AutoBPMN.AI: Conversational Process Modeling and Automation

Nataliia Klievtsova[1,*], Matthias Ehrendorfer[1], Juergen Mangler[1] and Stefanie Rinderle-Ma[1]

[1] *Technical University of Munich, TUM School of Computation, Information and Technology, Garching, Germany*

Abstract

We demonstrate the text-based modeling and redesign of processes in the process execution environment of the Cloud Process Execution Engine (CPEE) based on iterative conversations with an LLM such as gemini or gpt. The input can be text and an empty model as well as text and an existing model. The output is a newly created or redesigned process model in executable format, i.e., based on abstract syntax trees and displayed in BPMN-like fashion, exploiting the automatic layout capabilities of the CPEE. The tool is demonstrated for different scenarios such as the creation and redesign of the process models, as well as the subsequent process model execution.

1. Introduction

Process models are used to provide a blueprint of how processes should be performed in an organization. Therefore, they include information on steps needed to be done, their order, decisions needed to be taken and corresponding data, utilized data objects as input or output for certain tasks, roles allowed to perform certain tasks, etc. Depending on the layer on which the process models are created (cf. [1, p. 127]) the more information needs to be contained in the process models. While semantic or conceptual process modeling languages such as Business Process Model and Notation (BPMN) or Event-Driven Process Chain (EPC) focus on documenting and communicating processes, logic/formal languages such as Petri Nets or Refined Process Structured Trees (RPST) are well-suited to verify and analyze process properties (e.g., soundness, boundedness, or liveness). Finally, on the layer of graphical programming/execution languages such as Workflow Nets, Cloud Process Execution Engine Trees (CPEE-Trees), Business Process Execution Language for Web Services (BPEL), or Well-Structured Marking Nets (WSM-Nets) it is possible to directly execute the process model based on the execution semantics of the modeling languages.

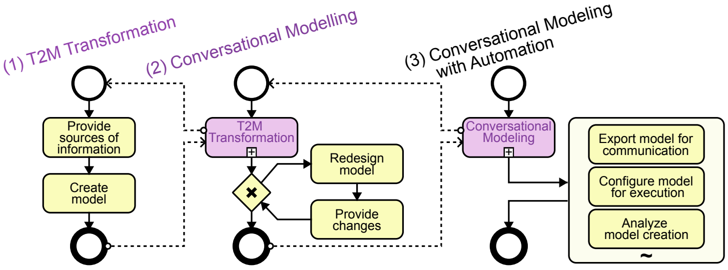

(1) T2M Transformation
(2) Conversational Modelling
(3) Conversational Modeling with Automation
Provide sources of information
Create model
T2M Transformation
Redesign model
Provide changes
Conversational Modeling
Export model for communication
Configure model for execution
Analyze model creation

Figure 1: Simplified Overview of Automated Generation of Process Models

Over the last 20 years, significant efforts have been devoted to automatically derive business process models from natural language text [2]. However, the resulting models remain at a conceptual level [3, 4]

Proceedings of the Best BPM Dissertation Award, Doctoral Consortium, and Demonstrations & Resources Forum co-located with 23rd International Conference on Business Process Management (BPM 2025), Seville, Spain, August 31st to September 5th, 2025.
[*] Corresponding author.
nataliia.klievtsova@tum.de (N. Klievtsova); matthias.ehrendorfer@tum.de (M. Ehrendorfer); juergen.mangler@tum.de (J. Mangler); stefanie.rinderle-ma@tum.de (S. Rinderle-Ma)
0009-0009-9010-2855 (N. Klievtsova); 0000-0002-7739-9123 (M. Ehrendorfer); 0000-0002-6332-5801 (J. Mangler); 0000-0001-5656-6108 (S. Rinderle-Ma)

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073

(see Fig. 1, (1) and (2)). For their further utilization these models have to be transformed across languages in different layers, which can be expensive, error-prone, and difficult. Hence, modeling processes directly at the execution layer may be desirable, but requires to consider execution properties such as including data objects necessary for determining how often a loop should be executed and thinking about the functionality of certain tasks. As process modeling can be already a challenge for domain experts who are often non-technical users, modeling at execution level might be too complex.

