. Jens Heidrich)! Ein Dank geht ebenfalls an die Unterstützer der Veranstaltung, d.h. die Gesellschaft für Informatik (GI e.V.) sowie die Central Europe Computer Measurement Group (ceCMG e.V.).

# **Der Fachausschuss und die Fachgruppen WI-VM, WI-PM, WI-PrdM stellen sich vor**

## **Fachausschuss WI-MAW:**

### ***Management der Anwendungsentwicklung und -wartung***

Anwendungssysteme sind aus Sicht der Wirtschaftsinformatik Aufgabenträger im Rahmen der Erfüllung der betrieblichen Gesamtaufgabe. Ihre Aufgabenstellungen werden aus den Unternehmenszielen und den strategischen Zielen der Informationsverarbeitung abgeleitet. Die Entwicklung von Anwendungssystemen erfolgt nicht "kontextfrei", sondern i.A. in einem bestimmten betrieblichen Umfeld. Dies bedeutet zum einen, dass sich das einzelne Anwendungssystem in bereichsübergreifende bzw. unternehmensweite Daten- und Funktionsmodelle oder Objektmodelle einordnen muss. Zum anderen existieren häufig bereits Anwendungen für andere betriebliche (Teil-)Aufgaben, mit denen das System zusammenarbeiten muss.

Der Fachausschuss beschäftigt sich aus dieser Sicht mit der Planung, der Entwicklung, der Einführung, dem Einsatz und der Wartung betrieblicher Anwendungssysteme. Im Vordergrund stehen Vorgehensweisen, Prinzipien und Methoden für die Anwendungsentwicklung im betrieblichen Umfeld sowie ihre Unterstützung durch Softwarewerkzeuge. Im Einzelnen setzt sich der Fachausschuss mit Themen wie den folgenden auseinander:

- Integration von Anwendungssystemen in eine existierende betriebliche DV-Landschaft;
- Sicherung der Investitionen in das Wirtschaftsgut Software; Bewertung von Vorgehensmodellen, Methoden und Werkzeugen zur Anwendungsentwicklung sowie Einsatzerfahrungen;
- Management von Softwareentwicklungsprojekten (Projektplanung, -durchführung und -kontrolle, Projektorganisation, Projektmanagementsysteme, Kosten/ Wirtschaftlichkeit),
- Software Produktmanagement, Configuration Management, Change Management, Migration Management, Reengineering.

Mitgliederzahl: ca. 500

#### **FA-Sprecher**

Prof. Dr. G. Herzwurm  
Universität Stuttgart  
Lehrstuhl für Allgemeine  
Betriebswirtschaftslehre und  
Wirtschaftsinformatik II  
(Unternehmenssoftware)

#### **stellv. FA-Sprecherin**

Dr.-Ing. Birgit Demuth  
Technische Universität Dresden  
Institut für Software- und  
Multimediatechnik

## **Fachgruppe WI-VM:**

### ***Vorgehensmodelle für die betriebliche Anwendungsentwicklung***

Betrachtungsgegenstand der Fachgruppe sind die als "Vorgehensmodelle" bezeichneten Beschreibungen der Aufbau- und Ablauforganisation von Projekten zur Entwicklung und Wartung von Anwendungssystemen. Solche Beschreibungen helfen, die Durchführung von Projekten innerhalb eines Unternehmens oder darüber hinaus zu standardisieren und zu verbessern. Der Begriff Anwendungssystem sei hier sehr weit gefasst: von technischen über betriebswirtschaftliche bis zu organisatorischen Systemen.

Um eine effektive und effiziente Gestaltung der Vorgehensmodelle und damit der Projekte zu erreichen, ist die Berücksichtigung der Schnittstellen zur Betriebswirtschaftslehre einerseits, insbesondere der Organisations- und der Managementlehre, und dem Software Engineering andererseits wesentlich.

