

RESEARCH

Open Access



LLM support for enterprise modelling based on integrating prompt and enterprise modelling meta-models

Kurt Sandkuhl^{1*} , Balbir S. Barn² and Souvik Barat³

*Correspondence:

Kurt Sandkuhl

kurt.sandkuhl@uni-rostock.de

¹Institute of Computer Science,
Rostock University, Albert-Einstein-
Str. 22, 18059 Rostock, Mecklenburg
Western-Pomerania, Germany

²Faculty of Science and Technology,
Middlesex University, The
Burroughs, London,
England NW4 4BT, UK

³Tata Consultancy Services
Research, Pune, Maharashtra, India

Abstract

Large language models (LLMs) are considered by many researchers to be a promising technology for automating routine tasks in conceptual modeling, such as in enterprise modeling. Enterprise Modelling (EM) is a significant method-based endeavour that is often seen as a prerequisite to broader organisational programmes of work, such as digital transformation. Examples of tasks in EM that could be supported by LLMs include generating models from natural language descriptions provided by domain experts and transforming models into text to support non-modeling experts in applying models. For such tasks, it would be beneficial to ensure consistent terminology use across enterprise models, LLM prompts, and natural language descriptions. The research question addressed in this paper is: In the context of enterprise modelling, what is required to conceptually integrate prompt and enterprise modelling meta-models? From this exploration, the main contributions are (1) a novel and first such example of a conceptual meta-model for prompt engineering that integrates the concepts of the modelling domain under consideration with concepts from the modelling language applied, and the input and output of prompts, and (2) a demonstration of the applicability of the meta-model for different EM languages. The application of this meta-model, created using design science principles, is examined through an evaluation strategy comprising three evaluation episodes: internal to development team, external in the validation context and external in the application context. The proposed meta-model is potentially significant as it paves a route for the design for prompt modelling tools in an enterprise modelling context.

Keywords Enterprise modelling, Large language model, Modelling method, ChatGPT, Prompt meta-model

1 Introduction

Language technologies, large language models (LLM) and neural text generators, such as OpenAI's ChatGPT or Google's Gemini, are considered by many business analysts and researchers as promising technology for automating routine tasks, leading to substantial improvement of productivity (see, e.g., [1] and [2]). Initial results from applying LLM



© The Author(s) 2026. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

in engineering disciplines also indicate the potential of this technology for supporting modelling processes. Examples include the use of ChatGPT in software engineering for UML modelling [3] and in design and manufacturing [4].

Enterprise Modelling is a significant method-based endeavour and is closely aligned and often seen as a prerequisite to broader organisational change programmes such as digital transformation [5]. Thus, Enterprise Modelling is a significant domain within which to explore the use of LLMs.

Our own previous work in this area [6] indicates that an LLM can be used to support Enterprise Modelling by substituting the domain expert in some of the preparatory tasks for early modelling phases or in the development of models for the general situation in an application domain.

If LLMs could generate enterprise models from existing textual descriptions of, for example, the current processes in an enterprise, this would save time, provided that a domain expert is available to evaluate the results of the LLM's output.

Encouraged by these results, this paper aims at a more detailed investigation of the potential of LLMs in Enterprise Modelling (EM) by taking the perspective of EM methods. We are motivated by the current evidence indicating that currently, prompt design is an artisanal activity, and there is a lack of a standardised meta-model or structured language for managing prompts, along with methods for prompt design. What is needed is an integrated meta-model for both tools and methods for prompt design. A further challenge noted by [7] is the platform-specific nature of prompt design. Prompts can show varying performance outcomes depending on a particular system. We anticipate that this concern will also be manifested through different EM methodologies. Addressing this gap would allow us to move from craft to engineering.

More concretely, our key conjecture is that a meta-model for prompt engineering integrating relevant domain concepts and the modelling language's meta-model would ease the use of LLMs in EM when used in the context of EM Methods. Currently, while there are meta-models both implicit and explicit for prompt engineering, there are none that consider the integration with a method. We thus formulate our primary research question as:

In the context of enterprise modelling methods, what is required to conceptually integrate prompt and enterprise modelling meta-models?

To that end, we report on our experiments and offer two main contributions arising from our investigations: (1) a meta-model for prompt engineering that integrates the concepts of the modelling domain under consideration with the notation of the modelling language applied and the input and output of prompts, and (2) the applicability of the meta-model for different EM languages.

To date, bringing together concepts for enterprise modelling for integration with prompt design remains an ongoing research challenge. Our model is one such proposal that has been evaluated through experimental design. The model is the substantially revised version of an initial meta-modeling attempt that covered the required conceptualization only insufficiently [8].

The paper is structured as follows: Sect. 2 introduces the background for our work from large language models and enterprise modelling. Section 3 introduces the research approach applied. Section 4 analyses related work in the areas of prompt engineering

and LLM use in enterprise modelling. Section 5 presents an illustrative example for applying LLM in enterprise modelling that also forms the basis for evaluating the prompt meta-model. Section 6 focuses on the development of the EM-aware prompt engineering meta-model and includes its experimental use. Section 7 provides a detailed evaluation of the proposed meta-model through the use of multiple validation episodes. In sect. 8, validity concerns are highlighted. Section 9 presents a concluding discussion on potential avenues for further work.

2 Theoretical background

The background for work presented in this paper is briefly summarized in this section. It originates from the fields of large language models and enterprise modelling.

2.1 Large language models

The OpenAI public release of ChatGPT-3 in November 2022 represented a key inflexion point in the mainstream use of AI as an aid to solving many types of problems. It has captured the imagination of both researchers and the public at large. A feature of the GPT model is the task-agnostic pre-training [9] which ensures enough information is encoded such that customisation is possible, in-context and at run-time, to enable handling of new tasks through prompts expressed in natural language [10].

Since then, the advances and deployment of LLMs has continued at pace along with supporting technology.

2.2 Enterprise modelling and methods

Enterprise Modelling (EM) is addressing the “systematic analysis and modelling of processes, organisation structures, product structures, IT-systems or any other perspective relevant for the modelling purpose” [11]. The variety of methods, languages and tools supporting EM is visible in work on research roadmaps and future directions from the information systems community (see, e.g., [12]) and from industrial organisation (e.g., [13]).

EM is meant to support organisations in coping with a broad range of challenges, including managing organisational change in dynamic market environments, aligning of organisational goals and information systems to support these goals, as well as explicating and consolidating knowledge from various stakeholder groups thus facilitating organisational learning. The prevalent role of EM is to provide methods, tools, and practices for capturing and visualising the current (“as-is”) situation and to develop the future (“to-be”) situation.

Given the complexity of enterprises, there appears to be a consensus in the academic literature on enterprise modelling that a key feature of an enterprise model is that it encompasses various perspectives [14]. Among the most prominent ones are the workflows or processes, products and services, organizational structure, information structures, or IT components of an enterprise. Depending on the modeling method, perspectives are represented by different sub-models or within the same model, as specified in the meta-model. Examples for meta-models used for multi-perspective enterprise modeling are MEMO [15], 4EM [16], and ArchiMate [17]. Furthermore, Karagiannis et al. [18] provide a comprehensive overview of modeling methods, including their meta-models.

Pertinent to this research is the current ongoing activity examining the impact of LLM use in EM. In sect. 4.2 we illustrate this impact by providing a summary of prior work on LLM use in EM with a concrete example where we selected 4EM [19], a multi-perspective EM language used in many universities for teaching EM. 4EM distinguishes several modelling perspectives that are summarized with their focus, issues to model and main components in Table 1.

Modelling methods and their constituents have been subject of research in the field of information systems development and, in particular, in method engineering since at least two decades. Method engineering (ME) is *the engineering discipline to design, construct and adapt methods, techniques and tools for the development of information systems* [20]. The need to adapt methods and tools according to organisational needs has been addressed by Situational Method Engineering (SME) [21]. SME is an ME approach that includes designing method parts supporting the realisation of some specific IS development activity as well as tailoring them based on local situational factors (e.g. the business sector, or size of the business). Each method part is represented according to a same template and adheres to a unique meta-model. Another practicable ME approach was proposed in [22]; it defines method parts as *method components*, consisting of a procedure guiding model development, concepts relevant to the model, and a notation. This sets a high attention on the concepts and their inclusion in the overall meta-model of the method.

3 Research approach

This paper is part of a research programme focused on developing technological support for integrating large language models (LLMs) into enterprise modelling (EM) methods. The study follows the design science research (DSR) paradigm [23], which begins with identifying relevant practical problems and proposing an initial artefact concept. Subsequent phases—requirements definition, design, evaluation, demonstration, and communication—iteratively refine this artefact using various research methods. In our case, the artefact is a meta-model that integrates prompting with EM to support LLM integration. As stated in the introduction, the main research question is:

In the context of enterprise modelling, what is required to conceptually integrate prompt and enterprise modelling meta-models?

Table 1 4EM perspectives and method components

	Focus	Issues to model	Components
Goals model (GM)	Vision and strategy	What does the organization want to achieve or to avoid and why?	Goal, Problem, External Constraint, Opportunity
Actors and resources model (ARM)	Organizational structure	Who is responsible for goals and processes? How are the actors interrelated?	Organization Unit, Actor, Role, Individual
Business process model (BPM)	Business operations	What are the business processes? How do they handle information and material?	Process, External Process, Information Set, Material Set
Technical components and requirements model (TRM)	Information system needs	What are the business requirements for the IS? How are they related to other models?	IS Component, IS Goal, IS Problem, IS Requirement
Products and services model (PM)	Products and services	What product and service components exist? What are the core features?	Product, Service, Component, Feature
Concepts model (CM)	Business ontology	What are the things and “phenomena” addressed in other sub-models?	Concept, Attribute

Relevance investigation in this paper includes two aspects: the analysis of related work from the scientific body of knowledge to confirm the lack of work on LLM integration into EM (Sect. 4) and an illustrative example for the use of LLM in EM (Sect. 5). This example helps to motivate the research on LLM integration into EM and why a prompt meta-model is required. The analysis of related work includes a systematic literature analysis [24] and the illustrative example, a quasi-experiment [25]. A purposive literature review focused on LLM prompts was used to develop the conceptual model described in Sect. 4.1. Concepts and their links to existing literature are described there. Additionally, we draw upon comparable prior work on component based development where similar challenges such as composability, re-use, libraries were at forefront of efforts to reduce costs in software development [26, 27]. *Requirements definition* started from the observations in the illustrative example and the method conceptualisation of 4EM. We performed an initial analysis of the relation between prompts, LLM output and EM methods to help structure and plan the research process.

Our experiments and the general use of LLMs indicate that the underlying and fundamental components of a method, such as the perspective of a problem domain under consideration, concepts and techniques remain intact. For us, the primary purpose of a LLM is a tool or support aid in the use of an enterprise modelling method. In such usage, the LLM may necessitate finer adjustments to methods; these changes primarily accommodate specific tasks or procedures, as opposed to inventing a new method, introducing new concepts in the meta-model, or changing the notation. However, concepts and notation are both expected to be important inputs to LLM use as they indicate the subject of LLM prompts and how the output should be structured, and thus are critical to maintaining consistency of outputs from the LLM. We recognise that there is potential to investigate whether integrating LLMs requires the addition of new method components, which may affect the overarching methodological framework underpinning methods and techniques.

As a result of the initial analysis, we identified two main perspectives for the artefact design: the organisational perspective focusing on procedures, roles and cooperation, and the technical perspective focusing on linking concepts and notations to LLM. While the organisational aspect has been the subject of previous work (see [28]), the technical perspective is the subject of this work. We decided not to stay on a purely conceptual level but to use the adaptation of the 4EM method as a use case and an initial validation for discussing LLM-oriented adaptations to EM methods. This results in a combination of deductive and inductive research. We start by investigating existing theories and approaches relevant to our research question; based on the identified theories, we design extensions or adaptations, apply our results to 4EM, and discuss implications for the general field of EM.

More concretely, the *Design and Evaluate* phase was conducted for the perspectives defined above: The technical perspective tackled in 6 focuses on linking the concepts and notations of method components to prompts for large language models. We propose a meta-model for prompt engineering for this purpose. The integrated meta-model is validated through three episodes focused on internal and external validation and are described in detail in Sect. 7.2.

4 Related work

4.1 Prompt engineering

Prompt engineering, as a term, originated from an online post about GPT3 and creative writing by Gwern Branwen, who suggested anthropomorphising LLMs through prompting the model to elicit required knowledge.¹ Prompting in this way could be seen as form of prompt programming where the “prompt is now a “program” which programs GPT-3 to do new things.”

Prompting is the practice of representing a task as a natural language utterance in order to query a language model for a response and is now a critical component of study for LLMs [29].

The fluid response of LLMs to prompts given to the system means that prompt-based prototyping allows non-Machine Learning (ML) experts to prototype ML functionality at lower cost and without the need to train models up front. Effectively, augmenting input with answered prompts becomes in-context learning. Against that, however, is a lack of clear guidance, techniques, and supporting tools for prompt design. Research literature in the text-to-image generation field also indicates that prompt engineering is a non-intuitive skill that is learned from extensive experimentation and trial and error [30].