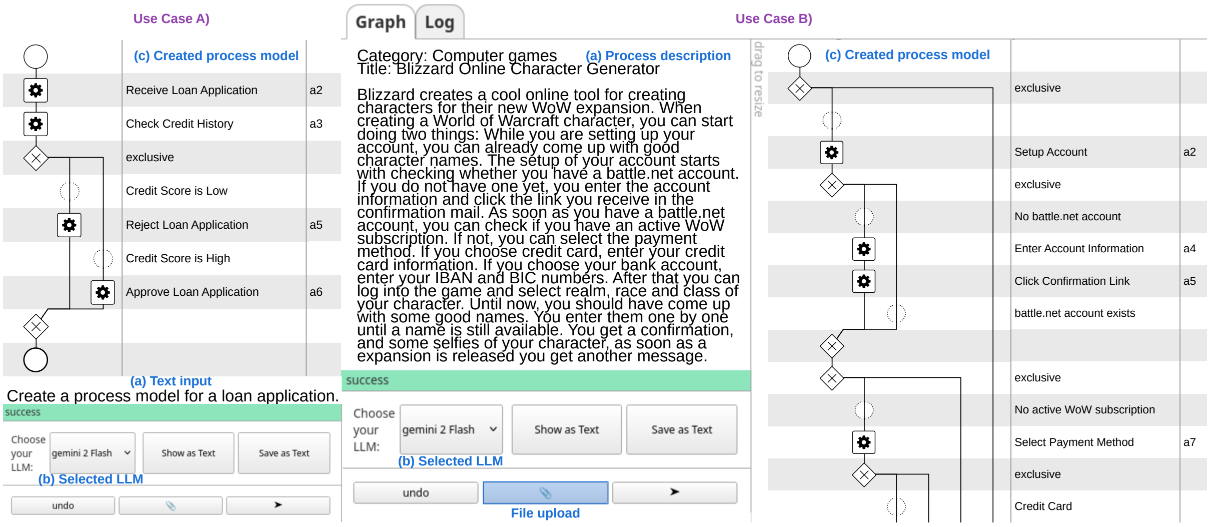
Thus, in this paper, we present a tool, bridging conceptual and executable layers, which allows domain experts to create and redesign process models directly in an execution environment as shown in Fig. 1, (3). The UI relies on natural text input in conversations with a chatbot (conversational process modeling and redesign) [5, 6]. The input comprises process descriptions, guidelines, and legal documents. The output is a process model which has been refined by the domain experts and is per definition sound and executable from a control flow perspective. It can then be used in several ways, including augmentation with data flow and endpoints for execution, transformation back to text, and simulation and analysis. Additionally, by logging how the process model is changed and based on which interactions with the users these changes have been achieved it is possible to analyze the process of process modeling [7].

2. Conversational Modeling and Automation with CPEE

In this section, we discuss the main characteristics and innovations of the LLM-based CPEE extension, which utilizes the concept of conversational process modeling. Since the primary goal of conversational modeling is to create and improve process models and descriptions through iterative exchange of question and answers between domain experts and chatbot, we demonstrate its application with **two scenarios:** (1) process model creation and (2) process model redesign.

Scenario 1) Process Model Creation can be driven by the use of documents and textual sources containing information about the process—such as regulatory documents or summaries based on domain expert interviews—or it can start from scratch, without any supplementary materials. Thus, Process model creation can be represented by two use cases.