Das Thema "Vorgehensmodelle" wird daher von der Fachgruppe aus verschiedenen Blickrichtungen betrachtet:

- Grundlagen: Begriffsdefinitionen, Bestandteile, (formale) Beschreibung von Vorgehensmodellen, Vorgehensmodell-Typen.
- Inhaltliche Bausteine: Konzepte, Methoden, Phasen, Projektmanagement, Qualitätssicherung.
- Werkzeugunterstützung: Vorgehensmodell-Driver, Meta-Modelle, Data-Dictionaries.
- Ökonomische, soziale und psychologische Aspekte: Einführung und Betrieb von Vorgehensmodellen, organisatorisches Umfeld.
- Beispiele aus der Praxis: Standard-Vorgehensmodelle in Organisationen, Branchen und für Anwendungstypen, spezielle Vorgehensmodelle von Unternehmen.
- Standardisierung von Vorgehensmodellen: V-Modell XT, Hermes

Die Fachgruppe fördert einen intensiven Gedankenaustausch durch die Pflege persönlicher Kontakte und unterstützt einen offenen und kritischen Dialog zwischen Wissenschaft und Praxis. Ein weiteres Ziel der Fachgruppe ist die Erarbeitung von Empfehlungen und Stellungnahmen zu den technischen, wirtschaftlichen, organisatorischen und sozialen Aspekten bei Auswahl und Einsatz von Vorgehensmodellen - dies insbesondere vor dem Hintergrund nationaler, europäischer und internationaler Normungs- und Standardisierungsbestrebungen. Weitere Informationen über Vorgehensmodelle und die Arbeit der Fachgruppe sind im Internet zu finden unter [www.vorgehensmodelle.de](http://www.vorgehensmodelle.de).

#### **FG-Sprecher**

Dr. rer. nat. Masud Fazal-Baqaie  
Next Data Service AG  
Berlin

#### **stellv. FG-Sprecher**

Dr. rer. nat. Enes Yigitbas  
Universität Paderborn  
Fachgruppe Datenbanken- und  
Informationssysteme

## **Fachgruppe WI-PM: *Projektmanagement***

Die Fachgruppe befasst sich mit dem Einsatz, der Verbreitung sowie der Weiterentwicklung des Projektmanagements. Neben Vertretern aus den Hochschulen sollen vor allem Praktiker die Arbeitsschwerpunkte der Fachgruppe definieren, Ergebnisse erarbeiten und Erfahrungen weitergeben. Für die Aufgabengebiete des Projektmanagements sollen Methoden, Werkzeuge und Techniken untersucht werden. Neben den klassischen Aufgabengebieten wie beispielsweise Projektorganisation, Aufwandschätzung, Projektverfolgung und Projektsteuerung stehen folgende Themen im Vordergrund:

### **Bedeutung und Dimensionierung des Projektmanagements.**

Die Bedeutung des DV-Projektmanagements als entscheidender Faktor für den Erfolg oder das Mißlingen von DV-Projekten wird von vielen Entscheidungsträgern unterschätzt. Daher sollte die grundsätzliche Bedeutung sowie der Nutzen einer angemessenen Ausstattung des Projektmanagements mit eigenen Ressourcen transparent gemacht werden.

### **Human Factors.**

In zahlreichen Projekten liegen die größten Projektrisiken bei den sogenannten Human Factors (oder "weichen" Faktoren). Der Umgang mit solchen Risiken erfordert Kompetenz bei Themen wie Motivation, Führung, Teamfähigkeit, Überwindung "politischer" Widerstände u.a.m.

### **Programm Management.**

Immer öfter gefordert wird das Management eines Portfolios von Projekten, wobei nicht alle Projekte des Portfolios eigentliche DV-Projekte zu sein brauchen. Solche Projektportfolios können beispielweise als Folge einer veränderten Unternehmensstrategie entstehen und sollen dann einen größeren Veränderungsprozess bewirken. Hauptaufgabe eines Programme Managements ist dabei die zielorientierte Steuerung der Abarbeitung des Projektportfolios, wobei insbesondere unternehmerische Gesichtspunkte zu beachten sind.

### **FG-Sprecher**

Alexander Volland  
Union IT-Services GmbH  
Frankfurt am Main

### **stellv. FG-Sprecher**

Prof. Dr. Axel Kalenborn  
Universität Trier

## **Fachgruppe WI-PrdM: *Software Produktmanagement***

Effizientes und effektives Management softwareintensiver Produkte ist zu einer kritischen Kernkompetenz von Unternehmen geworden. Unternehmen sind mit einer stetig wachsenden Anzahl von Herausforderungen konfrontiert, die durch unterschiedliche Lebenszyklen von Systemen und unterschiedliche Kritikalität im Systemeinsatz in immer mehr – und neuen – Anwendungsfeldern entstehen. Hybride Systeme, z.B. im Internet-of-Things, in Automobilen, Flugzeugen, Drohnen, medizinischen Geräten oder in der Unterhaltungselektronik geben Software eine nie dagewesene Bedeutung. Zusätzlich entstehen durch die vielfältigen Initiativen im Rahmen Digitalisierung neue Arbeits- und Geschäftsmodelle und eröffnen vollkommen neue, durch Software getriebene Möglichkeiten zur Innovation.