The core of this paper is the exploration of the use of LLMs and their effectiveness as “modelling assistants”. Our experience to date and our earlier work reported in 2023 [31] indicate that the conversational style of using prompts to construct models is promising. However, the conversational style of using prompts to interrogate an LLM is lacking guidelines, strategies, and an engineering approach. Fill et al. [32] report similar concerns, presenting informal guided prompts for simple experiments in entity modelling, business process design, and other conceptual modelling methods. They highlight the need for domain-specific languages to support few-shot prompting in complex, formal modelling frameworks—a focus also explored in this section.

A recent paper by Camara et al. [3] reports on experiments in evaluating the use of LLMs for software modelling (UML and OCL constraints). Their findings indicate that there is success in producing syntactically correct UML models but semantic errors are frequent.

Of note, is their observation on the role of model fragments that address a particular concern or implement a specific feature. These fragments could then be reused and modelling assistants could support that process. Modelling based on reuse will invariably involve methods or techniques that are technology dependent. Camara et al. do not address method awareness in their contribution. Our meta-model specifically addresses both these concerns.

The literature on prompt engineering highlights the following observations: Firstly, there is a lack of language/meta-model support underpinning prompt design. For many domains such as law [33], healthcare [34], generative art [30], and systems design (such as business process modelling [35] and OCL specification [36] the focus is on using examples derived from the prompt template approach advocated by Liu et al [29]. Formal descriptions such as a meta-model or a modelling language are largely absent. One exception to this are the results reported by Clariso and Cabot [7]. In their paper, they

¹ <https://gwern.net/gpt-3>.

present a meta-model of domain-specific language called *Impromptu* that addresses concerns around the platform-specific issues of prompt design. Prompts that work well for one platform may under-perform on another. *Impromptu* is potentially significant in that it supports platform independent specification of modular prompts that are re-usable. Hence there are overlaps with the meta-model presented in this paper.

A key difference is that our prompt meta-model is oriented towards enterprise modelling and related methods.

Secondly, related to the idea of an under-pinning meta-model, is the use of software tools for designing and managing the use of prompts. Fiannaca et al. identify the concern that prompts are used to solve complex problems but lack the strict grammar of traditional programming languages. They propose methods for extracting a semantically meaningful structure of natural language prompts but stop short of producing a description of that structure [37]. Others have proposed the need to have prompt catalogues that contain specifications of prompts that can be re-used in different contexts [38]. Again, there is an implicit meta-model underpinning these specifications.

The development of software tools to support visual design of prompts has been the focus of much activity and continues to evolve at a rapid rate. At the time of preparation of this paper, the foundational bases for prompting and design tools are exemplified through the examples of software tools below.

PromptChainer [10] provides a visual programming model/metaphor to support prompt chain authoring with a focus on transforming data between chain elements and for debugging prompt chains. Rapsai [39] like PromptChainer provides a visual programming platform featuring a node-graph editor to facilitate streamlining of end-end prototyping of multimedia applications that use ML. PromptSapper [40] is an AI Chain no-code interactive development environment that utilises block-based visual programming to support AI chain (c.f. prompt chain) design, testing and deployment. PromptSource [41] is a web-based system for creating, sharing and using natural language prompts. The environment offers a templating language and interface for defining data-linked prompts. The system is not specifically designed for a LLM such as GPT3.0 and GPT 4o.

LangChain² represents a popular technology that incorporates the development of domain-specific AI agents using retrieval augmented generation (RAG) (and therefore the prompt modelling needs required to utilise those AI agents. The tool implies a meta model behind the technology. Methods for building AI agents using Langchain are emerging, mostly through online tutorials rather than through method engineering approaches. LlamaIndex like, LangChain also facilitates the creation of RAG systems. In both cases RAG enhances LLMs with domain specific and contextual knowledge. While LangChain is potentially a more versatile platform, LlamaIndex's primary use case is document workflows such as organising, querying and summarising documents.³

Hence, the third important observation is the lack of comprehensive support for methods for prompt design and usage that go beyond click tutorials widely available. The literature provides an example of the provision of cookbook-style guidelines for prompts, such as that reported in [42]. Despite the technical advances, particularly in tool development using visual modelling paradigms, the approaches taken are method-agnostic.

²<https://www.langchain.com>.

³<https://www.ibm.com/think/topics/llamaindex-vs-langchain>.

Our proposal is part of a journey to similar tool environments but with one notable difference. Future tools need to be built around existing methods and practice to benefit from expert knowledge and therefore address some of the issues around validation of outcomes from an interaction session with a LLM. In particular, we think that a method driven use of LLMs would be a promising route to addressing the semantics concerns reported by Camara et al.

In sect. 6, we present a meta-model for prompt design that is integrated with the 4EM method. Our intention is that such a model can contribute to the basis of future tools that will support prompt design for enterprise modelling using the 4EM method. With such a proposition, prompt chaining, for example, then becomes a method dependent activity.

4.2 LLM use in enterprise modelling

The state of research into LLM use in enterprise modelling was investigated by performing a systematic literature review (SLR) based on the approach proposed by Kitchenham [24] in October 2025. The sub-research question for our work to identify LLM use in EM is as follows:

RQ-SLR: What previous scientific work is visible in publications about using neural text generators and large language models in enterprise modelling?

We used Scopus, IEEE Xplore and AISEL as literature databases to reach a wide coverage of computer science and information systems-related publication sources. The query was developed starting from the RQ-SLR and includes synonyms for the key terms from the research question. The final query is:

((“neural text” OR “ChatGPT” OR “LLM” OR “large language model” OR “foundation model”) AND (“enterprise modelling” OR “conceptual modelling” OR “process modelling”))

Table 2 summarises the number of hits for the query.

The 14 hits in IEEE Xplore were also included in Scopus. The hit in AISEL was one of our own papers. In Scopus, among the 124 hits for the query were 22 conference proceedings and 24 papers applying LLMs in mathematics, chemistry, education, law, biology, and economics. These papers were not relevant. Furthermore, fourteen papers focusing on specific aspects of LLM use in process modeling, such as user interaction, user behavior, or the implementation of information security, were excluded, as well as nine papers investigating quality improvement in Business Process Model and Notation (BPMN) models, including different aspects of syntactic, semantic, or pragmatic quality. Much work published also addresses the use of LLM for conceptual models outside the scope of our work, such as UML class diagrams, meta-models, business models, domain modeling, or capability maps. We also excluded papers proposing approaches for benchmarking of LLM for process modelling or improving specific technical features of LLM. The list of excluded papers and conference proceedings is provided as supplementary

Table 2 Results of the literature analysis

	No. of hits	Relevant
Scopus	124	20
AISEL	1	1
IEEE Xplore	14	1
Total		21

material to this paper. The remaining 21 relevant papers include our own previous work (four papers; see below) and 17 papers published in 2023 or later:

Klievtsova et al. [43] focus explicitly on process modeling and take the perspective of conversational modeling, defined as ‘creating and improving process models and process descriptions based on the iterative exchange of questions/answers between domain experts and chatbots. The paper contributes practical recommendations for LLM application in process modeling that might be relevant for the business processes perspective usually included in enterprise models. The paper by Fill et al. [32] is closely related to our work: the authors present the results of a series of experiments that investigate the engineering of prompts for generating and interpreting conceptual models. ChatGPT is used to generate ER, BPMN, UML, and Heraklit models and large parts of the models are correctly generated. One of the results of the paper is that the potential of LLMs for supporting modeling tasks is substantial if a textual problem description of the domain to be modeled exists. The paper also investigates the interpretation of models. In [44], a framework is proposed that leverages LLMs for the automated generation and iterative refinement of process models. The framework takes textual descriptions as a starting point and includes a prompting strategy, the generation of Partially Ordered Workflow Language (POWL) models, model refinement, and the correction of potential errors in the model. This paper is also related to our work but focuses on process models only and uses an intermediary language to represent the modeling results. A study of typical tasks in the business process management (BPM) lifecycle, such as process identification, discovery, analysis, or implementation, and a discussion on LLM application opportunities and future research directions is done in [45]. The paper intends to explore opportunities for LLM use and propose application scenarios, i.e., it does not contain any experiments or in-depth studies. Buchmann et al. [46] collect and structure what is expected from LLMs as support for semantics-driven systems engineering. Many expectations formulated in the paper are consistent with our view about the potential of LLMs in EM, like the support of multiple perspectives in EM or the importance of semantics. However, the paper does not present solutions and takes the wider scope of systems engineering. The use of LLMs in the context of the Legal Goal-oriented Requirements Language is focused on by de Kinderen and Winter [47].

Apaydin and Zisgen [48] investigate the creation of business process models from text, but have a focus on the development of a local LLM for this purpose. Hörner [49] proposes an approach for BPMN model generation and refinement based on user inputs in natural language with interactive refinement. Bennoit et al. [50] analyze recent publications in the field to identify potential use cases for LLM use in business process modeling. The work as such is related to our work, but has a focus on extending the BPM lifecycle. Klievtsova et al. [51] investigate the perception of users as an aid for qualitative assessment of process models generated by LLMs. Neuberger et al. [52] present work on prompting strategies for extracting process information from text. Although this paper, again, has a focus on processes, the results are relevant for our work. The research concludes that the number of example texts, the specificity of definitions, and the rigour of format instructions are identified as key to improving the accuracy of extracted information. As visible in Sect. 5.2, we also use clearly defined format instructions and provide definitions of what to extract in our prompts.

Various approaches for generating process models from text were proposed, which use different strategies. Licardo et al. [53] propose a method for automated extraction of BPMN models from textual descriptions using natural language processing tools and deep learning models. This is related to our work, but they focus on business process models only and use a subset of the BPMN modeling language. Liu et al. [29] have the same intention and propose to use markups in the text. Köpke and Safan [54] focus on supporting business process modeling with an online tool that optimizes resource use, such as LLM-token consumption. Nour et al. [55] also introduce an online tool and use a hybrid approach. Furthermore, Storey et al. [56] investigate the application potential of LLM in conceptual modeling with a focus on tasks related to software engineering. They provide an overview to the state of research, which does not show any approaches addressing multi-perspective models.

The combination of LLM and software agents (including large action models and large process models) is investigated in a feasibility study by [57] in the context of conceptual modeling, who conclude that there is substantial application potential. The topic of meta-modeling in the context of LLM use in conceptual modeling is only addressed by the work of Challa et al. [58]. However, they address the topic on a generic level by taking a meta-modeling platform as an example and proposing a tool architecture for LLM integration.

One of the results of the paper is that the potential of LLM for supporting modeling tasks is substantial if a textual problem description of the domain to be modeled exists. This is consistent with our observations. Another, more important, result is that there is no work about defining and integrating EM meta-models and prompt models or meta-models.

Table 3 shows the relevant papers, as described above, including the information about meta-model (MM) related aspects in the paper, which perspective of enterprise models or conceptual models is in focus, and the primary LLM application in the paper. The table also illustrates that most published work focuses on business process models (BPM), and only our own previous work considers enterprise models. Our previous work in the field includes a feasibility study on generating enterprise models from text [31], the initial, preliminary version of the meta-model [28], a toolchain for LLM-based model generation ?, and an application example [59].

5 Illustrative example

This section presents an example illustrating LLM use in enterprise modelling (EM) with three goals: (1) to motivate the investigation of LLMs in EM and the need for a prompt meta-model, (2) to derive requirements for a EM-aware meta-model, and (3) to serve as a running example for validating the prompt meta-model in Sect. 6. Specifically, we demonstrate how an LLM like ChatGPT can assist domain experts with established processes and inspire approaches to organisational change. The example is detailed from three perspectives: the modelling task and ChatGPT output (Sect. 5.1), a quasi-experimental view (Sect. 5.2), and meta-model requirements (Sect. 5.3).

5.1 Modelling task and ChatGPT output

We asked a researcher not familiar with the prompt meta-model but with 4EM and ChatGPT to develop an organisational modelling task that can be used to illustrate

Table 3 Results of the SLR: relevant papers

Paper	Year	Meta-model	EM pers	LLM application
Large language models for conceptual modeling: Assessment and application potential [56]	2025	No MM	overview to approaches	overview to potential
LLM, LAM, LxM Agent: From Talking to Acting Machines Insights from the Perspective of Conceptual Modeling [57]	2025	No MM	overview to applications	Differences of LLM, LLM agents, Large Action Models use
Efficient LLM-Based Conversational Process Modeling [54]	2025	process meta-model	BPM	Chatbot transforms user input; algorithm to create model
A Decomposed Hybrid Approach to Business Process Modeling with LLMs [55]	2025	No MM	BPM	LLM and algorithmic approach for model generation
A Universal Prompting Strategy for Extracting Process Model Information from Natural Language Text Using LLM [52]	2025	No MM	BPM	LLM for process information extraction from text; algorithm for model generation
AutoBPMN.AI: Conversational Process Modeling and Automation [51]	2025	No MM	BPM	text-based process modeling + redesign; iterative conversations with LLM
Towards an LLM-Based Conversational Framework for Business Process Modeling [49]	2025	No MM	BPM	BPMN generation and refinement; user input in natural language
Local Large Language Models for Business Process Modeling [48]	2025	No MM	BPM	pipeline for model generation and fine-tuned local LLM
Large language models: Expectations for semantics-driven systems engineering [46]	2024	No MM	no specific focus	expectations and research roadmap for semantic-driven systems eng
Identifying Use Cases for LLM in the Business Process Management Lifecycle [50]	2024	No MM	BPM	possible use cases for LLM in BPM
An Architecture for Integrating LLMs into Metamodeling Platforms: The Example of MM-AR [58]	2024	generic level; MM platform as example	no specific focus	tool architecture for LLM integration
A Method for Extracting BPMN Models from Textual Descriptions Using Natural Language Processing [53]	2024	No MM	BPM	procedural approach (method) that uses LLM
Towards Taming LLMs with Prompt Templates for Legal GRL Modeling [47]	2024	No MM	goal-oriented requirements language	development and reuse of prompt templates
Process Modeling with Large Language Models [44]	2024	No MM	BPM	generation of executable code for process model
Conceptual Modeling and LLMs: Impressions From First Experiments With ChatGPT [32]	2023	No MM	ER, BPM, UML	feasibility study of LLM use for model generation
Conversational Process Modelling: State of the Art, Applications, and Implications in Practice [43]	2023	No MM	BPM	approach overview; performance evaluation of LLM
LLMs for business process management: Opportunities and challenges [45]	2023	No MM	BPM	LLM in the business process management lifecycle

Interpretation: No MM - MM not addressed or presented.