Use Case A) Create Process Model "Out of the Blue" creates a process model (c) for a specific use case without further input, cf. (a) text input and (b) selected LLM in Fig. 2. For more information about process model and possible steps after model generation see Conversational Modeling and Automation with CPEE



Create a process model for a loan application.

success

Choose your LLM: gemini 2 Flash

Category: Computer games
Title: Blizzard Online Character Generator

Blizzard creates a cool online tool for creating characters for their new WoW expansion. When creating a World of Warcraft character, you can start doing two things: While you are setting up your account, you can already come up with good character names. The setup of your account starts with checking whether you have a battle.net account. If you do not have one yet, you enter the account information and click the link you receive in the confirmation mail. As soon as you have a battle.net account, you can check if you have an active WoW subscription. If not, you can select the payment method. If you choose credit card, enter your credit card information. If you choose your bank account, enter your IBAN and BIC numbers. After that you can log into the game and select realm, race and class of your character. Until now, you should have come up with some good names. You enter them one by one until a name is still available. You get a confirmation, and some selfies of your character, as soon as a expansion is released you get another message.

success

Choose your LLM: gemini 2 Flash

Figure 2: Create Process Model for Loan Application and from Gaming Process Description

Use Case B) Create Process Model Based on Text (T2M Transformation): Though Use Case A) might provide a starting point for some applications, a more guided creation of process models might be advisable in order to control the quality of the output [5]. One option is to work with more specified textual input such as process descriptions and/or regulatory documents. In Fig. 2 a gaming process description[1] is inserted (a), processed by LLM *gemini 2 Flash* (b), and the corresponding process model is created (snippet shown in (c)).

Scenario 2) Redesign Process Model: Use cases for Scenario 1) have illustrated the one-shot creation of process models out of the blue or from process descriptions. However, as the quality of the output process model is of utmost importance for many real-world domains, the iterative refinement of the models through conversational process model redesign [8, 6] is crucial. In particular, this can foster the interaction of domain experts with the model. Conceptually, conversational redesign is based in exploiting the established process change patterns [9] in matching them with textual user redesign requests by the LLM in order to achieve sound process models [8, 6]. Consider the abstract example depicted in Fig. 3. The process model is created by textual redesign requests step by step: i) *add task A*; ii) *add decision after task A with two branches, one branch contains task B, the other branch task C*; iii) *add task D after B and C*; iv) *delete task B*. The resulting process models after each step i)–iv) are shown in Fig. 3. The process of redesigning the model is stored in the process execution log resembling the process of process modeling [7].