In diesem dynamischen Umfeld findet softwaregetriebene Innovation an der Schnittstelle zwischen Informatik/Software Engineering und Wirtschaft statt, zwischen Forschung und industrieller Praxis. Das Produktmanagement umfasst hierbei die Entwicklung, Wartung und Evolution klassischer Softwarelösungen im gesamten Produktlebenszyklus, aber insbesondere auch innovative softwarebasierte Innovation. Die Fachgruppe befasst sich einerseits mit Konzepten, Methoden und Werkzeugen der Informatik/Wirtschaftsinformatik zur Gestaltung des Produktmanagements und der Produktinnovation. Andererseits wird insbesondere auch ein starker Fokus auf die praktische Anwendbarkeit theoretischer Konzepte gelegt.

Die Fachgruppe fördert auf dem genannten Gebiet den intensiven Gedankenaustausch, die Pflege persönlicher Kontakte und die Zusammenarbeit interessierter Personen und Gruppen. Dazu zählt u.a. die gegenseitige Information über Veranstaltungen, Projekte und Veröffentlichungen.

### **FG-Sprecher**

Prof. Dr. rer. pol. Andreas Helferich  
ISM International School of Management  
Campus Stuttgart

### **stellv. FG-Sprecher**

Dr. Dimitri Petrik  
Universität Stuttgart  
Abt. VIII: Wirtschaftsinformatik II  
(Software-intensive Business)

## **Mitglieder des Fachausschusses Management der Anwendungsentwicklung- und wartung (GI-MAW)**

Die Mitglieder des Leitungsgremiums des Fachausschusses finden Sie unter:

<https://fa-wi-maw.gi.de/fachausschuss/leitungsgremium>





# Impressum

Der Rundbrief des Fachausschusses *Management der Anwendungsentwicklung und -wartung (WI-MAW)* ist das Publikationsorgan des Fachausschusses sowie der Fachgruppen

WI-VM *Vorgehensmodelle für die betriebliche Anwendungsentwicklung*

WI-PM *Projektmanagement*

WI-PrdM *Software Produktmanagement*

Der Rundbrief erscheint einmal jährlich elektronisch. Durch den Rundbrief sollen wichtige Erfahrungen, neue Erkenntnisse und aktuelle Informationen unter den Mitgliedern ausgetauscht werden. Rundbriefbeiträge von Mitgliedern und Interessenten sind daher besonders willkommen. Es können Beiträge zu folgenden Rubriken eingereicht werden:

- Fachbeiträge: *Erfahrungsberichte; Theoretische Beiträge; Projektberichte (auch über laufende Projekte)*
- Informationen: *Buchbesprechungen; Tagungsberichte; Vorstellung von Arbeitsgruppen;*
- Leserbriefe: *Veranstaltungen; Call for Papers; Einladungen; Programme*

Es wird gebeten, Beiträge in elektronischer Form (Word) an die Rundbriefredaktion zu senden. Ein Ausdruck sollte keine Seitennummerierung enthalten.

Die Beiträge können in deutscher oder englischer Sprache abgefasst sein. Mit der Zusendung eines Beitrags ist das Einverständnis zur Veröffentlichung im Rundbrief verbunden. Jeder Beitrag wird ohne Begutachtung veröffentlicht.

<b>Herausgeber</b>	Fachausschuss <i>Management der Anwendungsentwicklung und -wartung</i>	
<b>Auflage</b>	500	
<b>Redaktion</b>	Christian Kop Institut für Artificial Intelligence and Cyber Security Universität Klagenfurt A-9020 Klagenfurt	E-mail: christian.kop@aau.at Tel.: +43 463 2700 3735 Fax: +43 463 2700 993735

Redaktionsschluß für das nächste Heft: 31.01.2025