LLM use in EM. For the example, we assume that ABC University has been subject to external assessment, which resulted in the recommendation to improve the research performance by establishing a stronger focus on interdisciplinary research and also improve the business processes in the department of the university responsible for supporting researchers in project acquisition and grants management. Following the

recommendation, the rector's office started an organisational change process. Part of the process is a more detailed specification of the goals to reach and an analysis of the grants office's current business processes for improvement potential. In this context, a team of domain experts for interdisciplinary research and grants management are part of an EM project that develops a goal model for the university and a business process model for the current processes in the grants office.

To prepare the EM team to model the business processes and provide inspiration for potential goals supporting interdisciplinarity, the EM team decides to use ChatGPT as a resource, develops suitable prompts, and asks domain experts to check ChatGPT's output. Table 11 in the Appendix shows the prompts. Prompt Goal-1 and BPM-1 were used for the tasks. Tables 12 and 13 provide excerpts of the ChatGPT output and the domain expert evaluation results for the two tasks. The domain expert evaluation had to distinguish between accurate output of ChatGPT, not accurate output (hallucination), output that was out of scope (i.e., no hallucination but not relevant). The complete ChatGPT output is provided in the appendix.

The evaluation of both ChatGPT outputs by the domain expert shows that (a) there is no wrong information (hallucination) included, (b) most output is accurate, but sometimes other wording should be used, and (c) the output that is out of scope still belongs to university tasks but in other departments. In conclusion, the domain expert characterised the output as a useful contribution and "saving time" as long as a domain expert is available to evaluate it. This confirms the results of our previous work [31] and illustrates why LLM use can make sense in an EM project.

5.2 Modelling task as Quasi-experiment

In the previous section, we only presented one part of the evaluation of the ChatGPT output: the domain expert's viewpoint. The modelling task used in the previous section, the ChatGPT output and the task of the domain expert were part of a quasi-experiment comparing the results of domain expert and ChatGPT systematically that also included 4EM as the target EM language. We used the same experimental design described in [31], which, in a nutshell, is as follows:

1. We developed a modelling task to be performed by both a domain expert and ChatGPT. The task focuses on one of the application fields for LLM use in EM identified in previous work: modelling the current situation in an enterprise and preparing changes.
2. Both ChatGPT and the domain expert were asked to "work" on the developed task and provide the results in 4EM. The domain expert was familiar with 4EM and the prompts for ChatGPT included information about the notation of the 4EM modelling language.
3. The results of ChatGPT and the domain expert were analysed by researchers. This analysis started with a harmonisation of the terminology used (if required) and afterwards focused on identifying differences and commonalities in these results.
4. The domain expert was first asked to check the harmonisation in terminology for correctness. This step was necessary to inform the domain experts about terminology changes made and avoid misunderstandings.
5. The domain expert afterwards was asked to check the content of the ChatGPT output

- (a) For the information from ChatGPT not contained in the domain expert's result, the expert was asked to decide if this information was not accurate (h – hallucination), accurate and missing in the domain expert's result (a/a – accurate and additional) or accurate but out of scope (a/o – accurate + out of scope)
 - (b) For the information from the domain expert's result but not in the ChatGPT, the domain expert was asked to decide if the missing information has to be considered as mandatory (m) or optional (o) information.
 - (c) For the information contained in both, the domain expert's result and ChatGPT, the domain expert was asked to determine if the meaning and intention could be seen as identical (i), similar to a large extent (s) or significantly different (d).
6. The researchers participating in the experiment used the ChatGPT output and the domain expert's result and prepared a separate 4EM model for both. Here, the result could be that there are no model mistakes, missing relations, or missing concepts.

The above analysis procedure was applied to the results from domain expert and the output of ChatGPT for the two tasks presented in the previous section. In both tasks, no differences in terminology had to be corrected. For the first task (goals to reach inter-disciplinarity), we used a second prompt (Goal-2, see Table 11 in the appendix) to refine the proposed goals. Afterwards, we produced a table comparing the output of ChatGPT with the result of the domain expert. The domain expert used this table and the output of ChatGPT to perform to compare the results and evaluate ChatGPT output. The Table 4 shows that the domain expert defined more goals than ChatGPT (e.g., establish organisational support); most of them were seen as mandatory for university operations. The processes identified by both, ChatGPT and domain expert, were all seen as similar or even identical. ChatGPT produced one goal not included in the domain expert's result (facilitate student-led initiatives) that the domain expert considered as valuable addition. All in all, the positive result of the previous section was confirmed in the experiment.

However, the experiment results also show the high importance of the prompts. The Goal-2 prompt, for example, was an ad-hoc extension of prompt Goal-2 as the Goal-1 output was not sufficiently detailed. Goal-1 included the description of the 4EM notation and the actual modelling task, although the notation information preferably should be presented in a separate prompt as it could be reused for other modelling task aiming at producing a goal model. Thus, prompt engineering requires more attention, and we argue that a prompt meta-model tailored to the use of ChatGPT in EM is required.

5.3 Meta-model requirements

The illustrative example discussed in the previous two sections and, in particular, the quasi-experiment showed a number of challenges in applying LLM for enterprise modelling purposes. Most of these challenges are related to the preparation of the prompts, the interpretation and terminological harmonisation of the output, and the transfer of the output into the representation in an enterprise model. In addition to general requirements, such as making prompts reusable, adaptable to different modelling tasks, possibility to capture proven patterns, suitability for different modelling languages and for tool implementation, some requirements to the envisioned EM-aware meta-model for prompt engineering can be identified that specifically relate to enterprise modelling.

Table 4 E1: Comparison of the goals for improving interdisciplinarity of ChatGPT and domain expert

Domain expert	ChatGPT	Domain expert evaluation
Establish organisational support for interdisciplinary research and education: Reach agreement with faculties on suitable organisational structure (staff function or virtual department?) Establish organisation unit and tasks in university bylaws Select and assign responsible researcher and staff Define organisational processes	(Not included in ChatGPT output)	Mandatory (m)
Establish a culture of interdisciplinary research: Organise interdisciplinary research workshops and seminars Provide support and recommendations for interdisciplinary curriculum development Define organisational policies that motivate and reward Interdisciplinary work	Sub Goal 2: Enhance faculty and student engagement in interdisciplinary projects. SG2.1: Develop incentive programs for interdisciplinary research and teaching. SG1.3: Implement regular interdisciplinary seminars and workshops	Similar to a large extent (s)
Acquire funding for interdisciplinary research and education: Establish internal university program for funding interdisciplinary work Identify external fundgivers and programs to apply for grants Acquire sponsorships from industry and foundations Acquire financial support from the State as strategic support	Sub-Goal 3: Secure funding for interdisciplinary initiatives. SG3.1: Apply for external grants dedicated to interdisciplinary research SG3.2: Establish a university fund specifically for supporting interdisciplinary projects. SG3.3: Foster partnerships with industry for joint research projects	Identical (i), but state funding is missing
Develop interdisciplinary research: Identify research areas with potential for interdisciplinary research Establish internal funding mechanism for interdisciplinary projects	Sub-Goal 1: Increase collaboration among faculties. SG1.1: Create interdisciplinary research clusters	Similar to a large extent (s)
Develop interdisciplinary education programs: Identify education areas with potential for interdisciplinary teaching Support pilot projects and acknowledgement structures Establish an internal funding mechanism for curriculum development	Sub-Goal 1: Increase collaboration among faculties. SG1.2: Establish joint degree programs across faculties. SG2.3: Provide training and resources for interdisciplinary curriculum development	Similar to a large extent (s)
(Not included in expert's goals)	SG2.2: Facilitate student-led interdisciplinary initiatives	a/a – accurate and additional

There are terms in the prompts and the LLM output that represent certain domain concepts; sometimes different terms for the same domain concept exist. The meta-model has to capture the semantic relation between domain concepts (DC), EM meta-model (EMM) and the domain-related terminology used in prompt and LLM output (DTP): an enterprise modelling language used in an application domain for a defined purpose obviously has to be suitable to represent the DC - either because matching classes exist or they can be represented as instances. DTP are used to express DC, i.e., they also represent DC in prompts or output. For natural language prompts and LLM output, the mapping on DC and, thus, also on EMM can be eased if a mapping of DTP on DC exists, for example, if DTP is part of the domain vocabulary that includes DC, or can be defined.

Furthermore, EM methods include different modelling components (e.g., modelling the current situation or identifying change needs) and modelling perspectives (e.g., process, goal or product modelling) (see Sect. 2.2). The components and perspectives require different prompts due to their different foci, but potentially also share prompts due to the use of the same modelling language. Both have to be supported by the EM-aware meta-model. Additionally, the definition of technical characteristics (e.g., requirements for the precision of the output) to be applied by the LLM may be required, which must also be possible in the meta-model.

6 EM-aware meta-model for prompt engineering

6.1 Presentation of the meta-model

Enterprise modelling languages have generally been specified by meta-models. In the context of integrating enterprise modelling practice with the use of LLM, we believe that enterprise modelling activities should be tailored to existing methods and practices to harness the expert knowledge embedded within the methodologies. In this section, we propose a meta-model-based EM-aware approach for enterprise modelling using LLMs.

In addition to the proposed meta-model for defining a structural approach to use LLM, our approach utilises the core concept of a meta-modelling framework to interoperate with any modelling technique in a seamless manner. In the case shown, we employ the 4EM method for underlying methodological support, and leverage the approach proposed by Goldkuhl et al. [22] to integrate meta-modelling techniques, 4EM methodology and LLM-based modelling.

Figure 1 first presents the overall package structure and the main types and their associations across the packages. Detail is omitted, that is later shown in Fig. 2. At this level, this diagram can be read as follows: Method Tasks (specific activities within an EM Method) consume and produce meta models (and their elements). Method Tasks also guide a Prompt Pattern comprising a series of Prompt Specs. Prompt Specs are expressed using Terms which reference either EM Meta Model elements or Domain Elements.

Figure 2 introduces the base meta-models representing existing knowledge in terms of meta-modelling and an example of an Enterprise Modelling method such as the 4EM method. The detailed model of 2 and its derivation from scientific literature and experience is summarised in Table 5.

Package *EM METHOD* specialised to support 4EM describes the core concepts of the 4EM method.

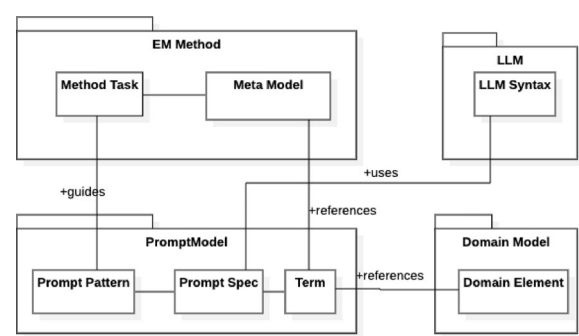


Fig. 1 Principal types, packages and associations

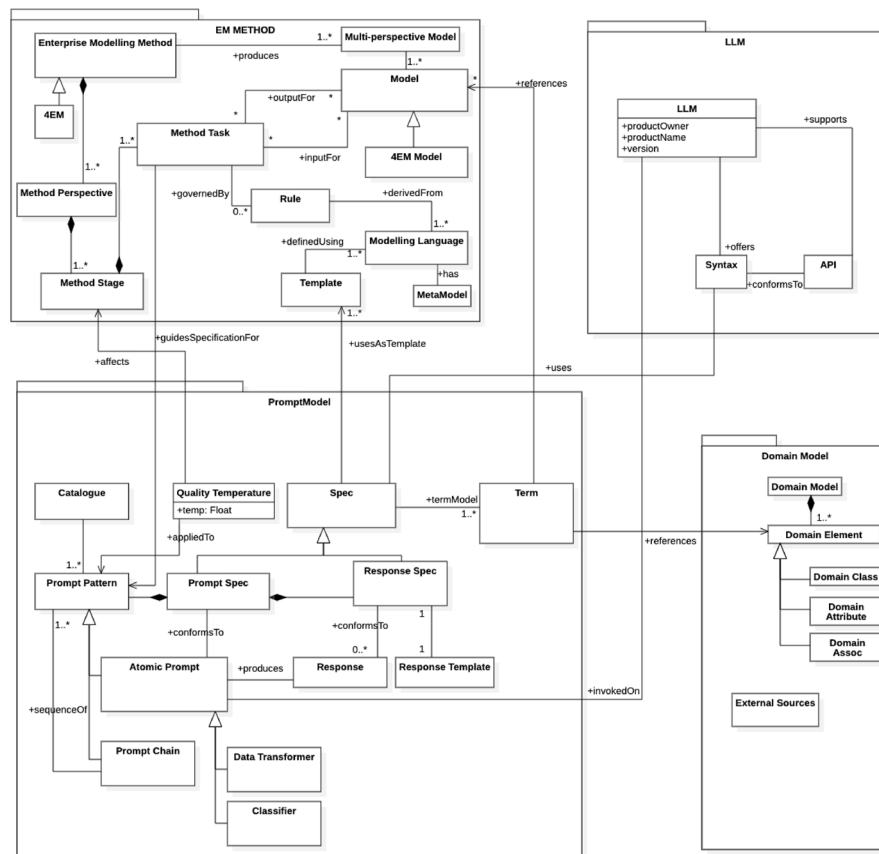


Fig. 2 Method driven prompt meta-model

In scientific literature, different ways on how to model modelling methodologies have been proposed. Examples are the ISO/EC International Standard 24744 [60], the meta-model underlying the MEMO approach [15] and the meta-models described in the context of situational method engineering [21]. Many of the approaches either focus on the modelling process, on the results of the modelling process or specific aspects of methods. In the context of our work, we decided to use the ISO 24744 standard as a fundamental technology as it covers the modelling process and the results of the process. Furthermore, as an international standard, we consider it not contradictory in the modelling community.

