JOT JOURNAL OF OBJECT TECHNOLOGY

Journal of Object Technology | RESEARCH ARTICLE

Enhancing Production Workflows by Leveraging BPMN to Model Inconsistencies — An Experience Report

Thomas Weber[†], Niklas D. Kuder[*,†], Thomas A. Völk[§], Jonas Schneider[*], Sebastian Weber[‡], and Anne Koziolek[†]

[*]Daedalus GmbH, Germany
[†]KASTEL, Karlsruhe Institute of Technology, Germany
[‡]FZI Research Center for Information Technology, Germany
[§]IPEK, Karlsruhe Institute of Technology, Germany

ABSTRACT The complexity of industrial workflows often necessitates the use of multiple models to adequately describe systems at various levels of abstraction. However, as the number of models involved in a process increases, so too does the potential for inconsistencies between models. These inconsistencies can impede workflow completion. Despite existing methods to manage workflows, there is a lack of approaches specifically addressing the modeling of inconsistencies between different models. In this paper, we propose a method to model inconsistencies between models, using specific event types within BPMN 2.0 diagrams, leveraging existing extensions to BPMN 2.0. We applied and evaluated this method in the manufacturing industry, specifically at a precision component manufacturer. We conducted nine expert interviews, and the insights we gained have helped us to identify 13 potential inconsistencies between the models used by the manufacturer. In our evaluation, the proposed modeling approach achieved a good level of usability. Our experience report adds real-world application to this practical solution for managing inconsistencies in industrial workflows. In this way, we contribute both to the field *and* to the practice of model-based engineering.

KEYWORDS Modeling methodologies, Inconsistency Modeling, Precision Manufacturing, Workflow Modeling, BPMN

1. Introduction

In the realm of manufacturing, e.g., for cyber-physical systems, a multitude of processes span across different entities, introducing the potential for inconsistencies, i.e., contradictory information, both within and between these processes. An inconsistency is thus one threat to the successful completion of a workflow, emerging from differences in two models, such as technical drawings or computer-aided design (CAD) models, describing overlapping parts of the system to be built. We want to model the possibility and the extent of inconsistencies between these models, in the workflows in which they are used. Different models are used to model a system, adhering to different paradigms. Inconsistencies between them should be addressed, because the existence of inconsistencies is a heuristic for eventual correctness (David et al. 2023). To comprehensively address these inconsistencies, we must first identify and conceptualize them.

The primary objective of this research is to designate a suitable modeling approach for modeling inconsistencies within process descriptions. This modeling approach should possess two attributes: machine-processability to enable the development of tools for ensuring artifact consistency, and human comprehensibility to facilitate communication about inconsistencies among stakeholders.

For practitioners, the communication about inconsistencies is the first and central part in resolving them. Existing research (Feldmann et al. 2015; Marchezan et al. 2023; Klare et al. 2021) is concerned with consistency management for specific use cases, instead of a modeling approach designed to be used by domain experts. Since recent studies have shown that communication is a big challenge in interdisciplinary manage-

JOT reference format:
Thomas Weber, Niklas D. Kuder, Thomas A. Völk, Jonas Schneider, Sebastian Weber, and Anne Koziolek. *Enhancing Production Workflows by Leveraging BPMN to Model Inconsistencies — An Experience Report.* Journal of Object Technology. Vol. 24, No. 2, 2025.
http://dx.doi.org/10.5381/jot.2025.24.2.a2

ment of inconsistency in practice (Albers et al. 2024, n.d.), we prioritize the communication about inconsistencies. Successful communication among stakeholders enables consistency management approaches for general use cases, e.g., in production engineering in advanced systems like cyber-physical systems. Thus, we chose a modeling approach designed for the communication between stakeholders from different disciplines and used existing extensions of that modeling approach in order to make the application of our proposed modeling as easy as possible with existing tools. Our decision, therefore, for Business Process Model and Notation 2.0 (BPMN)[1] (Brooke 1996), was motivated by the capacity of this modeling approach both to provide human comprehensibility for communication with stakeholders and to allow for machine-processability due to its formalization.

We evaluated the proposed modeling approach with an investigation into a real world domain, the contract manufacturing, thereby illuminating the potential sources of inconsistencies within them. Employing the selected modeling approach, it will enable the augmentation of existing mechanisms designed for consistency maintenance. Notably, this research is conducted in collaboration with a precision component contract manufacturer. The processes and inconsistencies are investigated through interviews conducted with manufacturer employees. This leads us to our research question:

RQ How can we model inconsistencies in a BPMN diagram?

To answer our research question, we first investigate and model the workflows at our industry partner, present our proposed approach to model inconsistencies, apply it to the workflows at the industry partner and evaluate it. Thus, our contributions are:

C1 BPMN models for all found workflows at a precision component contract manufacturer

C2 Modeling approach for inconsistencies in BPMN models

C3 Application of our modeling approach in a real world example

C4 Evaluation of our proposed modeling approach with a System Usability Scale (SUS) (Brooke 1996)

The SUS was conducted with experts at Daedalus GmbH with a score of 75. This illustrates its usability but also shows potential for improvement through discussion in the community.

We motivate our research in section 2 and explain our modeling approach choice of BPMN in subsection 2.2. The methodology for interviews we used is detailed in section 3. The results of the interviews, i.e., workflows and inconsistencies, are presented in section 4. Afterward, we discuss our evaluation in section 5, threats to the validity in section 6 and related work in section 7. We conclude our paper with a discussion in section 8, an outlook on future plans in section 9, and our conclusion in section 11.

[1] https://www.bpmn.org/

2. Motivation

Workflow diagrams are used to model, analyze and communicate workflows in companies (Aguilar-Savén 2004). They can be used to improve efficiency or to train new employees. There are many reasons for reduced efficiency of workflows, such as long waiting times for required raw materials. A non-obvious reason for reduced efficiency is inconsistencies that make it difficult or even impossible to complete a workflow successfully. Since inconsistencies may only be detected and understood with knowledge of a domain, communication with domain experts about inconsistencies is essential. Consistency is also pointed out as a current challenge by (Van Der Straeten et al. 2009).

Knowledge about inconsistencies is often implicit and only known to staff members if they either experienced it or have been informed about it by colleagues. To prevent the impact of the inconsistency, it should be explicitly specified in the workflow to ensure it is known to everyone executing the workflow. To help practitioners formulate and model the inconsistencies they know, we used BPMN, a well-known notation, and threats, a well-known analogy, both explained in subsection 2.2.

2.1. Motivating Example

Contract manufacturers do not always cover the whole production process for the customer. In manufacturing, the production process typically involves several stages, including design, material selection, machining, assembly, and surface treatment. The product to manufacture is described by a technical drawing. Each stage requires specific expertise and equipment, which is why some customers prefer to handle certain stages in-house or through specialized third-party providers. For instance, a customer might prefer to have the surface treatment, such as anodizing (oxidizing to protect metals from corrosion), carried out by their in-house anodizer. Anodizing is a crucial step in enhancing the durability and appearance of metal parts, and it requires precise control over the chemical processes involved.

In such cases, the customer typically does not create a new technical drawing. Instead, they provide the contract manufacturer with the original technical drawing that describes the end product, which includes the anodizing step. This drawing serves as a blueprint for the entire production process, detailing dimensions, materials, and finishing requirements. This leads to a situation where the technical drawing states that the part must be anodized, even though the customer has explicitly instructed the contract manufacturer not to perform this step. Once this information is entered into the contract manufacturer's internal system, a discrepancy arises: the definition of the part in the system conflicts with the technical drawing provided by the customer. The emergence of such an inconsistency is shown in Figure 1. The inconsistency is not visible in the workflow itself (for that reason, Figure 1 already contains the explicit specification of the inconsistency with our approach).