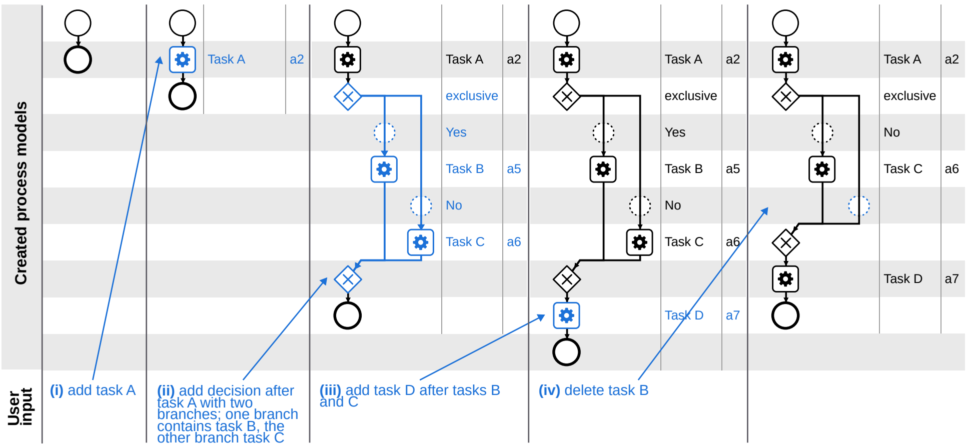


Figure 3: Create and Redesign Process Model

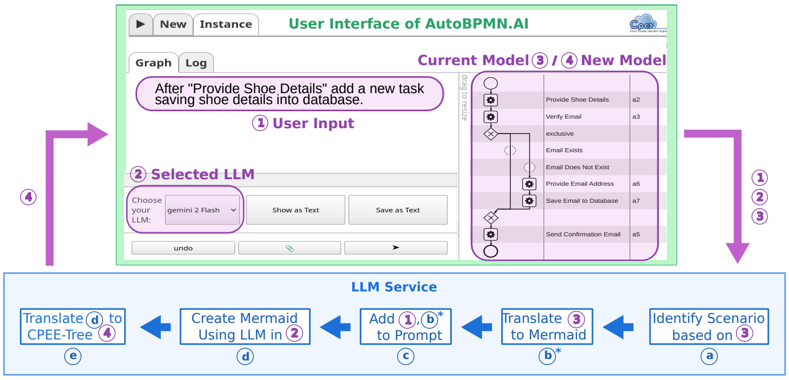
Conceptual Framework. The framework encompasses five steps (ⓐ to ⓔ) depicted in Fig. 4. To start with, a text describing how the model should look/change ①, the Selected LLM ②, and the Current Model ③ if it exists is sent to the LLM Service. Next, an appropriate scenario has to be identified ⓐ (i.e., either model creation or model redesign). Depending on the scenario, ③ is either converted to Mermaid[2] representation ⓑ or this step is skipped. Then, ① and, optionally, the result of ⓑ are inserted into the prompt ⓒ, and the LLM request is performed ⓓ (cf. [8, 10] for more information about prompt design, parameters, and examples). Finally, the output of ⓓ is transformed back into a CPEE-Tree ④ in ⓔ and passed back to the user interface for further usage/refinement.

Possible Subsequent Steps. After having obtained the final model the following steps can be taken:

- **Retrieve items for documentation and communication** to other participants of the process from models created in the scenarios described above (i.e., semantic/conceptual layer). The tool offers two ways to do this: (i) obtain a visual representation of the model ("save svg graph" in "Instance" tab shown in Fig. 4) and (ii) transform the model to text (again, using LLM) using the "Save as Text" button (again, in "Instance" tab shown in Fig. 4). An example of (ii) is shown in Fig. 5a Ⓐ.
- **Make the process model ready for execution** by adding information needed for execution (data objects, functionality and in-/output data objects for tasks, ...) directly to models created using the tool which only include control-flow but are already described in the form of a CPEE-Tree (which is a

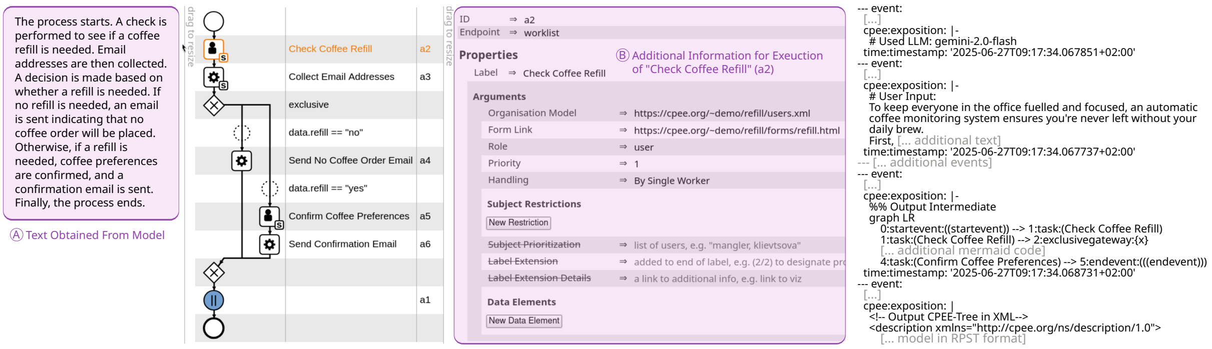
[1] taken from https://zenodo.org/records/7783492

[2] https://mermaid.js.org/, accessed 2025-06-26



After "Provide Shoe Details" add a new task saving shoe details into database.

Figure 4: Conceptual Framework



The process starts. A check is performed to see if a coffee refill is needed. Email addresses are then collected. A decision is made based on whether a refill is needed. If no refill is needed, an email is sent indicating that no coffee order will be placed. Otherwise, if a refill is needed, coffee preferences are confirmed, and a confirmation email is sent. Finally, the process ends.