The ISO standard defines a modelling method (methodology) as “specification of the process to follow together with the products to be used and generated” - with a model being one of the products. The process consists of (modelling) stages that are “coarse-grained work units [...] characterised by the abstraction level at which it works and the result that it aims to produce”. Stages consist of (modelling) tasks that are “a fine-grained work unit that focuses on what must be done to achieve a given purpose”. A modelling language is a “perceivable representation of a language, in order to share it with other so they can learn it”. This is “typically expressed by intension, using grammars or meta-models.”

This is captured by *Method Task* that is *governedBy* *Rule* which are *derivedFrom* one or more *Modelling Language* that has a *MetaModel* for expressing the intention.

Table 5 Overview of prompt model concepts

Concept	Citation	Notes
Catalogue	White et al.[38], Brown and Barn [26], Bach et al. [41]	Storage and retrieval of assets has been the subject of significant research activity. The original model of storing specifications has had limited success
Prompt Spec	Our proposal	Both Prompt Spec and Response Spec are described by a collection of Terms that provide a binding between a domain concept, and meta-model concept linked to a particular method modelling language output model
Response Spec	Our proposal	The co-related expected outcome of a Prompt Spec. Both Response Spec and Prompt Spec are subtypes of Spec
Atomic Prompt	A guide to the IEF [64]	This is denoted as a specific construct where a prompt does not decompose further and is standalone or used as part of a composition. A parallel is drawn from business process engineering
Response	Numerous	The conversational nature of LLMs such as ChatGPT have at their heart, responses. We propose that The text based response is based on a template and conforming to the Response Spec
Response Template	Our proposal and work such [37]	Although we call out this concept, it is implicit in other work such [37]. The template structure for a response
Prompt Chain	Wu et al. [10], Cheng et al. [40]	A sequence of Prompt Patterns that are either Prompt Chains or Atomic Prompts
Prompt Pattern	White et al[38], Gamma et al. [62]	Prompt Patterns can document successful approaches for systematically engineering different output and interaction goals when working with conversational LLMs. These are equivalent to the notion of Design Patterns for Software Engineering
Term	Our proposal	Prompting approaches specifically utilise the notion of terms as part of their structure. We formalise the concept and ensure that a Term references either a domain element (e.g. "Library"), a meta-model element (e.g. MMClass) and a Model produced by a Modelling Method
Data Transformer and Classifier	Wu et al. [10]	Subtypes of Atomic Prompt, we anticipate this to be growing type structure as case studies evolve
Package: LLM	Introduced for clarity	This package allows the prompt model to be tailored for a specific instance of an LLM. LLMs vary in their syntax, focus area and APIs offered. Versions are also important particularly when considering content used for training

Related to this, is the idea of level of formalisation of an enterprise method [61]. The ISO standard supports semantics, but our primary focus is in the structure and syntax of a modelling language. Additionally, as the prompt inputs and LLM outputs are text based, the propose prompt meta-model is not concerned with visual notations. We recognise that could be a future extension.

EM methods typically include multiple perspectives (see Sect. 2.2). Each perspective can be considered a sub-model of an enterprise model tailored for the concerns of a stakeholder group [14] and requiring certain modelling stages and tasks. Thus, we integrate perspectives into our meta-model as components of a methodology that include stages.

The Prompt Model Package, shown in the lower part of Fig. 2, is our primary contribution. The processes of design of a meta-model is a blend of knowledge, derived from existing research literature, technology development and evaluation - or knowledge gained through experiential practice and conjecture, where an assessment is made for likely value.

From the purposive literature review (focused on extracting concepts concerned with prompt design) we note the following.

White et al. [38] propose a pattern modelling language similar to that found in software patterns [62]. Prompt Patterns can document successful approaches for systematically

engineering different output and interaction goals when working with conversational LLMs. Accordingly, we propose a concept *Prompt Pattern*.

Prompt patterns can be collected into domain-specific catalogues. White et al. [38] collate their prompts into a notional catalogue concept of essential patterns to solve production of visualisations. Patterns in that catalogue follow a precise description of how to use a pattern. In our proposal, we advocate for a *Catalogue* concept which support classification of patterns into categories conforming to the underlying enterprise modelling method. In this case, the catalogue is typically derived based on the concept *Perspective* as described by Goldkuhl et al. [22]. Prompt catalogues are reminiscent of efforts from the late 90's where software reuse through component based design approaches to software development advocated the use of software component libraries for storage and retrieval [26]. Reflections on the on this paradigm by Mohammed et al. [63] point to reasons for failure such as: the unlikelihood of functional match, "not invented here syndrome", and architectural mismatch. These issues will also apply in the case of a prompt catalogue.

A Prompt Pattern is proposed to have two subtypes: *Atomic Prompt* and *Prompt Chain*. A *Prompt Chain* decomposes a complex task into a series of targeted sub-tasks that map to a distinct set of LLM steps [10, 40] that correspond to *Atomic Prompt* outputs. These linkages are specified through the *guidedBy* relationship linked to the *Method-4EM* concept (in *EM-4EM* package and *ofType* relationship linked to a *MetaModelElement* in the *MDE* package.

The *Quality Temperature* concept supports hyper parameters of a specific LLM and can be used to control the sensitivity of the outputs of the LLM. Quality Temperature applies to a Prompt Pattern and is may be configured at different stages and *Method Tasks*.

The *LLM* package provides a means of generalising to specific LLMs and the *Syntax* and *API* afforded by the LLM. For example, the experiments reported in this paper utilise the ChatGPT (a type of LLM) offered by the *ProductOwner* OpenAI, *Version GPT3.5*. A specific *LLM* is bound to a Prompt *Spec* via the uses association between *Spec* and *Syntax* that a specific *LLM* offers.⁴

Our *Prompt Spec* captures concepts (i.e., data, and method meta-model) conforming to the *Model* produced by the *Modelling Method*.

We adopt the pattern language proposed by White et al. to capture these behaviour and data specifications.

When using a modelling method, the cognitive action taking place, in dialogue with a domain expert, is the act of interpreting the semantics of a domain concept and then choosing the best representation of the domain concept from a palette of meta-model elements used by a method. For example, In designing a system for a Library, concepts such as Book and Author from the domain are identified and then used to denote the concept, as say an entity type, from a method. The same process is required when the domain expert is replaced by an LLM. The process is captured by the following concepts in the proposed meta-model.

The relationship between LLM-based modelling activity with an Enterprise Modelling method such as 4EM methodology is established through several associations: Firstly,

⁴One can use other LLMs by extending LLM class suitably.

The *Spec* uses *AsTemplate* to *Template* and therefore through navigations to the *Enterprise Modelling Method* (in our case instance 4EM). Secondly, A *Method Task guides-SpecificationFor* a *PromptPattern*. So we are able to bind the details of a task step in the 4EM method to a specific instance of a *PromptPattern*.

The data specification for the prompt *Spec* is expressed as a set of *Term* queries that are built from a *Term* model that references a *Model*, *Domain Element* and *Method Model Element*. Here, we using the concept of query as a way of stating that the collectively, the terms (types, attributes and associations) from the problem domain model and the enterprise modelling method domain model form the vocabulary for the *Spec* and inherited types.

Notably the *Term* uses the concepts from two abstraction layers for navigating concepts conforming to Work Procedure. From a behavioural perspective, *Prompt Spec* specifies either a *Prompt Pattern* or an *Atomic Prompt*. An Atomic Prompt is one that cannot be sub-divided further. There are parallels with Business Process Modelling. Elementary Processes that are atomic are those which are the smallest unit of business activity and which when complete, leave the business in a consistent state [64].

It is anticipated that Atomic Prompts can have many subtypes, and we identify two types as examples: *Data Transformer* and *Classifier*. For example, a data transformer might change a binary number into an octal base number. A classifier prompt might be used to test whether the supplied data in a prompt is a type of entity (also supplied as a parameter), e.g., is a "guest student" a type of student?

A *Response* arising from a *Prompt* is described in terms of *Method Model Elements* as per *Response Template*. *Response Template* is a textual specification of the desired grammar of the model element as suggested by Goldkuhl et al..Hence, the association *conforms to* between *Spec* and the *Modelling Language* provides the appropriate mechanism for this.

7 Evaluation of the meta-model

This section describes how the systematic evaluation of the meta-model against the objectives motivating our research was performed. Section 7.1 introduces the evaluation strategy applied and section 7.2 summarizes the different evaluation episodes and their results. Furthermore, two evaluation episodes are presented in detail: Sect. 7.3 applies the meta-model for restructuring the illustrative example from Sect. 5 and 7.4 presents the evaluation by a modelling expert.

7.1 Evaluation strategy

A conceptual meta model for the enterprise modelling community requires meaningful evaluation to assess its maturity. Here, the evaluation strategy is adopted from the work of Lincoln and Guba [65, p. 289 ff.] on "naturalistic inquiry".

Lincoln and Gruba consider the context of the validation and distinguish between validation that is internal to the development team and external to the developers, they further delineate validations between theory and practice and therefore combine these perspectives as two-by-three matrix as depicted in Table in 6.

Usually, validation starts on the "internal, development team" level with validation in theory followed by validation in practice, and proceeds "downward" in the matrix with

Table 6 Validation steps according to Lincoln and Guba

	Theory	Practice
<i>Internal, Development Team</i>	Validation against state of research, internal consistency checks, within the body of knowledge of method engineering and model-driven development	Prototype implementation for checking feasibility, test in lab environment
<i>External, in validation context</i>	Peer-review of publications describing approach and concepts, comparison to known best practices of the domain	Case studies with application partners using the artifacts for evaluation purposes, Application of the developed artifacts in cooperation / under instruction from developers
<i>External, in application context</i>	Development of extensions or enhancements of the concepts and approaches by external actors Application of the artifacts for creation of new theoretical knowledge, Comparison with related approaches	Use of the artifacts developed (e.g. algorithms, methods, software components) for solutions

Table 7 Summary of evaluation episode outcomes

	Theory	Practice
<i>Internal, Development Team</i>	Internal checks of consistent use of terminology and correct/complete relations in and between the packages; - Design alternatives; - Compatibility with relevant literature. (cf. section 4)	Instantiation of meta-model with prompts from earlier experiments reported in [31] and subsequent refinements of the meta model; again check for internal consistency
<i>External, in validation context</i>	Peer-review of two publications describing meta-model, application examples and related work [31, 28]. Reviews indicated stability of core concepts	Modelling and LLM experts performed evaluation procedure design for external evaluation (pilot study) (c.f. Section 7.4.)
<i>External, in application context</i>	Application of the meta-model to restructure LLM support for the illustrative modelling task described in Sect. 5 - Prompts and accompanying outputs are represented through instantiation of the prompt meta-model. - Some elements of the prompt model are not visible, e.g. data transformers. - Prompt specifications, templates and chains provide a means for repeatable ways of working. - Demonstrate use of different LLMs (ChatGPT 5.2; Gemini 3) - Demonstrate switch between modelling languages (from 4EM to ArchiMate) - Results in switch report over 80% similarity, and outputs as relevant, accurate and free from hallucination. See 7.3	Expert evaluation with application selected and designed by different experts. Some model ambiguity arising from multiple routes to instantiation. Experts report positively on instantiating the meta model from their prompts and results. No changes in meta model arising from the evaluations. Although it was noted that elements of the enterprise meta model could be made more explicit. All Experts reported that software tools implementing the functionality of the meta model would be welcome. Overall conclusion is that the meta-model can be used for its intended purpose for conceptual integration between prompt models and modelling languages. (c.f. 7.4.)

alternating theory and practice validation to “external, in application context”. Thus, the highest validation status would be reached if all cells in the matrix were covered.

As described in Table 6, Lincoln and Guba focus on the validation of research results, i.e., to check whether or not a certain result from research is appropriate for its purpose. In our case, the purpose of the meta-model is defined by the objectives (see introduction to this section). Validation and evaluation, even though different from each other, are very much linked. Evaluation is the process of assessing (and often computing) key characteristics of the research results, which can be used for validation purposes. As many of the meta-model objectives require measurements instead of only checking characteristics, we use the term evaluation episodes in the next section.