2.2. Choice of Modeling Approach

White et al. (White et al. 2004) compare the representation types of BPMN and UML activity diagrams on the basis of 21 workflow patterns with respect to their readability and the ability to represent these patterns. The authors conclude, that

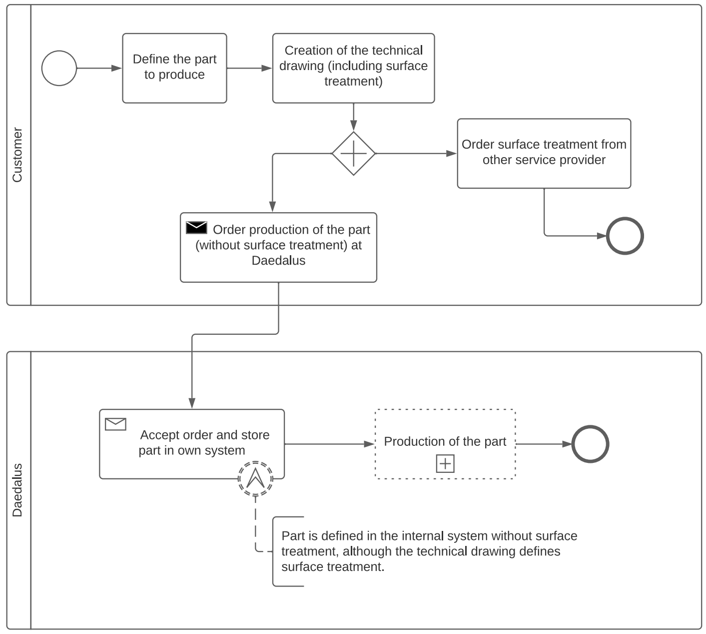

Figure 1 Exemplary workflow in which the inconsistency is non-obvious. The inconsistency is modeled as a non-interrupting escalation event, as proposed by our modeling approach. This workflow diagram describes the *Sales / Quoting* sub-process from Figure 3. The sub-activity *Production of the part* refers to the remainder of the overall workflow at Daedalus and its service providers, as illustrated in Figure 3.

activity diagrams and BPMN diagrams are very similar, but the two types of representations have different target groups and therefore the notation, among other things, also differs slightly. An example of this is the *start node* in UML compared to the *start event* in BPMN. The latter is relevant for the decision of the modeling approach in this paper, since we operate in the context of the manufacturing industry. In particular, this means that the stakeholders are not software developers, but rather domain experts in precision manufacturing.

Barker et al. (Barker & van Hemert 2008) provide a concise survey of existing workflow technology. Two important conclusions in their paper are, firstly, that we should not develop yet another workflow language because there are already many well-supported frameworks. Secondly, the used workflow language must be tailored to the domain under study, instead of being built by computer scientists for computer scientists. Because of both conclusions and our stakeholders being domain experts in precision manufacturing, we chose BPMN.

Since not only the processes, but also possible inconsistencies, which can arise in them, are to be represented, a further important argument for the use of the BPMN diagrams is the possibility of representing these inconsistencies. Meland and Gjære (Meland & Gjære 2012) show different possibilities to represent so-called *threats* in BPMN. A *threat* is a threat to the successful completion of a workflow. We can interpret the identified inconsistencies as such *threats* in order to represent them.

An event is represented by a circle in BPMN. Threats are modeled as *event types*. There are three events, which are represented in the upper half of Figure 2. *Start*-events start a process, *intermediate*-events take place in the middle of a process, and *end*-events end a process. Additionally, each kind of event can belong to an *event type*, e.g., *escalation* or *error*. For this work, especially the event type is relevant, because it is used to model the occurrence of inconsistencies. Error events are always interrupting, i.e., they require the immediate termination of the currently running process. Because the occurrence of an inconsistency does not require the immediate termination of the running process, the representation by *intermediate non-interrupting escalation events* is best suited.

We use non-interrupting *escalation events* to model inconsistencies that are tolerated in the workflow, i.e., that do not cause the workflow to fail if not resolved. The existence of inconsistencies is tolerated, as their removal imposes other challenges. The employees performing the workflow have to keep the inconsistencies in mind nevertheless. Interrupting escalation events are used to model inconsistencies that have to be resolved, otherwise the workflow would fail, but they do not have to be solved immediately. A workflow diagram containing an interrupting escalation event can thus be completed without abandoning its execution, but the inconsistency has to be resolved first. In our interviews, we found no inconsistencies that were unresolvable and thus required the abandonment of a workflow.

3. Elicitation Methodology

We designed our interviews using several guidelines (Fowler 1995; Taherdoost 2022; Marshall 2005), especially Fowler (Fowler 1995), who describes how interviews can be formulated effectively. Among other things, he mentions the following important aspects:

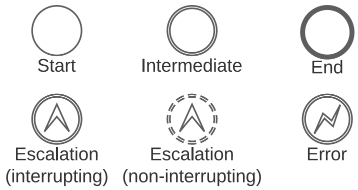

Figure 2 Different event types in BPMN.

Questions should be clear and precise In particular, no technical jargon should be used, but only "simple language". In this way, it can be achieved that all respondents understand the question and do not give an actually wrong answer out of uncertainty. This is especially relevant for our case, because we are interviewing experts of different domains.

Carefully consider preceding questions If several questions are asked consecutively in an interview, a preceding question can cause a bias on the subsequent questions. To avoid this, all questions should be asked as neutrally as possible.

Test questions before the interviews By testing the questions, possible problems with the wording, formatting, or structure of the questions are identified in time so that they do not affect the actual interview. This ensures that the questions are effective and provide reliable data.

Interviews We conducted three rounds of interviews. The first round was about identifying inconsistencies, using previously modeled BPMN workflows, prepared by us. The interviews in the first round did not include any inconsistencies and thus also no mention of our proposed modeling approach. Afterward, we used the information collected to model the inconsistencies in the BPMN diagram modeling the workflow the inconsistency may arise. These extended BPMN diagrams were used in the second round of interviews, which was concerned with the likelihood of inconsistencies occurring. The third round of interviews was a SUS (Brooke 1996) to assess the usefulness of our modeling approach, using selected inconsistencies modeled with our proposed modeling approach. The interviewees thus provided the inconsistency description, the likelihood of occurrence and their opinion about the modeling approach as result of the SUS. For the first two rounds of interviews, we conducted preliminary studies.

In order to obtain the best possible results in the expert interviews, we adhered to the aforementioned principles when conducting the interviews. Prior to the actual interviews, the interview questions were tested on a subset of the participants. In addition to the actual interview questions, the participants of the preliminary study received a meta-questionnaire, which was used to adjust the interview questions before the actual interviews. In the meta-questionnaire, the participants of the preliminary study gave an answer to the following three statements:

1. the questions were asked in an understandable manner and did not use technical jargon
2. the questions were asked in a neutral way and the previous questions did not cause bias on subsequent questions
3. a clear and unambiguous definition of the construct under study was provided.

Each of these statements could be answered with a scale as proposed in (Fowler 1995):

– Completely agree
– Generally agree
– Generally disagree
– Completely disagree

After initial interviews, it became apparent that a maximum of nine participants would be available for the questionnaires. This number resulted from the fact that no Daedalus employees should be interviewed who do not have a sufficient overview of the production process and insights in emerging inconsistencies. Of the selected participants, each had a rough overview of the overall process and was an expert in their respective area of expertise.

The results of the meta-questionnairs of the preliminary study were very good, and thus we decided to not modify the interview questions and include the results of the preliminary study into our final results. The first question "The questions were asked in an understandable way and not too much technical jargon was used." was answered "completely agree" 3 out of 3 times, the second question "The questions were asked neutrally, and the preceding questions did not cause bias on subsequent questions." was answered "completely agree" 2 times and "generally agree" once out of 3 times, and the last question "A clear and unambiguous definition of the construct under study was presented." was answered "completely agree" 3 out of 3 times. Thus, we decided to use the results of these first interviews, although they were part of the preliminary study, because the interview questions were answered independently and before the meta-questionnaire, and we did not change the interview questions for the other participants.