Ⓐ Text Obtained From Model

Check Coffee Refill	a2
Collect Email Addresses	a3
exclusive	
data.refill == "no"	
Send No Coffee Order Email	a4
data.refill == "yes"	
Confirm Coffee Preferences	a5
Send Confirmation Email	a6
	a1

ID ⇒ a2
Endpoint ⇒ worklist

Properties

Label ⇒ Check Coffee Refill

Ⓑ Additional Information for Exeuction of "Check Coffee Refill" (a2)

Arguments

Organisation Model	⇒ https://cpee.org/~demo/refill/users.xml
Form Link	⇒ https://cpee.org/~demo/refill/forms/refill.html
Role	⇒ user
Priority	⇒ 1
Handling	⇒ By Single Worker

Subject Restrictions

New Restriction

~~Subject Prioritization~~	⇒ list of users, e.g. "mangler, klievtsova"
~~Label Extension~~	⇒ added to end of label, e.g. (2/2) to designate pro
~~Label Extension Details~~	⇒ a link to additional info, e.g. link to viz

Data Elements

New Data Element

```
--- event:
  [...]
  cpee:exposition: |-
    # Used LLM: gemini-2.0-flash
  time:timestamp: '2025-06-27T09:17:34.067851+02:00'
--- event:
  [...]
  cpee:exposition: |-
    # User Input:
    To keep everyone in the office fuelled and focused, an automatic
    coffee monitoring system ensures you're never left without your
    daily brew.
    First, [... additional text]
  time:timestamp: '2025-06-27T09:17:34.067737+02:00'
--- [... additional events]
--- event:
  [...]
  cpee:exposition: |-
    %% Output Intermediate
    graph LR
      0:startevent:((startevent)) --> 1:task:(Check Coffee Refill)
      1:task:(Check Coffee Refill) --> 2:exclusivegateway:{x}
      [... additional mermaid code]
      4:task:(Confirm Coffee Preferences) --> 5:endevent:(((endevent)))
  time:timestamp: '2025-06-27T09:17:34.068731+02:00'
--- event:
  [...]
  cpee:exposition: |
    <!-- Output CPEE-Tree in XML-->
    <description xmlns="http://cpee.org/ns/description/1.0">
      [... model in RPST format]
```

(a) Model to Text Transformation and Additional Information for Execution

(b) Log of Process Modeling

Figure 5: Possible Subsequent Steps

graphical programming/execution language). This is shown in Fig. 5a Ⓑ.

- **Analyze how the process model has been created** based on information included in the log: User Input ①, Current Model ③ (CPEE-Tree representation), Selected LLM ②, Mermaid representation of Current Model (i.e., output of ⓑ), Mermaid representation of New Model (i.e., output of ⓒ), and New Model ④ (i.e., output from ⓓ) in CPEE-Tree representation (see Fig. 4). A shortened log is shown in Fig. 5b (extended log: https://autobpmn.ai/log_example.xes.yaml).

3. Maturity

Scientific maturity: The underlying concepts have been published in several outlets, i.e., text2model in [5], model2text in [11], conversational process modeling in [8, 6], and process automation and execution in CPEE in, e.g., [1].

CPEE maturity: The underlying process cloud execution engine CPEE is an open source engine; the engine and connected components have been downloaded 1.677.929 times so far[3]. CPEE has been developed into orchestration framework centurio.work which implements and operatively runs 16 different processes at 7 manufacturing companies [12].

Maturity of Conversational Process Modeling: The concepts of text2model transformations have been developed in the BPAIS[4] project funded by SAP Signavio; text2process transformation has become part of the modeling capabilities of the AI-assisted process modeler of SAP Signavio[5]. Moreover, we have conducted two user studies: the first one was conducted to evaluate text2model concepts with 40 participants with different experience levels in graphical modeling, e.g., UML, ER, BPMN, and

[3] https://rubygems.org/profiles/eTM, accessed 2025-06-25

[4] https://www.cs.cit.tum.de/bpm/projects/bpais/, accessed 2025-06-25

[5] https://news.sap.com/2025/03/sap-signavio-launches-ai-process-modeler-text-to-process/, accessed 2025-06-25

programming skills [13]. The second user study assesses the concepts of conversational process redesign with 64 participants. Finally, we are currently conducing a lab course at the Technical University of Munich on conversational process modeling and automation.