7.2 Evaluation episodes

Table 7 shows the performed episodes for evaluating our meta-model proposal and an overall summary of the outcomes.

A practical evaluation of the meta-model was performed in a joint working session with a modelling expert who was not part of the development team. The expert performed the tasks presented in Sect. 7.4 as a pilot study for the planned evaluation in application context (see below). In the pilot study, the expert selected the EM method to use and defined the modelling tasks requiring LLM use. During this pilot study, the expert performed all evaluation tasks but had the possibility to interact with the development team.

The theory part of the external evaluation in application context encompassed two activities: the application of the meta-model in the illustrative example presented in Sect. 5 with the purpose to analyse the LLM support for the modelling task, and demonstration of how to switch from one EM language (4EM) to a different EM language (ArchiMate) in the same example (see Sect. 7.3). The practice part was performed by three experts who selected the EM method to use, defined the modelling tasks and reported the results (see Sect. 7.4).

7.3 Evaluation case study

The aim of this section is to demonstrate the suitability of the proposed EM-aware meta-model for prompt engineering using the illustrative example presented in Sect. 5. At the same time, this forms an evaluation episode in the “external, in validation context” category. Three aspects are considered: (1) decomposition of the example and the mapping of prompts to the meta model; (2) how alternative modelling languages can be used to demonstrate use in different contexts; and (3) the usage of different LLMs.

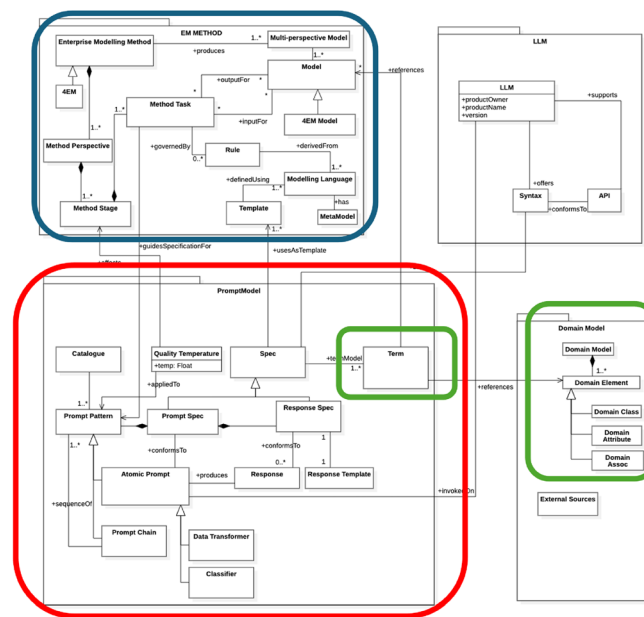
7.3.1 Example mappings to the meta-model

The prompt contents are mapped to the meta-model in Fig. 2 based on elements that are related to the modelling method, domain model, and the prompt meta-model packages. To illustrate the different conceptual parts in the meta-model and the prompts, we used different colours in 3: blue for 4EM, green for the organisational task, and the colour red for the prompt model only to illustrate terms in the prompt that are not part of any other conceptual part.

The modelling domain is obviously enterprise modelling (EM) or a partial perspective of EM such as goal modelling. The example focuses on organisational goals (the goal of improving interdisciplinarity) and concepts (the important terms and information required for the goal). The chosen modelling language, 4EM has in its meta-model, concepts to capture goals, concepts and their relations. Thus, the instantiation of the meta-model in the EM package reflects the 4EM concepts relevant for the organisational task. In prompt Goal-1, these concepts are “Goal”, “Opportunity”, “Problem”, “Threat”, “Weakness”, “Cause”, “Constraint”, including the relations “AND”, “OR”, “supports” and “hinders”.

The actual organisational task to be supported is to prepare a change process in a university towards more interdisciplinary research and education. The natural language task description is used as a request to the domain expert, and the LLM contains terms specifying the intention of the request and the required perspective. Taking a part of prompt Goal-1 as an example, we can identify terms as components of the prompt that are domain concepts and model elements:

- Prompt excerpt: The university aims to develop more interdisciplinary research and education programs. Produce a goal model for this purpose.



Prompt Goal-1

I am trying to capture vision and strategy of an organization using a goal model (GM) as follows: it has concepts of "Goal", "Opportunity", "Problem" (i.e., "Threat" and "Weakness"), "Cause", and "Constraint". Goals are refined by subgoals and they are connected with its parent goal using "AND" and "OR" relationships. All concepts may have binary relationships with other concepts of the type "supports" and "hinders". Produce vision and strategy of university ABC. The university has faculties for medicine, engineering sciences, social sciences, natural sciences, theology, law and life sciences. The university aims to develop more interdisciplinary research and education programs. Produce a goal model for this purpose - please provide a less verbose and detailed outcome. I want all concepts to be labelled and those labels should be used while describing all relationships.

ChatGPT-Output

Goal (G): Develop more interdisciplinary research and education programs. • **Sub-Goal 1 (SG1):** Increase collaboration among faculties. • **Sub-Goal 2 (SG2):** Enhance faculty and student engagement in interdisciplinary projects. • **Sub-Goal 3 (SG3):** Secure funding for interdisciplinary initiatives. **Opportunity (O):** • **O1:** Emerging global challenges that require interdisciplinary approaches. • **O2:** Availability of technology facilitating cross-disciplinary communication. **Problem (P):** • **Threat (T):** T1 - Changes in educational funding policies. • **Weakness (W):** W1 - Existing departmental silos. **Cause (C):** • **C1:** Lack of structured processes for interdisciplinary collaboration. • **C2:** Insufficient incentives for faculty to engage in interdisciplinary work. **Constraint (CN):** • **CN1:** Budget limitations. • **CN2:** Accreditation requirements for traditional disciplines.

Fig. 3 Decomposition of prompts

- Domain concepts in the excerpt: university, develop, interdisciplinary, research, education program
- Model element in the excerpt: goal

The ChatGPT output for the prompts also contains the 4EM concepts and concepts reflecting the organisational tasks.

In summary, the analysis shows that all terms in the prompts and the following output can be represented by instantiating the prompt meta-model. All conceptual parts of the meta-model are required, but a number of concepts in the prompt model are not visible, for example, data transformer prompts. Furthermore, the prompts contain different conceptual parts, i.e., they combine parts independent of the organisational task with parts only related to the organisational task. This makes reuse of prompts for other organisational tasks more difficult and calls for re-designing them in a way that adds structure and prepares reuse.

7.3.2 Systemising repeatable ways of prompting based on the meta-model

In engineering, the aim usually is to use a systematic (repeatable) way of working based on proven elements of solutions to solve an engineering problem. Thus, prompt engineering should define the systematic way of working and how to capture proven solution elements.

In the proposed meta-model, the catalogue of prompt patterns and the response specs reflect the solution elements. For the example, we have to represent the model elements of three different 4EM sub-models (goal/problem, business process, concepts model). This can be implemented by defining a suitable response template and response specs following the template (see Table 8). The response specs and templates are based on 4EM's meta-model [16]. The response specs' elements could be reused for other modelling tasks involving 4EM; the response templates for other modelling languages. Prompt patterns allow for defining prompts for model navigation, i.e., in the case of 4EM, what

Table 8 Prompt model elements defined when redesigning example

No.	Prompt model concept	Instance
P1	Response Template	< sub-model > has < concept list >. < concepts > are decomposed in < sub-concepts >. < concepts > are connected by < relationships > with < concepts >.
P2	Response Spec	Goal model (GM) has concepts "Goal", "Opportunity", "Problem" (i.e., "Threat" and "Weakness"), "Cause", and "Constraint"
P3		Goals are decomposed in subgoals. Subgoals are connected with its parent goal using "AND" and "OR" relationships. All concepts may have binary relationships with other concepts of the type "supports" and "hinders"
P4	Response Spec	Business Process model (BPM) has concepts "Process", "External Process", "Information Set", "Material Set", and "Role"
P5		Processes are decomposed in sub-processes. Processes and external processes "produce" and "consume" information sets or material sets. Processes and sub-processes are connected by "is responsible with role"
P6	Response Spec	I want all concepts to be labelled and those labels should be used while describing all relationships
P7	Prompt pattern	Produce < model type > for < purpose >.
P8	Model navigation::prompt pattern	Produce goal model for the organisation
P9	Model navigation::prompt pattern	Produce business process model for the organisation unit
P10	Model navigation::prompt pattern	Produce concepts model for the concepts included in the goal model
P11	Prompt pattern	Now the focus is to produce < model type > based on < earlier produced model type >.
P12	Model navigation::prompt pattern	Now the focus is to produce concepts model based on goal model
P13	Atomic Prompt	I am trying to capture goals for an organisation. The organisation is a university that has faculties for medicine, engineering sciences, social sciences, natural sciences, theology, law and life sciences. The university aims to develop more interdisciplinary research and education programs
P14	Atomic prompt	Many universities have an organisation unit responsible for all business processes related to external funding of research projects. Describe business processes in this organisation unit
	Prompt Chain	P13, P2, P3, P6, P8 P14, P4, P5
	Prompt Chain	P13, P2, P3, P6, P10

sub-model are addressed and have to be produced. The actual modelling task is represented as atomic tasks. Prompt chains are a sequence of the defined response specs, prompt patterns and atomic prompts.

Table 8 shows the prompts developed during the illustrative example's redesign. It should be noted that there are other possibilities for redesigning the example. For example, P1 could be split into several response templates (e.g., "< sub-model > has < concept list > ", "< concepts > are decomposed in < sub-concepts > ", and "< concepts > are connected by < relationships > with < concepts > "). Furthermore, < P4, P5 > are a repetition of < P2, P3 > for another 4EM sub-model. If more 4EM sub-models were to be included in the prompt, this repetition would have to be done for each additional sub-model, which could be expressed in another response template.

To validate the suitability of the defined prompt chain for the organisational modelling task, we used the new prompt chain with ChatGPT from a different account to avoid training effects from our previous requests. The domain expert familiar with the illustrative example (Sect. 5) is asked to compare the output of the new prompt chain with the initial output. Although the sequence of the sub-goals and the wording are not identical, the domain expert finds more than 90 % similarity. The most substantial difference is one additional sub-goal to "Enhance Faculty and Student Mobility Across Disciplines", which the domain expert considers accurate and relevant. While the similarity of the results are promising, the limitations of our experimental design requires us to consider these results as progress towards the goal of repeatability as a desirable engineering outcome.

7.3.3 Change of modelling language to ArchiMate

The next validation step is to substitute 4EM with another modelling language to demonstrate the suitability of the meta-model for multiple languages and the ease of adaptation. The ArchiMate modelling language was selected as it covers similar concepts to 4EM and is therefore a comparator. The first step of switching to ArchiMate is to adapt the response specs from 4EM to ArchiMate.

For the response specs related to the goal-related task (P2 and P3) and to the business process-related task (P4 and P5) the new specs for ArchiMate are:

- P2a: Goals are expressed with the concepts "goal", "outcome", "principle" and "requirement".
- P3a: Goals are decomposed in subgoals. Subgoals are connected with its parent goal using "composition" relationships. All concepts may have binary relationships with other concepts of the type "realisation".
- P4a: Business processes are expressed with the concepts "Business Process", "Business Event", "Business Object", and "Business Role".
- P5a: Business Processes are decomposed in sub-processes. Business Events and Business Processes "trigger" Business Processes. Business Processes "use" Business Objects. Business Role is connected with "assigned to" Business Process.

To validate the applicability of the ArchiMate-adapted prompt chain for the organisational modelling task, we used the ArchiMate-adapted prompt chain, i.e., P13, P2a, P3a, P6, P8, with ChatGPT from a different account and the domain expert is asked to compare the output of ArchiMate prompt chain with output of the 4EM prompt chain. Again, the sub-goals and the wording are not identical but the domain expert finds more

than 80 % similarity. The difference is sub-goal to "Enhance Faculty and Student Mobility Across Disciplines" from the 4EM prompt is replaced by "Enhance Infrastructure for Interdisciplinary Research" in the ArchiMate prompt output. As the ArchiMate modeling language uses outcome, principle and requirement instead of 4EM's problem, threat, opportunity, cause and constraint, the remaining output for the ArchiMate prompt is different from the 4EM prompt output. However, the domain experts consider the ArchiMate prompt output as relevant, accurate and free from hallucinations.

7.3.4 Change of LLM

Our meta model supports the possibility of the use of alternative LLMs through the *uses* relationship in 6. The evaluation episodes for this paper explore the use of ChatGPT and our claims are thus accordingly tempered by the primary use of ChatGPT rather than multiple LLMs. Elsewhere, our other research has indicated that for generalised situations different LLMs produce broadly similar results [66]. When prompting and the requirements are more granular, then differences emerge. A further challenge with switching between LLMs is the potential different affordances provided by LLMs. As the technology for LLMs continues to evolve, we envisage new functionality and capabilities emerging as LLMs seek to differentiate themselves. For example, temperature-like controls for extracting structured data might emerge. We anticipate that this will require further evolution of the Prompt Meta-Model. Given this, we recognize that limiting our expert evaluation experiments to one particular LLM is a necessary limitation and consequently our claims about generality are tempered accordingly. We leave the change of LLM to further experimental work for the future.