4. Results

The entire manufacturing process of Daedalus GmbH was divided into sub-processes. The description and modeling of these processes is our first contribution C1. Subsequently, 13 possible inconsistencies were identified on the basis of nine expert interviews. The inconsistencies are described in subsection 4.2 and were modeled using BPMN. The inconsistencies constitute our third contribution C3 and can also be found in our replication package (*Replication Package* 2024). Additionally, we identified 6 dimensions of inconsistencies.

4.1. Workflows

The entire process, with the details abstracted away by the sub-processes, is shown in Figure 3. The main process and the sub-processes are explained in the following and are our first contribution C1.

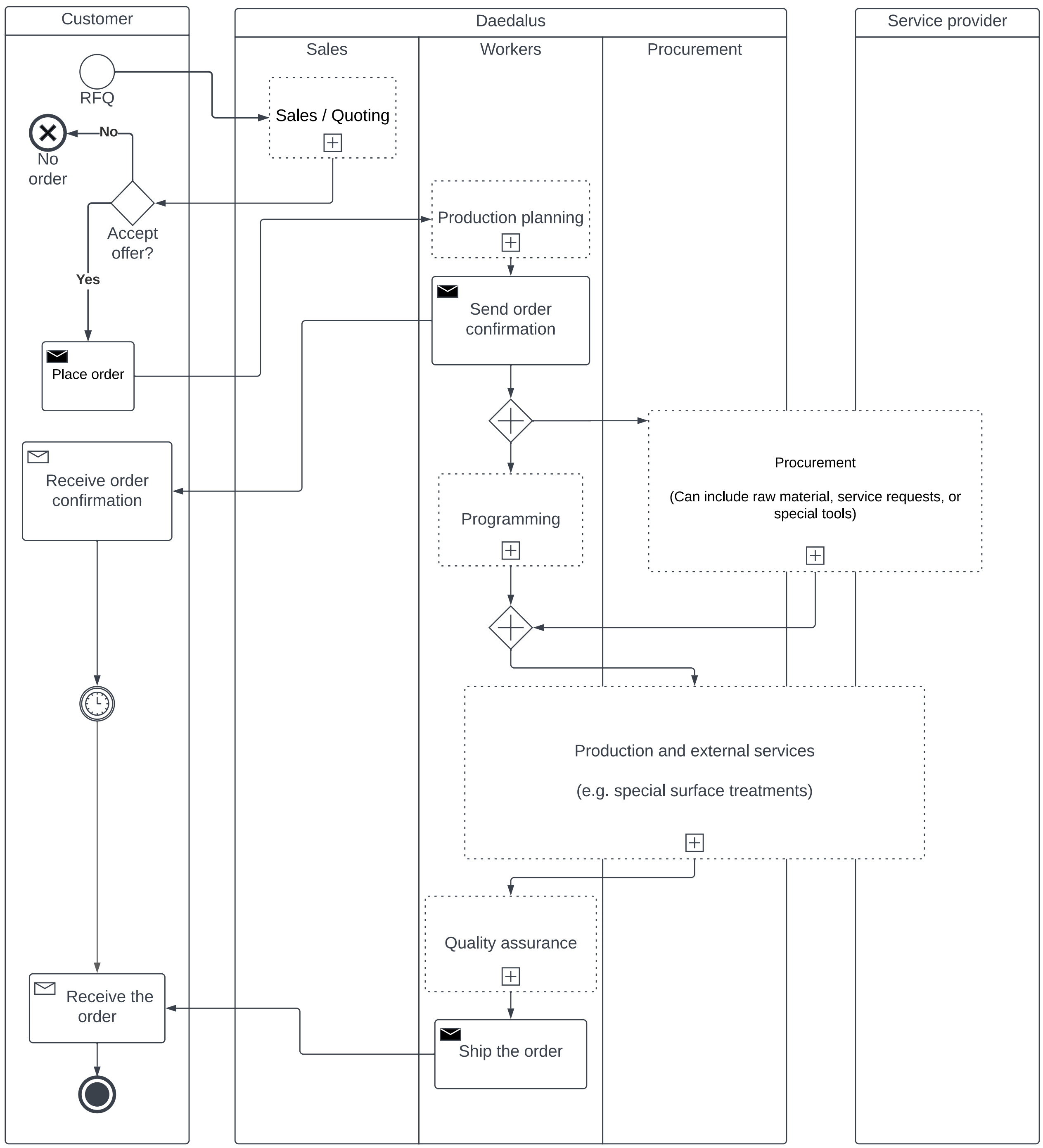

Figure 3 Overview of the workflow in precision manufacturing. The inconsistencies occur in the different subprocesses indicated by a dotted line and a plus sign in the task. The BPMN 2.0 diagrams of the subprocesses are available in our replication package (*Replication Package* 2024).

Main process The process starts with a Request For Quote (RFQ) from the customer. This request triggers the *Sales / Quoting* process at Daedalus, where the offer is reviewed and prices are defined. If the customer does not accept this offer, the process ends immediately. If an order is placed, the *Production planning* process is triggered. Following the completion of the *Production planning*, Daedalus *Send[s an] order confirmation*. The customer *Receive[s this] order confirmation*. After that, two processes are executed in parallel. On the one hand, required materials, special tools or external services are organized and purchased, and on the other hand, a programmer is assigned to write the Numerical Control (NC) program code required for the production of the ordered parts. After these tasks are done, the sub-process *Production and external services* can be started. If the order requires external services, these are applied here. After the parts have been completely manufactured and machined, *Quality assurance* can begin, and the order can be sent to the customer.

4.1.1. Sales / Quoting

The aim of the quoting process is to provide the customer with an offer for their order. Among other things, an initial definition for the desired component is saved here. Subsequent processes can lead to inconsistencies with this definition. The quoting process is divided into two subtasks: Quote Creation and Pricing.

Quote Creation In the Quote Creation task, the customer's order is created at Daedalus, but initially without prices. In addition, the specifications of the parts ordered by the customer are defined. This includes details such as the desired material, tolerances or additional surface treatments. Furthermore, the documents provided by the customer (technical drawing and CAD file) are uploaded to Daedalus' in-house system for each part of the order.

Pricing Once the parts have been defined, they are priced in the pricing task. Here, the worker has the opportunity to provide initial estimates of the required machines, production times and purchase prices in order to estimate the prices more accurately. This data can also be used later in production planning as a guide to create the production plan.

4.1.2. Production planning

The task of production planning is to provide each component ordered by the customer with a kind of recipe. This recipe is also called a production path, as it shows the exact route of each part through the factory. In addition, the production of the desired amount of each ordered part can be divided into several lots. Each lot can then be provided with its own quantity and a desired delivery date. It is possible that during this process, the production planner notices that information essential for production is missing. In this case, the customer must be consulted and the necessary information supplied. As a result, the original model of the component may have to be adapted.

Production path and lots When a customer orders 1200 components of the same type from Daedalus, but wants 100 components delivered each month over the course of a year, the order is split into lots. In this case, the order only consists of this one part, but it must be split up internally so that there are no unnecessary blockages on machines or avoidable storage costs. For this order, 12 lots are therefore created, each with a quantity of 100, in order to produce the desired order of 1200 components. In addition, each lot is given its own delivery date, which can then be taken into account when scheduling production. The production path can be defined independently for each lot so that it is possible to react to any exceptions, such as insufficient production capacity. In such a case, for example, for an order of 100 components, 80 components could be sent through the company's own factory and 20 components could be manufactured by outsourcing to a partner company. It is therefore possible to define several production paths for one component.