4. Videos and Tool

The tool is available at autobpmn.ai, where a new empty process instance can be created. For each instance the process model can be created/redesigned as described in Sect. 2. The user interface is realized by creating a new spin[6] of a CPEE cockpit[7] to create and redesign models using Conversational Modeling. The LLM Service which supports the Conversational Modeling outlined in blue in Fig. 4 is also available on github[8]. A video demonstrating the functionality of the tool is available at autobpmn.ai.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

[1] S. Rinderle-Ma, J. Mangler, D. Ritter, Fundamentals of Information Systems Interoperability, Springer, 2024.

[2] S. Schüler, S. Alpers, State of the art: Automatic generation of business process models, in: BPM 2023 International Workshops, Utrecht, The Netherlands, September 11-15, 2023, Revised Selected Papers, volume 492 of *Lecture Notes in Business Information Processing*, Springer, 2023, pp. 161–173.

[3] H. Kourani, A. Berti, D. Schuster, W. M. van der Aalst, Promoai: Process modeling with generative AI, in: Proceedings of the Thirty-ThirdInternational Joint Conference on Artificial Intelligence, IJCAI-2024, International Joint Conferences on Artificial Intelligence Organization, 2024.

[4] J. Köpke, A. Safan, Introducing the bpmn-chatbot for efficient llm-based process modeling, in: BPM Dempos, 2024, pp. 86–90. URL: https://ceur-ws.org/Vol-3758/paper-15.pdf.

[5] N. Klievtsova, J. Benzin, T. Kampik, J. Mangler, S. Rinderle-Ma, Conversational process modelling: State of the art, applications, and implications in practice, in: Business Process Management Forum, 2023, pp. 319–336. doi:10.1007/978-3-031-41623-1_19.

[6] N. Klievtsova, T. Kampik, J. Mangler, S. Rinderle-Ma, Conversational process model redesign, arXiv preprint arXiv:2505.05453 (2025).

[7] J. Claes, I. Vanderfeesten, J. Pinggera, H. A. Reijers, B. Weber, G. Poels, A visual analysis of the process of process modeling, Inf. Syst. E Bus. Manag. 13 (2015) 147–190.

[8] N. Klievtsova, T. Kampik, J. Mangler, S. Rinderle-Ma, Conversationally actionable process model creation, in: Cooperative Information Systems, 2024, pp. 39–55.

[9] B. Weber, M. Reichert, S. Rinderle-Ma, Change patterns and change support features - enhancing flexibility in process-aware information systems, Data Knowl. Eng. 66 (2008) 438–466.

[10] N. Klievtsova, J. Mangler, T. Kampik, J. Benzin, S. Rinderle-Ma, How can generative AI empower domain experts in creating process models?, in: 19th International Conference on Wirtschaftsinformatik (WI 2024), Würzburg, Germany, September 16-19, 2024, Proceedings, AISeL, 2024, p. 66.

[11] N. Klievtsova, J. Mangler, T. Kampik, S. Rinderle-Ma, Utilizing process models in the requirements engineering process through model2text transformation, in: RE, 2024, pp. 205–217.

[12] S. Rinderle-Ma, J. Mangler, Process automation and process mining in manufacturing, in: Business Process Management, 2021, pp. 3–14.

[13] N. Klievtsova, J. Benzin, J. Mangler, T. Kampik, S. Rinderle-Ma, Process modeler vs. chatbot: Is generative AI taking over process modeling?, in: Process Mining Workshops, 2024, pp. 637–649.

[6] https://github.com/etm/CPEE/blob/cpee2/cockpit/only_llm.html, accessed 2025-06-26

[7] https://cpee.org/, accessed 2025-06-26

[8] https://github.com/etm/cpee-llm, accessed 2025-06-26