Notwithstanding the position above, we conducted a further evaluation episode with two different LLMs: ChatGPT-5.2 and Google Gemini-3. To validate the possibility of applying different LLMs, we again used the prompt model elements for the organisational modelling task introduced in Table 8 as a basis. We again used a business process-related task, but changed the task to "student mobility" to avoid training effects. The resulting new atomic prompt P15 is: "Many universities have an organisation unit responsible for all business processes related to student mobility. Describe business processes in this organisation unit."

To demonstrate how specific features of LLMs can be controlled, we selected one feature offered by both LLMs and defined atomic prompts for this feature. The selected feature is the inclusion of results from Internet search for a query. When switched off in the settings of an individual account, ChatGPT and Gemini recommend different prompts to override, which leads to the following prompts: P16-GPT: Include results from Internet search; P16-Gemini: You are expected to include external results.

The resulting prompt chain for the evaluation episode is P15, P16, P4, P5. We used the new prompt chain with both LLMs. Excerpts of the results are shown in Fig. 4, which shows in the upper half a list of the top-level business processes returned by the LLMs and in the lower half an example for the refinement of one process. The domain expert familiar with the illustrative example (Sect. 5) and with 4EM is asked to check both outputs for compatibility with 4EM and also to compare the content. The Gemini-3 output is more concise than the ChatGPT-5.2 output. However, the expert concludes that both outputs contain correct and relevant business processes for the given task, are free from hallucinations, include results from Internet search, and adhere to the 4EM meta-model.

ChatGPT 5.2	Gemini 3
1) Mobility governance and portfolio management One sub-process 2) Outgoing student mobility (home university → abroad) Eight sub-processes 3) Incoming student mobility (abroad → host university) Six sub-processes 4) Cross-cutting process: casework, service desk, and stakeholder coordination No sub-processes 5) Data, systems, and reporting (the “back office”) Three sub-processes	1. Partnership and Agreement Management Three sub-processes 2. Outbound Mobility Process (Sending students abroad) Six sub-processes 3. Inbound Mobility Process (Receiving international students) Five sub-processes 4. Financial Management and Compliance Three sub-processes 5. Digital Transformation (Recent Trends) No sub-process
Process 1: Manage Mobility Framework Role responsible: Mobility Officer Sub-processes Identify Partner Institutions Negotiate Inter-Institutional Agreements Maintain Partner & Quota Catalogue Information Sets Partner Institution Profiles Inter-Institutional Agreements Mobility Quotas & Conditions Consumption / Production Consumes: External Partner Proposals (Information Set) Produces: Approved Mobility Agreements (Information Set) External Process Interaction External Process: Partner Institution Administration Exchanges: Agreement Drafts ↔ Signed Agreements	1. Process: Partnership and Agreement Management This process establishes the legal and academic framework for mobility. Role: International Relations Manager (IRM) Sub-processes: Partnership Negotiation: Discussing terms with foreign institutions. Agreement Drafting & Signing: Formalizing the contract. Quality Monitoring: Reviewing partnership performance. Produces (Information Sets): Inter-Institutional Agreement (IIA) Fact Sheet (Partnership details) Consumes (Information Sets): University Strategic Plan Academic Course Catalog External Process: “Partner University Legal Review” (performed by the foreign institution).

Fig. 4 Excerpt from ChatGPT and Gemini output for "student mobility" business processes

7.4 External validation and application

Our use of expert evaluation addressed practice aspects of the Lincoln and Guba matrix in both the external validation and external application context. Applying the meta-model requires a modelling expert familiar with enterprise modelling and basic LLM usage. The evaluation must occur in an unrestricted real-world context while allowing assessment against the meta-model’s objectives. To meet these criteria, we designed a task for EM experts with some LLM experience. Two evaluation rounds were conducted: the first in late 2024, and the second in spring 2025 following minor meta-model revisions. The evaluation task followed a five-step process discussed in this section: expert recruitment, selection of modelling method and example, preparation of prompts for modelling example, evaluation of meta-model’s prompt package, evaluation of other packages in meta-model. The steps and the guidance evaluation instrument is shown in Table 9 below.

One of the expert evaluations were conducted as a pilot study and therefore served as External validation in context / practice cell in the Lincoln and Guba matrix. The other expert evaluations all contribute to the External in application context / Practice cell.

7.4.1 Expert selection

Experts were drawn from our networks of contacts within the enterprise modelling community. There were two sets of evaluation experiments conducted and the expert profiles shown in Table 10 were noted. All had basic LLM experience.

7.4.2 Pilot study

The pilot study was conducted as a session where one of the developers of the meta-model was present when the expert worked on the defined tasks, observed the progress, and was available for questions.

The expert selected 4EM as modelling language, the modelling of the as-is organisation structure of a governmental institution as task, and the Actor & Resources Model of

Table 9 Evaluation steps and guidance

Step	Action	Guidance
0	Expert Selection and Task Description	Selected Expert must have solid experience in at least one Enterprise Modelling Language. Selected Expert must have basic experience in use of LLMs. Task Description: Evaluation of a meta model developed for the purpose of integrating prompt modelling and enterprise modelling. a) Develop prompts for use in enterprise modelling. b) Apply the meta model and provide expert opinion
1	Selection of Modelling Method and Example	Selection and development of prompts to be performed without specific information about meta model. Expert to select modelling method and modelling task where previous results are available
2	Prepare Prompts for Modelling Example	Using written task description document the activities. a) Identify tasks from different phases of selected modelling method. b) For each task define prompts considered useful for completing the task. c) Differentiate between prompts that are general for the task and those that are specific to the modelling example. d) Differentiate between prompts that are controlling the output of the LLM and those aimed at extracting information
3	Evaluation of Meta Model Prompt Package	Expert supplied with meta model description and provided with walkthrough of model. Expert to represent step 2 by using meta model and instantiating classes. Expert required to take notes on gaps, correctness and ambiguities. Analysis of instance models produced by experts. Contribution to group discussion
4	Evaluation of other packages in Meta Model	Interview with expert to collect questions and comments for other packages within the Meta Model. To address: missing concepts, missing relations, concept overlap

Table 10 Expert selection

Schedule	Academic expert	Industry expert
Late 2024	10 years of EM experience with Archimate, BPMN, UML and 4EM	7 years of EM, Archimate and BPMN
Spring 2025	Academic course delivery in EM and Conceptual Modelling	5 years of EM: Archimate and FIM

Prompt excerpt - Pilot Study

You are a consultant to the ministry of interior in the UK. Your first task is to document the **organizational structure** of the ministry, the different **units** and their individual **areas of responsibility**. Please use the actor and resources model (ARM) of the 4EM modelling language to document the organizational structure. ARM has concepts "**Organisation Unit**", "**Role**", and "**Individual**". With the "part of" relation, an "Organization Unit" can be decomposed in other "Organization Units". "Roles" are connected with an "Organization Unit" using the "responsible for" relationship.

ChatGPT-Output

The United Kingdom's Home Office, equivalent to the **Ministry** of the Interior, is structured to manage various aspects of national security, immigration, and law enforcement. [...]

1. Organization Unit: Ministry

- Part of: UK Government
- Responsible for: Overseeing immigration, security, law and order, policing in England and Wales, fire and rescue services in England, Border Force, visas and immigration, and the Security Service (MIS).

[...]

2. Organization Unit: Ministerial Leadership

- Part of: Home Office
- Responsible for: Setting strategic direction and policy for the department.

[...]

3. Organization Unit: Directorates

- Part of: Home Office
- Responsible for: Specific areas of policy and operational delivery.

- **Organization Unit: Border Force**

- Part of: Home Office
 - Responsible for: Securing the UK border by carrying out immigration and customs controls.

[...]

Fig. 5 Expert's prompt and ChatGPT's response in pilot study (excerpt)

4EM as target model (step 1). The expert designed the prompts, tested them with ChatGPT, and improved them (step 2). When the prompts were ready, the expert asked for an introduction to the meta-model, which was given using Fig. 2 and explanations of all packages, classes, and relations. During the explanation, the expert asked for an example

clarifying the difference between the Terms in Spec, Domain Element, and MetaModelElement. We used the task designed by the expert (prompt and response; see Fig. 5 for an excerpt) to provide the clarification as follows:

- Domain Elements: ministry, ministerial leadership, directorate, group
- MetaModelElement: organisation unit
- (Prompt) Spec: organisational structure, unit, area of responsibility
- Term in model: Home Office, Ministerial Leadership, Directorates, Border Force,...

When evaluating the instantiations (step 3) we did not discover any classes used in the wrong way and there were no additional clarification requests.

Parts of the evaluation of the other packages (step 4) were already performed during step 3, as discussed above. The expert did not report any missing or superfluous concepts. However, the expert questioned to suitability of Method Stage and Method Task and proposed to use Method Component / Method Chunk and Method Element / Activity instead. Our explanation of the envisioned compatibility to ISO 42010 satisfied the expert.

7.4.3 Expert evaluation

The expert participating in the first evaluation round selected the Federal Information Management (FIM) methodology. FIM is widespread in the German public sector. FIM is used to support the development of e-government services and for the sharing of service descriptions, data models, and (reference) processes between public actors. The expert decided to focus on the development of the core process within FIM and to use BPMN as modelling language (step 1). The actual modelling task was the application for a construction permit of an attachment to a family home. The expert documented the prompts and decomposed them as requested into general and application-specific ones. After the introduction to the meta-model, the expert managed to instantiate the meta-model without reporting missing or superfluous classes and without clarification needs (step 3).

When examining the prompt after the evaluation, we realized that the expert used the persona pattern and the alternative approaches pattern and instantiated this in the meta-model accordingly. As the BPMN modelling language already seems to be “known” to ChatGPT, the prompt did not include a response spec.

During the discussion with the expert about the meta-model (step 4), we discovered some parts of the model that could have been modelled differently. An example is the relation between DomainElement and Modelling Language. We decided to represent it by modelling that Term references to DomainElement and MetaModelElement, which different terms could be used in a meta-model and the prompts expressing the same domain element. However, this did not result in change needs. The temperature atomic prompt was not known to the experts.

In the second evaluation round, both experts used ArchiMate, but selected only a subset of the ArchiMate modelling language encompassing the concepts and relationships they frequently used in their own modelling. The industry expert decided to focus on the technology and application architectures. The university expert selected business, application, and data architecture (step 1). The actual modelling task in one case was

the IT-architecture of the manufacturing department of an SME, and in the other case was the enterprise architecture of a small retail company. Both experts documented the prompts and decomposed them as requested (step 2). After the introduction to the meta-model, both experts were able to instantiate the meta-model without complaints about flaws or missing classes (step 3). The discussion of the meta-model packages did not lead to any change needs. The university expert had the opinion that the elements of enterprise modelling meta-models could have been made more explicit to show the relation between terms used in prompts and concepts in the meta-model.

All experts agreed that a tool implementing functionality for using the meta-model would be more appropriate than the way of instantiating it proposed in the evaluation procedure.

The most important result of the evaluation is that the meta-model can be used for the intended purpose of representing reusable prompt elements and chains, and making the conceptual integration into the modelling language used explicitly.

In the pilot study, we realized that counting the different kinds of prompts in step 2 did not create any value or new knowledge, as the task for the domain expert was not complex enough to lead to new prompts. We considered removing this part of the step from the study design but preserved it for future use in potentially more complex tasks.

8 Validity considerations

Proposals for conceptual models are often challenged on validity grounds. It is recognised the essential limitation of a conceptual approach that has not been "implemented" in some form. In addition to the evaluation presented in Sect. 7, this paper offers the following argumentation: The internal validity (that is, the concepts, structure) [67] of the proposed prompt model is based on evidence from existing literature, our own expertise of many decades of meta-models and their subsequent implementation in software tools, and the evaluation experiments reported in the paper. Section 7.1 has provided a specific evaluation strategy on the work of Lincoln and Guba [65, p. 289 ff.] on "naturalistic inquiry". Outcomes from that are broadly coherent with those of proposals for validation supported by others such as Moody [68].

In the evaluation episodes, we primarily used qualitative approaches, such as expert evaluation or experiments. The reason for this decision stems from the nature of the main artefact to be evaluated - the meta-model. Understanding the details of the meta-model and applying it effectively requires substantial knowledge and experience in enterprise modeling and prompt engineering, which motivates evaluation by experts in the field. Several studies in quantitative research recommend that, in evaluation tasks with deep structure, insight scales with expertise, not with headcount (see, e.g., [69]). Therefore, we focused more on the competence of the experts than on the number of experts. We selected experts with proven expertise in the above areas, who, at the same time, were from outside the research communities of the authors to avoid bias. The total number of four experts (one expert for the pilot study and three additional experts for the actual evaluation) from our perspective creates a sufficient confidence level in the evaluation results. Future work will have to broaden the application cases for the meta-model (e.g., multi-perspective model generation, modeling consistency checking,

or model integration) with the involvement of additional experts and EM practitioners. Furthermore, the number of LLMs and differences in their technical features have to be increased in future work to further improve confidence in the broad applicability of the meta-model.

A gold standard for a conceptual model remains a software implementation, and that is a future endeavour for further work. External validity, that is, the generalisation and linkage to external environments, is partially through external review, and we have reported on aspects of that in Sect. 7.1, and also through software implementation. Generalisation to other LLMs has been considered through the LLM package, but further and repeated experiments are required. This is left as a task for future research.