Needs All necessary purchases (needs) are defined in production planning. These can be divided into material and service needs. The planning employee creates at least one material requirement during production planning and can also define various external steps individually for each lot within the production path, which are then automatically converted into service needs and processed in the procurement process. Such external steps are processes that the component has to go through outside the Daedalus production system. These can be, for example, surface treatments or special drillings that are to be carried out by partner companies. A service need in this case is the need for consultation with these partner companies. As soon as the orders have been confirmed with the partner companies, the service need is covered.

4.1.3. Procurement

During the procurement process, an employee looks for an open need and calls partner companies to fill that need. Such a need can be, for example, the material, the necessary external services, or special tools. After that, results (e.g., price, delivery date) are stored in the internal management system in the form of a purchase order.

4.1.4. Programming

The aim of this process is to write the NC program code for a component. The employee proceeds as follows: He reads the data stored up to that point about the component and successively develops the NC program code based on this. This is usually done iteratively. For each individual part of the program, the code is first written, then a machine is set up for testing and finally the part of the NC program code is tested on the machine. This process is repeated until the employee has developed the complete NC program code for the component. When the program is ready, another document, the setup sheet, is generated. In this setup sheet, the NC programmer records all the information required to set up a machine for this program so that any employee can set up the machine and start production. The employee then publishes the developed NC program and the setup sheet in a bundle within the company's internal system.

The process is completed as soon as the first good part has been manufactured. A good part here is a produced component that meets all quality requirements. As a final step, the NC programmer sets up a machine using the setup sheet and runs the developed NC program. Once the first part has been successfully manufactured, the process is complete. However, if the programmer notices any flaws, he adjusts the NC program and, if necessary, the setup sheet. In this process, the role of an NC programmer can also be interpreted as that of a project manager. The NC programmer has end-to-end responsibility for the program he has developed and can also make further changes after the first good part has been produced, which then increase the efficiency of the program. At any point in the process, the programmer may need to consult with the customer again.

4.1.5. Production and external services

This process cannot be clearly defined as it differs depending on the component. Some components are manufactured exclusively by Daedalus and sent directly to the customer. Other components require special drilling or surface treatment that cannot be carried out by Daedalus, or are not manufactured by Daedalus at all, but are sent to a partner company as a complete order. As a rule, however, the process contains the manufacturing and external services steps.

Manufacturing The NC program code (subsubsection 4.1.4) for the component is loaded onto the machine and the material ordered during the procurement process (subsubsection 4.1.3) is placed in the machine. In this way, one component after the other is manufactured for a specific lot.

External services If an external service is required during or after the production of a component, the corresponding component is sent to the partner company. Depending on the production path, it is then sent to another service provider or back to Daedalus.

4.1.6. Quality assurance

The purpose of this process is the inspection of the manufactured parts. The individual features of the part are measured, and an inspection report is created. If all features of the part are within the specified tolerances, it is sent to the customer together with the inspection report.

4.2. Inconsistencies

The inconsistencies found during the interviews are listed and explained in this section. To identify the inconsistencies, we first investigated the workflows and modeled them with BPMN. Afterward, we asked the interviewees in A summary of the inconsistencies and the processes in which they can arise is shown in Table 1.

4.2.1. Specification change after consultation

There may be a reason why the definition of the part provided by the customer cannot be accepted. Such reasons may be, for example:

- It is physically impossible to manufacture the part according to the customer's requirements
- The documents provided by the customer (technical drawing and CAD file) are already inconsistent
- The technical drawing is inconsistent in itself

In such a case, the customer is consulted and the specification of the component may be adapted accordingly. However, the customer rarely sends updated documents, but instead explains in a conversation or email what exactly needs to be changed on the part. As soon as this information is stored as a note, the definition of the part stored in the system is no longer consistent with the technical drawing or CAD file provided by the customer

4.2.2. Special requests from the customer

Daedalus does not always produce the entire part for the customer. The customer may, e.g., wish to have the surface treatment carried out by their own anodizer. In such a case, however, the customer rarely produces a new technical drawing, but instead provides Daedalus with the technical drawing that describes the end product. The technical drawing therefore states that the part is to be anodized, although Daedalus is explicitly not supposed to do this. As soon as the information is stored in Daedalus' internal system, this definition of the part is inconsistent with the technical drawing provided by the customer.

4.2.3. Transfer of data between systems

As there are no standardized interfaces between the internal systems of the individual parties, the transfer of customer data to Daedalus' internal system must sometimes be carried out manually. As a result, certain details may be overlooked or transferred incorrectly. In such a case, the internal definition of the part is inconsistent with the technical drawing provided by the customer.

4.2.4. Starting an order with incomplete design

A customer may place an order for a component that is not yet complete, e.g., due to time pressure. Production can be planned and the first NC programs written even though the part has not yet been fully designed. In such a case, the specification of the part may change slightly during planning. This may lead to an inconsistency between the definition of the part stored in the internal system and the updated technical drawing provided by the customer.

4.2.5. Incorrect definition of the production path

The production path is a sequence of individual processing steps that is defined for each part during production planning. As production planning is mainly carried out by domain experts, human errors can also occur here. For example, it is possible that the planning employee overlooks the specification of a surface treatment on the technical drawing. In this case, a production path is created without a surface treatment step, even though the part requires such a treatment.

4.2.6. Different raw material and color definitions

The raw material to be ordered can be identified in three different ways:

- Unique material number
- Unique DIN standard
- Unique description of the chemical composition

Different companies identify material in their system differently. It can therefore happen that a customer wants to have his parts manufactured from a material identified by a material number, and Daedalus stores this part with material identified by the chemical composition or the DIN standard in its own system. This can lead to structural inconsistencies, but these do not result in semantic inconsistencies. However, there are still employees in various companies who have been working in the industry

	4.1.1 Sales / Quoting	4.1.2 Production planning	4.1.3 Procurement	4.1.4 Programming	4.1.5 Production and external services	4.1.6 Quality assurance
4.2.1 Specification change after consultation	X	X		X		
4.2.2 Special requests from the customer	X					
4.2.3 Transfer of data between systems	X					
4.2.4 Starting an order with incomplete design	X					
4.2.5 Incorrect definition of the production path		X				
4.2.6 Different raw material and color definitions			X			
4.2.7 Information on raw material dimensions			X			
4.2.8 Linking raw material and tool orders			X			
4.2.9 Human error of the programmer				X		
4.2.10 Programming based on derivation of the technical drawing	X			X		
4.2.11 Features not specified				X		
4.2.12 Technical drawing to external service providers			X		X	
4.2.13 Creating test drawings						X

Table 1 Inconsistencies (4.2.1-4.2.13) found and the processes (4.1.1-4.1.6) they stem from.

for several years and still identify the materials using outdated descriptions. In such a case, the desired material is also stored using one of the three definitions mentioned, but there is an inconsistency between the raw material defined on the technical drawing and the raw material stored in the system. This inconsistency can also occur throughout the purchasing process if Daedalus and the supplier each use a different representation of the raw material.

4.2.7. Information on raw material dimensions During production planning, the required raw material dimensions per part are specified. The purchasing employee can decide during the purchasing process whether to order the material blocks in the specified size, or whether to order the material in bars and then saw it to size in the factory. The latter can often lead to significant savings, but in this case you may find an inconsistency in the definition of the production path and the ordering details of the raw material order.

4.2.8. Linking raw material and tool orders As special materials can only be processed with special tools, it is sometimes the case that a raw material order is accompanied by a corresponding tool order. Raw material orders are also subject to extremely high price fluctuations, meaning that the material required after the customer has placed the order can cost significantly more than planned. In such a case, the customer may agree to use a cheaper alternative material. If the raw material order is then changed, there may be an inconsistency between the current raw material order and the original tool order.

4.2.9. Human error of the programmer The NC program is created as a new model of the component. Human error can cause the resulting program code to be inconsistent with the technical drawing provided. Such a human error can be, e.g., a simple typing error.

4.2.10. Programming based on derivation of the technical drawing A technical drawing can either describe the finished component or the condition of the component after milling. Knowing which condition is described is essential, as some surface treatments can add a few micrometers. If the technical drawing describes the finished component, the programmer must take this information into account and write the NC program code so that the milled component is smaller than indicated on the technical drawing. After the surface treatment has been applied, the component then has exactly the desired dimensions. In this case, an NC program is created that is inconsistent with the technical drawing (or the CAD model sent by the customer), but this is a deliberate inconsistency. None of the models are incorrect. They merely describe different states of the component.

4.2.11. Features not specified As shown in Figure 4, the user may provide incomplete technical models. It can happen that a customer specifies important features of the component (e.g., fits, threads) only in the technical drawing, although the .STEP[2] format also supports such features. In this case, it

[2] Standard for the Exchange of Product model data

can happen that the programmer overlooks these details on the technical drawing, and they therefore do not appear in the NC program code, as this is derived from the .STEP file.

4.2.12. Technical drawing to external service providers

If Daedalus orders an external service such as special deep hole drilling or surface treatments, the service provider always requires the technical drawing as a description of the part. Due to existing inconsistencies, it is possible that this technical drawing is not up-to-date. In this case, the service provider receives an outdated technical drawing, which leads to an inconsistency between the internal definition of the part at Daedalus and the definition of the part at the service provider.

4.2.13. Creating test drawings One option for quality assurance is the initial sample test report. In order to define which features of a part are to be measured for the initial sample test report, so-called inspection drawings are created as a derivation of the saved technical drawing. Features are special characteristics of the component, such as a hole, fit, or thread. However, as the technical drawing does not always contain the most up-to-date information, the inspection drawings may also be incorrect. It is possible that the inspection drawings contain the correct information and are therefore inconsistent with the enclosed technical drawing, or that the inspection drawings are merely derived from the technical drawing and are therefore inconsistent with the definition of the part stored in the system.

4.3. Classification of Inconsistencies

We identified 6 dimensions, along which the inconsistencies can be classified. Not all inconsistencies can be classified in each dimension, but the classification enhances understanding by helping stakeholders identify patterns and common causes, which leads to more effective problem-solving. This classification also facilitates targeted solutions, allowing for specific strategies to address different types of inconsistencies. Examples for the types of inconsistencies are mentioned in brackets in the descriptions.

Source of Inconsistency Inconsistencies can originate from various sources, including customer specifications, internal processes, and communication channels. For example, changes or errors in customer-provided specifications (4.2.1, 4.2.2) and manual data entry errors within the company's internal processes (4.2.3, 4.2.4, 4.2.5) are common sources. Additionally, inconsistencies can arise from informal communication methods, such as verbal instructions or emails, which may not be accurately reflected in formal documentation (4.2.1, 4.2.2).

Type of Information Inconsistencies can be classified based on the type of information involved. Technical specifications, such as dimensions, materials, or surface treatments, are common areas where inconsistencies occur (4.2.6, 4.2.7, 4.2.10). Process information, including the sequence of production steps or the definition of production paths, can also be inconsistent (4.2.5).

Communication and Documentation Inconsistencies often stem from how information is communicated and documented. Verbal or informal communication can lead to discrepancies if not properly documented (4.2.1, 4.2.2). Formal documentation errors, such as outdated technical drawings or incorrect inspection drawings, also contribute to inconsistencies (4.2.11, 4.2.12, 4.2.13).

Human Factors Human errors are a significant source of inconsistencies. These can occur during data entry, programming, or planning (4.2.5, 4.2.9). Variations in how experienced employees interpret and apply specifications can also lead to inconsistencies (4.2.6).

Intentional vs. Unintentional Inconsistencies can be either intentional or unintentional. Intentional inconsistencies are deliberate deviations from specifications to achieve a desired outcome, such as adjustments for surface treatments (4.2.10). Unintentional inconsistencies, on the other hand, are unplanned discrepancies due to errors or miscommunications (4.2.3, 4.2.9).

Tolerance of Inconsistency Not all inconsistencies are resolved during the production process; some are tolerated. Tolerating an inconsistency usually involves annotating the information with the inconsistency or additional information, with which the effects of the inconsistency can be mitigated. For example, certain inconsistencies are tolerated if they do not significantly impact the final product (4.2.1, 4.2.6, 4.2.7, 4.2.10).

5. Evaluation

In this section, we evaluate our RQ, How can we model inconsistencies in a BPMN diagram? We place the focus here on the applicability and the usability of our proposed modeling approach. To address our RQ, we proposed a modeling approach for inconsistencies in BPMN diagrams (C2). We applied the approach at the industry partner to evaluate its applicability (C3). Our inconsistency modeling extended the general modeling of the workflows at the industry partner (C1). We used an SUS (Brooke 1996) to evaluate the usability of our modeling approach (C4).

Applicability Our proposed modeling approach was applicable for modeling all inconsistencies elicited in the workflows of the industry partner. For the elicitation of the inconsistencies in the interviews, we did not use our modeling approach, thus preventing the effect of not modeling an inconsistency because it cannot be modeled with the tools used to elicit it. Additionally, we evaluated the completeness of our inconsistency modeling by asking the interviewees about any missed inconsistencies. The interviewees did not mention any other inconsistencies besides those already described and afterward modeled. We also asked the interviewees to estimate the probability of the occurrence of the given inconsistencies in a product manufacturing process. The averaged probabilities are shown in Figure 5, with an average over all inconsistencies of 28%. Because we used a coarse-grained scale of 25% steps, this might be an overestimation. Nonetheless, our conclusion from these results is that we modeled no superfluous inconsistencies. Therefore, we consider our list of inconsistencies to be complete, and so our modeling approach is demonstrably applicable. In this way, our approach is a valuable addition to modeling research.

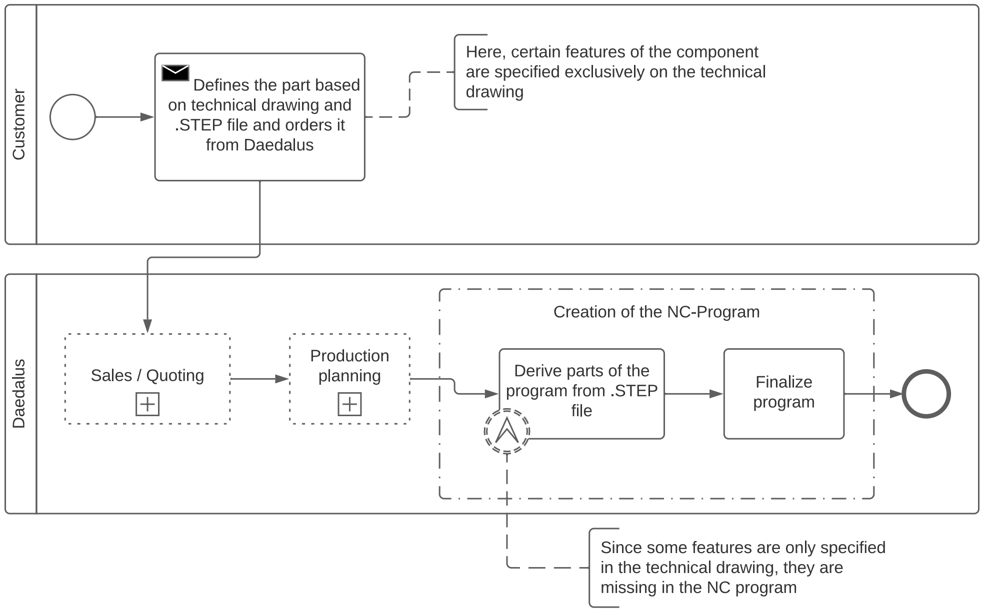

Figure 4 Example inconsistencies in the programming workflow modeled in BPMN 2.0. The sub-activity *Creation of the NC-Program* is part of the *Programming* sub-process, which follows the *Production planning*. The overall workflow is illustrated in Figure 3.