In considering the level of formalisation of an enterprise method [61]. The ISO standard supports semantics, but the primary focus here has been on the structure and syntax of a modelling language. As the prompt inputs and LLM outputs are text-based, the prompt model is not concerned with visual notations. This is noted as a future extension as the use of visual notations may impact the "correctness" of LLM outputs.

In supplying a meta model as part of a prompt, it is noted that current LLMs have limitations, such as those reported by [70], where "ChatGPT was unable to process the meta-models in their original size (757 KB of code)". In our work, we did not experience this limitation, but for very complex meta-models, this problem might occur in current LLM versions.

The remarkable power of LLMs is underscored with a recognition that organisations planning for their use should act with a heightened sense of accountability. Ethics dilemmas from the use of LLMs is a separate and important stream of work. A recent systematic review outlines the current landscape [71] is one of several recent studies. An earlier 2021 study outlines in detail ethical and social risks associated with the use of LLMs [72].

In this paper, ethics concerns should consider the impact on labour as the boosting of productivity may lead to a rise in unemployment as organisations seek to "bank the time savings" [73]. For the erstwhile Business Analyst performing enterprise modelling tasks, this is therefore a risk. Noting however, that as the use of AI in these settings evolves, it is possible that new roles may emerge such as Prompt Business Analyst.

9 Discussion and concluding remarks

The research presented in this paper aimed to investigate how enterprise modelling methods can be integrated with prompt modelling approaches to allow LLMs become proxy domain experts and modelling assistants. Following the presentation of our primary contribution proposal for a prompt modelling meta-model we have provided experimental data through three different validation steps. In summary, the three steps showed the following results: The intention of the meta-model is to establish the foundation for representing the content of prompt chains and the output of LLMs for the prompt chains. By decomposing the prompt from the illustrative example and its corresponding output into meta-model elements, we showed that it is possible to accommodate all parts in the meta-model. This is a very basic validation step: validation by instantiation.

Furthermore, the meta-model is supposed to form the basis for prompt engineering, which essentially includes the definition and composition of reusable prompt parts and a pre-defined way of representing outputs. The redesign of the illustrative example demonstrated this possibility.

Finally, we showed that the meta-model is not dependent on a specific EM language by switching from 4EM to ArchiMate, and not dependent on a certain LLM by using ChatGPT-4, ChatGPT-5.2 and Gemini-3. This is the precondition for integrating LLM into EM methods.

There are some limitations to our work in that the organisational and technical results so far only have been applied for a single EM method: 4EM. The package structure proposed in this work provides a means of separating concerns such that other methods can be incorporated. Future work will have to include investigation of other methods as part of the validation. We also report that our prompt meta-model is agnostic of LLMs in much the same way as [7]. Although the agnostic use of an LLM is noted, the experiments have been conducted with only a few LLMs and further experiments with alternative LLMs such as Claude and Llama are needed for comparison. Accordingly, our commentary and claims tempered with that position.

A further package structure of the prompt meta-model could add fine-tuning capabilities which are platform specific for a given LLM. We also draw out some of the key emerging ethical concerns from the use of LLMs for our particular domain of interest. This is potentially a significant additional opportunity. A further powerful validation of the prompt meta-model is feasible through the envisioned development of tool support. Tool support should also include the generation of response specs and response templates from meta-models of modelling languages. Such a software tool environment would support re-use prompt libraries and other typical component-like capabilities. Furthermore, work on LLM-based creation of goal-oriented models [74] indicates that aggregating results from several runs and instructing LLMs to use grammars yield promising results. Both techniques are candidates for future work to advance our approach. Our research agenda for software tooling remains an ongoing research endeavour.

A number of scientific studies investigate the shortcomings and disadvantages of LLMs, such as issues with result accuracy, bias from unbalanced training data, and trustworthiness concerns [75, 76]. For many of these shortcomings, strategies to address them have been proposed. For example, the accuracy and relevance of results can be improved in many application areas by complementing LLMs with vector databases that index essential information, embeddings, or graph databases that incorporate explicit domain knowledge. In our work, we do not discuss such strategies but keep a focus on prompting techniques and their conceptual integration with EM and domain vocabulary. This can be considered a limitation of our work and motivates future work on adding contextual training or adopting suitable vector embedding techniques to improve retrieval efficiency and effectiveness. However, our approach also has advantages. Intentionally, we minimized technology dependencies and based our approach solely on LLMs to facilitate easy LLM changes, prompt reuse, and handling of dynamic queries.

In an additional line of future work, we plan to investigate the differences and possibilities of applying or combining our meta-model with other approaches for generating

enterprise models, such as text-to-model or database-to-model approaches. Past work on generating enterprise models from data about business processes or organization structures originating from information systems or their databases has shown the feasibility of this approach, but was criticized for producing models overloaded with details and incomprehensible to decision-makers. LLM-generated models, in some cases, lack details or the grounding in current enterprise information. Our illustrative example and the evaluation episodes showed that, based on the meta-model, the development of reusable prompts for model generation in different EM languages can be supported. The meta-model could also be applied or extended to database content. Moreover, our focus on text-based prompts currently excludes other means of prompting, such as speech or visual notations. Extending the possibilities of prompting will serve wider user groups, but create new challenges in future work.

Another important area for future improvement of our work lies in enhancing model quality through contextualized and organization-specific knowledge. Our approach to enterprise modelling utilizes metamodel-guided chains of prompts and supports few-shot prompting. Both help to reduce the scope for hallucinations in the responses, and thus in the produced models as the prompts are fine-grained and contextual. However, the key challenge remains that responses from public LLMs are mostly generalized and inherently lack nuanced organization-specific characteristics. In contrast to enterprise modelling techniques that use internal data sources, such as systems of record (SoR), systems of insight (SoI) and data lakes as the basis for modelling, our approach, therefore, often fail to meet the desired quality standards, especially in regulated or specialized domains.

To address this gap, our future research will focus on developing an agentic workflow wherein a coordinated set of agents will: (a) explore and capture public knowledge using public LLMs, (b) enrich the model with proprietary insights by leveraging RAG architectures and embeddings for contextualization, and (c) validate the consistency and accuracy of the instantiations through advanced LLM reasoning and specialized chain-of-thought (CoT) techniques.

In the current form, our approach has proven to be efficient for a selected set of metamodels and is capable of supporting existing LLMs. There is also considerable scope to extend our framework to accommodate newer forms of LLMs that may require different prompt templates and styles. We are confident that these features significantly help lay the groundwork for AI-driven enterprise modelling. With the proposed future enhancements, we believe our approach will further broaden its applicability to a wider set of metamodels and prove valuable across diverse organizational contexts, including regulated and uncommon domains.

Appendix

See Tables [11](#), [12](#) and [13](#).

Table 11 Prompts used for the modelling task

No.	Prompt
Goal-1	I am trying to capture vision and strategy of an organisation using a goal model (GM) as follows: it has concepts of "Goal", "Opportunity", "Problem" (i.e., "Threat" and "Weakness"), "Cause", and "Constraint". Goals are refined by subgoals and they are connected with its parent goal using "AND" and "OR" relationships. All concepts may have binary relationships with other concepts of the type "supports" and "hinders". Produce vision and strategy of university ABC. The university has faculties for medicine, engineering sciences, social sciences, natural sciences, theology, law and life sciences. The university aims to develop more interdisciplinary research and education programs. Produce a goal model for this purpose - please provide a less verbose and detailed outcome. I want all concepts to be labelled and those labels should be used while describing all relationships
Goal-2	Please refine the sub-goals and show the relationships between the refined sub-goals
BPM-1	Many universities have an organisation unit responsible for all business processes related to external funding of research projects. Describe business processes in this organisation unit
BPM-2	Now I want to have a business process model (BPM) of this university. Such a model essentially describes What are the business processes? How do they handle information? and material? Essentially, a business process is assumed to consume input in terms of information and/or material and produce output of information and/or material. BPM components are process, external process, information set, and material set. Processes and external processes produce and consume information or material sets. Processes can be decomposed into sub-processes. It also includes the roles in the university responsible for the process or sub-process. Produce BPM for the university

Table 12 ChatGPT output for prompt BPM-1 (business processes in grants office), including domain expert evaluation

ChatGPT prompt or output	Expert evaluation
1. Identification and Dissemination of Funding Opportunities: Researching Funding Sources: Identifying potential funding sources. Announcements: Disseminating information about funding opportunities to faculty and researchers	Accurate and complete
2. Pre-Award Processes: Proposal Development: Assisting researchers in the development of project proposals. Compliance Checks: Ensuring proposals meet institutional and sponsor guidelines. Submission: Coordinating the proposal submission process, ensuring proposals on time and meet agency requirements	Accurate, but should be called preparation support
3. Award Negotiation and Acceptance: Negotiation: Engaging in negotiations with funding agencies over terms and conditions of awards. Acceptance: Facilitating the acceptance of awards. Account Setup: Establishing accounts for the management of awarded funds	'Negotiation' is only possible in theory but never happens in practice; other are accurate
4. Post-Award Management: Financial Administration: Managing the financial aspects of research projects. Compliance and Reporting: Ensuring compliance with all terms and conditions of the award. Amendments and Revisions: Handling requests for changes to project	accurate, but should be called operational management
5. Technology Transfer and Intellectual Property Management: IP Management: Managing intellectual property rights arising from research. Commercialisation Support: Assisting in the commercialisation of research outcomes	'Technology transfer' is out of scope, i.e., responsibility of another org. unit
6. Closure: Final Reporting: Coordinating the submission of final reports to the funding agency. Closeout: Ensuring all financial obligations are settled and that funds are fully expended or returned. Lessons Learned: Reviewing the project's outcomes and processes to improve future research administration practices	Accurate

Table 13 ChatGPT output for prompt Goal-1 (goals to reach interdisciplinarity), including domain expert evaluation

ChatGPT Output	Expert Evaluation
<i>Concepts and Labels</i>	
Goal (G): Develop more interdisciplinary research and education programs.	All sub-goals make sense but need refinement
Sub-Goal 1 (SG1): Increase collaboration among faculties.	
Sub-Goal 2 (SG2): Enhance faculty and student engagement in interdisciplinary projects.	
Sub-Goal 3 (SG3): Secure funding for interdisciplinary initiatives	
<i>Opportunity (O)</i>	
O1: Emerging global challenges that require interdisciplinary approaches.	O2 is not relevant / out of scope
O2: Availability of technology facilitating cross-disciplinary communication	
<i>Problem (P)</i>	
Threat (T): T1 - Changes in educational funding policies.	Accurate and relevant
Weakness (W): W1 - Existing departmental silos	
<i>Cause (C)</i>	
C1: Lack of structured processes for interdisciplinary collaboration.	All causes make sense
C2: Insufficient incentives for faculty to engage in interdisciplinary work	
<i>Constraint (CN)</i>	
CN1: Budget limitations.	Accurate and relevant
CN2: Accreditation requirements for traditional disciplines	

Supplementary Information

The online version contains supplementary material available at <https://doi.org/10.1007/s44163-026-01056-y>.

Supplementary file 1.
Supplementary file 2.
Supplementary file 3.
Supplementary file 4.
Supplementary file 5.

Author contributions

All authors jointly decided on the direction and structure of the paper, and also produced the introduction, validation and concluding remarks. BB provided the background section on LLM (2.1) and the related work on prompt engineering (4.1) with contributions from SB. KS wrote the background section on EM (2.2) and performed the literature analysis (4.2). BB and SB took the lead in developing the meta-model (6) and describing its different concepts; KS contributed. KS took the lead in describing the research approach (3), the illustrative example (5), and the evaluation of the meta-model (7); BB and SB contributed.

Funding

Open Access funding enabled and organized by Projekt DEAL. No funding project.

Data availability

The authors declare that the data supporting the findings of this study are available within the paper, its appendix, and in a supplementary information file. The paper contains the complete meta-model (concepts, model, and examples of instances) and the data related to the literature study. The paper and appendix present the prompts and the evaluation of their corresponding outputs. The supplementary information file contains the full protocol of the ChatGPT session used in the illustrative example.

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Clinical trial

Not applicable.

Competing interest

The authors declare no Conflict of interest.

Received: 8 November 2025 / Accepted: 17 February 2026

Published online: 13 March 2026

References

1. Makridakis S. The forthcoming artificial intelligence (ai) revolution: its impact on society and firms. *Futures*. 2017;90:46–60.
2. Eloundou T, Manning S, Mishkin P, Rock D. Gpts are gpts: an early look at the labor market impact potential of large language models. [Preprint]. 2023. Available from: arXiv preprint [arXiv:2303.10130](https://arxiv.org/abs/2303.10130).
3. Cámara J, Troya J, Burgueño L, Vallecillo A. On the assessment of generative ai in modeling tasks: an experience report with chatgpt and uml. *Softw Syst Model*. 2023;22(3):781–93.
4. Wang X, Anwer N, Dai Y, Liu A. Chatgpt for design, manufacturing, and education. *Procedia CIRP*. 2023;119:7–14.
5. van der Linden D, Ralyte J, Sandkuhl K, Zdravkovic J. Panel discussion: Enterprise modeling in the digital age. In: 13th IFIP WG 8.1 Working Conference on the Practice of Enterprise Modeling (PoEM 2020), Riga, Latvia. 2020. pp. 1–12.
6. Nast B, Görgen L, Müller E, Triller M, Sandkuhl K. Exploring large language models in enterprise modeling. *Discover Artif Intell*. 2025;5:1–32.
7. Clariso R, Cabot J. Model-driven prompt engineering. In: 2023 ACM/IEEE 26th International Conference on Model Driven Engineering Languages and Systems (MODELS). 2023. pp. 47–54.
8. Barn BS, Barat S, Sandkuhl K. Adaptation of enterprise modeling methods for large language models. In: IFIP Working Conference on The Practice of Enterprise Modeling. 2023. pp. 3–18.
9. Huang W, Abbeel P, Pathak D, Mordatch I. Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. In: International Conference on Machine Learning. 2022. pp. 9118–9147.
10. Wu T, et al. Promptchainer: chaining large language model prompts through visual programming. In: CHI Conference on Human Factors in Computing Systems Extended Abstracts. 2022. pp. 1–10.
11. Vernadat FB. Enterprise modelling and integration: from fact modelling to enterprise interoperability. Enterprise inter- and intra-organizational integration: Building international consensus. 2003. pp. 25–33.
12. Sandkuhl K, et al. From expert discipline to common practice: a vision and research agenda for extending the reach of enterprise modeling. *Bus Inform Syst Eng*. 2018;60:69–80.
13. Vernadat F. Enterprise modelling: research review and outlook. *Comput Ind*. 2020;122:103265.
14. Frank U. Multi-perspective enterprise modeling: foundational concepts, prospects and future research challenges. *Softw Syst Mod*. 2014;13:941–62.
15. Frank U. The memo meta modelling language (mml) and language architecture. ICB-research report: Tech. Rep; 2011.
16. Lantow B, Sandkuhl K, Stirna J. Enterprise modeling with 4em: perspectives and method. Domain-specific conceptual modeling: Concepts, methods and ADOxx tools. 2022. pp. 95–120.
17. Group TO. Archimate 3.2 specification. Tech. Rep. The Open Group. 2022.
18. Karagiannis D, Lee M, Hinkelmann K, Utz W. Domain-specific conceptual modeling: concepts, methods and ADOxx tools Springer Nature. 2022.
19. Sandkuhl K, Stirna J, Persson A, Wißotzki M. Enterprise modeling Springer. 2014.
20. Brinkkemper S. Method engineering: engineering of information systems development methods and tools. *Inf Softw Technol*. 1996;38:275–80.
21. Henderson-Sellers B, Ralyte J, Ågerfalk PJ, Rossi M. Situational method engineering Springer. 2014.
22. Goldkuhl G, Lind M, Seigerroth U. Method integration: the need for a learning perspective. *IEE Proc Soft*. 1998;145:113–8.
23. Hevner AR, March ST, Park J, Ram S. Design science in information systems research. *MIS Q*. 2004;28(1):75–106.
24. Kitchenham B. Procedures for performing systematic reviews. Keele UK Keele Univ. 2004;33:1–26.
25. Sjøberg DI, et al. A survey of controlled experiments in software engineering. *IEEE Trans Software Eng*. 2005;31:733–53.
26. Brown A, Barn B. Enterprise-scale cbd: building complex computer systems from components. STEP '99. Proceedings Ninth International Workshop Software Technology and Engineering Practice. 1999. pp. 82–91.
27. Barn B, Brown A, Cheesman J. Methods and tools for component based development. Proceedings. Technology of Object-Oriented Languages. TOOLS 26 (Cat. No.98EX176). 1998. pp. 385–395.
28. Barn B. Chatgpt could be your ally – really! MDX Minds. 2023. <https://mdxminds.com/2023/01/23/chatgpt-could-be-your-ally-really/>.
29. Liu P, et al. Pre-train, prompt, and predict: a systematic survey of prompting methods in natural language processing. *ACM Comput Surv*. 2023;55:1–35.
30. Openlaender J. Prompt engineering for text-based generative art. [preprint]. 2022. Available from: arXiv [arXiv:2204.13988](https://arxiv.org/abs/2204.13988).
31. Sandkuhl K, Barn B, Barat S. Neural text generators in enterprise modeling: Can chatgpt be used as proxy domain expert? Accepted for publication. In: Proceedings ISD 2023 conference, 2023.
32. Fill H-G, Fetteke P, Köpke J. Conceptual modeling and large language models: impressions from first experiments with chatgpt. *Enterprise Mod Inform Syst Architect (EMISAJ)*. 2023;18:1–15.
33. Trautmann D, Petrova A, Schilder F. Legal prompt engineering for multilingual legal judgement prediction. [preprint]. 2017. Available from : arXiv [arXiv:2212.02199](https://arxiv.org/abs/2212.02199).
34. Wang J, et al. Prompt engineering for healthcare: Methodologies and applications. arXiv [preprint]. 2023. Available from: [arXiv:2304.14670](https://arxiv.org/abs/2304.14670).
35. Busch K, Rochlitz A, Sola D, Leopold H. Just tell me: Prompt engineering in business process management. In: International Conference on Business Process Modeling, Development and Support. 2023. pp. 3–11.
36. Abukhalaf S, Hamdaqa M, Khomh F. On codex prompt engineering for ocl generation: An empirical study. [preprint]. 2023. Available from: arXiv preprint [arXiv:2303.16244](https://arxiv.org/abs/2303.16244).
37. Fiannaca AJ, Kulkarni C, Cai CJ, Terry M. Programming without a programming language: Challenges and opportunities for designing developer tools for prompt programming. In: Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems. 2023. pp. 1–7.

38. White J, et al. A prompt pattern catalog to enhance prompt engineering with chatgpt. [Preprint]. 2023. Available from: arXiv preprint [arXiv:2302.11382](https://arxiv.org/abs/2302.11382).
39. Du R, et al. Rapsai: Accelerating machine learning prototyping of multimedia applications through visual programming. In: Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems. 2023. pp. 1–23.
40. Cheng Y, et al. Prompt sapper: A llm-empowered production tool for building ai chains. [Preprint]. 2023. Available from: arXiv preprint [arXiv:2306.12028](https://arxiv.org/abs/2306.12028).
41. Bach SH, et al. Promptsources: An integrated development environment and repository for natural language prompts. [Preprint]. 2022. Available from: arXiv preprint [arXiv:2202.01279](https://arxiv.org/abs/2202.01279).
42. Ekin S. Prompt engineering for chatgpt: a quick guide to techniques, tips, and best practices. 2023.
43. Klievtsova N, Benzin J-V, Kampik T, Mangler J, Rinderle-Ma S. Conversational process modelling: State of the art, applications, and implications in practice. *Lecture Notes in Business Information Processing* 490 LNBI. 2023. pp. 319–336. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85172703496&doi=10.1007/978-3-031-20232-4_152.
44. Kourani H, Berti A, Schuster D, van der Aalst WMP. Process modeling with large language models. *Lecture Notes in Business Information Processing* 511 LNBI. 2024. pp. 229–244. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85197102930&doi=10.1007/978-3-031-20232-4_152.
45. Vidgof M, Bachhofner S, Mendling J. Large language models for business process management: Opportunities and challenges. In: *International Conference on Business Process Management*. 2023. pp. 107–123.
46. Buchmann R, et al. Large language models: expectations for semantics-driven systems engineering. *Data Knowl Eng*. 2024;152:102324 (<https://www.sciencedirect.com/science/article/pii/S0169023X2400048X>).
47. de Kinderen S, Winter K. Towards taming large language models with prompt templates for legal grl modeling. *Lecture Notes in Business Information Processing* 511 LNBI. 2024. pp. 213–228. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85197138654&doi=10.1007/978-3-031-20232-4_152.
48. Apaydin K, Zisgen Y. Local large language models for business process modeling. *Lecture Notes in Business Information Processing*. 2025;533:605–9. https://www.scopus.com/inward/record.uri?eid=2-s2.0-105002035733&doi=10.1007/978-3-031-20232-4_152.
49. Hörner LF. Towards an llm-based conversational framework for business process modeling: Research approach and preliminary results. *International Conference on Advanced Information Systems Engineering*. 2025. pp. 286–293.
50. Bennoit C, Greff T, Baum D, Bajwa IA. (eds). Identifying Use Cases for Large Language Models in the Business Process Management Lifecycle. 2024. <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85216075136&doi=10.1109/ICSE50999.2024.00010>.
51. Klievtsova N, Benzin JV, Mangler J, Kampik T, Rinderle-Ma S. Process modeler vs. chatbot: Is generative ai taking over process modeling? *Lecture Notes in Business Information Processing*. 2025;533, 637–649. https://www.scopus.com/inward/record.uri?eid=2-s2.0-105002040914&doi=10.1007/978-3-031-20232-4_152.
52. Neuberger J, Ackermann L, van der Aa H, Jablonski S. A universal prompting strategy for extracting process model information from natural language text using large language models. *Lecture Notes in Computer Science* 15238 LNCS. 2025. pp. 38–55. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85209570979&doi=10.1007/978-3-031-20232-4_152.
53. Licardo JT, Tanković N, Etinger D. A method for extracting bpmn models from textual descriptions using natural language processing. *Procedia Comput Sci*. 2024;239. <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85201266482&doi=10.1016/j.procs.2024.01.016>.
54. Köpke J, Safan A. Efficient llm-based conversational process modeling. *Lecture Notes in Business Information Processing* 534 LNBI. 2025. pp. 259–270. https://www.scopus.com/inward/record.uri?eid=2-s2.0-86000466769&doi=10.1007/978-3-031-20232-4_152.
55. Nour Eldin A, Assy N, Anesini O, Dalmás B, Gaaloul W. A decomposed hybrid approach to business process modeling with llms. *Lecture Notes in Computer Science* 15506 LNCS. 2025. pp. 243–260. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85218943963&doi=10.1007/978-3-031-20232-4_152.
56. Storey VC, et al. Large language models for conceptual modeling: Assessment and application potential. *Data Knowledge Eng*. 2025;160. <https://www.scopus.com/inward/record.uri?eid=2-s2.0-105009419852&doi=10.1016/j.datake.2025.100160>.
57. Fetteke P, Fill H-G Köpke J. Llm, lam, lxm agent: from talking to acting machines: Insights from the perspective of conceptual modeling. *Enterprise Modelling and Information Systems Architectures (EMISA)*. 2025. pp. 20.
58. Challa G, Gattini A, Muff F, Fill H-G. An architecture for integrating large language models into metamodeling platforms. In: *CEUR WS Proceedings*. 2024. pp. 3849.
59. Görgen L, Müller E, Triller M, Nast B, Sandkuhl K. Large language models in enterprise modeling: case study and experiences. *Int Conf Model-Driven Eng Soft Develop*. 2024;1:74–85.
60. Organization IS. Software engineering — metamodel for development methodologies. 2014. <https://www.iso.org/standard/62644.html>.
61. Bork D, Fill H-G. Formal Aspects of Enterprise Modeling Methods: A Comparison Framework. 2014 47th Hawaii International Conference on System Sciences (HICSS). 2014. pp. 3400–3409.
62. Gamma E, Helm R, Johnson R, Vlissides J. Design patterns: elements of reusable object-oriented software Pearson Deutschland GmbH. 1995.
63. Mohammadi H, Ghardallou W, Brick E, Mili A. On the persistent rumors of the programmer's imminent demise. *Softw Syst Model*. 2023;22:1969–76.
64. Instruments T. A guide to information engineering using the ief™. TI Part. 1990.
65. Lincoln Y, Guba E. *Naturalistic Inquiry* SAGE Publications Inc. 1985.
66. Kolev P-A, Pruss HH, Wilken JR, Sandkuhl K. Grass-root enterprise modelling: how large language models can help. In: *IFIP Working Conference on The Practice of Enterprise Modeling*. 2024. pp. 123–139.
67. Engström E, Runeson P. Software product line testing—a systematic mapping study. *Inf Softw Technol*. 2011;53:2–13.
68. Moody DL. Theoretical and practical issues in evaluating the quality of conceptual models: current state and future directions. *Data Knowl Eng*. 2005;55:243–76.
69. Dane E, Pratt MG. Exploring intuition and its role in managerial decision making. *Acad Manag Rev*. 2007;32:33–54.
70. Muff F, Fill H-G. Limitations of chatgpt in conceptual modeling: Insights from experiments in metamodeling. *Modellierung 2024 Satellite Events*. 2024. pp. 10–18420.
71. Jiao J, Afroogh S, Xu Y, Phillips C. Navigating llm ethics: advancements, challenges, and future directions. [Preprint]. 2024. Available from: arXiv preprint [arXiv:2406.18841](https://arxiv.org/abs/2406.18841).

72. Weidinger L, et al. Ethical and social risks of harm from language models. [Preprint]. 2021. Available from: [arXiv:2112.04359](https://arxiv.org/abs/2112.04359).
73. Sharps S, et al. The impact of AI on the labour market. 2024. url=<https://institute.global/insights/economic-prosperity/the-impact-of-ai-on-the-labour-market>.
74. Chen B, et al. On the use of gpt-4 for creating goal models: an exploratory study. 2In: 023 IEEE 31st International Requirements Engineering Conference Workshops (REW). 2023. pp. 262–271.
75. Huang X, et al. A survey of safety and trustworthiness of large language models through the lens of verification and validation. *Artif Intell Rev*. 2024;57:175.
76. Hajikhani A, Cole C. A critical review of large language models: sensitivity, bias, and the path toward specialized ai. *Quantitative Sci Studies*. 2024;5:736–56.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.