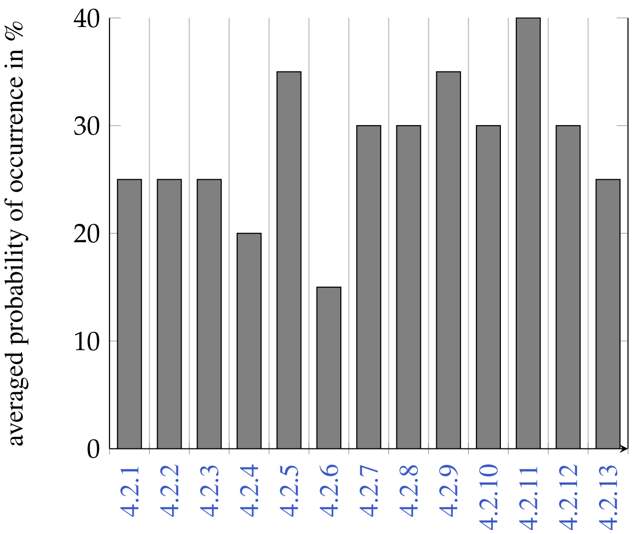

Figure 5 Averaged probability of occurrence of the inconsistencies (steps of 5%).

Usability An important aspect for modeling with domain experts is user-friendliness when new methods and tools are introduced. User-friendliness covers both the ease-to-use and the ease-to-understand. In this way, user-friendliness is included under the broader property of usability. Therefore, we decided to evaluate our modeling approach for its usability. To this end, we used an SUS. Further, we made the result of the SUS into our contribution C4, because we want to emphasize its use for the current evaluation as well as its use for future evaluations, especially as a satisfactorily comparable evaluation method for other approaches. SUSs are widely used to evaluate the usability of systems (Bangor et al. 2008), so we hope to initiate a fruitful discussion in the community by using the evaluation method., Furthermore, we plan to expand our evaluation or our modeling approach by applying it to describe more inconsistencies in different application domains as described in section 9.

The SUS consists of ten statements that users rate on a scale from 1 (strongly disagree) to 5 (strongly agree). The statements cover various aspects of usability, such as the complexity of the system, the ease of use, and the user's confidence in using the system. The SUS score is calculated by converting the raw scores and then summing them to produce a single score ranging from 0 to 100, where higher scores indicate better usability (Brooke 1996). The questionnaire for the SUS contains a short explanation of the concept of BPMN as well as of our proposed modeling approach for inconsistencies. We added to this two examples, i.e., the inconsistencies described in subsubsection 4.2.10 and subsubsection 4.2.9. The SUS was performed on a total of 7 participants and achieved a score of 75, which is a *good* result. The small number of participants is due to the limited number of Daedalus employees with suitable overview knowledge of inconsistencies occurring in the modeling of workflows. We plan to evaluate our modeling approach further in the context of bigger companies. Thus, we conclude, that the presented modeling approach serves as a promising way to model inconsistencies between models.

Closing remarks The practicality and positive reception of the chosen notation show a positive step forward. Nevertheless, improvements are still necessary. A notable obstacle lies in the inability to directly compare our modeling approach to existing modeling approaches because, to the best of our knowledge, there are none. Nonetheless, the work in this study exhibits potential and establishes a groundwork upon which future enhancements can be made, aiming to improve the efficacy and user-friendliness of the proposed modeling approach.

6. Threats to Validity

In this section, we discuss threats to validity (Runeson & Höst 2009).

Construct Validity We built our contributions partially using our other contributions. The inconsistencies, elicited in our second interview phase, are modeled based on the workflows, which we elicited in the first interview phase. To counteract that, we designed the interviews we conducted based on several guidelines (Fowler 1995; Taherdoost 2022; Marshall 2005) (cf. Section 3) to address this threat. To avoid bias because of the ordering, we used a preliminary study to eliminate possible sequence effects, i.e., the influence of the order of the questions on the answers of the interviewees. To mitigate threats to our usability evaluation, we chose the established and commonly used metric of a SUS. While we think that avoiding a sequence bias is possible for the inconsistency elicitation, the data acquisition regarding the probability of occurrence would be affected by the ordering of the inconsistencies. To avoid this effect, we randomized the order of questions for the probability of occurrence. The SUS contained examples chosen from the inconsistencies elicited in the second phase, and thus our selection of inconsistencies could affect the results of the SUS. The threats we faced are inherent to work with a mid-size company, because the number of employees knowledgeable enough about the processes and inconsistencies is limited. In future work, we plan to collaborate with bigger companies and perform a comparative study where one group elicits the inconsistencies and the other one uses the modeling approach.

Internal Validity The interviewees have not identified any further workflows or inconsistencies in the interviews. For now, we assume that our workflows and our inconsistencies are complete. However, to reduce bias towards a single domain, we want to increase the number of participants and domains in future work. Our evaluation of the probability of occurrence of the inconsistencies used coarse-grained steps, which might have resulted in an overestimation.

We selected two inconsistencies modeled with our modeling approach to present in the SUS, which could influence the results of the SUS. We want to mitigate this risk in future work by choosing multiple examples. Additionally, the participants were in part not familiar with BPMN, which could influence the results. Furthermore, a SUS does not measure the actual but the perceived success (Drew et al. 2018). With the limited scope of our study, we could not evaluate this distinction between actual and perceived success, because we did not evaluate whether the modeled inconsistencies improved the interviewees knowledge about inconsistencies in the workflows. We plan to conduct further studies in future work to evaluate the distinction between actual and perceived success.

External Validity Firstly, the limited number of participants raises concerns about the generalizability of the findings, as the sample size might not be representative of the broader workforce in this industry. Secondly, the exclusive focus on a single company restricts the external validity of the study. Despite these limitations, the study offers valuable insights within its scope. We plan to mitigate these risks by applying our method in studies with more participants and different domains in future work.

Reliability We utilized several guidelines (Fowler 1995; Taherdoost 2022; Marshall 2005) (cf. Section 3) to improve the reliability of our study. Because our results are domain and to an extent also company specific, the reproducibility of the results is limited. We tried to mitigate risks regarding the interviews by using pre-studies and examples. Nonetheless, we acknowledge potential bias, such as a different understanding of inconsistencies, which impacts the reliability to some extent.

7. Related Work

To the best of our knowledge, there is no modeling approach for BPMN, that allows to model inconsistencies. We have therefore decided to include (in)consistency management in a dual role, on the one hand as motivation for our research, and on the other hand as related work.

(Zarour et al. 2020) conducted a systematic literature review on the current state of the art of BPMN extensions. The authors examined 49 extensions and compared them according to a set of criteria. The domains of the extensions include security, complexity, knowledge management, manufacturing, and event handling among others, but none of them are concerned with modeling inconsistencies between models. To the best of our knowledge, there is no modeling approach for BPMN to model inconsistencies.

(Liebel et al. 2018) conducted a survey on model-based engineering in the industry. They focused on embedded systems and identified current challenges. The main challenges are interoperability difficulties between tools, high training efforts for developers and usability issues. We also identified these challenges in the preparation of our interviews, and therefore decided to use existing and well-established tools and their extensions.

In contrast to the modeling of inconsistencies within workflows, there is research about inconsistencies between workflows. (Awadid & Nurcan 2019) performed a systematic literature review about requirements for consistency among business process models. The existence of multiple variants of the same business process introduces the possibility of inconsistencies.

Several frameworks for consistency management have been proposed so far, e.g., (Feldmann et al. 2015; Marchezan et al. 2023; Klare et al. 2021), but they are focused on the practical part of resolving inconsistencies for specific use cases, instead of their description. While their cases can be seen as a motivation of our work and improve consistency management, we focused on one hand on providing a modeling approach for inconsistencies and on the other hand we wanted to keep it independent of the domain.

The coupling between process models and textual descriptions has been researched by (van der Aa et al. 2015). As both model categories can occur in an organization, they can describe the same business process and can become inconsistent. The authors present an approach to automatically identify inconsistencies between a process model and a corresponding textual description. Their work provides an interesting related work, as their approach itself can be seen as a business process of the creation of business processes and corresponding textual descriptions and the possible inconsistencies between them can be modeled with our proposed method.

The identification of an inconsistency in a workflow is a necessary step for its correction. (Herzig et al. 2014) use graph representation for models and identify inconsistencies by means of pattern matching. The inconsistencies found may, if they cannot be eliminated, be modeled with our approach.

(Kühn et al. 2023) provide a formalized classification schema for model consistency. For future work, we plan to use their classification for the inconsistencies we found and reason about consistency types that did not occur in the interviews and whether they can occur in our scenario.

Coming from a product engineering motivation, (Albers et al. 2024) analyzed inconsistency in product development from a physical engineering perspective. While the setting does not involve the modelling of workflows, it gives preliminary insights that not only the process itself, but also the corresponding setting and the necessary development activities are important to understand the management and communication of inconsistency in CPS engineering environments.

This work is an extension of our previous work (Kuder et al. 2024), where we presented the new idea of the modeling approach in the context of software architecture. We extend it in this work in several directions. We provide workflow and inconsistency descriptions and an evaluation, as well as a discussion of the threats to validity and a general discussion. We also extended the argumentation for the modeling approach.

8. Discussion

The presented modeling approach is a proposal for modeling inconsistencies in BPMN models. We evaluated it with a SUS, which we conducted with employees from the Daedalus GmbH. The resulting score of *75* illustrates the potential of our modeling approach for inconsistencies.

The exclusive focus on a single company restricts the external validity of the study, making it challenging to generalize the conclusions to other organizations within the same industry or different sectors, but this study lays the groundwork for future research endeavors and also serves as an impulse for the community, inspiring further scholarly engagement and stimulating discussions in related fields. In CPS engineering, not only software engineers, but also engineers from different domains and different education backgrounds are found. We should include their perspectives into the discussion of tools we can give practitioners to be able to communicate efficiently about inconsistencies. Our modeling approach is a promising starting point for the modeling of inconsistencies between technical models.

With clear definitions of inconsistent models, the specific inconsistencies between those models, and problematic workflows, software engineers can develop general solutions for pressing problems. This offers opportunities, e.g., in Product-Production-CoDesign by formalizing inconsistency between existing products and production systems and enabling continuous knowledge transfer (Albers et al. 2022). Especially by understanding the production system as one part of the (cyber-)physical product lifecycle, this works sets up an example for exploring inconsistencies occurring in product development. This is pressing, as the production system itself is not static but evolves in generations within the product engineering environment of companies and is directly linked to the lifecycle of CPS (Albers et al. 2016). Therefore, this work sets up an example for exploring inconsistencies occuring in product engineering from a production process point of view.

Furthermore, we provide the inconsistencies found as well as the complete workflows found at the precision component contract manufacturer in our replication package (*Replication Package* 2024). We encourage other researchers to use that dataset to model the inconsistencies found with their own modeling approach. The usage of a SUS enables the simple comparison of different approaches.

This dual role, i.e., providing real world data and a modeling approach, positions the research as a catalyst for both methodological advancement and collaborative knowledge exchange within the academic and professional community.

The approach is first and foremost tailored to the modeling of inconsistencies in workflows. The interviews and the interview process design can be re-used to conduct similar studies in different domains. The modeled inconsistencies could, depending on the domain and application, be used to automate inconsistency handling. While there are other modeling approaches that could be used, we think our proposed modeling approach avoid hindering the ease of use and understanding that may arise when directly relating artifacts.

9. Future Plans

For future work, we plan to mitigate the mentioned threats to validity by conducting additional interviews in the context of different companies with an increased number of participants, especially to separate the understanding of the workflows from the understanding of the inconsistencies. Furthermore, we want to separate the assessment of the usability of our modeling approach from modeling with it.

This separation is only possible with a much greater number of participants, which we unfortunately did not have in this study. However, we consider the results of this study as a good stepping stone for further research. Additionally, we plan to expand our research in several ways in the future. Firstly, we plan to apply a SUS to other methods to model inconsistencies in a BPMN diagram, once they become available. Additionally, we want to extend our modeling approach to different parts of the cyber-physical product lifecycle. The data gained will allow us to evaluate our modeling approach more thoroughly and develop it further.

While we think our approach is suitable, a new extension to BPMN may be even better to model inconsistencies, especially working in interdisciplinary scenarios in the engineering of cyber-physical systems. One has to keep in mind, that a new approach can always lead to interoperability problems, training costs, and user-friendliness issues. Secondly, we want to expand our approach in the direction of the work of van der Aa et al. (van der Aa et al. 2015) and develop a process for the identification of inconsistencies. The process will include (semi-)automatic inconsistency identification processes based on modeled workflows, as well as manual inconsistency identification based on interviews. Additionally, we want to formalize the inconsistencies modeled with the classification from Kühn et al. (Kühn et al. 2023).

The results of this paper can be used to explicitly document inconsistencies in workflows and enable the modeling community to develop general solutions for consistency management systems. The inconsistency modeling will especially be used and developed further, in different projects of the Collaborative Research Centre "Convide" (Reussner et al. 2023). In General, this study lays the groundwork for future research endeavors and also serves as an impulse for the community, inspiring further scholarly engagement and stimulating discussions in related fields.

10. Data Availability

Our data is available as a replication package (*Replication Package* 2024). Because the interviewees were German, the data we collected is also in German. If you need any derived artifacts, e.g., translated diagrams, please contact the authors of this paper.

11. Summary

This experience report from the manufacturing sector explored the pervasive issue of inconsistencies within workflows. Our special attention focused on inconsistencies in precision component contract manufacturing and the impediments these inconsistencies create for workflow completion. Inconsistencies often arise from contradictory information between different models (e.g., technical drawings and CAD models), and poor communication exacerbates these issues.

To enhance stakeholder communication while also modeling inconsistencies, we employed BPMN 2.0 diagrams and leveraged existing extensions. The results of our extensive expert interviews identified 13 potential inconsistencies within a precision manufacturer's workflows. While there is certainly potential for improvement, our proposed modeling approach achieved an overall good level of usability, scoring 75 on the System Usability Scale (Brooke 1996).

Our four principal contributions are as follows. First, we provide detailed BPMN models for all workflows of the precision manufacturer, and identify dimensions to classify the inconsistencies. Second, we introduce a novel approach for modeling inconsistencies within the BPMN models, levering the potential of existing extensions, and thereby assisting practitioners in identifying and addressing relevant issues. Third, we demonstrate the practical application of our modeling approach in a real-world setting, showcasing its relevance and effectiveness. Fourth, our evaluation underscores the practical utility of our approach while also indicating potential future enhancements.

Our findings highlight the critical importance of addressing inconsistencies to improve workflow efficiency. When left unmanaged, inconsistencies can lead to inefficiency and significant disruptions. Therefore, a method is needed that can make inconsistencies explicit while also providing a structured way to model and communicate them. Our modeling approach achieves this, contributing to the facilitation of mutual understanding and issue resolution for stakeholders from different disciplines.

In future work, we will refine our approach further based on more feedback from practitioners and additional usability testing. We aim to expand the applicability of our modeling approach to other domains beyond manufacturing, thereby broadening its impact. Furthermore, we plan to develop automated tools to support consistency management. We anticipate that automation will bring greater efficiency and effectiveness to the handling of inconsistencies, ultimately making workflows more robust.

Our research bridges the gap between machine processability and human comprehensibility. By addressing the challenge of inconsistencies head-on, we advance workflow management practices and ultimately support the development of more reliable and efficient systems across industry sectors.

Acknowledgments

This work was supported by funding from the topic Engineering Secure Systems of the Helmholtz Association (HGF), KASTEL Security Research Labs and the Collaborative Research Center "Convide" (CRC-1608) funded by the German Research Foundation (DFG). Thanks to our textician.

References

Aguilar-Savén, R. S. (2004). Business process modelling: Review and framework. *International Journal of production economics*, *90*(2), 129–149.

Albers, A., Koziolek, A., Völk, T. A., Klippert, M., Pfaff, F., Stolpmann, R., & Schwarz, S. E. (2024). Identification of inconsistencies in agile cps engineering with formula student. In I. Bitran, S. Conn, A. Mitsis, P. Ritala, M. Torkkeli, & M. Trabelsi (Eds.), *Proceedings of the xxxv ispim innovation conference* (pp. 1–15).

Albers, A., Lanza, G., Klippert, M., Schäfer, L., Frey, A., Hellweg, F., … others (2022). Product-production-codesign: An approach on integrated product and production engineering across generations and life cycles. *Procedia CIRP*, *109*, 167–172.

Albers, A., Reiss, N., Bursac, N., & Richter, T. (2016). Ipem-integrated product engineering model in context of product generation engineering. doi: 10.5445/IR/1000060192

Albers, A., Völk, T. A., Pfaff, F., & Nowak, K. (n.d.). *Inconsistency situations in engineering of cyber-physical systems.* Zenodo. doi: 10.5281/zenodo.13968934

Awadid, A., & Nurcan, S. (2019). Consistency requirements in business process modeling: a thorough overview. *Software & Systems Modeling*, *18*, 1097–1115.

Bangor, A., Kortum, P. T., & Miller, J. T. (2008). An empirical evaluation of the system usability scale. *Intl. Journal of Human–Computer Interaction*, *24*(6), 574–594.

Barker, A., & van Hemert, J. (2008). Scientific workflow: A survey and research directions. In (Vols. Wyrzykowski, Roman and Dongarra, Jack and Karczewski, Konrad and Wasniewski, Jerzy, pp. 746–753). Berlin, Heidelberg: Springer Berlin Heidelberg.

Brooke, J. (1996). Sus: a "quick and dirty'usability. *Usability evaluation in industry*, *189*(3), 189–194.

David, I., Vangheluwe, H., & Syriani, E. (2023). Model consistency as a heuristic for eventual correctness. *Journal of Computer Languages*, *76*, 101223.

Drew, M. R., Falcone, B., & Baccus, W. L. (2018). What does the system usability scale (sus) measure? validation using think aloud verbalization and behavioral metrics. In *Design, user experience, and usability: Theory and practice: 7th international conference, duxu 2018, held as part of hci international 2018, las vegas, nv, usa, july 15-20, 2018, proceedings, part i 7* (pp. 356–366).

Feldmann, S., Herzig, S. J., Kernschmidt, K., Wolfenstetter, T., Kammerl, D., Qamar, A., … Vogel-Heuser, B. (2015). Towards effective management of inconsistencies in model-based engineering of automated production systems. , 916-923. (15th IFAC Symposium on Information Control Problems in Manufacturing) doi: https://doi.org/10.1016/j.ifacol.2015.06.200

Fowler, F. J. (1995). *Improving survey questions: Design and evaluation*. Sage.

Herzig, S. J., Qamar, A., & Paredis, C. J. (2014). An approach to identifying inconsistencies in model-based systems engineering. *Procedia Computer Science*, *28*, 354–362.

Klare, H., Kramer, M. E., Langhammer, M., Werle, D., Burger, E., & Reussner, R. (2021). Enabling consistency in view-based system development—the vitruvius approach. *JSS*, *171*, 110815.

Kuder, N. D., Weber, T., Schneider, J., Weber, S., Völk, T. A., Albers, A., & Koziolek, A. (2024). Inconsistencies in production workflows and how to model them. In *2024 ieee 21st international conference on software architecture companion (icsa-c)*. doi: 10.1109/ICSA-C63560.2024.00021

Kühn, T., Fuchß, D., Corallo, S., König, L., Burger, E., Keim, J., … Reussner, R. (2023). *A formalized classification schema for model consistency* (Vol. 2023; Tech. Rep. No. 1). Karlsruher Institut für Technologie (KIT). doi: 10.5445/IR/1000161127

Liebel, G., Marko, N., Tichy, M., Leitner, A., & Hansson, J. (2018). Model-based engineering in the embedded systems domain: an industrial survey on the state-of-practice. *Software & Systems Modeling*, *17*, 91–113.

Marchezan, L., Assunção, W. K. G., Herac, E., Keplinger, F., Egyed, A., & Lauwerys, C. (2023). Fulfilling industrial needs for consistency among engineering artifacts. In *Icse-seip* (p. 246-257). doi: 10.1109/ICSE-SEIP58684.2023.00028

Marshall, G. (2005). The purpose, design and administration of a questionnaire for data collection. *Radiography*.

Meland, P. H., & Gjære, E. A. (2012). Representing threats in bpmn 2.0. In *2012 seventh international conference on availability, reliability and security* (p. 542-550).

Replication package. (2024). doi: 10.35097/YVtjSZhutJKPvdBi

Reussner, R., Schaefer, I., Beckert, B., Koziolek, A., & Burger, E. (2023). Consistency in the view-based development of cyber-physical systems (convide). In *2023 acm/ieee international conference on model driven engineering languages and systems companion (models-c)* (pp. 83–84).

Runeson, P., & Höst, M. (2009). Guidelines for conducting and reporting case study research in software engineering. *Empirical software engineering*, *14*, 131–164.

Taherdoost, H. (2022). Designing a questionnaire for a research paper: A comprehensive guide to design and develop an effective questionnaire. *Asian Journal of Managerial Science*.

van der Aa, H., Leopold, H., & Reijers, H. A. (2015). Detecting inconsistencies between process models and textual descriptions. In *Business process management* (pp. 90–105). Springer International Publishing.

Van Der Straeten, R., Mens, T., & Van Baelen, S. (2009). Challenges in model-driven software engineering. In M. R. V. Chaudron (Ed.), *Models in software engineering* (pp. 35–47). Berlin, Heidelberg: Springer Berlin Heidelberg.

White, S. A., et al. (2004). Process modeling notations and workflow patterns. *Workflow handbook*, *2004*(265-294), 12.

Zarour, K., Benmerzoug, D., Guermouche, N., & Drira, K. (2020). A systematic literature review on bpmn extensions. *Business Process Management Journal*, *26*(6), 1473–1503.

About the authors

Thomas Weber is PhD researcher at Karlsruhe Institute of Technology. You can contact the author at thomas.weber@kit.edu or visit https://dsis.kastel.kit.edu/staff_Thomas_Weber.php.

Niklas D. Kuder is a student at Karlsruhe Institute of Technology. You can contact the author at niklas.kuder@student.kit.edu.

Thomas A. Völk is PhD researcher at Karlsruhe Institute of Technology. You can contact the author at thomas.voelk@kit.edu or visit https://www.ipek.kit.edu/21_12970.php.

Jonas Schneider is founder and CEO of Daedalus GmbH. You can contact the author at jonas@daedalus.de or visit https://www.daedalus.de/team/jonas-schneider.

Sebastian Weber is a researcher at the FZI Research Center for Information Technology. You can contact the author at sebastian.weber@fzi.de or visit https://www.fzi.de/en/team/sebastian-weber/.

Anne Koziolek is a professor at Karlsruhe Institute of Technology. You can contact the author at koziolek@kit.edu or visit https://mcse.kastel.kit.edu/staff_Koziolek_Anne.php.